

Embroidermodder

2.0.0 alpha

Generated by Doxygen 1.9.6

| | |
|---|-----------|
| 1 Overview | 1 |
| 1.0.1 License | 1 |
| 2 About | 2 |
| 2.1 The Embroidermodder Project and Team | 2 |
| 2.1.1 "Core Development Team" | 2 |
| 2.2 for Embroidermodder 2, libembroidery and all other related code | 3 |
| 2.2.1 Embroidermodder 1 | 3 |
| 2.2.2 Features | 3 |
| 2.2.3 "Build and Install" | 4 |
| 2.2.4 History | 5 |
| 2.3 Contact us | 5 |
| 3 Downloads | 5 |
| 3.1 Alpha Build | 5 |
| 4 Tutorials | 6 |
| 4.1 Basic Features | 6 |
| 4.1.1 Move a single stitch in an existing pattern | 6 |
| 4.2 Altering a Single Stitch (2022-09-19) | 6 |
| 4.2.1 Convert one pattern to another format | 6 |
| 4.3 Advanced Features | 6 |
| 5 Post History | 6 |
| 5.1 Fast Forward (2014-02-13) | 7 |
| 5.2 January 2022 Development Notes For Embroidermodder 2 (2022-01-31) | 10 |
| 5.2.1 Broad Development Goals | 10 |
| 5.2.2 The New Settings System | 10 |
| 5.2.3 Reducing Reliance on Qt5 | 10 |
| 5.2.4 Palettes | 11 |
| 5.2.5 Conclusion | 11 |
| 5.3 June 2022 Backer Update (2022-06-22) | 11 |
| 5.3.1 Fill Algorithms | 11 |
| 5.3.2 Working Render Algorithm | 11 |
| 5.3.3 Timetable | 12 |
| 5.4 New Website! (2013-09-09) | 12 |
| 5.5 Crowdfunding Campaign Coming Soon! (2013-09-09) | 12 |
| 5.6 Our Kickstarter Crowdfunding Campaign is LIVE! (2014-03-17) | 13 |
| 5.7 Fast Forward (2014-02-13) | 13 |
| 6 Changelog | 13 |
| 7 Ideas | 13 |
| 8 Formats | 14 |

| | |
|--|-----------|
| 8.1 Overview | 14 |
| 8.1.1 Read/Write Support Levels | 14 |
| 8.1.2 Table of Format Support Levels | 15 |
| 8.1.3 Format Support | 16 |
| 9 Geometry and Algorithms | 17 |
| 9.1 To Do | 17 |
| 9.1.1 Development | 17 |
| 9.1.2 Testing | 18 |
| 9.1.3 Contributing | 18 |
| 9.1.4 Embroidermodder Project Coding Standards | 18 |
| 9.1.5 Version Control | 19 |
| 9.1.6 Donations | 19 |
| 9.1.7 Embroidermodder Project Coding Standards | 19 |
| 9.1.8 Ideas | 20 |
| 9.1.9 Electronics development | 23 |
| 9.1.10 Development | 23 |
| 9.2 Embroiderbot and Libembroidery on Embedded Systems | 23 |
| 9.2.1 Compatible Boards | 23 |
| 9.2.2 Arduino Considerations | 24 |
| 9.2.3 Space | 24 |
| 9.2.4 Tables | 24 |
| 9.2.5 Current Pattern Memory Management | 24 |
| 9.2.6 Special Notes | 25 |
| 9.2.7 The Assembly Split | 25 |
| 9.3 The Embroider Command Line Program | 25 |
| 9.3.1 Embroider pipeline | 25 |
| 9.3.2 embroider CLI | 25 |
| 10 GNU Free Documentation License | 26 |
| 11 Contributor Covenant Code of Conduct | 31 |
| 11.1 Our Pledge | 31 |
| 11.2 Our Standards | 31 |
| 11.3 Enforcement Responsibilities | 32 |
| 11.4 Scope | 32 |
| 11.5 Enforcement | 32 |
| 11.6 Enforcement Guidelines | 32 |
| 11.6.1 1. Correction | 32 |
| 11.6.2 2. Warning | 32 |
| 11.6.3 3. Temporary Ban | 33 |
| 11.6.4 4. Permanent Ban | 33 |
| 11.7 Attribution | 33 |

| | |
|--|-----------|
| 12 Privacy Policy for Embroidery Viewer | 33 |
| 12.0.1 CONTACT US | 34 |
| 13 Todo List | 34 |
| 14 Hierarchical Index | 37 |
| 14.1 Class Hierarchy | 37 |
| 15 Class Index | 41 |
| 15.1 Class List | 41 |
| 16 File Index | 44 |
| 16.1 File List | 44 |
| 17 Class Documentation | 48 |
| 17.1 _bcf_directory Struct Reference | 48 |
| 17.1.1 Detailed Description | 48 |
| 17.1.2 Member Data Documentation | 49 |
| 17.2 _bcf_directory_entry Struct Reference | 49 |
| 17.2.1 Member Data Documentation | 49 |
| 17.3 _bcf_file Struct Reference | 51 |
| 17.3.1 Member Data Documentation | 51 |
| 17.4 _bcf_file_difat Struct Reference | 52 |
| 17.4.1 Member Data Documentation | 52 |
| 17.5 _bcf_file_fat Struct Reference | 52 |
| 17.5.1 Member Data Documentation | 52 |
| 17.6 _bcf_file_header Struct Reference | 53 |
| 17.6.1 Detailed Description | 53 |
| 17.6.2 Member Data Documentation | 53 |
| 17.7 _vp3Hoop Struct Reference | 55 |
| 17.7.1 Member Data Documentation | 56 |
| 17.8 Action__ Struct Reference | 58 |
| 17.8.1 Detailed Description | 58 |
| 17.8.2 Member Data Documentation | 58 |
| 17.9 Application Class Reference | 59 |
| 17.9.1 Detailed Description | 59 |
| 17.9.2 Constructor & Destructor Documentation | 59 |
| 17.9.3 Member Function Documentation | 60 |
| 17.9.4 Member Data Documentation | 60 |
| 17.10 ArcObject Class Reference | 60 |
| 17.10.1 Member Enumeration Documentation | 63 |
| 17.10.2 Constructor & Destructor Documentation | 64 |
| 17.10.3 Member Function Documentation | 65 |
| 17.10.4 Member Data Documentation | 72 |

| | |
|--|-----|
| 17.11 BaseObject Class Reference | 73 |
| 17.11.1 Member Enumeration Documentation | 74 |
| 17.11.2 Constructor & Destructor Documentation | 74 |
| 17.11.3 Member Function Documentation | 74 |
| 17.11.4 Member Data Documentation | 79 |
| 17.12 CircleObject Class Reference | 79 |
| 17.12.1 Member Enumeration Documentation | 81 |
| 17.12.2 Constructor & Destructor Documentation | 82 |
| 17.12.3 Member Function Documentation | 82 |
| 17.13 CmdPrompt Class Reference | 85 |
| 17.13.1 Detailed Description | 87 |
| 17.13.2 Constructor & Destructor Documentation | 87 |
| 17.13.3 Member Function Documentation | 87 |
| 17.13.4 Member Data Documentation | 93 |
| 17.14 CmdPromptHandle Class Reference | 93 |
| 17.14.1 Detailed Description | 94 |
| 17.14.2 Constructor & Destructor Documentation | 94 |
| 17.14.3 Member Function Documentation | 94 |
| 17.14.4 Member Data Documentation | 95 |
| 17.15 CmdPromptHistory Class Reference | 95 |
| 17.15.1 Detailed Description | 96 |
| 17.15.2 Constructor & Destructor Documentation | 96 |
| 17.15.3 Member Function Documentation | 96 |
| 17.15.4 Member Data Documentation | 97 |
| 17.16 CmdPromptInput Class Reference | 98 |
| 17.16.1 Constructor & Destructor Documentation | 99 |
| 17.16.2 Member Function Documentation | 99 |
| 17.16.3 Member Data Documentation | 104 |
| 17.17 CmdPromptSplitter Class Reference | 105 |
| 17.17.1 Detailed Description | 105 |
| 17.17.2 Constructor & Destructor Documentation | 105 |
| 17.17.3 Member Function Documentation | 105 |
| 17.18 Compress Struct Reference | 106 |
| 17.18.1 Member Data Documentation | 106 |
| 17.19 DimLeaderObject Class Reference | 107 |
| 17.19.1 Member Enumeration Documentation | 110 |
| 17.19.2 Constructor & Destructor Documentation | 110 |
| 17.19.3 Member Function Documentation | 111 |
| 17.19.4 Member Data Documentation | 114 |
| 17.20 EllipseObject Class Reference | 115 |
| 17.20.1 Member Enumeration Documentation | 117 |
| 17.20.2 Constructor & Destructor Documentation | 117 |

| | |
|--|-----|
| 17.20.3 Member Function Documentation | 118 |
| 17.21 EmbAlignedDim_Struct Reference | 121 |
| 17.21.1 Member Data Documentation | 121 |
| 17.22 EmbAngularDim_Struct Reference | 121 |
| 17.22.1 Member Data Documentation | 121 |
| 17.23 EmbArc_Struct Reference | 121 |
| 17.23.1 Detailed Description | 122 |
| 17.23.2 Member Data Documentation | 122 |
| 17.24 EmbArcLengthDim_Struct Reference | 122 |
| 17.24.1 Member Data Documentation | 122 |
| 17.25 EmbArray_Struct Reference | 123 |
| 17.25.1 Member Data Documentation | 123 |
| 17.26 EmbBezier_Struct Reference | 124 |
| 17.26.1 Member Data Documentation | 124 |
| 17.27 EmbBlock_Struct Reference | 124 |
| 17.27.1 Member Data Documentation | 124 |
| 17.28 EmbCircle_Struct Reference | 125 |
| 17.28.1 Member Data Documentation | 125 |
| 17.29 EmbColor_Struct Reference | 125 |
| 17.29.1 Detailed Description | 125 |
| 17.29.2 Member Data Documentation | 126 |
| 17.30 EmbDetailsDialog Class Reference | 126 |
| 17.30.1 Detailed Description | 126 |
| 17.30.2 Constructor & Destructor Documentation | 126 |
| 17.30.3 Member Function Documentation | 127 |
| 17.30.4 Member Data Documentation | 127 |
| 17.31 EmbDiameterDim_Struct Reference | 128 |
| 17.31.1 Member Data Documentation | 128 |
| 17.32 EmbEllipse_Struct Reference | 128 |
| 17.32.1 Member Data Documentation | 129 |
| 17.33 EmbFormatList_Struct Reference | 129 |
| 17.33.1 Member Data Documentation | 129 |
| 17.34 EmbGeometry_Struct Reference | 130 |
| 17.34.1 Member Data Documentation | 131 |
| 17.35 EmblImage_Struct Reference | 132 |
| 17.35.1 Member Data Documentation | 133 |
| 17.36 EmblInfiniteLine_Struct Reference | 133 |
| 17.36.1 Member Data Documentation | 134 |
| 17.37 EmbLayer_Struct Reference | 134 |
| 17.37.1 Member Data Documentation | 134 |
| 17.38 EmbLeaderDim_Struct Reference | 134 |
| 17.38.1 Member Data Documentation | 135 |

| | |
|---|-----|
| 17.39 EmbLine_ Struct Reference | 135 |
| 17.39.1 Member Data Documentation | 135 |
| 17.40 EmbLinearDim_ Struct Reference | 136 |
| 17.40.1 Member Data Documentation | 136 |
| 17.41 EmbOrdinateDim_ Struct Reference | 136 |
| 17.41.1 Member Data Documentation | 136 |
| 17.42 EmbPath_ Struct Reference | 136 |
| 17.42.1 Member Data Documentation | 137 |
| 17.43 EmbPattern_ Struct Reference | 137 |
| 17.43.1 Member Data Documentation | 138 |
| 17.44 EmbPoint_ Struct Reference | 138 |
| 17.44.1 Member Data Documentation | 139 |
| 17.45 EmbRadiusDim_ Struct Reference | 139 |
| 17.45.1 Member Data Documentation | 139 |
| 17.46 EmbRay_ Struct Reference | 140 |
| 17.46.1 Member Data Documentation | 140 |
| 17.47 EmbRect_ Struct Reference | 140 |
| 17.47.1 Member Data Documentation | 140 |
| 17.48 EmbSatinOutline_ Struct Reference | 141 |
| 17.48.1 Member Data Documentation | 141 |
| 17.49 EmbSpline_ Struct Reference | 142 |
| 17.49.1 Member Data Documentation | 142 |
| 17.50 EmbStitch_ Struct Reference | 142 |
| 17.50.1 Member Data Documentation | 142 |
| 17.51 EmbTextMulti_ Struct Reference | 143 |
| 17.51.1 Member Data Documentation | 143 |
| 17.52 EmbTextSingle_ Struct Reference | 143 |
| 17.52.1 Member Data Documentation | 144 |
| 17.53 EmbThread_ Struct Reference | 144 |
| 17.53.1 Member Data Documentation | 144 |
| 17.54 EmbTime_ Struct Reference | 145 |
| 17.54.1 Member Data Documentation | 145 |
| 17.55 EmbVector_ Struct Reference | 146 |
| 17.55.1 Detailed Description | 146 |
| 17.55.2 Member Data Documentation | 146 |
| 17.56 EmbView_ Struct Reference | 146 |
| 17.56.1 Detailed Description | 147 |
| 17.56.2 EmbViews | 147 |
| 17.56.3 Member Data Documentation | 147 |
| 17.57 hoop_padding Struct Reference | 150 |
| 17.57.1 Member Data Documentation | 150 |
| 17.58 Huffman Struct Reference | 151 |

| | |
|--|-----|
| 17.58.1 Member Data Documentation | 151 |
| 17.59 ImageObject Class Reference | 152 |
| 17.59.1 Member Enumeration Documentation | 154 |
| 17.59.2 Constructor & Destructor Documentation | 154 |
| 17.59.3 Member Function Documentation | 155 |
| 17.60 ImageWidget Class Reference | 157 |
| 17.60.1 Detailed Description | 157 |
| 17.60.2 Constructor & Destructor Documentation | 157 |
| 17.60.3 Member Function Documentation | 158 |
| 17.60.4 Member Data Documentation | 159 |
| 17.61 LayerManager Class Reference | 159 |
| 17.61.1 Detailed Description | 159 |
| 17.61.2 Constructor & Destructor Documentation | 159 |
| 17.61.3 Member Function Documentation | 160 |
| 17.61.4 Member Data Documentation | 160 |
| 17.62 LineObject Class Reference | 161 |
| 17.62.1 Member Enumeration Documentation | 163 |
| 17.62.2 Constructor & Destructor Documentation | 163 |
| 17.62.3 Member Function Documentation | 164 |
| 17.63 LSYSTEM Struct Reference | 167 |
| 17.63.1 Member Data Documentation | 167 |
| 17.64 MainWindow Class Reference | 167 |
| 17.64.1 Detailed Description | 177 |
| 17.64.2 Constructor & Destructor Documentation | 177 |
| 17.64.3 Member Function Documentation | 177 |
| 17.64.4 Member Data Documentation | 207 |
| 17.65 MdiArea Class Reference | 221 |
| 17.65.1 Constructor & Destructor Documentation | 222 |
| 17.65.2 Member Function Documentation | 222 |
| 17.65.3 Member Data Documentation | 225 |
| 17.66 MdiWindow Class Reference | 225 |
| 17.66.1 Constructor & Destructor Documentation | 227 |
| 17.66.2 Member Function Documentation | 228 |
| 17.66.3 Member Data Documentation | 234 |
| 17.67 Parameter_Struct Reference | 235 |
| 17.67.1 Member Data Documentation | 235 |
| 17.68 PathObject Class Reference | 236 |
| 17.68.1 Member Enumeration Documentation | 238 |
| 17.68.2 Constructor & Destructor Documentation | 238 |
| 17.68.3 Member Function Documentation | 238 |
| 17.68.4 Member Data Documentation | 241 |
| 17.69 PointObject Class Reference | 241 |

| | |
|--|-----|
| 17.69.1 Member Enumeration Documentation | 243 |
| 17.69.2 Constructor & Destructor Documentation | 243 |
| 17.69.3 Member Function Documentation | 243 |
| 17.70 PolygonObject Class Reference | 245 |
| 17.70.1 Member Enumeration Documentation | 247 |
| 17.70.2 Constructor & Destructor Documentation | 248 |
| 17.70.3 Member Function Documentation | 248 |
| 17.70.4 Member Data Documentation | 250 |
| 17.71 PolylineObject Class Reference | 251 |
| 17.71.1 Member Enumeration Documentation | 253 |
| 17.71.2 Constructor & Destructor Documentation | 253 |
| 17.71.3 Member Function Documentation | 254 |
| 17.71.4 Member Data Documentation | 256 |
| 17.72 PreviewDialog Class Reference | 256 |
| 17.72.1 Constructor & Destructor Documentation | 256 |
| 17.72.2 Member Data Documentation | 257 |
| 17.73 PropertyEditor Class Reference | 257 |
| 17.73.1 Constructor & Destructor Documentation | 259 |
| 17.73.2 Member Function Documentation | 260 |
| 17.73.3 Member Data Documentation | 266 |
| 17.74 RectObject Class Reference | 270 |
| 17.74.1 Member Enumeration Documentation | 272 |
| 17.74.2 Constructor & Destructor Documentation | 273 |
| 17.74.3 Member Function Documentation | 273 |
| 17.75 SaveObject Class Reference | 276 |
| 17.75.1 Constructor & Destructor Documentation | 277 |
| 17.75.2 Member Function Documentation | 277 |
| 17.75.3 Member Data Documentation | 282 |
| 17.76 SelectBox Class Reference | 282 |
| 17.76.1 Constructor & Destructor Documentation | 283 |
| 17.76.2 Member Function Documentation | 283 |
| 17.76.3 Member Data Documentation | 284 |
| 17.77 Settings_Struct Reference | 285 |
| 17.77.1 Detailed Description | 287 |
| 17.77.2 Member Data Documentation | 287 |
| 17.78 Settings_Dialog Class Reference | 294 |
| 17.78.1 Constructor & Destructor Documentation | 299 |
| 17.78.2 Member Function Documentation | 299 |
| 17.78.3 Member Data Documentation | 308 |
| 17.79 StatusBar Class Reference | 315 |
| 17.79.1 Constructor & Destructor Documentation | 316 |
| 17.79.2 Member Function Documentation | 316 |

| | |
|--|-----|
| 17.79.3 Member Data Documentation | 316 |
| 17.80 StatusBarButton Class Reference | 317 |
| 17.80.1 Constructor & Destructor Documentation | 317 |
| 17.80.2 Member Function Documentation | 318 |
| 17.80.3 Member Data Documentation | 319 |
| 17.81 StxThread_ Struct Reference | 319 |
| 17.81.1 Member Data Documentation | 320 |
| 17.82 SubDescriptor_ Struct Reference | 320 |
| 17.82.1 Member Data Documentation | 320 |
| 17.83 SvgAttribute_ Struct Reference | 321 |
| 17.83.1 Member Data Documentation | 321 |
| 17.84 TextSingleObject Class Reference | 321 |
| 17.84.1 Member Enumeration Documentation | 323 |
| 17.84.2 Constructor & Destructor Documentation | 323 |
| 17.84.3 Member Function Documentation | 324 |
| 17.84.4 Member Data Documentation | 326 |
| 17.85 thread_color_ Struct Reference | 327 |
| 17.85.1 Member Data Documentation | 327 |
| 17.86 ThredExtension_ Struct Reference | 327 |
| 17.86.1 Member Data Documentation | 327 |
| 17.87 ThredHeader_ Struct Reference | 328 |
| 17.87.1 Member Data Documentation | 328 |
| 17.88 UiObject_ Struct Reference | 329 |
| 17.88.1 Detailed Description | 329 |
| 17.88.2 Member Data Documentation | 329 |
| 17.89 UndoableAddCommand Class Reference | 331 |
| 17.89.1 Constructor & Destructor Documentation | 331 |
| 17.89.2 Member Function Documentation | 331 |
| 17.89.3 Member Data Documentation | 332 |
| 17.90 UndoableDeleteCommand Class Reference | 332 |
| 17.90.1 Constructor & Destructor Documentation | 332 |
| 17.90.2 Member Function Documentation | 332 |
| 17.90.3 Member Data Documentation | 332 |
| 17.91 UndoableGripEditCommand Class Reference | 333 |
| 17.91.1 Constructor & Destructor Documentation | 333 |
| 17.91.2 Member Function Documentation | 333 |
| 17.91.3 Member Data Documentation | 333 |
| 17.92 UndoableMirrorCommand Class Reference | 334 |
| 17.92.1 Constructor & Destructor Documentation | 334 |
| 17.92.2 Member Function Documentation | 334 |
| 17.92.3 Member Data Documentation | 334 |
| 17.93 UndoableMoveCommand Class Reference | 335 |

| | |
|--|------------|
| 17.93.1 Constructor & Destructor Documentation | 335 |
| 17.93.2 Member Function Documentation | 335 |
| 17.93.3 Member Data Documentation | 335 |
| 17.94 UndoableNavCommand Class Reference | 336 |
| 17.94.1 Constructor & Destructor Documentation | 336 |
| 17.94.2 Member Function Documentation | 336 |
| 17.94.3 Member Data Documentation | 336 |
| 17.95 UndoableRotateCommand Class Reference | 337 |
| 17.95.1 Constructor & Destructor Documentation | 337 |
| 17.95.2 Member Function Documentation | 337 |
| 17.95.3 Member Data Documentation | 338 |
| 17.96 UndoableScaleCommand Class Reference | 338 |
| 17.96.1 Constructor & Destructor Documentation | 338 |
| 17.96.2 Member Function Documentation | 339 |
| 17.96.3 Member Data Documentation | 339 |
| 17.97 UndoEditor Class Reference | 339 |
| 17.97.1 Constructor & Destructor Documentation | 340 |
| 17.97.2 Member Function Documentation | 340 |
| 17.97.3 Member Data Documentation | 340 |
| 17.98 UndoHistory_ Struct Reference | 341 |
| 17.98.1 Detailed Description | 341 |
| 17.98.2 Member Data Documentation | 341 |
| 17.99 View Class Reference | 341 |
| 17.99.1 Constructor & Destructor Documentation | 344 |
| 17.99.2 Member Function Documentation | 345 |
| 17.99.3 Member Data Documentation | 352 |
| 17.100 VipHeader_ Struct Reference | 355 |
| 17.100.1 Member Data Documentation | 355 |
| 18 File Documentation | 356 |
| 18.1 CODE_OF_CONDUCT.md File Reference | 356 |
| 18.2 embroidermodder2/cmdprompt.cpp File Reference | 356 |
| 18.3 embroidermodder2/embdetails-dialog.cpp File Reference | 356 |
| 18.4 embroidermodder2/embroidermodder.cpp File Reference | 356 |
| 18.4.1 Function Documentation | 357 |
| 18.4.2 Variable Documentation | 357 |
| 18.5 embroidermodder2/embroidermodder.h File Reference | 357 |
| 18.5.1 Detailed Description | 361 |
| 18.5.2 Typedef Documentation | 361 |
| 18.5.3 EmbViews | 361 |
| 18.5.4 Enumeration Type Documentation | 362 |
| 18.5.5 Function Documentation | 367 |

| | |
|---|-----|
| 18.5.6 Variable Documentation | 367 |
| 18.6 embroidermodder.h | 368 |
| 18.7 embroidermodder2/imagewidget.cpp File Reference | 404 |
| 18.8 embroidermodder2/layer-manager.cpp File Reference | 404 |
| 18.8.1 Detailed Description | 404 |
| 18.9 embroidermodder2/mainwindow-commands.cpp File Reference | 404 |
| 18.10 embroidermodder2/mainwindow-menus.cpp File Reference | 404 |
| 18.11 embroidermodder2/mainwindow-settings.cpp File Reference | 404 |
| 18.11.1 Function Documentation | 405 |
| 18.12 embroidermodder2/mainwindow-toolbars.cpp File Reference | 406 |
| 18.12.1 Function Documentation | 406 |
| 18.12.2 Variable Documentation | 406 |
| 18.13 embroidermodder2/mainwindow.cpp File Reference | 407 |
| 18.13.1 Typedef Documentation | 408 |
| 18.13.2 Function Documentation | 408 |
| 18.13.3 Variable Documentation | 409 |
| 18.14 embroidermodder2/mdiarea.cpp File Reference | 411 |
| 18.15 embroidermodder2/mdiwindow.cpp File Reference | 411 |
| 18.16 embroidermodder2/object-arc.cpp File Reference | 411 |
| 18.16.1 Function Documentation | 411 |
| 18.17 embroidermodder2/object-base.cpp File Reference | 411 |
| 18.18 embroidermodder2/object-circle.cpp File Reference | 411 |
| 18.19 embroidermodder2/object-dimleader.cpp File Reference | 411 |
| 18.20 embroidermodder2/object-ellipse.cpp File Reference | 411 |
| 18.21 embroidermodder2/object-image.cpp File Reference | 411 |
| 18.22 embroidermodder2/object-line.cpp File Reference | 411 |
| 18.23 embroidermodder2/object-path.cpp File Reference | 412 |
| 18.24 embroidermodder2/object-point.cpp File Reference | 412 |
| 18.25 embroidermodder2/object-polygon.cpp File Reference | 412 |
| 18.26 embroidermodder2/object-polyline.cpp File Reference | 412 |
| 18.27 embroidermodder2/object-rect.cpp File Reference | 412 |
| 18.27.1 Detailed Description | 412 |
| 18.28 embroidermodder2/object-save.cpp File Reference | 412 |
| 18.29 embroidermodder2/object-textsingle.cpp File Reference | 412 |
| 18.30 embroidermodder2/preview-dialog.cpp File Reference | 412 |
| 18.31 embroidermodder2/property-editor.cpp File Reference | 412 |
| 18.31.1 Variable Documentation | 416 |
| 18.32 embroidermodder2/README.md File Reference | 428 |
| 18.33 embroidermodder2/selectbox.cpp File Reference | 428 |
| 18.34 embroidermodder2/settings-dialog.cpp File Reference | 428 |
| 18.35 embroidermodder2/statusbar-button.cpp File Reference | 428 |
| 18.36 embroidermodder2/statusbar.cpp File Reference | 428 |

| | |
|--|-----|
| 18.37 <code>embroidermodder2/undo-commands.cpp</code> File Reference | 428 |
| 18.38 <code>embroidermodder2/undo-editor.cpp</code> File Reference | 428 |
| 18.38.1 Detailed Description | 429 |
| 18.39 <code>embroidermodder2/view.cpp</code> File Reference | 429 |
| 18.39.1 Detailed Description | 429 |
| 18.40 <code>extern/libembroidery/src/array.c</code> File Reference | 429 |
| 18.40.1 Function Documentation | 429 |
| 18.41 <code>extern/libembroidery/src/compress.c</code> File Reference | 431 |
| 18.41.1 Detailed Description | 431 |
| 18.41.2 Function Documentation | 431 |
| 18.41.3 Variable Documentation | 433 |
| 18.42 <code>extern/libembroidery/src/embroidery.h</code> File Reference | 433 |
| 18.42.1 Macro Definition Documentation | 440 |
| 18.42.2 Typedef Documentation | 447 |
| 18.42.3 Function Documentation | 450 |
| 18.42.4 Variable Documentation | 465 |
| 18.43 <code>embroidery.h</code> | 466 |
| 18.44 <code>extern/libembroidery/src/embroidery_internal.h</code> File Reference | 473 |
| 18.44.1 Macro Definition Documentation | 481 |
| 18.44.2 Typedef Documentation | 489 |
| 18.44.3 Enumeration Type Documentation | 490 |
| 18.44.4 Function Documentation | 490 |
| 18.44.5 Variable Documentation | 513 |
| 18.45 <code>embroidery_internal.h</code> | 513 |
| 18.46 <code>extern/libembroidery/src/encoding.c</code> File Reference | 520 |
| 18.46.1 Detailed Description | 521 |
| 18.46.2 Function Documentation | 521 |
| 18.47 <code>extern/libembroidery/src/fill.c</code> File Reference | 523 |
| 18.47.1 Function Documentation | 524 |
| 18.47.2 Variable Documentation | 527 |
| 18.48 <code>extern/libembroidery/src/formats.c</code> File Reference | 528 |
| 18.48.1 Function Documentation | 529 |
| 18.48.2 Variable Documentation | 531 |
| 18.49 <code>extern/libembroidery/src/formats/format_100.c</code> File Reference | 532 |
| 18.49.1 Detailed Description | 532 |
| 18.49.2 Function Documentation | 532 |
| 18.50 <code>extern/libembroidery/src/formats/format_10o.c</code> File Reference | 532 |
| 18.50.1 Detailed Description | 532 |
| 18.50.2 Function Documentation | 533 |
| 18.51 <code>extern/libembroidery/src/formats/format_art.c</code> File Reference | 533 |
| 18.51.1 Detailed Description | 533 |
| 18.51.2 Function Documentation | 533 |

| | |
|---|-----|
| 18.52 <code>extern/libembroidery/src/formats/format_bmc.c</code> File Reference | 533 |
| 18.52.1 Detailed Description | 534 |
| 18.52.2 Function Documentation | 534 |
| 18.53 <code>extern/libembroidery/src/formats/format_bro.c</code> File Reference | 534 |
| 18.53.1 Detailed Description | 534 |
| 18.53.2 Function Documentation | 534 |
| 18.54 <code>extern/libembroidery/src/formats/format_cnd.c</code> File Reference | 534 |
| 18.54.1 Detailed Description | 535 |
| 18.54.2 Function Documentation | 535 |
| 18.55 <code>extern/libembroidery/src/formats/format_col.c</code> File Reference | 535 |
| 18.55.1 Detailed Description | 535 |
| 18.55.2 Function Documentation | 536 |
| 18.56 <code>extern/libembroidery/src/formats/format_csd.c</code> File Reference | 536 |
| 18.56.1 Detailed Description | 536 |
| 18.56.2 Macro Definition Documentation | 536 |
| 18.56.3 Function Documentation | 536 |
| 18.56.4 Variable Documentation | 537 |
| 18.57 <code>extern/libembroidery/src/formats/format_csv.c</code> File Reference | 537 |
| 18.57.1 Detailed Description | 538 |
| 18.57.2 Function Documentation | 538 |
| 18.58 <code>extern/libembroidery/src/formats/format_dat.c</code> File Reference | 538 |
| 18.58.1 Function Documentation | 538 |
| 18.59 <code>extern/libembroidery/src/formats/format_dem.c</code> File Reference | 539 |
| 18.59.1 Detailed Description | 539 |
| 18.59.2 Function Documentation | 539 |
| 18.60 <code>extern/libembroidery/src/formats/format_dsb.c</code> File Reference | 539 |
| 18.60.1 Detailed Description | 539 |
| 18.60.2 Function Documentation | 540 |
| 18.61 <code>extern/libembroidery/src/formats/format_dst.c</code> File Reference | 540 |
| 18.61.1 Detailed Description | 540 |
| 18.61.2 Macro Definition Documentation | 541 |
| 18.61.3 Function Documentation | 541 |
| 18.62 <code>extern/libembroidery/src/formats/format_dsz.c</code> File Reference | 542 |
| 18.62.1 Function Documentation | 542 |
| 18.63 <code>extern/libembroidery/src/formats/format_dxf.c</code> File Reference | 542 |
| 18.63.1 Function Documentation | 543 |
| 18.64 <code>extern/libembroidery/src/formats/format_edr.c</code> File Reference | 543 |
| 18.64.1 Function Documentation | 543 |
| 18.65 <code>extern/libembroidery/src/formats/format_emd.c</code> File Reference | 544 |
| 18.65.1 Detailed Description | 544 |
| 18.65.2 Function Documentation | 544 |
| 18.66 <code>extern/libembroidery/src/formats/format_exp.c</code> File Reference | 544 |

| | |
|--|-----|
| 18.66.1 Function Documentation | 544 |
| 18.67 extern/libembroidery/src/formats/format_exy.c File Reference | 545 |
| 18.67.1 Function Documentation | 545 |
| 18.68 extern/libembroidery/src/formats/format_eyc.c File Reference | 545 |
| 18.68.1 Function Documentation | 545 |
| 18.69 extern/libembroidery/src/formats/format_fxy.c File Reference | 546 |
| 18.69.1 Function Documentation | 546 |
| 18.70 extern/libembroidery/src/formats/format_gc.c File Reference | 546 |
| 18.70.1 Function Documentation | 546 |
| 18.71 extern/libembroidery/src/formats/format_gnc.c File Reference | 547 |
| 18.71.1 Function Documentation | 547 |
| 18.72 extern/libembroidery/src/formats/format_gt.c File Reference | 547 |
| 18.72.1 Function Documentation | 547 |
| 18.73 extern/libembroidery/src/formats/format_hus.c File Reference | 548 |
| 18.73.1 Function Documentation | 548 |
| 18.74 extern/libembroidery/src/formats/format_inb.c File Reference | 549 |
| 18.74.1 Function Documentation | 549 |
| 18.75 extern/libembroidery/src/formats/format_inf.c File Reference | 549 |
| 18.75.1 Function Documentation | 549 |
| 18.76 extern/libembroidery/src/formats/format_jef.c File Reference | 550 |
| 18.76.1 Function Documentation | 550 |
| 18.77 extern/libembroidery/src/formats/format_ksm.c File Reference | 551 |
| 18.77.1 Function Documentation | 551 |
| 18.78 extern/libembroidery/src/formats/format_max.c File Reference | 551 |
| 18.78.1 Function Documentation | 552 |
| 18.78.2 Variable Documentation | 552 |
| 18.79 extern/libembroidery/src/formats/format_mit.c File Reference | 552 |
| 18.79.1 Function Documentation | 552 |
| 18.80 extern/libembroidery/src/formats/format_new.c File Reference | 553 |
| 18.80.1 Function Documentation | 553 |
| 18.81 extern/libembroidery/src/formats/format_ofm.c File Reference | 553 |
| 18.81.1 Function Documentation | 554 |
| 18.82 extern/libembroidery/src/formats/format_pcd.c File Reference | 554 |
| 18.82.1 Function Documentation | 555 |
| 18.83 extern/libembroidery/src/formats/format_pcm.c File Reference | 555 |
| 18.83.1 Function Documentation | 555 |
| 18.84 extern/libembroidery/src/formats/format_pcq.c File Reference | 555 |
| 18.84.1 Function Documentation | 556 |
| 18.85 extern/libembroidery/src/formats/format_pcs.c File Reference | 556 |
| 18.85.1 Function Documentation | 556 |
| 18.86 extern/libembroidery/src/formats/format_pec.c File Reference | 556 |
| 18.86.1 Function Documentation | 557 |

| | |
|---|-----|
| 18.87 extern/libembroidery/src/formats/format_pel.c File Reference | 558 |
| 18.87.1 Function Documentation | 558 |
| 18.88 extern/libembroidery/src/formats/format_pem.c File Reference | 558 |
| 18.88.1 Function Documentation | 558 |
| 18.89 extern/libembroidery/src/formats/format_pes.c File Reference | 559 |
| 18.89.1 Function Documentation | 559 |
| 18.89.2 Variable Documentation | 561 |
| 18.90 extern/libembroidery/src/formats/format_phb.c File Reference | 561 |
| 18.90.1 Function Documentation | 561 |
| 18.91 extern/libembroidery/src/formats/format_phc.c File Reference | 562 |
| 18.91.1 Function Documentation | 562 |
| 18.92 extern/libembroidery/src/formats/format_plt.c File Reference | 562 |
| 18.92.1 Function Documentation | 562 |
| 18.93 extern/libembroidery/src/formats/format_rgb.c File Reference | 563 |
| 18.93.1 Function Documentation | 563 |
| 18.94 extern/libembroidery/src/formats/format_sew.c File Reference | 563 |
| 18.94.1 Function Documentation | 563 |
| 18.95 extern/libembroidery/src/formats/format_shv.c File Reference | 564 |
| 18.95.1 Function Documentation | 564 |
| 18.96 extern/libembroidery/src/formats/format_sst.c File Reference | 564 |
| 18.96.1 Function Documentation | 564 |
| 18.97 extern/libembroidery/src/formats/format_stx.c File Reference | 565 |
| 18.97.1 Function Documentation | 565 |
| 18.98 extern/libembroidery/src/formats/format_svg.c File Reference | 565 |
| 18.98.1 Function Documentation | 566 |
| 18.98.2 Variable Documentation | 566 |
| 18.99 extern/libembroidery/src/formats/format_t01.c File Reference | 567 |
| 18.99.1 Function Documentation | 567 |
| 18.100 extern/libembroidery/src/formats/format_t09.c File Reference | 567 |
| 18.100.1 Function Documentation | 567 |
| 18.101 extern/libembroidery/src/formats/format.tap.c File Reference | 568 |
| 18.101.1 Function Documentation | 568 |
| 18.102 extern/libembroidery/src/formats/format_thr.c File Reference | 568 |
| 18.102.1 Function Documentation | 568 |
| 18.103 extern/libembroidery/src/formats/format_txt.c File Reference | 569 |
| 18.103.1 Function Documentation | 569 |
| 18.104 extern/libembroidery/src/formats/format_u00.c File Reference | 569 |
| 18.104.1 Function Documentation | 569 |
| 18.105 extern/libembroidery/src/formats/format_u01.c File Reference | 570 |
| 18.105.1 Function Documentation | 570 |
| 18.106 extern/libembroidery/src/formats/format_vip.c File Reference | 570 |
| 18.106.1 Function Documentation | 571 |

| | |
|---|-----|
| 18.106.2 Variable Documentation | 571 |
| 18.107 extern/libembroidery/src/formats/format_vp3.c File Reference | 572 |
| 18.107.1 Function Documentation | 572 |
| 18.108 extern/libembroidery/src/formats/format_xxx.c File Reference | 573 |
| 18.108.1 Function Documentation | 573 |
| 18.109 extern/libembroidery/src/formats/format_zsk.c File Reference | 574 |
| 18.109.1 Detailed Description | 574 |
| 18.109.2 Function Documentation | 574 |
| 18.110 extern/libembroidery/src/geometry.c File Reference | 574 |
| 18.110.1 Function Documentation | 575 |
| 18.111 extern/libembroidery/src/geometry/arc.c File Reference | 576 |
| 18.111.1 Function Documentation | 576 |
| 18.112 extern/libembroidery/src/geometry/circle.c File Reference | 579 |
| 18.112.1 Function Documentation | 580 |
| 18.113 extern/libembroidery/src/geometry/ellipse.c File Reference | 580 |
| 18.113.1 Function Documentation | 581 |
| 18.114 extern/libembroidery/src/geometry/functions.c File Reference | 582 |
| 18.114.1 Function Documentation | 582 |
| 18.115 extern/libembroidery/src/geometry/line.c File Reference | 583 |
| 18.115.1 Function Documentation | 583 |
| 18.116 extern/libembroidery/src/geometry/path.c File Reference | 583 |
| 18.117 extern/libembroidery/src/geometry/polygon.c File Reference | 583 |
| 18.118 extern/libembroidery/src/geometry/polyline.c File Reference | 583 |
| 18.119 extern/libembroidery/src/geometry/rect.c File Reference | 584 |
| 18.119.1 Function Documentation | 584 |
| 18.120 extern/libembroidery/src/geometry/text.c File Reference | 584 |
| 18.120.1 Function Documentation | 584 |
| 18.121 extern/libembroidery/src/geometry/vector.c File Reference | 586 |
| 18.121.1 Function Documentation | 586 |
| 18.122 extern/libembroidery/src/image.c File Reference | 588 |
| 18.122.1 Detailed Description | 588 |
| 18.122.2 Function Documentation | 588 |
| 18.123 extern/libembroidery/src/main.c File Reference | 589 |
| 18.123.1 Macro Definition Documentation | 591 |
| 18.123.2 Function Documentation | 594 |
| 18.123.3 Variable Documentation | 600 |
| 18.124 extern/libembroidery/src/pattern.c File Reference | 600 |
| 18.124.1 Detailed Description | 601 |
| 18.124.2 Function Documentation | 601 |
| 18.125 extern/libembroidery/src/thread-color.c File Reference | 605 |
| 18.125.1 Function Documentation | 606 |
| 18.125.2 Variable Documentation | 606 |

| | |
|---|------------|
| 18.126 privacy_policy.md File Reference | 607 |
| Bibliography | 608 |
| Index | 609 |

1 Overview

Version

2.0.0-alpha

Author

The Embroidermodder Team

(UNDER MAJOR RESTRUCTURING, PLEASE WAIT FOR VERSION 2)

<http://www.libembroidery.org>

Embroidermodder is a free machine embroidery application. The newest version, Embroidermodder 2 can:

- edit and create embroidery designs
- estimate the amount of thread and machine time needed to stitch a design
- convert embroidery files to a variety of formats
- upscale or downscale designs
- run on Windows, Mac and Linux

Embroidermodder 2 is very much a work in progress since we're doing a ground up rewrite to an interface in C using the GUI toolkit SDL2. The reasoning for this is detailed in the issues tab.

For a more in-depth look at what we are developing read our [website](#) which includes these docs as well as the up-to date printer-friendly versions. These discuss recent changes, plans and has user and developer guides for all the Embroidermodder projects.

To see what we're focussing on right now, see the [Open Collective News](#).

1.0.1 License

The source code is under the terms of the zlib license: see `LICENSE.md` in the source code directory.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

A copy of the license is included in the section entitled "GNU Free Documentation License".

2 About

2.1 The Embroidermodder Project and Team

The *Embroidermodder 2* project is a collection of small software utilities for manipulating, converting and creating embroidery files in all major embroidery machine formats. The program *Embroidermodder 2* itself is a larger graphical user interface (GUI) which is at the heart of the project.

The tools and associated documents are:

- This website (www.libembroidery.org), which is maintained [here](#).
- [The manual](#) covering all these projects.
- The GUI (`embroidermodder`), maintained [here](#).
- The core library of low-level functions: [libembroidery](#).
- The CLI `embroider` which is part of [libembroidery](#).
- Mobile embroidery format viewers and tools ([EmbroideryMobile](#))).
- Specs for an open hardware embroidery machine called Embroiderbot (not started yet) which is also part of [libembroidery](#).

They all tools to make the standard user experience of working with an embroidery machine better without expensive software which is locked to specific manufacturers and formats. But ultimately we hope that the core *Embroidermodder 2* is a practical, ever-present tool in larger workshops, small cottage industry workshops and personal hobbyist's bedrooms.

Embroidermodder 2 is licensed under the zlib license and we aim to keep all of our tools open source and free of charge. If you would like to support the project check out our [Open Collective](#) group. If you would like to help, please join us on GitHub. This document is written as developer training as well helping new users (see the last sections) so this is the place to learn how to start changing the code.

The Embroidermodder Team is the collection of people who've submitted patches, artwork and documentation to our three projects. The team was established by Jonathan Greig and Josh Varga. The full list is actively maintained below.

2.1.1 "Core Development Team"

Embroidermodder 2:

- [Jonathan Greig](#)
- [Josh Varga](#)
- [Robin Swift](#)

Embroidermodder 1:

- [Josh Varga](#)
- [Mark Pontius](#)

2.2 for Embroidermodder 2, libembroidery and all other related code

If you have contributed and wish to be added to this list, alter the [README on Embroidermodder github page](#) and we'll copy it to the libembroidery source code since that is credited to "The Embroidermodder Team".

2.2.1 Embroidermodder 1

The Embroidermodder Team is also inspired by the original Embroidermodder that was built by Mark Pontius and the same Josh Varga on SourceForge which unfortunately appears to have died from linkrot. We may create a distribution on here to be the official "legacy" Embroidermodder code but likely in a separate repository because it's GNU GPL v3 and this code is written to be zlib (that is, permissive licensed) all the way down.

One reason why this is useful is that the rewrite by Jonathan Greig, John Varga and Robin Swift for Embroidermodder 2 should have no regressions: no features present in v1 should be missing in v2.

2.2.2 Features

Embroidermodder 2 has many advanced features that enable you to create awesome designs quicker, tweak existing designs to perfection, and can be fully customized to fit your workflow.

A summary of these features:

- Cross Platform
- Realistic rendering
- Various grid types and auto-adjusting rulers
- Many measurement tools
- Add text to any design
- Supports many formats
- Batch Conversion
- Scripting API

2.2.2.1 Cross Platform If you use multiple operating systems, it's important to choose software that works on all of them.

Embroidermodder 2 runs on Windows, Linux and Mac OS X. Let's not forget the [Raspberry Pi](#).

2.2.2.2 Realistic Rendering It is important to be able to visualize what a design will look like when stitched and our pseudo `3D" realistic rendering helps achieve this.

Realistic rendering sample #1:

Realistic rendering sample #2:

Realistic rendering sample #3:

Various grid types and auto-adjusting rulers

Making use of the automatically adjusting ruler in conjunction with the grid will ensure your design is properly sized and fits within your embroidery hoop area.

Use rectangular, circular or isometric grids to construct your masterpiece!

Multiple grids and rulers in action:

2.2.2.3 Realistic Rendering Taking measurements is a critical part of creating great designs. Whether you are designing mission critical embroidered space suits for NASA or some other far out design for your next meet-up, you will have precise measurement tools at your command to make it happen. You can locate individual points or find distances between any 2 points anywhere in the design!

Take quick and accurate measurements:

2.2.2.4 Add text to any design Need to make company apparel for all of your employees with individual names on them? No sweat. Just simply add text to your existing design or create one from scratch, quickly and easily. Didn't get it the right size or made a typo? No problem. Just select the text and update it with the property editor.

Add text and adjust its properties quickly:

2.2.2.5 Supports many formats Embroidery machines all accept different formats. There are so many formats available that it can sometimes be confusing whether a design will work with your machine.

Embroidermodder 2 supports a wide variety of embroidery formats as well as several vector formats, such as SVG and DXF. This allows you to worry less about which designs you can use.

2.2.2.6 Batch Conversion Need to send a client several different formats? Just use libembroidery-convert, our command line utility which supports batch file conversion.

There are a multitude of formats to choose from:

2.2.2.7 Scripting API If you've got programming skills and there is a feature that isn't currently available that you absolutely cannot live without, you have the capability to create your own custom commands for Embroidermodder 2. We provide an QtScript API which exposes various application functionality so that it is possible to extend the application without requiring a new release. If you have created a command that you think is worth including in the next release, just [contact us](#) and we will review it for functionality, bugs, and finally inclusion.

An Embroidermodder 2 command excerpt:

2.2.3 "Build and Install"

Assuming you already have the SDL2 libraries you can proceed to using the fast build, which assumes you want to build and test locally.

The fast build should be:

```
bash build.sh
```

or, on Windows:

```
.\build.bat
```

Then run using the `run.bat` or `run.sh` scripts in the build/ directory.

Otherwise, follow the instructions below.

If you plan to install the dev version to your system (we recommend you wait for the official installers and beta release first) then use the CMake build instead.

2.2.3.1 Install on Desktop We recommend that if you want to install the development version you use the CMake build. Like this:

```
git submodule init  
git submodule update  
  
mkdir build  
cd build  
cmake ..  
cmake --build .  
sudo cmake --install .
```

These lines are written into the file:

```
./build_install.sh
```

On Windows use the next section.

2.2.4 History

Embroidermodder 1 was started by Mark Pontius in 2004 while staying up all night with his son in his first couple months. When Mark returned to his day job, he lacked the time to continue the project. Mark made the decision to focus on his family and work, and in 2005, Mark gave full control of the project to Josh Varga so that Embroidermodder could continue its growth.

Embroidermodder 2 was conceived in mid 2011 when Jonathan Greig and Josh Varga discussed the possibility of making a cross-platform version. It is currently in active development and will run on GNU/Linux, Mac OS X, Microsoft Windows and Raspberry Pi.

All `Embroidermodder` `downloads` are hosted on SourceForge.

The `source code for Embroidermodder 1` has always been hosted on Sourceforge.

The `source code for Embroidermodder 2` was moved to GitHub on July 18, 2013.

The `website for Embroidermodder` was moved to GitHub on September 9, 2013.

2.3 Contact us

For general questions email: `embroidermodder at gmail.com`

To request a new feature `open an issue on the main Embroidermodder GitHub repository`. We'll move it to the correct repository.

3 Downloads

3.1 Alpha Build

This is a highly experimental build: we recommend users wait for the beta release when the basic features are functional.

Visit our `GitHub Releases page` for the current build. Unfortunately, earlier builds went down with the Sourceforge page we hosted them on.

4 Tutorials

4.1 Basic Features

4.1.1 Move a single stitch in an existing pattern

1. In the `File` menu, click `Open . . .`. When the open dialog appears find and select your file by double clicking the name of the file. Alternatively, left click the file once then click the `Open` button.
- 2.
3. In the 'File' menu

TIP: For users who prefer

4.2 Altering a Single Stitch (2022-09-19)

In this tutorial we cover the scenarios where:

1. you have a design that you want to move a few points in but make no major changes
2. you have a design that you want to remove a few stitches from
3. you have a design that has some stitches in the wrong colour

In all cases, open Embroidermodder 2 and use `File > Open` to get the open file dialog.

(TO BE COMPLETED.)

4.2.1 Convert one pattern to another format

1. In the `File` menu, click `Open . . .`.
2. The
3. In the dropdown menu within the save dialog select the

4.3 Advanced Features

5 Post History

\seciton open-collective Open Collective and New Plan (2021-12-19)

Hi, welcome to our first update after switching to Open Collective.

I hope that we can get people excited about open software and hardware for embroidery again. Clearly there was some real talent and effort from Jonathon, Mark and other contributors in making this happen. Hopefully, Josh and I can make these programs a standard fixture of garage workshops by making them easy to install, modify, build and distribute.

I've sketched out a timetable for the development of Embroidermodder, I feel I should share it with everyone:

| Date | Event |
|-------------------------------------|---|
| Dec 2021 - Jan 2022 | libembroidery 1.0 features, particularly the basic file format support and fills Bugfixing, Testing, QA for libembroidery |
| 31st of Jan 2022 | libembroidery 1.0 will be released, then updates will slow down and the Embroidermodder 2 development version will be fixed to the API of this version. |
| Feb 2022 | An overview of what has changed will be written up for the website as a news update Better documentation of libembroidery. |
| Feb-April 2022 | Finish the conversion to new GUI toolkit |
| April-May 2022 | Finish all the targets in the Design, or assign them to 2.1. |
| May-June 2022 | Stop pushing new features. Bugfixing, Testing, QA for Embroidermodder 2 |
| Summer Solstice (21st of June) 2022 | Embroidermodder 2 is officially released. Distribute NSIS installer, debian package, fedora package, mac bundle and source archives |
| July 2022 | News and Documentation work for Embroidermodder 2 |

Embroidermodder 2 is a zlib licensed software and we endeavour to keep it free and well documented. Check out our [main github page](#) to see.

Cheers,

Robin

5.1 Fast Forward (2014-02-13)

There have been some changes that happened over the past several months:

The launch of the Kickstarter campaign was delayed but is still going to happen. Unless there are other unforeseen setbacks, the revised plan is to launch our campaign on March 17.

We have started setup of a nightly build/continuous integration system using Travis-CI. This is important as it will allow quicker bug fixes and increased stability long term. Currently this is only available for 64-bit Linux builds but will eventually include OSX, Windows and Arduino.

We have also improved the stability and improved the API of our backend library: libembroidery. The API is still changing but we now can produce standalone static and shared versions of libembroidery. Another notable change is that there are now experimental Delphi and Lazarus(FreePascal) bindings in the works for libembroidery, contributed by [x2nie](#).

Last but not least, I have a demonstration of our libembroidery code embedded on an Arduino. The video below simulates the stitching by drawing it to a 2.8" TFT display (v1) made by [Adafruit](https://adafruit.com). It's really fast so I slowed it down for the video. Instead of drawing, we could change it to control motors and other hardware to create an open source embroidery machine. This is what the brain of an embroidery machine looks like and it just needs a heart and body. Enjoy the [video](https://www.youtube.com/watch?v=KqiKfn4lxBk)! [frameborder="0" allowfullscreen></iframe>](https://www.youtube.com/embed/KqiKfn4lxBk) Jonathan [Permanent link to this article](news0.html::demo) @section jan22news January 2022 Development Notes For Embroidermodder 2 (2022-01-31) Hi backers, since the last update development has been focused on mostly not user-facing aspects of the program. This is what we planned in the timetable, aiming to release the libembroidery library for the end of the month. The idea being that if we can fix an API for libembroidery early, it will focus development into a program that remains more consistent by the height of the Summer. Unfortunately, this was too ambitious. But we can discuss what has changed, and why a later date is better than focusing on hitting this milestone. @subsection broad-development-goals Broad Development Goals We want: * A small codebase. * A separation of code and data in procedural style. * A separation of basic library

functions from GUI code.

- * A build system without non-standard dependencies (i.e. not including windows.h, X11, GL).
- * A software that runs without installation
- * A software that can keep records of its state that can be loaded on reboot

To meet these goals we are making a C/FreeGLUT application:

- * Runs on more systems because it has less (and less high level) dependencies.
- * Requires less specialist knowledge from our potential new developers of Qt, C++ and the build system.
- * Compiles with less faff since the only dependencies are the graphics libraries of the host system (meeting broad development goal 4).

Other decisions made to meet these goals are detailed below.

@subsection new-settings-system The New Settings System In the attempt to convert the mostly C++ codebase to C, we developed a basic system for storing the data of Embroidermodder (including all the icons) as a single JSON file weighing in at about 6Mb. This allows the second broad development goal to be met.

@subsection reducing-reliance-on-qt5 Reducing Reliance on Qt5 The FreeGLUT variant of the code (in gui.c) will eventually be the software. The broad development goals should make that very complex change a sequence of smaller, less complex changes. For example: breaking apart highly object-oriented code that relies on high level Qt function calls into data and code and then separating out the code parts into C and C++ functions.

@subsection color-palettes Color Palettes Using a custom script we converted all the icons into xpm, then using another reduced their palettes down to 16 colors + transparency. This hasn't affected their appearance significantly, and makes making a global GUI palette feasible. With a global GUI palette we can make icon themes be a simple remap of the palette.

@subsection conclusion Conclusion Overall, the software is easier to understand. But not easy enough for it to be worth committing to extensive documentation that will likely have to change. This also makes deciding on a API difficult. Based on this, a short term aim can be writing more high level documentation like this update to clarify (even just to the team) why we made the decisions we have.

Cheers, Robin Swift
The Embroidermodder Team

@section June 2022 Backer Update (2022-06-22) Hi backers, since it's mainly me developing as a hobbyist at the moment successes are rare so I thought it would be good to share these with you.

@subsection working-fill-algorithm Working Fill Algorithms I've managed to get two fills working from the command line using the commands:

```
@icode $ embroider --fill input_image.ppm 130 output.dst $ embroider --cross-stitch input_image.ppm 130 output.dst @endcode
```

So given this version of the banner logo: It is cross-stitched like this:

There are major limitations, it requires an input that has depth 8 and is 1000x1000 pixels and it only takes in ppm format images. Your other format images could be converted to this format using:

```
@icode $ convert input.png -depth 8 -geometry 1000x1000 output.ppm @endcode
```

A command like this will be embedded into the software to allow more image types but it would require the user to install imagemagick first. Here's another example using the imagemagick wizard:

@subsection working-render-algorithm Working Render Algorithm The renders shown above were also made by the program using the commands:

```
@icode $ embroider --render input.dst output.ppm $ convert output.ppm output.png @endcode
```

@subsection timetable Timetable As for the timetable, obviously that was too ambitious given the lack of a large team and the slow growth of interest. A tentative timetable is on the README, but that's uncertain as long as the major rewrite to C/SDL2 is underway, hopefully it'll still be a 2.0 release this year. I'll try and share some more examples next month!

Cheers, Robin

@subpage kickstarter-live Our Kickstarter Crowdfunding Campaign is LIVE! (2014-03-17) \author Jonathan and Josh <a href=" <https://www.kickstarter.com/projects/redteam316/embroidermodder-2-for-windows-mac-linux-pi-and-ard>">link We have launched our crowdfunding campaign on Kickstarter! It will be running until Sunday, April 20th. The way Kickstarter operates is that we need to reach our funding goal to receive any funds. There are downloads available for Windows (32-bit), Linux (32 and 64-bit), Mac OS X (64-bit) and Raspberry Pi (Raspbian) on the Kickstarter page. There has been a ton of work done to get to this point and to ensure a timely and stable delivery, this campaign needs to succeed. We really want to shake up the embroidery world and we hope you agree. The link to our campaign is: <a href=" <https://www.kickstarter.com/projects/redteam316/embroidermodder-2-for-windows-mac-linux-pi-and-ard>">https://www.kickstarter.com/projects/redteam316/embroidermodder-2-for-windows-mac-linux-pi-and-ard.

... also check out our shamrockin' embroidery design created with Embroidermodder 2 using work-in-progress manual satin command! Lucky you! <a href=" <https://github.com/Embroidermodder/Embroidermodder/raw/master/embroidermodder2/samples/shamrockin.dst>">Download Here. "/> –Jonathan and Josh Permanent link to this article

@subpage new-website New Website! (2013-09-09) \author Jonathan The Embroidermodder website now has a fresh new look. The content has been updated to reflect the upcoming version, Embroidermodder 2. Check out our features page for a summary of what to expect with Embroidermodder 2. ↵

The background is a tiled image of an actual design that was stitched out during the pre-alpha stage. ↵ It was created by Nina Paley and Theodore Gray using Mathematica in conjunction with our software. ↵ They have graciously allowed us to use it for the project in whichever way we wish. We thought it looked so good, that it has become the new theme for Embroidermodder 2. To check out some of the more interesting embroidery projects they are working on, <a href=" <http://blog.ninapaley.com/>">look here. The old website which was for Embroidermodder 1 has been preserved and can be found <a href=" <http://embroidermodder.sourceforge.net/embroidermodder1.html>">here for anyone interested. –Jonathan Permanent link to this article @subsection crowdfunding-1 Crowdfunding Campaign Coming Soon! \author Jonathan <a href=" <http://www.kickstarter.com/projects/redteam316/369640335?token=57f7685e>">link There has been a considerable amount of development time put into Embroidermodder 2 over the past several months. To be able to keep up this momentum, there needs to be at least one full time developer working on it. We are planning on launching a Kickstarter campaign in early October if everything goes according to plan. We also plan to release an alpha version during this timeframe, so there are many good things on the horizon! Successful funding will have a major impact on how soon the final version will be released. The preview link to our campaign is <a href=" <http://www.kickstarter.com/projects/redteam316/369640335?token=57f7685e>">here, feel free to leave feedback and spread the word via Twitter, Facebook, email, or word of mouth. Keep an eye out, because it's coming! –Jonathan (2013-09-09) @ref crowdfunding-1 "Permanent link to this article" @section open-collective Open Collective and New Plan (2021-12-19) Hi, welcome to our first update after switching to Open Collective. I hope that we can get people excited about open software and hardware for embroidery again. Clearly there was some real talent and effort from Jonathon, Mark and other contributors in making this happen. Hopefully, Josh and I can make these programs a standard fixture of garage workshops by making them easy to install, modify, build and distribute. I've sketched out a timetable for the development of Embroidermodder, I feel I should share it with everyone: <table class="markdownTable"> <tr class="markdownTableHead"> <th class="markdownTableHeadNone"> Date

Event

Dec 2021 - Jan 2022

libembroidery 1.0 features, particularly the basic file format support and fills Bugfixing, Testing, QA for libembroidery
31st of Jan 2022

libembroidery 1.0 will be released, then updates will slow down and the Embroidermodder 2 development version will be fixed to the API of this version.

Feb 2022

An overview of what has changed will be written up for the website as a news update Better documentation of libembroidery.

Feb-April 2022

Finish the conversion to new GUI toolkit

April-May 2022

Finish all the targets in the Design, or assign them to 2.1.

May-June 2022

Stop pushing new features. Bugfixing, Testing, QA for Embroidermodder 2

Summer Solstice (21st of June) 2022

Embroidermodder 2 is officially released. Distribute NSIS installer, debian package, fedora package, mac bundle and source archives

July 2022

News and Documentation work for Embroidermodder 2

Embroidermodder 2 is a zlib licensed software and we endeavour to keep it free and well documented. Check out our [main github page](#) to see.

Cheers,

Robin

5.2 January 2022 Development Notes For Embroidermodder 2 (2022-01-31)

Hi backers,

since the last update development has been focused on mostly not user-facing aspects of the program. This is what we planned in the timetable, aiming to release the libembroidery library for the end of the month.

The idea being that if we can fix an API for libembroidery early, it will focus development into a program that remains more consistent by the height of the Summer.

Unfortunately, this was too ambitious. But we can discuss what has changed, and why a later date is better than focusing on hitting this milestone.

5.2.1 Broad Development Goals

We want:

- A small codebase.
- A separation of code and data in procedural style.
- A separation of basic library functions from GUI code.
- A build system without non-standard dependencies (i.e. not including windows.h, X11, GL).
- A software that runs without installation
- A software that can keep records of its state that can be loaded on reboot

To meet these goals we are making a C/FreeGLUT application:

- Runs on more systems because it has less (and less high level) dependencies.
- Requires less specialist knowledge from our potential new developers of Qt, C++ and the build system.
- Compiles with less faff since the only dependencies are the graphics libraries of the host system (meeting broad development goal 4).

Other decisions made to meet these goals are detailed below.

5.2.2 The New Settings System

In the attempt to convert the mostly C++ codebase to C, we developed a basic system for storing the data of Embroidermodder (including all the icons) as a single JSON file weighing in at about 6Mb.

This allows the second broad development goal to be met.

5.2.3 Reducing Reliance on Qt5

The FreeGLUT variant of the code (in gui.c) will eventually be the software.

The broad development goals should make that very complex change a sequence of smaller, less complex changes. For example: breaking apart highly object-oriented code that relies on high level Qt function calls into data and code and then separating out the code parts into C and C++ functions.

5.2.4 Palettes

Using a custom script we converted all the icons into xpm, then using another reduced their palettes down to 16 colors + transparency.

This hasn't affected their appearance significantly, and makes making a global GUI palette feasible. With a global GUI palette we can make icon themes be a simple remap of the palette.

5.2.5 Conclusion

Overall, the software is easier to understand. But not easy enough for it to be worth committing to extensive documentation that will likely have to change. This also makes deciding on a API difficult.

Based on this, a short term aim can be writing more high level documentation like this update to clarify (even just to the team) why we made the decisions we have.

Cheers,

Robin Swift

The Embroidermodder Team

5.3 June 2022 Backer Update (2022-06-22)

Hi backers,

since it's mainly me developing as a hobbyist at the moment successes are rare so I thought it would be good to share these with you.

5.3.1 Fill Algorithms

I've managed to get two fills working from the command line using the commands:

```
$ embroider --fill input_image.ppm 130 output.dst
$ embroider --cross-stitch input_image.ppm 130 output.dst
```

So given this version of the banner logo:

It is cross-stitched like this:

There are major limitations, it requires an input that has depth 8 and is 1000x1000 pixels and it only takes in ppm format images. Your other format images could be converted to this format using:

```
$ convert input.png -depth 8 -geometry 1000x1000 output.ppm
```

A command like this will be embedded into the software to allow more image types but it would require the user to install imagemagick first.

Here's another example using the imagemagick wizard:

5.3.2 Working Render Algorithm

The renders shown above were also made by the program using the commands:

```
$ embroider --render input.dst output.ppm
$ convert output.ppm output.png
```

5.3.3 Timetable

As for the timetable, obviously that was too ambitious given the lack of a large team and the slow growth of interest. A tentative timetable is on the README, but that's uncertain as long as the major rewrite to C/SDL2 is underway, hopefully it'll still be a 2.0 release this year.

I'll try and share some more examples next month!

Cheers,

Robin

5.4 New Website! (2013-09-09)

The Embroidermodder website now has a fresh new look. The content has been updated to reflect the upcoming version, Embroidermodder 2. Check out our [features page](#) for a summary of what to expect with Embroidermodder 2.

The background is a tiled image of an actual design that was stitched out during the pre-alpha stage. It was created by Nina Paley and Theodore Gray using Mathematica in conjunction with our software. They have graciously allowed us to use it for the project in whichever way we wish. We thought it looked so good, that it has become the new theme for Embroidermodder 2. To check out some of the more interesting embroidery projects they are working on, [look here](#).

The old website which was for Embroidermodder 1 has been preserved and can be found [here](#) for anyone interested.

–Jonathan

[Permanent link to this article](#)

5.5 Crowdfunding Campaign Coming Soon! (2013-09-09)

[link](#)

There has been a considerable amount of development time put into Embroidermodder 2 over the past several months. To be able to keep up this momentum, there needs to be at least one full time developer working on it. We are planning on launching a Kickstarter campaign in early October if everything goes according to plan. We also plan to release an alpha version during this timeframe, so there are many good things on the horizon! Successful funding will have a major impact on how soon the final version will be released.

The preview link to our campaign is [here](#), feel free to leave feedback and spread the word via Twitter, Facebook, email, or word of mouth. Keep an eye out, because it's coming!

–Jonathan

[Permanent link to this article](#)

5.6 Our Kickstarter Crowdfunding Campaign is LIVE! (2014-03-17)

[link](#)

We have launched our crowdfunding campaign on Kickstarter! It will be running until Sunday, April 20th. The way Kickstarter operates is that we need to reach our funding goal to receive any funds. There are downloads available for Windows (32-bit), Linux (32 and 64-bit), Mac OS X (64-bit) and Raspberry Pi (Raspbian) on the Kickstarter page. There has been a ton of work done to get to this point and to ensure a timely and stable delivery, this campaign needs to succeed. We really want to shake up the embroidery world and we hope you agree. The link to our campaign is: <https://www.kickstarter.com/projects/redteam316/embroidermodder-2-for-windows-mac-linux-pi-and-ard>

... also check out our shamrockin' embroidery design created with Embroidermodder 2 using work-in-progress manual satin command!

Lucky you! [Download Here.](#)

–Jonathan and Josh

[Permanent link to this article](#)

5.7 Fast Forward (2014-02-13)

There have been some changes that happened over the past several months:

The launch of the Kickstarter campaign was delayed but is still going to happen. Unless there are other unforeseen setbacks, the revised plan is to launch our campaign on March 17.

We have started setup of a nightly build/continuous integration system using Travis-CI. This is important as it will allow quicker bug fixes and increased stability long term. Currently this is only available for 64-bit Linux builds but will eventually include OSX, Windows and Arduino.

We have also improved the stability and improved the API of our backend library: libembroidery. The API is still changing but we now can produce standalone static and shared versions of libembroidery. Another notable change is that there are now experimental Delphi and Lazarus(FreePascal) bindings in the works for libembroidery, contributed by [x2nie](#).

Last but not least, I have a demonstration of our libembroidery code embedded on an Arduino. The video below simulates the stitching by drawing it to a 2.8" TFT display (v1) made by [Adafruit](https://adafruit.com). It's really fast so I slowed it down for the video. Instead of drawing, we could change it to control motors and other hardware to create an open source embroidery machine. This is what the brain of an embroidery machine looks like and it just needs a heart and body. Enjoy the [video](https://www.youtube.com/watch?v=KqiKfn4lxBk)! [Permanent link to this article](#).

6 Changelog

7 Ideas

Stuff that is now supposed to be generated by Doxygen:

Todo Bibliography style to plainnat.

Todo Serif font for printed docs.

Todo US letter paper version of printed docs.

8 Formats

8.1 Overview

8.1.1 Read/Write Support Levels

The table of read/write format support levels uses the status levels described here:

| Status Label | Description |
|--------------|--|
| rw-none | Either the format produces no output, reporting an error. Or it produces a Tajima dst file as an alternative. |
| rw-poor | A file somewhat similar to our examples is produced. We don't know how well it runs on machines in practice as we don't have any user reports or personal tests. |
| rw-basic | Simple files in this format run well on machines that use this format. |
| rw-standard | Files with non-standard features work on machines and we have good documentation on the format. |
| rw-reliable | All known features don't cause crashes. Almost all work as expected. |
| rw-complete | All known features of the format work on machines that use this format. Translations from and to this format preserve all features present in both. |

These can be split into r-basic w-none, for example, if they don't match.

So all formats can, in principle, have good read and good write support, because it's defined in relation to files that we have described the formats for.

| Status Label | Description |
|---------------|---|
| test-none | No tests have been written to test the specifics of the format. |
| test-basic | Stitch Lists and/or colors have read/write tests. |
| test-thorough | All features of that format has at least one test. |
| test-fuzz | Can test the format for uses of features that we haven't thought of by feeding in nonsense that is designed to push possibly dangerous weaknesses to reveal themselves. |
| test-complete | Both thorough and fuzz testing is covered. |

8.1.1.1 Test Support Levels So all formats can, in principle, have complete testing support, because it's defined in relation to files that we have described the formats for.

| Status Label | Description |
|--------------|--|
| doc-none | We haven't researched this beyond finding example files. |
| doc-basic | We have a rough sketch of the size and contents of the header if there is one. We know the basic stitch encoding (if there is one), but not necessarily all stitch features. |
| doc-standard | We know some good sources and/or have tested all the features that appear to exist. They mostly work the way we have described. |
| doc-good | All features that were described somewhere have been covered here or we have thoroughly tested our ideas against other softwares and hardwares and they work as expected. |
| doc-complete | There is a known official description and our description covers all the same features. |

8.1.1.2 Documentation Support Levels Not all formats can have complete documentation because it's based on what information is publically available. So the total score is reported in the table below based on what level we think is available.

8.1.1.3 Overall Support Since the overall support level is the combination of these 4 factors, but rather than summing up their values it's an issue of the minimum support of the 4.

| Status Label | Description |
|--------------|--|
| read-only | If write support is none and read support is not none. |
| write-only | If read support is none and write support is not none. |
| unstable | If both read and write support are not none but testing or documentation is none. |
| basic | If all ratings are better than none. |
| reliable | If all ratings are better than basic. |
| complete | If all ratings could not reasonably be better (for example any improvements rely on information that we may never have access to). This is the only status that can be revoked, since if the format changes or new documentation is released it is no longer complete. |
| experimental | For all other scenarios. |

8.1.2 Table of Format Support Levels

Overview of documentation support by format.

| Format | Ratings | Score |
|---|-----------------------------------|--------------|
| Toyota Embroidery Format (.100) | rw-basic doc-none test-none | unstable |
| Toyota Embroidery Format (.10o) | rw-basic doc-none test-none | unstable |
| Bernina Embroidery Format (.art) | rw-none doc-none test-none | experimental |
| Bitmap Cache Embroidery Format (.bmc) | r-basic w-none doc-none test-none | unstable |
| Bits and Volts Embroidery Format (.bro) | rw-none doc-none test-none | experimental |
| Melco Embroidery Format (.cnd) | rw-none doc-none test-none | experimental |
| Embroidery Thread Color Format (.col) | rw-basic doc-none test-none | experimental |
| Singer Embroidery Format (.csd) | rw-none doc-none test-none | experimental |
| Comma Separated Values (.csv) | rw-none doc-none test-none | experimental |
| Barudan Embroidery Format (.dat) | rw-none doc-none test-none | experimental |
| Melco Embroidery Format (.dem) | rw-none doc-none test-none | experimental |
| Barudan Embroidery Format (.dsb) | rw-none doc-none test-none | experimental |
| Tajima Embroidery Format (.dst) | rw-none doc-none test-none | experimental |
| ZSK USA Embroidery Format (.dsz) | rw-none doc-none test-none | experimental |
| Drawing Exchange Format (.dxf) | rw-none doc-none test-none | experimental |
| Embird Embroidery Format (.edr) | rw-none doc-none test-none | experimental |
| Elna Embroidery Format (.emd) | rw-none doc-none test-none | experimental |
| Melco Embroidery Format (.exp) | rw-none doc-none test-none | experimental |
| Eltac Embroidery Format (.exy) | rw-none doc-none test-none | experimental |
| Sierra Expanded Embroidery Format (.eys) | rw-none doc-none test-none | experimental |
| Fortron Embroidery Format (.fxy) | rw-none doc-none test-none | experimental |
| Smoothie G-Code Embroidery Format (.gc) | rw-none doc-none test-none | experimental |
| Great Notions Embroidery Format (.gnc) | rw-none doc-none test-none | experimental |
| Gold Thread Embroidery Format (.gt) | rw-none doc-none test-none | experimental |
| Husqvarna Viking Embroidery Format (.hus) | rw-none doc-none test-none | experimental |

| Format | Ratings | Score |
|---|----------------------------|--------------|
| Inbro Embroidery Format (.inb) | rw-none doc-none test-none | experimental |
| Embroidery Color Format (.inf) | rw-none doc-none test-none | experimental |
| Janome Embroidery Format (.jef) | rw-none doc-none test-none | experimental |
| Pfaff Embroidery Format (.ksm) | rw-none doc-none test-none | experimental |
| Pfaff Embroidery Format (.max) | rw-none doc-none test-none | experimental |
| Mitsubishi Embroidery Format (.mit) | rw-none doc-none test-none | experimental |
| Ameco Embroidery Format (.new) | rw-none doc-none test-none | experimental |
| Melco Embroidery Format (.ofm) | rw-none doc-none test-none | experimental |
| Pfaff Embroidery Format (.pcd) | rw-none doc-none test-none | experimental |
| Pfaff Embroidery Format (.pcm) | rw-none doc-none test-none | experimental |
| Pfaff Embroidery Format (.pcq) | rw-none doc-none test-none | experimental |
| Pfaff Embroidery Format (.pcs) | rw-none doc-none test-none | experimental |
| Brother Embroidery Format (.pec) | rw-none doc-none test-none | experimental |
| Brother Embroidery Format (.pel) | rw-none doc-none test-none | experimental |
| Brother Embroidery Format (.pem) | rw-none doc-none test-none | experimental |
| Brother Embroidery Format (.pes) | rw-none doc-none test-none | experimental |
| Brother Embroidery Format (.phb) | rw-none doc-none test-none | experimental |
| Brother Embroidery Format (.phc) | rw-none doc-none test-none | experimental |
| AutoCAD Embroidery Format (.plt) | rw-none doc-none test-none | experimental |
| RGB Embroidery Format (.rgb) | rw-none doc-none test-none | experimental |
| Janome Embroidery Format (.sew) | rw-none doc-none test-none | experimental |
| Husqvarna Viking Embroidery Format (.shv) | rw-none doc-none test-none | experimental |
| Sunstar Embroidery Format (.sst) | rw-none doc-none test-none | experimental |
| Data Stitch Embroidery Format (.stx) | rw-none doc-none test-none | experimental |
| Scalable Vector Graphics (.svg) | rw-none doc-none test-none | experimental |
| Pfaff Embroidery Format (.t01) | rw-none doc-none test-none | experimental |
| Pfaff Embroidery Format (.t09) | rw-none doc-none test-none | experimental |
| Happy Embroidery Format (.tap) | rw-none doc-none test-none | experimental |
| ThredWorks Embroidery Format (.thr) | rw-none doc-none test-none | experimental |
| Text File (.txt) | rw-none doc-none test-none | experimental |
| Barudan Embroidery Format (.u00) | rw-none doc-none test-none | experimental |
| Barudan Embroidery Format (.u01) | rw-none doc-none test-none | experimental |
| Pfaff Embroidery Format (.vip) | rw-none doc-none test-none | experimental |
| Pfaff Embroidery Format (.vp3) | rw-none doc-none test-none | experimental |
| Singer Embroidery Format (.xxx) | rw-none doc-none test-none | experimental |
| ZSK USA Embroidery Format (.zsk) | rw-none doc-none test-none | experimental |

8.1.3 Format Support

```
| FORMAT | READ | WRITE | NOTES |-----|-----|-----| | 10o | YES || read (need to fix external color
loading) (maybe find out what ctrl | code flags of 0x10, 0x08, 0x04, and 0x02 mean) || 100 || | none (4 byte codes)
61 00 10 09 (type, type2, x, y ?) x | y (signed char) || 100 || | none (4 byte codes) 61 00 10 09 (type, type2, x, y ?)
x & y (signed char) || art || | none || bro | YES || read (complete)(maybe figure out detail of header) || cnd ||
none || col || | (color file no design) read(final) write(final) || csd | YES || read (complete) || dat || | read () |
dem || | none (looks like just encrypted cnd) || dsb | YES || read (unknown how well) (stitch data looks same as
10o) || dst | YES || read (complete) / write(unknown) || dsz | YES || read (unknown) || dxf || | read (Port to
C. needs refactored) || edr || | read (C version is broken) / write (complete) || emd || | read (unknown) || exp |
YES || read (unknown) / write(unknown) || exy | YES || read (need to fix external color loading) || fxy | YES || |
read (need to fix external color loading) || gnc || | none || gt || | read (need to fix external color loading) || hus
```

```
| YES || read (unknown) / write (C version is broken) || inb | YES || read (buggy?) || jef | YES || write (need
to fix the offsets when it is moving to another spot) || ksm | YES || read (unknown) / write (unknown) || pcd ||
||| pcm |||| pcq ||| read (Port to C) || pcs | BUGGY || read (buggy / colors are not correct / after reading,
writing any other format is messed up) || pec ||| read / write (without embedded images, sometimes overlooks
some stitches leaving a gap) || pel ||| none || pem ||| none || pes | YES |||| phb |||| phc ||||| rgb ||
||| sew | YES ||| shv ||| read (C version is broken) || sst ||| none || svg | YES ||| tap | YES || read
(unknown) || u01 ||||| vip | YES ||| vp3 | YES ||| xxx | YES ||| zsk ||| read (complete) |
```

Todo Josh, Review this section and move any info still valid or needing work into TODO comments in the actual libembroidery code. Many items in this list are out of date and do not reflect the current status of libembroidery. When finished, delete this file.

- Test that all formats read data in correct scale (format details should match other programs)
- Add which formats to work with to preferences.
- Check for memory leaks
- Update all formats without color to check for edr or rgb files
- Fix issues with DST (VERY important that DST work well)

Todo Support for Singer FHE, CHE (Compucon) formats?

9 Geometry and Algorithms

9.1 To Do

Todo (Arduino) Fix emb-outline files

Todo (Arduino) Fix thread-color files

Todo (Arduino) Logging of Last Stitch Location to External USB Storage(commonly available and easily replaced)
...wait until TRE is available to avoid rework

Todo (Arduino) inotool.org - seems like the logical solution for Nightly/CI builds

Todo (Arduino) Smoothieboard experiments

Todo (testing) looping test that reads 10 times while running valgrind. See \texttt{embPattern_loadExternalColorFile()}\ Arduino leak note for more info.

9.1.1 Development

If you wish to develop with us you can chat via the contact email on the [website]\url{ <https://libembroidery.org>} or in the issues tab on the [github page]\url{ <https://github.com/Embroidermodder/Embroidermodder/issues>}. People have been polite and friendly in these conversations and I (Robin) have really enjoyed them. If we do have any arguments please note we have a [Code of Conduct] [CODE_OF_CONDUCT.md](#) so there is a consistent policy to enforce when dealing with these arguments.

The first thing you should try is building from source using the [build advice](build) above. Then read some of the [manual] \url{ https://libembroidery.org/embroidermodder_2.0_manual.pdf} to get the general layout of the source code and what we are currently planning.

9.1.2 Testing

To find unfixed errors run the tests by launching from the command line with:

```
$ embroidermodder --test
```

then dig through the output. It's currently not worth reporting the errors, since there are so many but if you can fix anything reported here you can submit a PR.

9.1.3 Contributing

9.1.3.1 Funding The easiest way to help is to fund development (see the Donate button above), since we can't afford to spend a lot of time developing and only have limited kit to test out libembroidery on.

9.1.3.2 Programming and Engineering Should you want to get into the code itself:

- Low level C developers are needed for the base library libembroidery.
- Low level assembly programmers are needed for translating some of libembroidery to Embroider↔Bot.
- Hardware Engineers to help design our own kitbashed embroidery machine EmbroiderBot, one of the original project aims in 2013.
- Scheme developers and C/SDL developers to help build the GUI.
- Scheme developers to help add designs for generating of custom stitch-filled emblems like the heart or dolphin. Note that this happens in Embroidermodder not libembroidery (which assumes that you already have a function available).

9.1.3.3 Writing We also need people familiar with the software and the general machine embroidery ecosystem to contribute to the [documentation](#).

We need researchers to find references for the documentation: colour tables, machine specifications etc. The history is murky and often very poorly maintained so if you know anything from working in the industry that you can share: it'd be appreciated!

9.1.4 Embroidermodder Project Coding Standards

A basic set of guidelines to use when submitting code.

Code structure is more important than style, so first we advise you read 'Design' and experimenting before getting into the specifics of code style.

9.1.4.1 Where Code Goes Anything that deals with the specifics of embroidery file formats, threads, rendering to images, embroidery machinery or command line interfaces should go in libembroidery not here.

9.1.4.2 Non-compiled Files Go

Todo Like most user interfaces Embroidermodder is mostly data, so here we will have a list describing where each CSV goes.

9.1.4.3 in which we break style on purpose Most style guides advise you to keep functions short. We make a few pointed exceptions to this where the overall health and functionality of the source code should benefit.

The `actuator` function will always be a mess and it should be: we're keeping the total source lines of code down by encoding all user action into a discrete sequence of strings that are all below `\texttt{_STRING_LENGTH}` in length. See the section on the actuator (TODO) describing why any other solution we could think here would mean more code without a payoff in speed of execution or clarity.

9.1.5 Version Control

Being an open source project, developers can grab the latest code at any time and attempt to build it themselves. We try our best to ensure that it will build smoothly at any time, although occasionally we do break the build. In these instances, please provide a patch, pull request which fixes the issue or open an issue and notify us of the problem, as we may not be aware of it and we can build fine.

Try to group commits based on what they are related to: features/bugs/comments/graphics/commands/etc...

9.1.6 Donations

Creating software that interfaces with hardware is costly. A summary of some of the costs involved:

- Developer time for 2 core developers
- Computer equipment and parts
- Embroidery machinery
- Various electronics for kitbashing Embroiderbot
- Consumable materials (thread, fabric, stabilizer, etc...)

If you have found our software useful, please consider funding further development by donating to the project on Open Collective (`\url{ https://opencollective.com/embroidermodder }`).

9.1.7 Embroidermodder Project Coding Standards

Rather than maintain our own standard for style, please defer to the Python's PEP 7 [3] for C style and emulating that in C++.

A basic set of guidelines to use when submitting code. Defer to the PEP7 standard with the following additions:

- All files and directories shall be lowercase and contain no spaces.
- Structs and class names should use LeadingCapitals.
- Enums and constants should be BLOCK_CAPITALS.
- Class members and functions without a parent class should be `snake_case`. With the exception of when one of the words is a "class" name from libembroidery in which case it has the middle capitals like this: `embArray_add`.
- Don't use exceptions.
- Don't use ternary operator `(?:)` in place of if/else.
- Don't repeat a variable name that already occurs in an outer scope.

9.1.7.1 Version Control Being an open source project, developers can grab the latest code at any time and attempt to build it themselves. We try our best to ensure that it will build smoothly at any time, although occasionally we do break the build. In these instances, please provide a patch, pull request which fixes the issue or open an issue and notify us of the problem, as we may not be aware of it and we can build fine.

Try to group commits based on what they are related to: features/bugs/comments/graphics/commands/etc...

9.1.7.2 Comments When writing code, sometimes there are items that we know can be improved, incomplete or need special clarification. In these cases, use the types of comments shown below. They are pretty standard and are highlighted by many editors to make reviewing code easier. We also use shell scripts to parse the code to find all of these occurrences so someone wanting to go on a bug hunt will be able to easily see which areas of the code need more love.

libembroidery and Embroidermodder are written in C and adheres to C89 standards. This means that any C99 or C++ comments will show up as errors when compiling with gcc. In any C code, you must use:

```
/* Use C Style Comments within code blocks.  
 *  
 * Use Doxygen style code blocks to place todo, bug, hack, warning,  
 * and note items like this:  
 *  
 * \todo EXAMPLE: This code clearly needs more work or further review.  
 *  
 * \bug This code is definitely wrong. It needs fixed.  
 *  
 * \hack This code shouldn't be written this way or I don't  
 * feel right about it. There may a better solution  
 *  
 * \warning Think twice (or more times) before changing this code.  
 * I put this here for a good reason.  
 *  
 * \note This comment is much more important than lesser comments.  
 */
```

9.1.8 Ideas

9.1.8.1 Why this document I've been trying to make this document indirectly through the Github issues page and the website we're building but I think a straightforward, plain-text file needs to be the ultimate backup for this. Then I can have a printout while I'm working on the project.

9.1.8.2 Qt and dependencies I'm switching to SDL2 (which is a whole other conversation) which means we can ship it with the source code package meaning only a basic build environment is necessary to build it.

9.1.8.3 Documentation Can we treat the website being a duplicate of the docs a non-starter? I'd be happier with tex/pdf only and (I know this is counter-intuitive) one per project.

9.1.8.4 Social Platform So... all the issues and project boards etc. being on Github is all well and good assuming that we have our own copies. But we don't if Github goes down or some other major player takes over the space and we have to move (again, since this started on SourceForge).

This file is a backup for that which is why I'm repeating myself between them.

9.1.8.5 Identify the meaning of these TODO items

- Saving CSV/SVG (rt) + CSV read/write UNKNOWN interpreted as COLOR bug #179
- Lego Mindstorms NXT/EV3 ports and/or commands

9.1.8.6 Progress Chart The chart of successful from-to conversions (previously a separate issue) is something that should appear in the README.

9.1.8.7 Standard The criteria for a good Pull Request from an outside developer has these properties, from most to least important:

- No regressions on testing.
- Add a feature, bug fix or documentation that is already agreed on through GitHub issues or some other way with a core developer.
- No GUI specific code should be in libembroidery, that's for Embroidermodder.
- Pedantic/ansi C unless there's a good reason to use another language.
- Meet the style above (i.e. [PEP 7, Code Lay-out](#)). We'll just fix the style if the code's good and it's not a lot of work.
- `embroider` should be in POSIX style as a command line program.
- No dependancies that aren't "standard", i.e. use only the C Standard Library.

9.1.8.8 Image Fitting A currently unsolved problem in development that warrants further research is the scenario where a user wants to feed `embroider` an image that can then be .

9.1.8.9 To Place A *right-handed coordinate system* is one where up is positive and right is positive. Left-handed is up is positive, left is positive. Screens often use down is positive, right is positive, including the OpenGL standard so when switching between graphics formats and stitch formats we need to use a vertical flip (`embPattern__flip`).

`0x20` is the space symbol, so when padding either 0 or space is preferred and in the case of space use the literal '`'`'.

9.1.8.10 To Do We currently need help with:

- Thorough descriptions of each embroidery format.
- Finding resources for each of the branded thread libraries (along with a full citation for documentation).
- Finding resources for each geometric algorithm used (along with a full citation for documentation).
- Completing the full `--full-test-suite` with no segfaults and at least a clear error message (for example not implemented yet).
- Identifying best guesses for filling in missing information when going from, say `.csv` to a late `.pes` version. What should the default be when the data doesn't clarify?
- Improving the written documentation.

- Funding, see the Sponsor button above. We can treat this as work and put far more hours in with broad support in small donations from people who want specific features.

Beyond this the development targets are categories sorted into:

- Basic Features
- Code quality and user friendliness
- embroider CLI
- Documentation
- GUI
- electronics development

9.1.8.11 Basic Features

- Incorporate `#if` defined parts of `libembroidery.c`.
- Interpret how to write formats that have a read mode from the source code and vice versa.
- Document the specifics of the file formats here for embroidery machine specific formats. Find websites and other sources that break down the binary formats we currently don't understand.
- Find more and better documentation of the structure of the headers for the formats we do understand.

9.1.8.12 Code quality and user friendliness

- Document all structs, macros and functions (will contribute directly on the web version).
- Incorporate experimental code, improve support for language bindings.
- Make `stitch x, y` into an `EmbVector`.

9.1.8.13 Documentation

- Create csv data files for thread tables.
- Convert tex to markdown, make tex an output of `build.bash`.
- Run `sloccount` on `extern/` and `.` (and `)`) so we know the current scale of the project, aim to get this number low. Report the total as part of the documentation.
- Try to get as much of the source code that we maintain into C as possible so new developers don't need to learn multiple languages to have an effect. This bars the embedded parts of the code.

9.1.8.14 GUI

- Make EmbroideryMobile (Android) also backend to `libembroidery` with a Java wrapper.
- Make EmbroideryMobile (iOS) also backend to `libembroidery` with a Swift wrapper.
- Share some of the MobileViewer and iMobileViewer layout with the main EM2. Perhaps combine those 3 into the Embroidermodder repository so there are 4 repositories total.
- Convert layout data to JSON format and use cJSON for parsing.

9.1.9 Electronics development

- Currently experimenting with Fritzing8, upload netlists to embroiderbot when they can run simulations using the asm in libembroidery.
- Create a common assembly for data that is the same across chipsets libembroidery_data_internal.s.
- Make the defines part of `embroidery.h` all systems and the function list c code only. That way we can share some development between assembly and C versions.

9.1.10 Development

9.1.10.1 Contributing If you're interested in getting involved, here's some guidance for new developers. Currently The Embroidermodder Team is all hobbyists with an interest in making embroidery machines more open and user friendly. If you'd like to support us in some other way you can donate to our Open Collective page (click the Donate button) so we can spend more time working on the project.

All code written for libembroidery should be ANSI C89 compliant if it is C. Using other languages should only be used where necessary to support bindings.

9.1.10.2 Debug If you wish to help with development, run this debug script and send us the error log.

```
#!/bin/bash
rm -fr libembroidery-debug
git clone http://github.com/embroidermodder/libembroidery libembroidery-debug
cd libembroidery-debug
cmake -DCMAKE_BUILD_TYPE=DEBUG .
cmake --build . --config=DEBUG
valgrind ./embroider --full-test-suite
```

While we will attempt to maintain good results from this script as part of normal development it should be the first point of failure on any system we haven't tested or format we understand less.

9.1.10.3 Binary download We need a current `embroider` command line program download, so people can update without building.

9.2 Embroiderbot and Libembroidery on Embedded Systems

The libembroidery library is designed to support embedded environments, so it can be used in CNC applications.

9.2.1 Compatible Boards

We recommend using an Arduino greater specs. That being said, we have had success using an Arduino Uno R3 but this will likely require further optimization and other improvements to ensure continued compatibility with the Uno. See below for more information.

9.2.2 Arduino Considerations

There are two main concerns here: Flash Storage and SRAM.

libembroidery continually outgrows the 32KB of Flash storage on the Arduino Uno and every time this occurs, a decision has to be made as to what capabilities should be included or omitted. While reading files is the main focus on arduino, writing files may also play a bigger role in the future. Long term, it would be most practical to handle the inclusion or omission of any feature via a single configuration header file that the user can modify to suit their needs.

SRAM is in extremely limited supply and it will deplete quickly so any dynamic allocation should occur early during the setup phase of the sketch and sparingly or not at all later in the sketch. To help minimize SRAM consumption on Arduino and ensure libembroidery can be used in any way the sketch creator desires, it is required that any sketch using libembroidery must implement event handlers. See the ino-event source and header files for more information.

There is also an excellent article by Bill Earl on the Adafruit Learning System which covers these topics in more depth: <http://learn.adafruit.com/memories-of-an-arduino?view=all>.

9.2.3 Space

Since a stitch takes 3 bytes of storage and many patterns use more than 10k stitches, we can't assume that the pattern will fit in memory. Therefore we will need to buffer the current pattern on and off storage in small chunks. By the same reasoning, we can't load all of one struct before looping so we will need functions similar to binaryRead↔Int16 for each struct.

This means the EmbArray approach won't work since we need to load each element and dynamic memory management is unnecessary because the arrays lie in storage.

Todo Replace EmbArray functions with embPattern load functions.

9.2.4 Tables

All thread tables and large text blocks are too big to compile directly into the source code. Instead we can package the library with a data packet that is compiled from an assembly program in raw format so the specific padding can be controlled.

In the user section above we will make it clear that this file needs to be loaded on the pattern USB/SD card or the program won't function.

Todo Start file with a list of offsets to data with a corresponding table to load into with macro constants for each label needed.

9.2.5 Current Pattern Memory Management

It will be simpler to make one file per EmbArray so we keep an EmbFile* and a length, so no malloc call is necessary. So there needs to be a consistent tmpfile naming scheme.

Todo For each pattern generate a random string of hexadecimal and append it to the filenames like stitch↔List_A16F.dat. Need to check for a file which indicates that this string has been used already.

9.2.6 Special Notes

Due to historical reasons and to remain compatible with the Arduino 1.0 IDE, this folder must be called ``utility''. Refer to the arduino build process for more info: <https://arduino.github.io/arduino-cli/0.19/sketch-build-process/>.

libembroidery relies on the Arduino SD library for reading files. See the ino-file source and header files for more information.

9.2.7 The Assembly Split

One problem to the problem of supporting both systems with abundant memory (such as a 2010s or later desktop) and with scarce memory (such as embedded systems) is that they don't share the same assembly language. To deal with this: there will be two equivalent software which are hand engineered to be similar but one will be in C and the other in the assembly dialects we support.

All assembly will be intended for embedded systems only, since a slightly smaller set of features will be supported. However, we will write a x86 version since that can be tested.

That way the work that has been done to simplify the C code can be applied to the assembly versions.

9.3 The Embroider Command Line Program

Todo Move back to libembroidery now we have the combined docs build.

9.3.1 Embroider pipeline

Adjectives apply to every following noun so

```
embroider --satin 0.3,0.6 --thickness 2 --circle 10,20,5 \
--border 3 --disc 30,40,10 --arc 30,50,10,60 output.pes
```

Creates:

- a circle with properties: thickness 2, satin 0.3,0.6
- a disc with properties:
- an arc with properties:

in that order then writes them to the output file `output.pes`.

9.3.2 `embroider` CLI

- Make `-circle` flag to add a circle to the current pattern.
- Make `-rect` flag to add a rectangle to the current pattern.
- Make `-fill` flag to set the current satin fill algorithm for the current geometry. (for example `-fill crosses -circle 11,13,10` fills a circle with center 11mm, 13mm with radius 10mm with crosses).
- Make `-ellipse` flag to add to ellipse to the current pattern.
- Make `-bezier` flag to add a bezier curve to the current pattern.

10 GNU Free Documentation License

Version 1.3, 3 November 2008

Copyright (C) 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc. <https://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

10.0.0.0.1 0. PREAMBLE The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

10.0.0.0.2 1. APPLICABILITY AND DEFINITIONS This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has

been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, La \leftarrow TeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

10.0.0.3 2. VERBATIM COPYING You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

10.0.0.4 3. COPYING IN QUANTITY If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

10.0.0.0.5 4. MODIFICATIONS You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

10.0.0.6 5. COMBINING DOCUMENTS You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

10.0.0.7 6. COLLECTIONS OF DOCUMENTS You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

10.0.0.8 7. AGGREGATION WITH INDEPENDENT WORKS A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

10.0.0.9 8. TRANSLATION Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

10.0.0.0.10 9. TERMINATION You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10.0.0.0.11 10. FUTURE REVISIONS OF THIS LICENSE The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <https://www.gnu.org/licenses/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

10.0.0.0.12 11. RELICENSING "Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

10.0.0.1 ADDENDUM: How to use this License for your documents To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (C) YEAR YOUR NAME.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.3
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.
A copy of the license is included in the section entitled "GNU
Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with ... Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the
Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

11 Contributor Covenant Code of Conduct

11.1 Our Pledge

We as members, contributors, and leaders pledge to make participation in our community a harassment-free experience for everyone, regardless of age, body size, visible or invisible disability, ethnicity, sex characteristics, gender identity and expression, level of experience, education, socio-economic status, nationality, personal appearance, race, religion, or sexual identity and orientation.

We pledge to act and interact in ways that contribute to an open, welcoming, diverse, inclusive, and healthy community.

11.2 Our Standards

Examples of behavior that contributes to a positive environment for our community include:

- Demonstrating empathy and kindness toward other people
- Being respectful of differing opinions, viewpoints, and experiences
- Giving and gracefully accepting constructive feedback
- Accepting responsibility and apologizing to those affected by our mistakes, and learning from the experience
- Focusing on what is best not just for us as individuals, but for the overall community

Examples of unacceptable behavior include:

- The use of sexualized language or imagery, and sexual attention or advances of any kind
- Trolling, insulting or derogatory comments, and personal or political attacks
- Public or private harassment
- Publishing others' private information, such as a physical or email address, without their explicit permission
- Other conduct which could reasonably be considered inappropriate in a professional setting

11.3 Enforcement Responsibilities

Community leaders are responsible for clarifying and enforcing our standards of acceptable behavior and will take appropriate and fair corrective action in response to any behavior that they deem inappropriate, threatening, offensive, or harmful.

Community leaders have the right and responsibility to remove, edit, or reject comments, commits, code, wiki edits, issues, and other contributions that are not aligned to this Code of Conduct, and will communicate reasons for moderation decisions when appropriate.

11.4 Scope

This Code of Conduct applies within all community spaces, and also applies when an individual is officially representing the community in public spaces. Examples of representing our community include using an official e-mail address, posting via an official social media account, or acting as an appointed representative at an online or offline event.

11.5 Enforcement

Instances of abusive, harassing, or otherwise unacceptable behavior may be reported to the community leaders responsible for enforcement at embroidermodder@gmail.com. All complaints will be reviewed and investigated promptly and fairly.

All community leaders are obligated to respect the privacy and security of the reporter of any incident.

11.6 Enforcement Guidelines

Community leaders will follow these Community Impact Guidelines in determining the consequences for any action they deem in violation of this Code of Conduct:

11.6.1 1. Correction

Community Impact: Use of inappropriate language or other behavior deemed unprofessional or unwelcome in the community.

Consequence: A private, written warning from community leaders, providing clarity around the nature of the violation and an explanation of why the behavior was inappropriate. A public apology may be requested.

11.6.2 2. Warning

Community Impact: A violation through a single incident or series of actions.

Consequence: A warning with consequences for continued behavior. No interaction with the people involved, including unsolicited interaction with those enforcing the Code of Conduct, for a specified period of time. This includes avoiding interactions in community spaces as well as external channels like social media. Violating these terms may lead to a temporary or permanent ban.

11.6.3 3. Temporary Ban

Community Impact: A serious violation of community standards, including sustained inappropriate behavior.

Consequence: A temporary ban from any sort of interaction or public communication with the community for a specified period of time. No public or private interaction with the people involved, including unsolicited interaction with those enforcing the Code of Conduct, is allowed during this period. Violating these terms may lead to a permanent ban.

11.6.4 4. Permanent Ban

Community Impact: Demonstrating a pattern of violation of community standards, including sustained inappropriate behavior, harassment of an individual, or aggression toward or disparagement of classes of individuals.

Consequence: A permanent ban from any sort of public interaction within the community.

11.7 Attribution

This Code of Conduct is adapted from the [Contributor Covenant](https://www.contributor-covenant.org/version/2/0/code_of_conduct.html), version 2.0, available at https://www.contributor-covenant.org/version/2/0/code_of_conduct.html.

Community Impact Guidelines were inspired by Mozilla's code of conduct enforcement ladder.

For answers to common questions about this code of conduct, see the FAQ at <https://www.contributor-covenant.org/faq>. Translations are available at <https://www.contributor-covenant.org/translations>.

12 Privacy Policy for Embroidery Viewer

Last updated December 15, 2021

Embroidermodder ("we" or "us" or "our") respects the privacy of our users ("user" or "you"). This Privacy Policy explains how we collect, use, disclose, and safeguard your information when you visit our mobile application (the "Application"). Please read this Privacy Policy carefully. IF YOU DO NOT AGREE WITH THE TERMS OF THIS PRIVACY POLICY, PLEASE DO NOT ACCESS THE APPLICATION.

We reserve the right to make changes to this Privacy Policy at any time and for any reason. We will alert you about any changes by updating the "Last updated" date of this Privacy Policy. You are encouraged to periodically review this Privacy Policy to stay informed of updates. You will be deemed to have been made aware of, will be subject to, and will be deemed to have accepted the changes in any revised Privacy Policy by your continued use of the Application after the date such revised Privacy Policy is posted.

This Privacy Policy does not apply to the third-party online/mobile store from which you install the Application or make payments. We are not responsible for any of the data collected by any such third party.

We do not knowingly collect information from anyone other than what is already provided by the app store. If you become aware of any data we have collected, please contact us using the contact information provided below.

12.0.1 CONTACT US

If you have questions or comments about this Privacy Policy, please contact us at:

Embroidermodder@gmail.com

13 Todo List

Member `ArcObject::gripEdit (const QPointF &before, const QPointF &after)`

gripEdit() for ArcObject

Member `ArcObject::updateRubber (QPainter *painter=0)`

Arc Rubber Modes

updateRubber() gripping for ArcObject

Member `bcf_directory`

possibly add a directory tree in the future.

Member `bcf_file_header`

CLSID should be a separate type.

Member `binaryWriteInt (FILE *f, int data)`

replace with emblnt_read

Member `binaryWriteIntBE (FILE *f, int data)`

replace with emblnt_read

Member `binaryWriteShort (FILE *f, short data)`

replace with emblnt_read

Member `binaryWriteUInt (FILE *f, unsigned int data)`

replace with emblnt_read

Member `binaryWriteUIntBE (FILE *f, unsigned int data)`

replace with emblnt_read

Member `binaryWriteUShort (FILE *f, unsigned short data)`

replace with emblnt_read

Member `binaryWriteUShortBE (FILE *f, unsigned short data)`

replace with emblnt_read

Member `copy_trim (char const *s)`

decription

Member `decode_t01_record (unsigned char b[3], int *x, int *y, int *flags)`

remove the unused return argument.

Member `EllipseObject::gripEdit (const QPointF &before, const QPointF &after)`

gripEdit() for EllipseObject

Member `embArc_print (EmbArc arc)`

move to arc.c

Member `embGeometry_vulcanize (EmbGeometry *obj)`

Review. This could be controlled by a simple flag.

Member `embPattern_correctForMaxStitchLength (EmbPattern *p, EmbReal maxStitchLength, EmbReal maxJumpLength)`

The params determine the max XY movement rather than the length. They need renamed or clarified further.

Member `embPattern_stitchEllipse` (`EmbPattern *p, EmbEllipse ellipse, int thread_index, int style`)

finish stitchEllipse

Member `embPattern_stitchPath` (`EmbPattern *p, EmbPath path, int thread_index, int style`)

finish stitch path

Member `embPattern_stitchPolygon` (`EmbPattern *p, EmbPolygon polygon, int thread_index, int style`)

finish stitch polygon

Member `embPattern_stitchPolyline` (`EmbPattern *p, EmbPolyline polyline, int thread_index, int style`)

finish stitch polyline

Member `embVector_multiply` (`EmbVector vector, EmbReal magnitude, EmbVector *result`)

make result return argument.

Member `embVector_normalize` (`EmbVector vector, EmbVector *result`)

make result return argument.

File `format_art.c`

Find a source.

File `format_bmc.c`

Find a source.

File `format_cnd.c`

Find a source.

Page **Formats**

Support for Singer FHE, CHE (Compucon) formats?

Josh, Review this section and move any info still valid or needing work into TODO comments in the actual libembroidery code. Many items in this list are out of date and do not reflect the current status of libembroidery. When finished, delete this file.

Member `formatTable [numberOfFormats]`

This list needs reviewed in case some stitch formats also can contain object data (EMBFORMAT_↔ STCHANDOBJ). *

Member `fread_int32_be` (`FILE *f`)

replace with emblInt_read

Member `fread_uint16` (`FILE *f`)

replace with emblInt_read

Member `generate_dragon_curve` (`char *state, int iterations`)

find citation for paper folding method

Page **Geometry and Algorithms**

(Arduino) Fix emb-outline files

(Arduino) Fix thread-color files

(Arduino) inotool.org - seems like the logical solution for Nightly/CI builds

For each pattern generate a random string of hexadecimal and append it to the filenames like `stitchList\↔_A16F.dat`. Need to check for a file which indicates that this string has been used already.

Start file with a list of offsets to data with a corresponding table to load into with macro constants for each label needed.

Replace EmbArray functions with embPattern load functions.

Move back to libembroidery now we have the combined docs build.

Like most user interfaces Embroidermodder is mostly data, so here we will have a list describing where each CSV goes.

(testing) looping test that reads 10 times while running valgrind. See \texttt{\{embPattern_loadExternalColorFile()\}}

Arduino leak note for more info.

(Arduino) Smoothieboard experiments

(Arduino) Logging of Last Stitch Location to External USB Storage(commonly available and easily replaced)
...wait until TRE is available to avoid rework

Page Ideas

US letter paper version of printed docs.

Serif font for printed docs.

Bibliography style to plainnat.

Member Index

document this.

Member **MainWindow::createAllActions ()**

Set What's This Context Help to statusTip for now so there is some infos there. Make custom whats this context help popup with more descriptive help than just the status bar/tip one liner(short but not real long) with a hyperlink in the custom popup at the bottom to open full help file description. Ex: like wxPython AGW's SuperToolTip.
ACTION->setWhatsThis(statusTip);

Finish All Commands ... <.<

Member **MainWindow::validFileFormat (const QString &fileName)**

check the file exists on the system, rename to validFile?

Member **MdiWindow::loadFile (const QString &fileName)**

reincorporate embPattern_moveStitchListToPolylines(p); //TODO: Test more

Member **MdiWindow::saveBMC ()**

Should BMC be limited to ~32KB or is this a mix up with Bitmap Cache?

Is there/should there be other embedded data in the bitmap besides the image itself?

Save a Brother PEL image (An 8bpp, 130x113 pixel monochromatic? bitmap image) Why 8bpp when only 1bpp is needed?

Member **PropertyEditor::createComboBoxSelected ()**

document this

Member **PropertyEditor::createGroupBoxGeneral ()**

use proper icons for toolButtons

Member **PropertyEditor::createGroupBoxGeometryArc ()**

use proper icons for toolButtons

Member **PropertyEditor::createGroupBoxGeometryBlock ()**

use proper icons for toolButtons

mapSignal for blocks

Member **PropertyEditor::createGroupBoxGeometryCircle ()**

use proper icons for toolButtons

Member **PropertyEditor::createGroupBoxGeometryEllipse ()**

use proper icons for toolButtons

Member **PropertyEditor::createGroupBoxGeometryImage ()**

use proper icons for toolButtons

mapSignal for images

Member **PropertyEditor::createGroupBoxGeometryInfiniteLine ()**

use proper icons for toolButtons

mapSignal for infinite lines

Member **PropertyEditor::createGroupBoxMiscArc ()**

use proper icons for toolButtons

Member [PropertyEditor::createGroupBoxMisclImage \(\)](#)

use proper icons for toolButtons

Member [PropertyEditor::createToolButtonQSelect \(\)](#)

document this

Member [PropertyEditor::eventFilter \(QObject *obj, QEvent *event\)](#)

document this

Member [PropertyEditor::~PropertyEditor \(\)](#)

document this

Member [RectObject::allGripPoints \(\)](#)

make return value a std::vector<std::string>

Member [SaveObject::save \(const QString &fileName\)](#)

Before saving to a stitch only format, Embroidermodder needs to calculate the optimal path to minimize jump stitches. Also based upon which layer needs to be stitched first, the path to the next object needs to be hidden beneath fills that will come later. When finding the optimal path, we need to take into account the color of the thread, as we do not want to try to hide dark colored stitches beneath light colored fills.

Member [SaveObject::toPolyline \(EmbPattern *pattern, const QPointF &objPos, const QPainterPath &objPath, const QString &layer, const QColor &color, const QString &lineType, const QString &lineWeight\)](#)

FIX EmbPolyline* polyObject = embPolyline_init(pointList, color_out, 1); //TODO: proper lineType embPattern->_addPolylineAbs(pattern, polyObject);

Member [SubDescriptor_::colorCode](#)

better variable naming

Member [SubDescriptor_::someInt](#)

better variable naming

Member [SubDescriptor_::someOtherInt](#)

better variable naming

Member [UndoHistory](#)

document this.

14 Hierarchical Index

14.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

| | |
|--------------------------------------|----|
| _bcf_directory | 48 |
| _bcf_directory_entry | 49 |
| _bcf_file | 51 |
| _bcf_file_difat | 52 |
| _bcf_file_fat | 52 |
| _bcf_file_header | 53 |
| _vp3Hoop | 55 |
| Action__ | 58 |

| | |
|--------------------------|-----|
| Compress | 106 |
| EmbAlignedDim_ | 121 |
| EmbAngularDim_ | 121 |
| EmbArc_ | 121 |
| EmbArcLengthDim_ | 122 |
| EmbArray_ | 123 |
| EmbBezier_ | 124 |
| EmbBlock_ | 124 |
| EmbCircle_ | 125 |
| EmbColor_ | 125 |
| EmbDiameterDim_ | 128 |
| EmbEllipse_ | 128 |
| EmbFormatList_ | 129 |
| EmbGeometry_ | 130 |
| EmblImage_ | 132 |
| EmblInfiniteLine_ | 133 |
| EmbLayer_ | 134 |
| EmbLeaderDim_ | 134 |
| EmbLine_ | 135 |
| EmbLinearDim_ | 136 |
| EmbOrdinateDim_ | 136 |
| EmbPath_ | 136 |
| EmbPattern_ | 137 |
| EmbPoint_ | 138 |
| EmbRadiusDim_ | 139 |
| EmbRay_ | 140 |
| EmbRect_ | 140 |
| EmbSatinOutline_ | 141 |
| EmbSpline_ | 142 |
| EmbStitch_ | 142 |
| EmbTextMulti_ | 143 |
| EmbTextSingle_ | 143 |

| | |
|-------------------------|-----|
| EmbThread_ | 144 |
| EmbTime_ | 145 |
| EmbVector_ | 146 |
| EmbView_ | 146 |
| hoop_padding | 150 |
| Huffman | 151 |
| LSYSTEM | 167 |
| Parameter_ | 235 |
| QApplication | |
| Application | 59 |
| QDialog | |
| EmbDetailsDialog | 126 |
| LayerManager | 159 |
| Settings_Dialog | 294 |
| QDockWidget | |
| PropertyEditor | 257 |
| UndoEditor | 339 |
| QFileDialog | |
| PreviewDialog | 256 |
| QGraphicsPathItem | |
| BaseObject | 73 |
| ArcObject | 60 |
| CircleObject | 79 |
| DimLeaderObject | 107 |
| EllipseObject | 115 |
| ImageObject | 152 |
| LineObject | 161 |
| PathObject | 236 |
| PointObject | 241 |
| PolygonObject | 245 |
| PolylineObject | 251 |
| RectObject | 270 |
| TextSingleObject | 321 |
| QGraphicsView | |

| | |
|----------------------------------|-----|
| View | 341 |
| QLineEdit | |
| CmdPromptInput | 98 |
| QMainWindow | |
| MainWindow | 167 |
| QMdiArea | |
| MdiArea | 221 |
| QMdiSubWindow | |
| MdiWindow | 225 |
| QObject | |
| SaveObject | 276 |
| QRubberBand | |
| SelectBox | 282 |
| QSplitter | |
| CmdPromptSplitter | 105 |
| QSplitterHandle | |
| CmdPromptHandle | 93 |
| QStatusBar | |
| StatusBar | 315 |
| QTextBrowser | |
| CmdPromptHistory | 95 |
| QToolButton | |
| StatusBarButton | 317 |
| QUndoCommand | |
| UndoableAddCommand | 331 |
| UndoableDeleteCommand | 332 |
| UndoableGripEditCommand | 333 |
| UndoableMirrorCommand | 334 |
| UndoableMoveCommand | 335 |
| UndoableNavCommand | 336 |
| UndoableRotateCommand | 337 |
| UndoableScaleCommand | 338 |
| QWidget | |
| CmdPrompt | 85 |
| ImageWidget | 157 |
| Settings_ | 285 |
| StxThread_ | 319 |
| SubDescriptor_ | 320 |

| | |
|------------------------|-----|
| SvgAttribute_ | 321 |
| thread_color_ | 327 |
| ThredExtension_ | 327 |
| ThredHeader_ | 328 |
| UiObject_ | 329 |
| UndoHistory_ | 341 |
| VipHeader_ | 355 |

15 Class Index

15.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

| | |
|----------------------------------|-----|
| _bcf_directory | 48 |
| _bcf_directory_entry | 49 |
| _bcf_file | 51 |
| _bcf_file_difat | 52 |
| _bcf_file_fat | 52 |
| _bcf_file_header | 53 |
| _vp3Hoop | 55 |
| Action__ | 58 |
| Application | 59 |
| ArcObject | 60 |
| BaseObject | 73 |
| CircleObject | 79 |
| CmdPrompt | 85 |
| CmdPromptHandle | 93 |
| CmdPromptHistory | 95 |
| The Command Prompt History class | |
| CmdPromptInput | 98 |
| CmdPromptSplitter | 105 |
| Compress | 106 |

| | |
|---|-----|
| DimLeaderObject | 107 |
| EllipseObject | 115 |
| EmbAlignedDim_ | 121 |
| EmbAngularDim_ | 121 |
| EmbArc_ Absolute position (not relative) | 121 |
| EmbArcLengthDim_ | 122 |
| EmbArray_ | 123 |
| EmbBezier_ | 124 |
| EmbBlock_ | 124 |
| EmbCircle_ | 125 |
| EmbColor_ | 125 |
| EmbDetailsDialog 126 | |
| EmbDiameterDim_ | 128 |
| EmbEllipse_ | 128 |
| EmbFormatList_ | 129 |
| EmbGeometry_ | 130 |
| EmblImage_ | 132 |
| EmblInfiniteLine_ | 133 |
| EmbLayer_ | 134 |
| EmbLeaderDim_ | 134 |
| EmbLine_ | 135 |
| EmbLinearDim_ | 136 |
| EmbOrdinateDim_ | 136 |
| EmbPath_ | 136 |
| EmbPattern_ | 137 |
| EmbPoint_ | 138 |
| EmbRadiusDim_ | 139 |
| EmbRay_ | 140 |
| EmbRect_ | 140 |
| EmbSatinOutline_ | 141 |
| EmbSpline_ | 142 |

| | |
|-----------------------------|-----|
| EmbStitch_ | 142 |
| EmbTextMulti_ | 143 |
| EmbTextSingle_ | 143 |
| EmbThread_ | 144 |
| EmbTime_ | 145 |
| EmbVector_ | 146 |
| EmbView_ | 146 |
| hoop_padding | 150 |
| Huffman | 151 |
| ImageObject | 152 |
| ImageWidget | 157 |
| LayerManager | 159 |
| LineObject | 161 |
| LSYSTEM | 167 |
| MainWindow | 167 |
| The MainWindow class | 167 |
| MdiArea | 221 |
| MdiWindow | 225 |
| Parameter_ | 235 |
| PathObject | 236 |
| PointObject | 241 |
| PolygonObject | 245 |
| PolylineObject | 251 |
| PreviewDialog | 256 |
| PropertyEditor | 257 |
| RectObject | 270 |
| SaveObject | 276 |
| SelectBox | 282 |
| Settings_ | 285 |
| Settings System | 285 |
| Settings_Dialog | 294 |
| StatusBar | 315 |

| | |
|--|-----|
| StatusBarButton | 317 |
| StxThread_ | 319 |
| SubDescriptor_ | 320 |
| SvgAttribute_ | 321 |
| TextSingleObject | 321 |
| thread_color_ | 327 |
| ThredExtension_ | 327 |
| ThredHeader_ | 328 |
| UiObject_ | |
| This covers the inbuilt designs: Dolphin, Snowflake and Heart. Covers Rotate, Scale and Point UI events | 329 |
| UndoableAddCommand | 331 |
| UndoableDeleteCommand | 332 |
| UndoableGripEditCommand | 333 |
| UndoableMirrorCommand | 334 |
| UndoableMoveCommand | 335 |
| UndoableNavCommand | 336 |
| UndoableRotateCommand | 337 |
| UndoableScaleCommand | 338 |
| UndoEditor | 339 |
| UndoHistory_ | 341 |
| View | 341 |
| VipHeader_ | 355 |

16 File Index

16.1 File List

Here is a list of all files with brief descriptions:

| | |
|---|-----|
| embroidermodder2/cmdprompt.cpp | 356 |
| embroidermodder2/embdetails-dialog.cpp | 356 |
| embroidermodder2/embroidermodder.cpp | 356 |
| embroidermodder2/embroidermodder.h | 357 |
| embroidermodder2/imagewidget.cpp | 404 |

| | |
|--|-----|
| embroidermodder2/layer-manager.cpp | 404 |
| embroidermodder2/mainwindow-commands.cpp | 404 |
| embroidermodder2/mainwindow-menus.cpp | 404 |
| embroidermodder2/mainwindow-settings.cpp | 404 |
| embroidermodder2/mainwindow-toolbars.cpp | 406 |
| embroidermodder2/mainwindow.cpp | 407 |
| embroidermodder2/mdiarea.cpp | 411 |
| embroidermodder2/mdiwindow.cpp | 411 |
| embroidermodder2/object-arc.cpp | 411 |
| embroidermodder2/object-base.cpp | 411 |
| embroidermodder2/object-circle.cpp | 411 |
| embroidermodder2/object-dimleader.cpp | 411 |
| embroidermodder2/object-ellipse.cpp | 411 |
| embroidermodder2/object-image.cpp | 411 |
| embroidermodder2/object-line.cpp | 411 |
| embroidermodder2/object-path.cpp | 412 |
| embroidermodder2/object-point.cpp | 412 |
| embroidermodder2/object-polygon.cpp | 412 |
| embroidermodder2/object-polyline.cpp | 412 |
| embroidermodder2/object-rect.cpp | 412 |
| embroidermodder2/object-save.cpp | 412 |
| embroidermodder2/object-textsingle.cpp | 412 |
| embroidermodder2/preview-dialog.cpp | 412 |
| embroidermodder2/property-editor.cpp | 412 |
| embroidermodder2/selectbox.cpp | 428 |
| embroidermodder2/settings-dialog.cpp | 428 |
| embroidermodder2/statusbar-button.cpp | 428 |
| embroidermodder2/statusbar.cpp | 428 |
| embroidermodder2/undo-commands.cpp | 428 |
| embroidermodder2/undo-editor.cpp | 428 |
| embroidermodder2/view.cpp | 429 |
| extern/libembroidery/src/array.c | 429 |

| | |
|---|-----|
| extern/libembroidery/src/ compress.c | 431 |
| extern/libembroidery/src/ embroidery.h | 433 |
| extern/libembroidery/src/ embroidery_internal.h | 473 |
| extern/libembroidery/src/ encoding.c | 520 |
| extern/libembroidery/src/ fill.c | 523 |
| extern/libembroidery/src/ formats.c | 528 |
| extern/libembroidery/src/ geometry.c | 574 |
| extern/libembroidery/src/ image.c | 588 |
| extern/libembroidery/src/ main.c | 589 |
| extern/libembroidery/src/ pattern.c | 600 |
| extern/libembroidery/src/ thread-color.c | 605 |
| extern/libembroidery/src/formats/ format_100.c | 532 |
| extern/libembroidery/src/formats/ format_10o.c | 532 |
| extern/libembroidery/src/formats/ format_art.c | 533 |
| extern/libembroidery/src/formats/ format_bmc.c | 533 |
| extern/libembroidery/src/formats/ format_bro.c | 534 |
| extern/libembroidery/src/formats/ format_cnd.c | 534 |
| extern/libembroidery/src/formats/ format_col.c | 535 |
| extern/libembroidery/src/formats/ format_csd.c | 536 |
| extern/libembroidery/src/formats/ format_csv.c | 537 |
| extern/libembroidery/src/formats/ format_dat.c | 538 |
| extern/libembroidery/src/formats/ format_dem.c | 539 |
| extern/libembroidery/src/formats/ format_dsb.c | 539 |
| extern/libembroidery/src/formats/ format_dst.c | 540 |
| extern/libembroidery/src/formats/ format_dsz.c | 542 |
| extern/libembroidery/src/formats/ format_dxf.c | 542 |
| extern/libembroidery/src/formats/ format_edr.c | 543 |
| extern/libembroidery/src/formats/ format_emd.c | 544 |
| extern/libembroidery/src/formats/ format_exp.c | 544 |
| extern/libembroidery/src/formats/ format_exy.c | 545 |
| extern/libembroidery/src/formats/ format_eyc.c | 545 |
| extern/libembroidery/src/formats/ format_fxy.c | 546 |

| | |
|--|-----|
| extern/libembroidery/src/formats/ format_gc.c | 546 |
| extern/libembroidery/src/formats/ format_gnc.c | 547 |
| extern/libembroidery/src/formats/ format_gt.c | 547 |
| extern/libembroidery/src/formats/ format_hus.c | 548 |
| extern/libembroidery/src/formats/ format_inb.c | 549 |
| extern/libembroidery/src/formats/ format_inf.c | 549 |
| extern/libembroidery/src/formats/ format_jef.c | 550 |
| extern/libembroidery/src/formats/ format_ksm.c | 551 |
| extern/libembroidery/src/formats/ format_max.c | 551 |
| extern/libembroidery/src/formats/ format_mit.c | 552 |
| extern/libembroidery/src/formats/ format_new.c | 553 |
| extern/libembroidery/src/formats/ format_ofm.c | 553 |
| extern/libembroidery/src/formats/ format_pcd.c | 554 |
| extern/libembroidery/src/formats/ format_pcm.c | 555 |
| extern/libembroidery/src/formats/ format_pcq.c | 555 |
| extern/libembroidery/src/formats/ format_pcs.c | 556 |
| extern/libembroidery/src/formats/ format_pec.c | 556 |
| extern/libembroidery/src/formats/ format_pel.c | 558 |
| extern/libembroidery/src/formats/ format_pem.c | 558 |
| extern/libembroidery/src/formats/ format_pes.c | 559 |
| extern/libembroidery/src/formats/ format_phb.c | 561 |
| extern/libembroidery/src/formats/ format_phc.c | 562 |
| extern/libembroidery/src/formats/ format_plt.c | 562 |
| extern/libembroidery/src/formats/ format_rgb.c | 563 |
| extern/libembroidery/src/formats/ format_sew.c | 563 |
| extern/libembroidery/src/formats/ format_shv.c | 564 |
| extern/libembroidery/src/formats/ format_sst.c | 564 |
| extern/libembroidery/src/formats/ format_stx.c | 565 |
| extern/libembroidery/src/formats/ format_svg.c | 565 |
| extern/libembroidery/src/formats/ format_t01.c | 567 |
| extern/libembroidery/src/formats/ format_t09.c | 567 |
| extern/libembroidery/src/formats/ format.tap.c | 568 |

| | |
|---|-----|
| extern/libembroidery/src/formats/format_thr.c | 568 |
| extern/libembroidery/src/formats/format_txt.c | 569 |
| extern/libembroidery/src/formats/format_u00.c | 569 |
| extern/libembroidery/src/formats/format_u01.c | 570 |
| extern/libembroidery/src/formats/format_vip.c | 570 |
| extern/libembroidery/src/formats/format_vp3.c | 572 |
| extern/libembroidery/src/formats/format_xxx.c | 573 |
| extern/libembroidery/src/formats/format_zsk.c | 574 |
| extern/libembroidery/src/geometry/arc.c | 576 |
| extern/libembroidery/src/geometry/circle.c | 579 |
| extern/libembroidery/src/geometry/ellipse.c | 580 |
| extern/libembroidery/src/geometry/functions.c | 582 |
| extern/libembroidery/src/geometry/line.c | 583 |
| extern/libembroidery/src/geometry/path.c | 583 |
| extern/libembroidery/src/geometry/polygon.c | 583 |
| extern/libembroidery/src/geometry/polyline.c | 583 |
| extern/libembroidery/src/geometry/rect.c | 584 |
| extern/libembroidery/src/geometry/text.c | 584 |
| extern/libembroidery/src/geometry/vector.c | 586 |

17 Class Documentation

17.1 _bcf_directory Struct Reference

```
#include <embroidery_internal.h>
```

Public Attributes

- `bcf_directory_entry * dirEntries`
- `unsigned int maxNumberOfDirectoryEntries`

17.1.1 Detailed Description

Todo possibly add a directory tree in the future.

17.1.2 Member Data Documentation

17.1.2.1 `dirEntries` `bcf_directory_entry*` `dirEntries`

17.1.2.2 `maxNumberOfDirectoryEntries` `unsigned int` `maxNumberOfDirectoryEntries`

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery_internal.h`

17.2 `_bcf_directory_entry` Struct Reference

```
#include <embroidery_internal.h>
```

Public Attributes

- `char directoryEntryName [32]`
- `unsigned short directoryEntryNameLength`
- `unsigned char objectType`
- `unsigned char colorFlag`
- `unsigned int leftSiblingId`
- `unsigned int rightSiblingId`
- `unsigned int childId`
- `unsigned char CLSID [16]`
- `unsigned int stateBits`
- `EmbTime creationTime`
- `EmbTime modifiedTime`
- `unsigned int startingSectorLocation`
- `unsigned long streamSize`
- `unsigned int streamSizeHigh`
- `struct _bcf_directory_entry * next`

17.2.1 Member Data Documentation

17.2.1.1 `childId` `unsigned int` `childId`

17.2.1.2 `CLSID` `unsigned char` `CLSID[16]`

17.2.1.3 colorFlag unsigned char colorFlag

17.2.1.4 creationTime [EmbTime](#) creationTime

17.2.1.5 directoryEntryName char directoryEntryName[32]

17.2.1.6 directoryEntryNameLength unsigned short directoryEntryNameLength

17.2.1.7 leftSiblingId unsigned int leftSiblingId

17.2.1.8 modifiedTime [EmbTime](#) modifiedTime

17.2.1.9 next struct [_bcf_directory_entry](#)* next

17.2.1.10 objectType unsigned char objectType

17.2.1.11 rightSiblingId unsigned int rightSiblingId

17.2.1.12 startingSectorLocation unsigned int startingSectorLocation

17.2.1.13 stateBits unsigned int stateBits

17.2.1.14 streamSize unsigned long streamSize

17.2.1.15 streamSizeHigh unsigned int streamSizeHigh

The documentation for this struct was generated from the following file:

- [extern/libembroidery/src/embroidery_internal.h](#)

17.3 _bcf_file Struct Reference

```
#include <embroidery_internal.h>
```

Public Attributes

- [bcf_file_header header](#)
- [bcf_file_difat * difat](#)
- [bcf_file_fat * fat](#)
- [bcf_directory * directory](#)

17.3.1 Member Data Documentation

17.3.1.1 difat [bcf_file_difat*](#) difat

The header for the CompoundFile

17.3.1.2 directory [bcf_directory*](#) directory

The File Allocation Table for the Compound File

17.3.1.3 fat [bcf_file_fat*](#) fat

The "Double Indirect FAT" for the CompoundFile

17.3.1.4 header [bcf_file_header](#) header

The documentation for this struct was generated from the following file:

- [extern/libembroidery/src/embroidery_internal.h](#)

17.4 `_bcf_file_difat` Struct Reference

```
#include <embroidery_internal.h>
```

Public Attributes

- unsigned int `fatSectorCount`
- unsigned int `fatSectorEntries` [109]
- unsigned int `sectorSize`

17.4.1 Member Data Documentation

17.4.1.1 `fatSectorCount` unsigned int `fatSectorCount`

17.4.1.2 `fatSectorEntries` unsigned int `fatSectorEntries[109]`

17.4.1.3 `sectorSize` unsigned int `sectorSize`

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery_internal.h`

17.5 `_bcf_file_fat` Struct Reference

```
#include <embroidery_internal.h>
```

Public Attributes

- int `fatEntryCount`
- unsigned int `fatEntries` [255]
- unsigned int `numberOfEntriesInFatSector`

17.5.1 Member Data Documentation

17.5.1.1 fatEntries unsigned int fatEntries[255]

17.5.1.2 fatEntryCount int fatEntryCount

17.5.1.3 numberOfEntriesInFatSector unsigned int numberOfEntriesInFatSector

The documentation for this struct was generated from the following file:

- [extern/libembroidery/src/embroidery_internal.h](#)

17.6 _bcf_file_header Struct Reference

```
#include <embroidery_internal.h>
```

Public Attributes

- unsigned char [signature](#) [8]
- unsigned char [CLSID](#) [16]
- unsigned short [minorVersion](#)
- unsigned short [majorVersion](#)
- unsigned short [byteOrder](#)
- unsigned short [sectorShift](#)
- unsigned short [miniSectorShift](#)
- unsigned short [reserved1](#)
- unsigned int [reserved2](#)
- unsigned int [numberOfDirectorySectors](#)
- unsigned int [numberOfFATSectors](#)
- unsigned int [firstDirectorySectorLocation](#)
- unsigned int [transactionSignatureNumber](#)
- unsigned int [miniStreamCutoffSize](#)
- unsigned int [firstMiniFATSectorLocation](#)
- unsigned int [numberOfMiniFatSectors](#)
- unsigned int [firstDifatSectorLocation](#)
- unsigned int [numberOfDifatSectors](#)

17.6.1 Detailed Description

Todo CLSID should be a separate type.

17.6.2 Member Data Documentation

17.6.2.1 byteOrder unsigned short byteOrder

17.6.2.2 CLSID unsigned char CLSID[16]

17.6.2.3 firstDifatSectorLocation unsigned int firstDifatSectorLocation

17.6.2.4 firstDirectorySectorLocation unsigned int firstDirectorySectorLocation

17.6.2.5 firstMiniFATSectorLocation unsigned int firstMiniFATSectorLocation

17.6.2.6 majorVersion unsigned short majorVersion

17.6.2.7 miniSectorShift unsigned short miniSectorShift

17.6.2.8 miniStreamCutoffSize unsigned int miniStreamCutoffSize

17.6.2.9 minorVersion unsigned short minorVersion

17.6.2.10 numberOfDifatSectors unsigned int numberOfDifatSectors

17.6.2.11 numberOfDirectorySectors unsigned int numberOfDirectorySectors

17.6.2.12 numberOfFATSectors unsigned int numberOfFATSectors

17.6.2.13 numberOfMiniFatSectors unsigned int numberOfMiniFatSectors

17.6.2.14 reserved1 unsigned short reserved1

17.6.2.15 reserved2 unsigned int reserved2

17.6.2.16 sectorShift unsigned short sectorShift

17.6.2.17 signature unsigned char signature[8]

17.6.2.18 transactionSignatureNumber unsigned int transactionSignatureNumber

The documentation for this struct was generated from the following file:

- [extern/libembroidery/src/embroidery_internal.h](#)

17.7 _vp3Hoop Struct Reference

```
#include <embroidery_internal.h>
```

Public Attributes

- int [right](#)
- int [bottom](#)
- int [left](#)
- int [top](#)
- int [threadLength](#)
- char [unknown2](#)
- unsigned char [numberOfColors](#)
- unsigned short [unknown3](#)
- int [unknown4](#)
- int [numberOfBytesRemaining](#)
- int [xOffset](#)
- int [yOffset](#)
- unsigned char [byte1](#)
- unsigned char [byte2](#)
- unsigned char [byte3](#)
- int [right2](#)
- int [left2](#)
- int [bottom2](#)
- int [top2](#)
- int [width](#)
- int [height](#)

17.7.1 Member Data Documentation

17.7.1.1 bottom int bottom

17.7.1.2 bottom2 int bottom2

17.7.1.3 byte1 unsigned char byte1

17.7.1.4 byte2 unsigned char byte2

17.7.1.5 byte3 unsigned char byte3

17.7.1.6 height int height

17.7.1.7 left int left

17.7.1.8 left2 int left2

17.7.1.9 numberOfBytesRemaining int numberOfBytesRemaining

17.7.1.10 numberOfColors unsigned char numberOfColors

17.7.1.11 right int right

17.7.1.12 right2 int right2

17.7.1.13 threadLength int threadLength

17.7.1.14 top int top

17.7.1.15 top2 int top2

17.7.1.16 unknown2 char unknown2

17.7.1.17 unknown3 unsigned short unknown3

17.7.1.18 unknown4 int unknown4

17.7.1.19 width int width

17.7.1.20 xOffset int xOffset

17.7.1.21 yOffset int yOffset

The documentation for this struct was generated from the following file:

- [extern/libembroidery/src/embroidery_internal.h](#)

17.8 Action__ Struct Reference

```
#include <embroidermodder.h>
```

Public Attributes

- int `hash`
- std::string `icon`
- std::string `tooltip`
- std::string `statustip`
- std::string `shortcut`
- std::vector< std::string > `aliases`
- std::vector< std::string > `script`
- std::string `menu_name`
- int `menu_position`
- std::string `toolbar_name`
- int `toolbar_position`

17.8.1 Detailed Description

17.8.2 Member Data Documentation

17.8.2.1 aliases std::vector<std::string> `aliases`

17.8.2.2 hash int `hash`

17.8.2.3 icon std::string `icon`

17.8.2.4 menu_name std::string `menu_name`

17.8.2.5 menu_position int `menu_position`

17.8.2.6 script std::vector<std::string> `script`

17.8.2.7 shortcut std::string shortcut

17.8.2.8 statustip std::string statustip

17.8.2.9 toolbar_name std::string toolbar_name

17.8.2.10 toolbar_position int toolbar_position

17.8.2.11 tooltip std::string tooltip

The documentation for this struct was generated from the following file:

- [embroidermodder2/embroidermodder.h](#)

17.9 Application Class Reference

```
#include <embroidermodder.h>
```

Public Member Functions

- [Application](#) (int argc, char **argv)
Application::Application.
- void [setMainWin](#) (MainWindow *mainWin)

Public Attributes

- [MainWindow * _mainWin](#)

Protected Member Functions

- virtual bool [event](#) (QEvent *e)
Application::event.

17.9.1 Detailed Description

Note

On Mac, if the user drops a file on the app's Dock icon, or uses Open As, then this is how the app actually opens the file.

17.9.2 Constructor & Destructor Documentation

17.9.2.1 Application() [Application](#) (

```
    int argc,  
    char ** argv )
```

Application::Application.

Parameters

| | |
|-------------|----------------------|
| <i>argc</i> | <input type="text"/> |
| <i>argv</i> | <input type="text"/> |

17.9.3 Member Function Documentation

17.9.3.1 event() `bool event (QEvent * event) [protected], [virtual]`

[Application::event](#).

Parameters

| | |
|--------------|----------------------|
| <i>event</i> | <input type="text"/> |
|--------------|----------------------|

Returns

17.9.3.2 setMainWin() `void setMainWin (MainWindow * mainWin) [inline]`

17.9.4 Member Data Documentation

17.9.4.1 _mainWin [MainWindow* _mainWin](#)

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/embroidermodder.cpp](#)

17.10 ArcObject Class Reference

```
#include <embroidermodder.h>
```

Public Types

- enum { [Type](#) = OBJ_TYPE_ARC }

Public Types inherited from BaseObject

- enum { `Type` = `OBJ_TYPE_BASE` }

Public Member Functions

- `ArcObject (EmbArc arc, QRgb rgb, QGraphicsItem *parent=0)`
ArcObject::ArcObject.
- `ArcObject (EmbReal startX, EmbReal startY, EmbReal midX, EmbReal midY, EmbReal endX, EmbReal endY, QRgb rgb, QGraphicsItem *parent=0)`
ArcObject::ArcObject.
- `ArcObject (ArcObject *obj, QGraphicsItem *parent=0)`
ArcObject::ArcObject.
- `~ArcObject ()`
ArcObject::~ArcObject.
- `virtual int type () const`
- `void init (EmbReal startX, EmbReal startY, EmbReal midX, EmbReal midY, EmbReal endX, EmbReal endY, QRgb rgb, Qt::PenStyle lineType)`
ArcObject::init.
- `void updatePath ()`
ArcObject::updatePath.
- `void calculateArcData (EmbReal startX, EmbReal startY, EmbReal midX, EmbReal midY, EmbReal endX, EmbReal endY)`
ArcObject::calculateArcData.
- `void updateArcRect (EmbReal radius)`
ArcObject::updateArcRect.
- `EmbReal objectRadius () const`
- `EmbReal objectStartAngle () const`
ArcObject::objectStartAngle.
- `EmbReal objectEndAngle () const`
ArcObject::objectEndAngle.
- `QPointF objectStartPoint () const`
ArcObject::objectStartPoint.
- `EmbReal objectStartX () const`
ArcObject::objectStartX.
- `EmbReal objectStartY () const`
ArcObject::objectStartY.
- `QPointF objectMidPoint () const`
ArcObject::objectMidPoint.
- `EmbReal objectMidX () const`
ArcObject::objectMidX.
- `EmbReal objectMidY () const`
ArcObject::objectMidY.
- `QPointF objectEndPoint () const`
ArcObject::objectEndPoint.
- `EmbReal objectEndX () const`
ArcObject::objectEndX.
- `EmbReal objectEndY () const`
ArcObject::objectEndY.
- `EmbReal objectArea () const`

- *ArcObject::objectArea.*
- EmbReal **objectArcLength** () const
 - ArcObject::objectArcLength.*
- EmbReal **objectChord** () const
 - ArcObject::objectChord.*
- EmbReal **objectIncludedAngle** () const
 - ArcObject::objectIncludedAngle.*
- bool **objectClockwise** () const
 - ArcObject::objectClockwise.*
- void **setObjectRadius** (EmbReal radius)
- void **setObjectStartAngle** (EmbReal angle)
- void **setObjectEndAngle** (EmbReal angle)
- void **setObjectStartPoint** (const QPointF &point)
- void **setObjectStartPoint** (EmbReal pointX, EmbReal pointY)
- void **setObjectMidPoint** (const QPointF &point)
- void **setObjectMidPoint** (EmbReal pointX, EmbReal pointY)
- void **setObjectEndPoint** (const QPointF &point)
 - ArcObject::setObjectEndPoint.*
- void **setObjectEndPoint** (EmbReal pointX, EmbReal pointY)
 - ArcObject::setObjectEndPoint.*
- void **updateRubber** (QPainter *painter=0)
 - ArcObject::updateRubber.*
- virtual void **vulcanize** ()
 - ArcObject::vulcanize.*
- virtual QPointF **mouseSnapPoint** (const QPointF &mousePoint)
 - ArcObject::mouseSnapPoint.*
- virtual QList< QPointF > **allGripPoints** ()
 - ArcObject::allGripPoints.*
- virtual void **gripEdit** (const QPointF &before, const QPointF &after)
 - ArcObject::gripEdit.*

Public Member Functions inherited from [BaseObject](#)

- [BaseObject](#) (QGraphicsItem *parent=0)
- virtual ~[BaseObject](#) ()
- virtual int **type** () const
- qint64 **objectID** () const
- QPen **objectPen** () const
- QColor **objectColor** () const
- QRgb **objectColorRGB** () const
- Qt::PenStyle **objectLineType** () const
- EmbReal **objectLineWidth** () const
- QPainterPath **objectPath** () const
- int **objectRubberMode** () const
- QPointF **objectRubberPoint** (const QString &key) const
- QString **objectRubberText** (const QString &key) const
- QPointF **objectCenter** () const
- EmbReal **objectCenterX** () const
- EmbReal **objectCenterY** () const
- void **setObjectCenter** (EmbVector center)
- void **setObjectCenterX** (EmbReal centerX)

- void `setObjectCenterY` (`EmbReal` centerY)
- `QRectF rect` () const
- void `setRect` (const `QRectF` &r)
- void `setRect` (`EmbReal` x, `EmbReal` y, `EmbReal` w, `EmbReal` h)
- `QLineF line` () const
- void `setLine` (const `QLineF` &li)
- void `setLine` (`EmbReal` x1, `EmbReal` y1, `EmbReal` x2, `EmbReal` y2)
- void `setObjectColor` (const `QColor` &color)
- void `setObjectColorRGB` (`QRgb` rgb)
- void `setObjectLineType` (`Qt::PenStyle` lineType)
- void `setObjectLineWidth` (`EmbReal` lineWidth)
- void `setObjectPath` (const `QPainterPath` &p)
- void `setObjectRubberMode` (int mode)
- void `setObjectRubberPoint` (const `QString` &key, const `QPointF` &point)
- void `setObjectRubberText` (const `QString` &key, const `QString` &txt)
- virtual `QRectF boundingRect` () const
- virtual `QPainterPath shape` () const
- void `drawRubberLine` (const `QLineF` &rubLine, `QPainter` *painter=0, const `char` *colorFromScene=0)
- virtual void `vulcanize` ()=0
- virtual `QPointF mouseSnapPoint` (const `QPointF` &mousePoint)=0
- virtual `QList< QPointF >` `allGripPoints` ()=0
- virtual void `gripEdit` (const `QPointF` &before, const `QPointF` &after)=0

Public Attributes

- `QPointF arcStartPoint`
- `QPointF arcMidPoint`
- `QPointF arcEndPoint`

Public Attributes inherited from `BaseObject`

- `QPen objPen`
- `QPen lwtPen`
- `QLineF objLine`
- int `objRubberMode`
- `QHash< QString, QPointF >` `objRubberPoints`
- `QHash< QString, QString >` `objRubberTexts`
- `qint64 objID`

Protected Member Functions

- void `paint` (`QPainter` *, const `QStyleOptionGraphicsItem` *, `QWidget` *)
ArcObject::paint.

Protected Member Functions inherited from `BaseObject`

- `QPen lineWeightPen` () const
- void `realRender` (`QPainter` *painter, const `QPainterPath` &renderPath)

17.10.1 Member Enumeration Documentation

17.10.1.1 anonymous enum anonymous enum

Enumerator

| | |
|------|---------------------------------|
| Type | <input type="button" value=""/> |
|------|---------------------------------|

17.10.2 Constructor & Destructor Documentation**17.10.2.1 ArcObject() [1/3]** [ArcObject](#) (

```
    EmbArc arc,  
    QRgb rgb,  
    QGraphicsItem * parent = 0 )
```

[ArcObject::ArcObject](#).

Parameters

| | |
|---------------|---------------------------------|
| <i>arc</i> | <input type="button" value=""/> |
| <i>rgb</i> | <input type="button" value=""/> |
| <i>parent</i> | <input type="button" value=""/> |

17.10.2.2 ArcObject() [2/3] [ArcObject](#) (

```
    EmbReal startX,  
    EmbReal startY,  
    EmbReal midX,  
    EmbReal midY,  
    EmbReal endX,  
    EmbReal endY,  
    QRgb rgb,  
    QGraphicsItem * parent = 0 )
```

[ArcObject::ArcObject](#).

Parameters

| | |
|---------------|---------------------------------|
| <i>startX</i> | <input type="button" value=""/> |
| <i>startY</i> | <input type="button" value=""/> |
| <i>midX</i> | <input type="button" value=""/> |
| <i>midY</i> | <input type="button" value=""/> |
| <i>endX</i> | <input type="button" value=""/> |
| <i>endY</i> | <input type="button" value=""/> |
| <i>rgb</i> | <input type="button" value=""/> |
| <i>parent</i> | <input type="button" value=""/> |

17.10.2.3 ArcObject() [3/3] `ArcObject (`

```
    ArcObject * obj,  
    QGraphicsItem * parent = 0 )
```

`ArcObject::ArcObject.`

Parameters

| | |
|---------------------|--|
| <code>obj</code> | |
| <code>parent</code> | |

17.10.2.4 ~ArcObject() `~ArcObject ()`

`ArcObject::~ArcObject.`

17.10.3 Member Function Documentation**17.10.3.1 allGripPoints()** `QList< QPointF > allGripPoints () [virtual]`

`ArcObject::allGripPoints.`

Returns

Implements `BaseObject`.

17.10.3.2 calculateArcData() `void calculateArcData (`

```
    EmbReal startX,  
    EmbReal startY,  
    EmbReal midX,  
    EmbReal midY,  
    EmbReal endX,  
    EmbReal endY )
```

`ArcObject::calculateArcData.`

Parameters

| | |
|---------------------|--|
| <code>startX</code> | |
| <code>startY</code> | |
| <code>midX</code> | |
| <code>midY</code> | |
| <code>endX</code> | |
| <code>endY</code> | |

```
17.10.3.3 gripEdit() void gripEdit (
    const QPointF & before,
    const QPointF & after ) [virtual]
```

[ArcObject::gripEdit](#).

Parameters

| | |
|---------------|--|
| <i>before</i> | |
| <i>after</i> | |

Todo `gripEdit()` for [ArcObject](#)

Implements [BaseObject](#).

```
17.10.3.4 init() void init (
    EmbReal startX,
    EmbReal startY,
    EmbReal midX,
    EmbReal midY,
    EmbReal endX,
    EmbReal endY,
    QRgb rgb,
    Qt::PenStyle lineType )
```

[ArcObject::init](#).

Parameters

| | |
|-----------------|--|
| <i>startX</i> | |
| <i>startY</i> | |
| <i>midX</i> | |
| <i>midY</i> | |
| <i>endX</i> | |
| <i>endY</i> | |
| <i>rgb</i> | |
| <i>lineType</i> | |

```
17.10.3.5 mouseSnapPoint() QPointF mouseSnapPoint (
    const QPointF & mousePoint ) [virtual]
```

[ArcObject::mouseSnapPoint](#).

Parameters

mousePoint

Returns

the closest snap point to the mouse point.

Implements [BaseObject](#).

17.10.3.6 objectArcLength() `EmbReal objectArcLength () const`

[ArcObject::objectArcLength](#).

Returns**17.10.3.7 objectArea()** `EmbReal objectArea () const`

[ArcObject::objectArea](#).

Returns**17.10.3.8 objectChord()** `EmbReal objectChord () const`

[ArcObject::objectChord](#).

Returns**17.10.3.9 objectClockwise()** `bool objectClockwise () const`

[ArcObject::objectClockwise](#).

Returns

17.10.3.10 objectEndAngle() `EmbReal` `objectEndAngle() const`

`ArcObject::objectEndAngle.`

Returns

17.10.3.11 objectEndPoint() `QPointF` `objectEndPoint() const`

`ArcObject::objectEndPoint.`

Returns

17.10.3.12 objectEndX() `EmbReal` `objectEndX() const`

`ArcObject::objectEndX.`

Returns

17.10.3.13 objectEndY() `EmbReal` `objectEndY() const`

`ArcObject::objectEndY.`

Returns

17.10.3.14 objectIncludedAngle() `EmbReal` `objectIncludedAngle() const`

`ArcObject::objectIncludedAngle.`

Returns

17.10.3.15 objectMidPoint() `QPointF objectMidPoint () const`
`ArcObject::objectMidPoint.`

Returns

17.10.3.16 objectMidX() `EmbReal objectMidX () const`
`ArcObject::objectMidX.`

Returns

17.10.3.17 objectMidY() `EmbReal objectMidY () const`
`ArcObject::objectMidY.`

Returns

17.10.3.18 objectRadius() `EmbReal objectRadius () const [inline]`

17.10.3.19 objectStartAngle() `EmbReal objectStartAngle () const`
`ArcObject::objectStartAngle.`

Returns

17.10.3.20 objectStartPoint() `QPointF objectStartPoint () const`
`ArcObject::objectStartPoint.`

Returns

17.10.3.21 objectStartX() `EmbReal` `objectStartX () const`

[ArcObject::objectStartX.](#)

Returns

17.10.3.22 objectStartY() `EmbReal` `objectStartY () const`

[ArcObject::objectStartY.](#)

Returns

17.10.3.23 paint() `void` `paint (`
 `QPainter * painter,`
 `const QStyleOptionGraphicsItem * option,`
 `QWidget *) [protected]`

[ArcObject::paint.](#)

Parameters

| | |
|----------------------|--|
| <code>painter</code> | |
| <code>option</code> | |

17.10.3.24 setObjectEndAngle() `void` `setObjectEndAngle (`
 `EmbReal angle)`**17.10.3.25 setObjectEndPoint()** `[1/2]` `void` `setObjectEndPoint (`
 `const QPointF & point)`

[ArcObject::setObjectEndPoint.](#)

Parameters

| | |
|--------------------|--|
| <code>point</code> | |
|--------------------|--|

17.10.3.26 setObjectEndPoint() [2/2] void setObjectEndPoint (

| | |
|-----------------|--|
| EmReal pointX, | |
| EmReal pointY) | |

ArcObject::setObjectEndPoint.

Parameters

| | |
|---------------|--|
| <i>pointX</i> | |
| <i>pointY</i> | |

17.10.3.27 setObjectMidPoint() [1/2] void setObjectMidPoint (

| | |
|-------------------------|--|
| const QPointF & point) | |
|-------------------------|--|

17.10.3.28 setObjectMidPoint() [2/2] void setObjectMidPoint (

| | |
|-----------------|--|
| EmReal pointX, | |
| EmReal pointY) | |

17.10.3.29 setObjectRadius() void setObjectRadius (

| | |
|-----------------|--|
| EmReal radius) | |
|-----------------|--|

17.10.3.30 setObjectStartAngle() void setObjectStartAngle (

| | |
|----------------|--|
| EmReal angle) | |
|----------------|--|

17.10.3.31 setObjectStartPoint() [1/2] void setObjectStartPoint (

| | |
|-------------------------|--|
| const QPointF & point) | |
|-------------------------|--|

17.10.3.32 setObjectStartPoint() [2/2] void setObjectStartPoint (

| | |
|-----------------|--|
| EmReal pointX, | |
| EmReal pointY) | |

17.10.3.33 type() virtual int type () const [inline], [virtual]
Reimplemented from [BaseObject](#).

17.10.3.34 updateArcRect() void updateArcRect (

| | |
|-----------------|--|
| EmReal radius) | |
|-----------------|--|

ArcObject::updateArcRect.

Parameters

| | |
|---------------|----------------------|
| <i>radius</i> | <input type="text"/> |
|---------------|----------------------|

17.10.3.35 updatePath() void updatePath ()

[ArcObject::updatePath](#).

17.10.3.36 updateRubber() void updateRubber (
 QPainter * painter = 0)

[ArcObject::updateRubber](#).

Parameters

| | |
|----------------|----------------------|
| <i>painter</i> | <input type="text"/> |
|----------------|----------------------|

Todo Arc Rubber Modes

Todo [updateRubber\(\)](#) gripping for ArcObject

17.10.3.37 vulcanize() void vulcanize () [virtual]

[ArcObject::vulcanize](#).

Implements [BaseObject](#).

17.10.4 Member Data Documentation**17.10.4.1 arcEndPoint** QPointF arcEndPoint**17.10.4.2 arcMidPoint** QPointF arcMidPoint

17.10.4.3 arcStartPoint QPointF arcStartPoint

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/object-arc.cpp](#)

17.11 BaseObject Class Reference

```
#include <embroidermodder.h>
```

Public Types

- enum { [Type](#) = OBJ_TYPE_BASE }

Public Member Functions

- [BaseObject](#) (QGraphicsItem *parent=0)
- virtual ~[BaseObject](#) ()
- virtual int [type](#) () const
- qint64 [objectID](#) () const
- QPen [objectPen](#) () const
- QColor [objectColor](#) () const
- QRgb [objectColorRGB](#) () const
- Qt::PenStyle [objectLineType](#) () const
- EmbReal [objectLineWidth](#) () const
- QPainterPath [objectPath](#) () const
- int [objectRubberMode](#) () const
- QPointF [objectRubberPoint](#) (const QString &key) const
- QString [objectRubberText](#) (const QString &key) const
- QPointF [objectCenter](#) () const
- EmbReal [objectCenterX](#) () const
- EmbReal [objectCenterY](#) () const
- void [setObjectCenter](#) (EmbVector center)
- void [setObjectCenterX](#) (EmbReal centerX)
- void [setObjectCenterY](#) (EmbReal centerY)
- QRectF [rect](#) () const
- void [setRect](#) (const QRectF &r)
- void [setRect](#) (EmbReal x, EmbReal y, EmbReal w, EmbReal h)
- QLineF [line](#) () const
- void [setLine](#) (const QLineF &li)
- void [setLine](#) (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2)
- void [setObjectColor](#) (const QColor &color)
- void [setObjectColorRGB](#) (QRgb rgb)
- void [setObjectLineType](#) (Qt::PenStyle lineType)
- void [setObjectLineWidth](#) (EmbReal lineWidth)
- void [setObjectPath](#) (const QPainterPath &p)
- void [setObjectRubberMode](#) (int mode)
- void [setObjectRubberPoint](#) (const QString &key, const QPointF &point)
- void [setObjectRubberText](#) (const QString &key, const QString &txt)
- virtual QRectF [boundingRect](#) () const
- virtual QPainterPath [shape](#) () const
- void [drawRubberLine](#) (const QLineF &rubLine, QPainter *painter=0, const char *colorFromScene=0)
- virtual void [vulcanize](#) ()=0
- virtual QPointF [mouseSnapPoint](#) (const QPointF &mousePoint)=0
- virtual QList< QPointF > [allGripPoints](#) ()=0
- virtual void [gripEdit](#) (const QPointF &before, const QPointF &after)=0

Public Attributes

- QPen `objPen`
- QPen `lwtPen`
- QLineF `objLine`
- int `objRubberMode`
- QHash<QString, QPointF> `objRubberPoints`
- QHash<QString, QString> `objRubberTexts`
- qint64 `objID`

Protected Member Functions

- QPen `lineWeightPen () const`
- void `realRender (QPainter *painter, const QPainterPath &renderPath)`

17.11.1 Member Enumeration Documentation

17.11.1.1 anonymous enum anonymous enum

Enumerator

| | |
|------|---------------------------------|
| Type | <input type="button" value=""/> |
|------|---------------------------------|

17.11.2 Constructor & Destructor Documentation

17.11.2.1 BaseObject() `BaseObject (` `QGraphicsItem * parent = 0)`

17.11.2.2 ~BaseObject() `~BaseObject () [virtual]`

17.11.3 Member Function Documentation

17.11.3.1 allGripPoints() `virtual QList<QPointF> allGripPoints () [pure virtual]`

Implemented in `ArcObject`, `CircleObject`, `DimLeaderObject`, `EllipseObject`, `ImageObject`, `LineObject`, `PathObject`, `PointObject`, `PolygonObject`, `PolylineObject`, `RectObject`, and `TextSingleObject`.

17.11.3.2 boundingRect() `QRectF boundingRect () const [virtual]`

17.11.3.3 drawRubberLine() `void drawRubberLine (`
 `const QLineF & rubLine,`
 `QPainter * painter = 0,`
 `const char * colorFromScene = 0)`

17.11.3.4 gripEdit() `virtual void gripEdit (`
 `const QPointF & before,`
 `const QPointF & after) [pure virtual]`

Implemented in [ArcObject](#), [CircleObject](#), [DimLeaderObject](#), [EllipseObject](#), [ImageObject](#), [LineObject](#), [PathObject](#), [PointObject](#), [PolygonObject](#), [PolylineObject](#), [RectObject](#), and [TextSingleObject](#).

17.11.3.5 line() `QLineF line () const [inline]`

17.11.3.6 lineWeightPen() `QPen lineWeightPen () const [inline], [protected]`

17.11.3.7 mouseSnapPoint() `virtual QPointF mouseSnapPoint (`
 `const QPointF & mousePoint) [pure virtual]`

Implemented in [ArcObject](#), [CircleObject](#), [DimLeaderObject](#), [EllipseObject](#), [ImageObject](#), [LineObject](#), [PathObject](#), [PointObject](#), [PolygonObject](#), [PolylineObject](#), [RectObject](#), and [TextSingleObject](#).

17.11.3.8 objectCenter() `QPointF objectCenter () const [inline]`

17.11.3.9 objectCenterX() `EmbReal objectCenterX () const [inline]`

17.11.3.10 objectCenterY() `EmbReal objectCenterY () const [inline]`

17.11.3.11 `objectColor()` QColor objectColor () const [inline]

17.11.3.12 `objectColorRGB()` QRgb objectColorRGB () const [inline]

17.11.3.13 `objectID()` qint64 objectID () const [inline]

17.11.3.14 `objectLineType()` Qt::PenStyle objectLineType () const [inline]

17.11.3.15 `objectLineWidth()` EmbReal objectLineWidth () const [inline]

17.11.3.16 `objectPath()` QPainterPath objectPath () const [inline]

17.11.3.17 `objectPen()` QPen objectPen () const [inline]

17.11.3.18 `objectRubberMode()` int objectRubberMode () const [inline]

17.11.3.19 `objectRubberPoint()` QPointF objectRubberPoint (const QString & key) const

17.11.3.20 `objectRubberText()` QString objectRubberText (const QString & key) const

17.11.3.21 `realRender()` void realRender (QPainter * painter, const QPainterPath & renderPath) [protected]

17.11.3.22 rect() QRectF rect () const [inline]

17.11.3.23 setLine() [1/2] void setLine (const QLineF & li) [inline]

17.11.3.24 setLine() [2/2] void setLine (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2) [inline]

17.11.3.25 setObjectCenter() void setObjectCenter (EmbVector center) [inline]

17.11.3.26 setObjectCenterX() void setObjectCenterX (EmbReal centerX) [inline]

17.11.3.27 setObjectCenterY() void setObjectCenterY (EmbReal centerY) [inline]

17.11.3.28 setObjectColor() void setObjectColor (const QColor & color)

17.11.3.29 setObjectColorRGB() void setObjectColorRGB (QRgb rgb)

17.11.3.30 setObjectLineType() void setObjectLineType (Qt::PenStyle lineType)

17.11.3.31 `setObjectLineWeight()` void setObjectLineWeight (EmbReal lineWeight)

17.11.3.32 `setObjectPath()` void setObjectPath (const QPainterPath & p) [inline]

17.11.3.33 `setObjectRubberMode()` void setObjectRubberMode (int mode) [inline]

17.11.3.34 `setObjectRubberPoint()` void setObjectRubberPoint (const QString & key, const QPointF & point) [inline]

17.11.3.35 `setObjectRubberText()` void setObjectRubberText (const QString & key, const QString & txt) [inline]

17.11.3.36 `setRect()` [1/2] void setRect (const QRectF & r) [inline]

17.11.3.37 `setRect()` [2/2] void setRect (EmbReal x, EmbReal y, EmbReal w, EmbReal h) [inline]

17.11.3.38 `shape()` virtual QPainterPath shape () const [inline], [virtual]

17.11.3.39 `type()` virtual int type () const [inline], [virtual]

Reimplemented in [ArcObject](#), [CircleObject](#), [DimLeaderObject](#), [EllipseObject](#), [ImageObject](#), [LineObject](#), [PathObject](#), [PointObject](#), [PolygonObject](#), [PolylineObject](#), [RectObject](#), and [TextSingleObject](#).

17.11.3.40 vulcanize() virtual void vulcanize () [pure virtual]

Implemented in [ArcObject](#), [CircleObject](#), [DimLeaderObject](#), [EllipseObject](#), [ImageObject](#), [LineObject](#), [PathObject](#), [PointObject](#), [PolygonObject](#), [PolylineObject](#), [RectObject](#), and [TextSingleObject](#).

17.11.4 Member Data Documentation

17.11.4.1 lwtPen QPen lwtPen

17.11.4.2 objID qint64 objID

17.11.4.3 objLine QLineF objLine

17.11.4.4 objPen QPen objPen

17.11.4.5 objRubberMode int objRubberMode

17.11.4.6 objRubberPoints QHash<QString, QPointF> objRubberPoints

17.11.4.7 objRubberTexts QHash<QString, QString> objRubberTexts

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/object-base.cpp](#)

17.12 CircleObject Class Reference

```
#include <embroidermodder.h>
```

Public Types

- enum { `Type` = `OBJ_TYPE_CIRCLE` }

Public Types inherited from `BaseObject`

- enum { `Type` = `OBJ_TYPE_BASE` }

Public Member Functions

- `CircleObject` (`EmbReal` `centerX`, `EmbReal` `centerY`, `EmbReal` `radius`, `QRgb` `rgb`, `QGraphicsItem` *`parent`=0)
- `CircleObject` (`CircleObject` *`obj`, `QGraphicsItem` *`parent`=0)
- `~CircleObject` ()
- void `init` (`EmbReal` `centerX`, `EmbReal` `centerY`, `EmbReal` `radius`, `QRgb` `rgb`, `Qt::PenStyle` `lineType`)
- void `updatePath` ()
- virtual int `type` () const
- `QPainterPath` `objectSavePath` () const
- `EmbReal` `objectRadius` () const
- `EmbReal` `objectDiameter` () const
- `EmbReal` `objectArea` () const
- `EmbReal` `objectCircumference` () const
- `QPointF` `objectQuadrant0` () const
- `QPointF` `objectQuadrant90` () const
- `QPointF` `objectQuadrant180` () const
- `QPointF` `objectQuadrant270` () const
- void `setObjectRadius` (`EmbReal` `radius`)
- void `setObjectDiameter` (`EmbReal` `diameter`)
- void `setObjectArea` (`EmbReal` `area`)
- void `setObjectCircumference` (`EmbReal` `circumference`)
- void `updateRubber` (`QPainter` *`painter`=0)
- virtual void `vulcanize` ()
- virtual `QPointF` `mouseSnapPoint` (const `QPointF` &`mousePoint`)
- virtual `QList<QPointF>` `allGripPoints` ()
- virtual void `gripEdit` (const `QPointF` &`before`, const `QPointF` &`after`)

Public Member Functions inherited from `BaseObject`

- `BaseObject` (`QGraphicsItem` *`parent`=0)
- virtual `~BaseObject` ()
- virtual int `type` () const
- `qint64` `objectID` () const
- `QPen` `objectPen` () const
- `QColor` `objectColor` () const
- `QRgb` `objectColorRGB` () const
- `Qt::PenStyle` `objectLineType` () const
- `EmbReal` `objectLineWidth` () const
- `QPainterPath` `objectPath` () const
- int `objectRubberMode` () const
- `QPointF` `objectRubberPoint` (const `QString` &`key`) const
- `QString` `objectRubberText` (const `QString` &`key`) const
- `QPointF` `objectCenter` () const

- `EmbReal objectCenterX () const`
- `EmbReal objectCenterY () const`
- `void setObjectCenter (EmbVector center)`
- `void setObjectCenterX (EmbReal centerX)`
- `void setObjectCenterY (EmbReal centerY)`
- `QRectF rect () const`
- `void setRect (const QRectF &r)`
- `void setRect (EmbReal x, EmbReal y, EmbReal w, EmbReal h)`
- `QLineF line () const`
- `void setLine (const QLineF &li)`
- `void setLine (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2)`
- `void setObjectColor (const QColor &color)`
- `void setObjectColorRGB (QRgb rgb)`
- `void setObjectLineType (Qt::PenStyle lineType)`
- `void setObjectLineWeight (EmbReal lineWeight)`
- `void setObjectPath (const QPainterPath &p)`
- `void setObjectRubberMode (int mode)`
- `void setObjectRubberPoint (const QString &key, const QPointF &point)`
- `void setObjectRubberText (const QString &key, const QString &txt)`
- `virtual QRectF boundingRect () const`
- `virtual QPainterPath shape () const`
- `void drawRubberLine (const QLineF &rubLine, QPainter *painter=0, const char *colorFromScene=0)`
- `virtual void vulcanize ()=0`
- `virtual QPointF mouseSnapPoint (const QPointF &mousePoint)=0`
- `virtual QList< QPointF > allGripPoints ()=0`
- `virtual void gripEdit (const QPointF &before, const QPointF &after)=0`

Protected Member Functions

- `void paint (QPainter *, const QStyleOptionGraphicsItem *, QWidget *)`

Protected Member Functions inherited from `BaseObject`

- `QPen lineWeightPen () const`
- `void realRender (QPainter *painter, const QPainterPath &renderPath)`

Additional Inherited Members

Public Attributes inherited from `BaseObject`

- `QPen objPen`
- `QPen lwtPen`
- `QLineF objLine`
- `int objRubberMode`
- `QHash< QString, QPointF > objRubberPoints`
- `QHash< QString, QString > objRubberTexts`
- `qint64 objID`

17.12.1 Member Enumeration Documentation

17.12.1.1 anonymous enum anonymous enum

Enumerator

| | |
|------|---------------------------------|
| Type | <input type="button" value=""/> |
|------|---------------------------------|

17.12.2 Constructor & Destructor Documentation

17.12.2.1 `CircleObject()` [1/2] `CircleObject` (

```
    EmbReal centerX,  
    EmbReal centerY,  
    EmbReal radius,  
    QRgb rgb,  
    QGraphicsItem * parent = 0 )
```

17.12.2.2 `CircleObject()` [2/2] `CircleObject` (

```
    CircleObject * obj,  
    QGraphicsItem * parent = 0 )
```

17.12.2.3 `~CircleObject()` `~CircleObject` ()

17.12.3 Member Function Documentation

17.12.3.1 `allGripPoints()` `QList< QPointF > allGripPoints ()` [virtual]

Implements [BaseObject](#).

17.12.3.2 `gripEdit()` `void gripEdit (` `const QPointF & before,` `const QPointF & after)` [virtual]

Implements [BaseObject](#).

17.12.3.3 init() void init (

```
EmbReal centerX,
EmbReal centerY,
EmbReal radius,
QRgb rgb,
Qt::PenStyle lineType )
```

17.12.3.4 mouseSnapPoint() QPointF mouseSnapPoint (

```
const QPointF & mousePoint ) [virtual]
```

Implements [BaseObject](#).

17.12.3.5 objectArea() EmbReal objectArea () const [inline]

17.12.3.6 objectCircumference() EmbReal objectCircumference () const [inline]

17.12.3.7 objectDiameter() EmbReal objectDiameter () const [inline]

17.12.3.8 objectQuadrant0() QPointF objectQuadrant0 () const [inline]

17.12.3.9 objectQuadrant180() QPointF objectQuadrant180 () const [inline]

17.12.3.10 objectQuadrant270() QPointF objectQuadrant270 () const [inline]

17.12.3.11 objectQuadrant90() QPointF objectQuadrant90 () const [inline]

17.12.3.12 objectRadius() EmbReal objectRadius () const [inline]

17.12.3.13 `objectSavePath()` `QPainterPath objectSavePath () const`

17.12.3.14 `paint()` `void paint (`
 `QPainter * painter,`
 `const QStyleOptionGraphicsItem * option,`
 `QWidget *) [protected]`

17.12.3.15 `setObjectArea()` `void setObjectArea (`
 `EmbReal area)`

17.12.3.16 `setObjectCircumference()` `void setObjectCircumference (`
 `EmbReal circumference)`

17.12.3.17 `setObjectDiameter()` `void setObjectDiameter (`
 `EmbReal diameter)`

17.12.3.18 `setObjectRadius()` `void setObjectRadius (`
 `EmbReal radius)`

17.12.3.19 `type()` `virtual int type () const [inline], [virtual]`

Reimplemented from [BaseObject](#).

17.12.3.20 `updatePath()` `void updatePath ()`

17.12.3.21 `updateRubber()` `void updateRubber (`
 `QPainter * painter = 0)`

17.12.3.22 vulcanize() void vulcanize () [virtual]

Implements [BaseObject](#).

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/object-circle.cpp](#)

17.13 CmdPrompt Class Reference

```
#include <embroidermodder.h>
```

Public Slots

- [QString getHistory \(\)](#)
- [QString getPrefix \(\)](#)
- [QString getCurrentText \(\)](#)
- [void setCurrentText \(const QString &txt\)](#)
- [void setHistory \(const QString &txt\)](#)
- [void setPrefix \(const QString &txt\)](#)
- [void appendHistory \(const QString &txt\)](#)
- [void startResizingTheHistory \(int y\)](#)
- [void stopResizingTheHistory \(int y\)](#)
- [void resizeTheHistory \(int y\)](#)
- [void addCommand \(const QString &alias, const QString &cmd\)](#)
- [void endCommand \(\)](#)
- [bool isCommandActive \(\)](#)
- [QString activeCommand \(\)](#)
- [QString lastCommand \(\)](#)
- [void processInput \(\)](#)
- [void enableRapidFire \(\)](#)
- [void disableRapidFire \(\)](#)
- [bool isRapidFireEnabled \(\)](#)
- [void alert \(const QString &txt\)](#)
- [void startBlinking \(\)](#)
- [void stopBlinking \(\)](#)
- [void blink \(\)](#)
- [void setPromptTextColor \(const QColor &\)](#)
- [void setPromptBackgroundColor \(const QColor &\)](#)
- [void setPromptFontFamily \(const QString &\)](#)
- [void setPromptFontStyle \(const QString &\)](#)
- [void setPromptFontSize \(int\)](#)
- [void floatingChanged \(bool\)](#)
- [void saveHistory \(const QString &fileName, bool html\)](#)

Signals

- void `appendTheHistory` (const QString &txt, int prefixLength)
- void `startCommand` (const QString &cmd)
- void `runCommand` (const QString &cmd, const QString &cmdtxt)
- void `deletePressed` ()
- void `tabPressed` ()
- void `escapePressed` ()
- void `upPressed` ()
- void `downPressed` ()
- void `F1Pressed` ()
- void `F2Pressed` ()
- void `F3Pressed` ()
- void `F4Pressed` ()
- void `F5Pressed` ()
- void `F6Pressed` ()
- void `F7Pressed` ()
- void `F8Pressed` ()
- void `F9Pressed` ()
- void `F10Pressed` ()
- void `F11Pressed` ()
- void `F12Pressed` ()
- void `cutPressed` ()
- void `copyPressed` ()
- void `pastePressed` ()
- void `selectAllPressed` ()
- void `undoPressed` ()
- void `redoPressed` ()
- void `shiftPressed` ()
- void `shiftReleased` ()
- void `showSettings` ()
- void `historyAppended` (const QString &txt)

Public Member Functions

- `CmdPrompt` (QWidget *parent=0)
- `~CmdPrompt` ()
- void `updateStyle` ()

Public Attributes

- `CmdPromptInput` * `promptInput`
- `CmdPromptHistory` * `promptHistory`
- `QVBoxLayout` * `promptVBoxLayout`
- `QFrame` * `promptDivider`
- `CmdPromptSplitter` * `promptSplitter`
- `QHash<QString, QString>` * `styleHash`
- `QTimer` * `blinkTimer`
- bool `blinkState`

17.13.1 Detailed Description**17.13.2 Constructor & Destructor Documentation****17.13.2.1 CmdPrompt()** `CmdPrompt (QWidget * parent = 0)`**17.13.2.2 ~CmdPrompt()** `~CmdPrompt ()`**17.13.3 Member Function Documentation****17.13.3.1 activeCommand** `QString activeCommand () [inline], [slot]`**17.13.3.2 addCommand** `void addCommand (const QString & alias, const QString & cmd) [inline], [slot]`**17.13.3.3 alert** `void alert (const QString & txt) [slot]`**17.13.3.4 appendHistory** `void appendHistory (const QString & txt) [slot]`**17.13.3.5 appendTheHistory** `void appendTheHistory (const QString & txt, int prefixLength) [signal]`**17.13.3.6 blink** `void blink () [slot]`

17.13.3.7 `copyPressed` void copyPressed () [signal]

17.13.3.8 `cutPressed` void cutPressed () [signal]

17.13.3.9 `deletePressed` void deletePressed () [signal]

17.13.3.10 `disableRapidFire` void disableRapidFire () [inline], [slot]

17.13.3.11 `downPressed` void downPressed () [signal]

17.13.3.12 `enableRapidFire` void enableRapidFire () [inline], [slot]

17.13.3.13 `endCommand` void endCommand () [inline], [slot]

17.13.3.14 `escapePressed` void escapePressed () [signal]

17.13.3.15 `F10Pressed` void F10Pressed () [signal]

17.13.3.16 `F11Pressed` void F11Pressed () [signal]

17.13.3.17 `F12Pressed` void F12Pressed () [signal]

17.13.3.18 F1Pressed void F1Pressed () [signal]

17.13.3.19 F2Pressed void F2Pressed () [signal]

17.13.3.20 F3Pressed void F3Pressed () [signal]

17.13.3.21 F4Pressed void F4Pressed () [signal]

17.13.3.22 F5Pressed void F5Pressed () [signal]

17.13.3.23 F6Pressed void F6Pressed () [signal]

17.13.3.24 F7Pressed void F7Pressed () [signal]

17.13.3.25 F8Pressed void F8Pressed () [signal]

17.13.3.26 F9Pressed void F9Pressed () [signal]

17.13.3.27 floatingChanged void floatingChanged (bool *isFloating*) [slot]

17.13.3.28 getCurrentText QString getCurrentText () [inline], [slot]

17.13.3.29 `getHistory` `QString getHistory () [inline], [slot]`

17.13.3.30 `getPrefix` `QString getPrefix () [inline], [slot]`

17.13.3.31 `historyAppended` `void historyAppended (const QString & txt) [signal]`

17.13.3.32 `isCommandActive` `bool isCommandActive () [inline], [slot]`

17.13.3.33 `isRapidFireEnabled` `bool isRapidFireEnabled () [inline], [slot]`

17.13.3.34 `lastCommand` `QString lastCommand () [inline], [slot]`

17.13.3.35 `pastePressed` `void pastePressed () [signal]`

17.13.3.36 `processInput` `void processInput () [inline], [slot]`

17.13.3.37 `redoPressed` `void redoPressed () [signal]`

17.13.3.38 `resizeTheHistory` `void resizeTheHistory (int y) [inline], [slot]`

17.13.3.39 `runCommand` `void runCommand (const QString & cmd, const QString & cmdtxt) [signal]`

17.13.3.40 `saveHistory` void saveHistory (const QString & *fileName*, bool *html*) [slot]

17.13.3.41 `selectAllPressed` void selectAllPressed () [signal]

17.13.3.42 `setCurrentText` void setCurrentText (const QString & *txt*) [inline], [slot]

17.13.3.43 `setHistory` void setHistory (const QString & *txt*) [inline], [slot]

17.13.3.44 `setPrefix` void setPrefix (const QString & *txt*) [slot]

17.13.3.45 `setPromptBackgroundColor` void setPromptBackgroundColor (const QColor & *color*) [slot]

17.13.3.46 `setPromptFontFamily` void setPromptFontFamily (const QString & *family*) [slot]

17.13.3.47 `setPromptFontSize` void setPromptFontSize (int *size*) [slot]

17.13.3.48 `setPromptFontStyle` void setPromptFontStyle (const QString & *style*) [slot]

17.13.3.49 `setPromptTextColor` void setPromptTextColor (const QColor & *color*) [slot]

17.13.3.50 shiftPressed void shiftPressed () [signal]

17.13.3.51 shiftReleased void shiftReleased () [signal]

17.13.3.52 showSettings void showSettings () [signal]

17.13.3.53 startBlinking void startBlinking () [slot]

17.13.3.54 startCommand void startCommand (const QString & cmd) [signal]

17.13.3.55 startResizingTheHistory void startResizingTheHistory (int y) [inline], [slot]

17.13.3.56 stopBlinking void stopBlinking () [slot]

17.13.3.57 stopResizingTheHistory void stopResizingTheHistory (int y) [inline], [slot]

17.13.3.58 tabPressed void tabPressed () [signal]

17.13.3.59 undoPressed void undoPressed () [signal]

17.13.3.60 updateStyle() void updateStyle ()

17.13.3.61 upPressed void upPressed () [signal]

17.13.4 Member Data Documentation

17.13.4.1 blinkState bool blinkState

17.13.4.2 blinkTimer QTimer* blinkTimer

17.13.4.3 promptDivider QFrame* promptDivider

17.13.4.4 promptHistory CmdPromptHistory* promptHistory

17.13.4.5 promptInput CmdPromptInput* promptInput

17.13.4.6 promptSplitter CmdPromptSplitter* promptSplitter

17.13.4.7 promptVBoxLayout QVBoxLayout* promptVBoxLayout

17.13.4.8 styleHash QHash<QString, QString>* styleHash

The documentation for this class was generated from the following files:

- embroidermodder2/[embroidermodder.h](#)
- embroidermodder2/[cmdprompt.cpp](#)

17.14 CmdPromptHandle Class Reference

```
#include <embroidermodder.h>
```

Signals

- void `handlePressed` (int y)
- void `handleReleased` (int y)
- void `handleMoved` (int y)

Public Member Functions

- `CmdPromptHandle` (Qt::Orientation orientation, QSplitter *parent)
- `~CmdPromptHandle` ()

Public Attributes

- int `pressY`
- int `releaseY`
- int `moveY`

Protected Member Functions

- void `mousePressEvent` (QMouseEvent *e)
- void `mouseReleaseEvent` (QMouseEvent *e)
- void `mouseMoveEvent` (QMouseEvent *e)

17.14.1 Detailed Description

17.14.2 Constructor & Destructor Documentation

17.14.2.1 `CmdPromptHandle()` `CmdPromptHandle` (
 Qt::Orientation *orientation*,
 QSplitter * *parent*)

17.14.2.2 `~CmdPromptHandle()` `~CmdPromptHandle` ()

17.14.3 Member Function Documentation

17.14.3.1 `handleMoved` `void handleMoved` (
 int *y*) [signal]

17.14.3.2 handlePressed void handlePressed (int y) [signal]

17.14.3.3 handleReleased void handleReleased (int y) [signal]

17.14.3.4 mouseMoveEvent() void mouseMoveEvent (QMouseEvent * e) [protected]

17.14.3.5 mousePressEvent() void mousePressEvent (QMouseEvent * e) [protected]

17.14.3.6 mouseReleaseEvent() void mouseReleaseEvent (QMouseEvent * e) [protected]

17.14.4 Member Data Documentation

17.14.4.1 moveY int moveY

17.14.4.2 pressY int pressY

17.14.4.3 releaseY int releaseY

The documentation for this class was generated from the following files:

- embroidermodder2/[embroidermodder.h](#)
- embroidermodder2/[cmdprompt.cpp](#)

17.15 CmdPromptHistory Class Reference

The Command Prompt History class.

```
#include <embroidermodder.h>
```

Public Slots

- void [appendHistory](#) (const QString &txt, int prefixLength)
- void [startResizeHistory](#) (int y)
- void [stopResizeHistory](#) (int y)
- void [resizeHistory](#) (int y)

Signals

- void [historyAppended](#) (const QString &txt)

Public Member Functions

- [CmdPromptHistory](#) (QWidget *parent=0)
- [~CmdPromptHistory](#) ()
- QString [applyFormatting](#) (const QString &txt, int prefixLength)

Public Attributes

- int [tmpHeight](#)

Protected Member Functions

- void [contextMenuEvent](#) (QContextMenuEvent *event)

17.15.1 Detailed Description

The Command Prompt History class.

17.15.2 Constructor & Destructor Documentation

17.15.2.1 [CmdPromptHistory\(\)](#) [CmdPromptHistory](#) (

```
QWidget * parent = 0 )
```

17.15.2.2 [~CmdPromptHistory\(\)](#) [~CmdPromptHistory](#) ()

17.15.3 Member Function Documentation

17.15.3.1 appendHistory void appendHistory (const QString & txt, int prefixLength) [slot]

17.15.3.2 applyFormatting() QString applyFormatting (const QString & txt, int prefixLength)

17.15.3.3 contextMenuEvent() void contextMenuEvent (QContextMenuEvent * event) [protected]

17.15.3.4 historyAppended void historyAppended (const QString & txt) [signal]

17.15.3.5 resizeHistory void resizeHistory (int y) [slot]

17.15.3.6 startResizeHistory void startResizeHistory (int y) [slot]

17.15.3.7 stopResizeHistory void stopResizeHistory (int y) [slot]

17.15.4 Member Data Documentation

17.15.4.1 tmpHeight int tmpHeight

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/cmdprompt.cpp](#)

17.16 CmdPromptInput Class Reference

```
#include <embroidermodder.h>
```

Public Slots

- void `addCommand` (const QString &alias, const QString &cmd)
- void `endCommand` ()
- void `processInput` (void)
- void `checkSelection` ()
- void `updateCurrentText` (const QString &txt)
- void `checkEditedText` (const QString &txt)
- void `checkChangedText` (const QString &txt)
- void `checkCursorPosition` (int oldpos, int newpos)

Signals

- void `appendHistory` (const QString &txt, int prefixLength)
- void `startCommand` (const QString &cmd)
- void `runCommand` (const QString &cmd, const QString &cmdtxt)
- void `deletePressed` ()
- void `tabPressed` ()
- void `escapePressed` ()
- void `upPressed` ()
- void `downPressed` ()
- void `F1Pressed` ()
- void `F2Pressed` ()
- void `F3Pressed` ()
- void `F4Pressed` ()
- void `F5Pressed` ()
- void `F6Pressed` ()
- void `F7Pressed` ()
- void `F8Pressed` ()
- void `F9Pressed` ()
- void `F10Pressed` ()
- void `F11Pressed` ()
- void `F12Pressed` ()
- void `cutPressed` ()
- void `copyPressed` ()
- void `pastePressed` ()
- void `selectAllPressed` ()
- void `undoPressed` ()
- void `redoPressed` ()
- void `shiftPressed` ()
- void `shiftReleased` ()
- void `showSettings` ()
- void `stopBlinking` ()

Public Member Functions

- `CmdPromptInput (QWidget *parent=0)`
- `~CmdPromptInput ()`
- `void changeFormatting (const QList< QTextLayout::FormatRange > &formats)`
- `void clearFormatting ()`
- `void applyFormatting ()`

Public Attributes

- `QString curText`
- `QString defaultPrefix`
- `QString prefix`
- `QString lastCmd`
- `QString curCmd`
- `bool cmdActive`
- `bool rapidFireEnabled`
- `bool isBlinking`
- `QHash< QString, QString > * aliasHash`

Protected Member Functions

- `void contextMenuEvent (QContextMenuEvent *event)`
- `bool eventFilter (QObject *obj, QEvent *event)`

Private Slots

- `void copyClip ()`
- `void pasteClip ()`

17.16.1 Constructor & Destructor Documentation

17.16.1.1 CmdPromptInput() `CmdPromptInput (`
`QWidget * parent = 0)`

17.16.1.2 ~CmdPromptInput() `~CmdPromptInput ()`

17.16.2 Member Function Documentation

17.16.2.1 addCommand void addCommand (const QString & alias, const QString & cmd) [slot]

17.16.2.2 appendHistory void appendHistory (const QString & txt, int prefixLength) [signal]

17.16.2.3 applyFormatting() void applyFormatting ()

17.16.2.4 changeFormatting() void changeFormatting (const QList< QTextLayout::FormatRange > & formats)

17.16.2.5 checkChangedText void checkChangedText (const QString & txt) [slot]

17.16.2.6 checkCursorPosition void checkCursorPosition (int oldpos, int newpos) [slot]

17.16.2.7 checkEditedText void checkEditedText (const QString & txt) [slot]

17.16.2.8 checkSelection void checkSelection () [slot]

17.16.2.9 clearFormatting() void clearFormatting ()

17.16.2.10 contextMenuEvent() void contextMenuEvent (QContextMenuEvent * event) [protected]

17.16.2.11 copyClip void copyClip () [private], [slot]

17.16.2.12 copyPressed void copyPressed () [signal]

17.16.2.13 cutPressed void cutPressed () [signal]

17.16.2.14 deletePressed void deletePressed () [signal]

17.16.2.15 downPressed void downPressed () [signal]

17.16.2.16 endCommand void endCommand () [slot]

17.16.2.17 escapePressed void escapePressed () [signal]

17.16.2.18 eventFilter() bool eventFilter (QObject * obj, QEEvent * event) [protected]

17.16.2.19 F10Pressed void F10Pressed () [signal]

17.16.2.20 F11Pressed void F11Pressed () [signal]

17.16.2.21 F12Pressed void F12Pressed () [signal]

17.16.2.22 F1Pressed void F1Pressed () [signal]

17.16.2.23 F2Pressed void F2Pressed () [signal]

17.16.2.24 F3Pressed void F3Pressed () [signal]

17.16.2.25 F4Pressed void F4Pressed () [signal]

17.16.2.26 F5Pressed void F5Pressed () [signal]

17.16.2.27 F6Pressed void F6Pressed () [signal]

17.16.2.28 F7Pressed void F7Pressed () [signal]

17.16.2.29 F8Pressed void F8Pressed () [signal]

17.16.2.30 F9Pressed void F9Pressed () [signal]

17.16.2.31 pasteClip void pasteClip () [private], [slot]

17.16.2.32 pastePressed void pastePressed () [signal]

17.16.2.33 processInput void processInput (void) [slot]

17.16.2.34 redoPressed void redoPressed () [signal]

17.16.2.35 runCommand void runCommand (const QString & cmd, const QString & cmdtxt) [signal]

17.16.2.36 selectAllPressed void selectAllPressed () [signal]

17.16.2.37 shiftPressed void shiftPressed () [signal]

17.16.2.38 shiftReleased void shiftReleased () [signal]

17.16.2.39 showSettings void showSettings () [signal]

17.16.2.40 startCommand void startCommand (const QString & cmd) [signal]

17.16.2.41 stopBlinking void stopBlinking () [signal]

17.16.2.42 tabPressed void tabPressed () [signal]

17.16.2.43 undoPressed void undoPressed () [signal]

17.16.2.44 updateCurrentText void updateCurrentText (const QString & txt) [slot]

17.16.2.45 upPressed void upPressed () [signal]

17.16.3 Member Data Documentation

17.16.3.1 aliasHash QHash<QString, QString>* aliasHash

17.16.3.2 cmdActive bool cmdActive

17.16.3.3 curCmd QString curCmd

17.16.3.4 curText QString curText

17.16.3.5 defaultPrefix QString defaultPrefix

17.16.3.6 isBlinking bool isBlinking

17.16.3.7 lastCmd QString lastCmd

17.16.3.8 prefix QString prefix

17.16.3.9 rapidFireEnabled bool rapidFireEnabled

The documentation for this class was generated from the following files:

- embroiderymodder2/[embroidermodder.h](#)
- embroiderymodder2/[cmdprompt.cpp](#)

17.17 CmdPromptSplitter Class Reference

```
#include <embroidermodder.h>
```

Signals

- void [pressResizeHistory](#) (int y)
- void [releaseResizeHistory](#) (int y)
- void [moveResizeHistory](#) (int y)

Public Member Functions

- [CmdPromptSplitter](#) (QWidget *parent=0)
- [~CmdPromptSplitter](#) ()

Protected Member Functions

- [QSplitterHandle * createHandle](#) ()

17.17.1 Detailed Description

17.17.2 Constructor & Destructor Documentation

17.17.2.1 CmdPromptSplitter() [CmdPromptSplitter](#) (QWidget * parent = 0)**17.17.2.2 ~CmdPromptSplitter()** [~CmdPromptSplitter](#) ()

17.17.3 Member Function Documentation

17.17.3.1 `createHandle()` `QSplitterHandle * createHandle () [protected]`

17.17.3.2 `moveResizeHistory()` `void moveResizeHistory (int y) [signal]`

17.17.3.3 `pressResizeHistory()` `void pressResizeHistory (int y) [signal]`

17.17.3.4 `releaseResizeHistory()` `void releaseResizeHistory (int y) [signal]`

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/cmdprompt.cpp](#)

17.18 Compress Struct Reference

```
#include <embroidery_internal.h>
```

Public Attributes

- `int bit_position`
- `char * input_data`
- `int input_length`
- `int bits_total`
- `int block_elements`
- `huffman character_length_huffman`
- `huffman character_huffman`
- `huffman distance_huffman`

17.18.1 Member Data Documentation

17.18.1.1 `bit_position` `int bit_position`

17.18.1.2 bits_total int bits_total

17.18.1.3 block_elements int block_elements

17.18.1.4 character_huffman [huffman](#) character_huffman

17.18.1.5 character_length_huffman [huffman](#) character_length_huffman

17.18.1.6 distance_huffman [huffman](#) distance_huffman

17.18.1.7 input_data char* input_data

17.18.1.8 input_length int input_length

The documentation for this struct was generated from the following file:

- [extern/libembroidery/src/embroidery_internal.h](#)

17.19 DimLeaderObject Class Reference

```
#include <embroidermodder.h>
```

Public Types

- enum [ArrowStyle](#) {
 [NoArrow](#) , [Open](#) , [Closed](#) , [Dot](#) ,
 [Box](#) , [Tick](#) }
- enum [lineStyle](#) { [NoLine](#) , [Flared](#) , [Fletching](#) }
- enum { [Type](#) = [OBJ_TYPE_DIMLEADER](#) }

Public Types inherited from [BaseObject](#)

- enum { [Type](#) = [OBJ_TYPE_BASE](#) }

Public Member Functions

- `DimLeaderObject (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, QRgb rgb, QGraphicsItem *parent=0)`
- `DimLeaderObject (DimLeaderObject *obj, QGraphicsItem *parent=0)`
- `~DimLeaderObject ()`
- `void init (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, QRgb rgb, Qt::PenStyle lineType)`
- `void updateLeader ()`
- `virtual int type () const`
- `QPointF objectEndPoint1 () const`
- `QPointF objectEndPoint2 () const`
- `QPointF objectMidPoint () const`
- `EmbReal objectX1 () const`
- `EmbReal objectY1 () const`
- `EmbReal objectX2 () const`
- `EmbReal objectY2 () const`
- `EmbReal objectDeltaX () const`
- `EmbReal objectDeltaY () const`
- `EmbReal objectAngle () const`
- `EmbReal objectLength () const`
- `void setObjectEndPoint1 (const QPointF &endPt1)`
- `void setObjectEndPoint1 (EmbReal x1, EmbReal y1)`
- `void setObjectEndPoint2 (const QPointF &endPt2)`
- `void setObjectEndPoint2 (EmbReal x2, EmbReal y2)`
- `void setObjectX1 (EmbReal x)`
- `void setObjectY1 (EmbReal y)`
- `void setObjectX2 (EmbReal x)`
- `void setObjectY2 (EmbReal y)`
- `void updateRubber (QPainter *painter=0)`
- `virtual void vulcanize ()`
- `virtual QPointF mouseSnapPoint (const QPointF &mousePoint)`
- `virtual QList< QPointF > allGripPoints ()`
- `virtual void gripEdit (const QPointF &before, const QPointF &after)`

Public Member Functions inherited from `BaseObject`

- `BaseObject (QGraphicsItem *parent=0)`
- `virtual ~BaseObject ()`
- `virtual int type () const`
- `qint64 objectID () const`
- `QPen objectPen () const`
- `QColor objectColor () const`
- `QRgb objectColorRGB () const`
- `Qt::PenStyle objectLineType () const`
- `EmbReal objectLineWidth () const`
- `QPainterPath objectPath () const`
- `int objectRubberMode () const`
- `QPointF objectRubberPoint (const QString &key) const`
- `QString objectRubberText (const QString &key) const`
- `QPointF objectCenter () const`
- `EmbReal objectCenterX () const`
- `EmbReal objectCenterY () const`
- `void setObjectCenter (EmbVector center)`

- void `setObjectCenterX` (`EmbReal` centerX)
- void `setObjectCenterY` (`EmbReal` centerY)
- `QRectF rect` () const
- void `setRect` (const `QRectF` &r)
- void `setRect` (`EmbReal` x, `EmbReal` y, `EmbReal` w, `EmbReal` h)
- `QLineF line` () const
- void `setLine` (const `QLineF` &li)
- void `setLine` (`EmbReal` x1, `EmbReal` y1, `EmbReal` x2, `EmbReal` y2)
- void `setObjectColor` (const `QColor` &color)
- void `setObjectColorRGB` (`QRgb` rgb)
- void `setObjectLineType` (`Qt::PenStyle` lineType)
- void `setObjectLineWeight` (`EmbReal` lineWeight)
- void `setObjectPath` (const `QPainterPath` &p)
- void `setObjectRubberMode` (int mode)
- void `setObjectRubberPoint` (const `QString` &key, const `QPointF` &point)
- void `setObjectRubberText` (const `QString` &key, const `QString` &txt)
- virtual `QRectF boundingRect` () const
- virtual `QPainterPath shape` () const
- void `drawRubberLine` (const `QLineF` &rubLine, `QPainter` *painter=0, const `char` *colorFromScene=0)
- virtual void `vulcanize` ()=0
- virtual `QPointF mouseSnapPoint` (const `QPointF` &mousePoint)=0
- virtual `QList<QPointF>` `allGripPoints` ()=0
- virtual void `gripEdit` (const `QPointF` &before, const `QPointF` &after)=0

Public Attributes

- bool `curved`
- bool `filled`
- `QPainterPath` `lineStylePath`
- `QPainterPath` `arrowStylePath`
- `EmbReal` `arrowStyleAngle`
- `EmbReal` `arrowStyleLength`
- `EmbReal` `lineStyleAngle`
- `EmbReal` `lineStyleLength`

Public Attributes inherited from `BaseObject`

- `QPen` `objPen`
- `QPen` `lwtPen`
- `QLineF` `objLine`
- int `objRubberMode`
- `QHash<QString, QPointF>` `objRubberPoints`
- `QHash<QString, QString>` `objRubberTexts`
- `qint64` `objID`

Protected Member Functions

- void `paint` (`QPainter` *, const `QStyleOptionGraphicsItem` *, `QWidget` *)

Protected Member Functions inherited from [BaseObject](#)

- QPen [lineWeightPen \(\) const](#)
- void [realRender \(QPainter *painter, const QPainterPath &renderPath\)](#)

17.19.1 Member Enumeration Documentation**17.19.1.1 anonymous enum [anonymous enum](#)**

Enumerator

| | |
|------|---------------------------------|
| Type | <input type="button" value=""/> |
|------|---------------------------------|

17.19.1.2 ArrowStyle [enum ArrowStyle](#)

Enumerator

| | |
|---------|---------------------------------|
| NoArrow | <input type="button" value=""/> |
| Open | <input type="button" value=""/> |
| Closed | <input type="button" value=""/> |
| Dot | <input type="button" value=""/> |
| Box | <input type="button" value=""/> |
| Tick | <input type="button" value=""/> |

17.19.1.3 lineStyle [enum lineStyle](#)

Enumerator

| | |
|-----------|---------------------------------|
| NoLine | <input type="button" value=""/> |
| Flared | <input type="button" value=""/> |
| Fletching | <input type="button" value=""/> |

17.19.2 Constructor & Destructor Documentation**17.19.2.1 DimLeaderObject() [1/2] [DimLeaderObject](#) (**
EmReal x1,

```
    EmbReal y1,
    EmbReal x2,
    EmbReal y2,
    QRgb rgb,
    QGraphicsItem * parent = 0 )
```

17.19.2.2 DimLeaderObject() [2/2] `DimLeaderObject (`
 `DimLeaderObject * obj,`
 `QGraphicsItem * parent = 0)`

17.19.2.3 ~DimLeaderObject() `~DimLeaderObject ()`

17.19.3 Member Function Documentation

17.19.3.1 allGripPoints() `QList< QPointF > allGripPoints () [virtual]`

Implements [BaseObject](#).

17.19.3.2 gripEdit() `void gripEdit (`
 `const QPointF & before,`
 `const QPointF & after) [virtual]`

Implements [BaseObject](#).

17.19.3.3 init() `void init (`
 `EmbReal x1,`
 `EmbReal y1,`
 `EmbReal x2,`
 `EmbReal y2,`
 `QRgb rgb,`
 `Qt::PenStyle lineType)`

17.19.3.4 mouseSnapPoint() `QPointF mouseSnapPoint (`
 `const QPointF & mousePoint) [virtual]`

Implements [BaseObject](#).

17.19.3.5 objectAngle() `EmbReal` `objectAngle () const`

17.19.3.6 objectDeltaX() `EmbReal` `objectDeltaX () const [inline]`

17.19.3.7 objectDeltaY() `EmbReal` `objectDeltaY () const [inline]`

17.19.3.8 objectEndPoint1() `QPointF` `objectEndPoint1 () const`

17.19.3.9 objectEndPoint2() `QPointF` `objectEndPoint2 () const`

17.19.3.10 objectLength() `EmbReal` `objectLength () const [inline]`

17.19.3.11 objectMidPoint() `QPointF` `objectMidPoint () const`

17.19.3.12 objectX1() `EmbReal` `objectX1 () const [inline]`

17.19.3.13 objectX2() `EmbReal` `objectX2 () const [inline]`

17.19.3.14 objectY1() `EmbReal` `objectY1 () const [inline]`

17.19.3.15 objectY2() `EmbReal` `objectY2 () const [inline]`

17.19.3.16 `paint()` void paint (

```
QPainter * painter,
const QStyleOptionGraphicsItem * option,
QWidget * ) [protected]
```

17.19.3.17 `setObjectEndPoint1()` [1/2] void setObjectEndPoint1 (

```
const QPointF & endPt1 )
```

17.19.3.18 `setObjectEndPoint1()` [2/2] void setObjectEndPoint1 (

```
EmbReal x1,
EmbReal y1 )
```

17.19.3.19 `setObjectEndPoint2()` [1/2] void setObjectEndPoint2 (

```
const QPointF & endPt2 )
```

17.19.3.20 `setObjectEndPoint2()` [2/2] void setObjectEndPoint2 (

```
EmbReal x2,
EmbReal y2 )
```

17.19.3.21 `setObjectX1()` void setObjectX1 (

```
EmbReal x ) [inline]
```

17.19.3.22 `setObjectX2()` void setObjectX2 (

```
EmbReal x ) [inline]
```

17.19.3.23 `setObjectY1()` void setObjectY1 (

```
EmbReal y ) [inline]
```

17.19.3.24 `setObjectY2()` void setObjectY2 (

```
EmbReal y ) [inline]
```

17.19.3.25 `type()` `virtual int type () const [inline], [virtual]`

Reimplemented from [BaseObject](#).

17.19.3.26 `updateLeader()` `void updateLeader ()`

17.19.3.27 `updateRubber()` `void updateRubber (`
`QPainter * painter = 0)`

17.19.3.28 `vulcanize()` `void vulcanize () [virtual]`

Implements [BaseObject](#).

17.19.4 Member Data Documentation

17.19.4.1 `arrowStyleAngle` [EmbReal](#) `arrowStyleAngle`

17.19.4.2 `arrowStyleLength` [EmbReal](#) `arrowStyleLength`

17.19.4.3 `arrowStylePath` `QPainterPath arrowStylePath`

17.19.4.4 `curved` `bool curved`

17.19.4.5 `filled` `bool filled`

17.19.4.6 `lineStyleAngle` [EmbReal](#) `lineStyleAngle`

17.19.4.7 `lineStyleLength` `EmbReal` `lineStyleLength`

17.19.4.8 `lineStylePath` `QPainterPath` `lineStylePath`

The documentation for this class was generated from the following files:

- `embroidermodder2/embroidermodder.h`
- `embroidermodder2/object-dimleader.cpp`

17.20 EllipseObject Class Reference

```
#include <embroidermodder.h>
```

Public Types

- enum { `Type` = `OBJ_TYPE_ELLIPSE` }

Public Types inherited from `BaseObject`

- enum { `Type` = `OBJ_TYPE_BASE` }

Public Member Functions

- `EllipseObject` (`EmbReal` `centerX`, `EmbReal` `centerY`, `EmbReal` `width`, `EmbReal` `height`, `QRgb` `rgb`, `QGraphicsItem` *`parent`=0)
- `EllipseObject` (`EllipseObject` *`obj`, `QGraphicsItem` *`parent`=0)
- `~EllipseObject` ()
- void `init` (`EmbReal` `centerX`, `EmbReal` `centerY`, `EmbReal` `width`, `EmbReal` `height`, `QRgb` `rgb`, `Qt::PenStyle` `lineType`)
- void `updatePath` ()
- virtual int `type` () const
- `QPainterPath` `objectSavePath` () const
- `EmbReal` `objectRadiusMajor` () const
- `EmbReal` `objectRadiusMinor` () const
- `EmbReal` `objectDiameterMajor` () const
- `EmbReal` `objectDiameterMinor` () const
- `EmbReal` `objectWidth` () const
- `EmbReal` `objectHeight` () const
- `QPointF` `objectQuadrant0` () const
- `QPointF` `objectQuadrant90` () const
- `QPointF` `objectQuadrant180` () const
- `QPointF` `objectQuadrant270` () const
- void `setObjectSize` (`EmbReal` `width`, `EmbReal` `height`)
- void `setObjectRadiusMajor` (`EmbReal` `radius`)
- void `setObjectRadiusMinor` (`EmbReal` `radius`)
- void `setObjectDiameterMajor` (`EmbReal` `diameter`)
- void `setObjectDiameterMinor` (`EmbReal` `diameter`)
- void `updateRubber` (`QPainter` *`painter`=0)
- virtual void `vulcanize` ()
- virtual `QPointF` `mouseSnapPoint` (const `QPointF` &`mousePoint`)
Returns the closest snap point to the mouse point.
- virtual `QList<QPointF>` `allGripPoints` ()
- virtual void `gripEdit` (const `QPointF` &`before`, const `QPointF` &`after`)

Public Member Functions inherited from [BaseObject](#)

- [BaseObject](#) (QGraphicsItem *parent=0)
- virtual [~BaseObject](#) ()
- virtual int [type](#) () const
- qint64 [objectID](#) () const
- QPen [objectPen](#) () const
- QColor [objectColor](#) () const
- QRgb [objectColorRGB](#) () const
- Qt::PenStyle [objectLineType](#) () const
- [EmbReal objectLineWidth](#) () const
- QPainterPath [objectPath](#) () const
- int [objectRubberMode](#) () const
- QPointF [objectRubberPoint](#) (const QString &key) const
- QString [objectRubberText](#) (const QString &key) const
- QPointF [objectCenter](#) () const
- [EmbReal objectCenterX](#) () const
- [EmbReal objectCenterY](#) () const
- void [setObjectCenter](#) ([EmbVector](#) center)
- void [setObjectCenterX](#) ([EmbReal](#) centerX)
- void [setObjectCenterY](#) ([EmbReal](#) centerY)
- QRectF [rect](#) () const
- void [setRect](#) (const QRectF &r)
- void [setRect](#) ([EmbReal](#) x, [EmbReal](#) y, [EmbReal](#) w, [EmbReal](#) h)
- QLineF [line](#) () const
- void [setLine](#) (const QLineF &l)
- void [setLine](#) ([EmbReal](#) x1, [EmbReal](#) y1, [EmbReal](#) x2, [EmbReal](#) y2)
- void [setObjectColor](#) (const QColor &color)
- void [setObjectColorRGB](#) (QRgb rgb)
- void [setObjectLineType](#) (Qt::PenStyle lineType)
- void [setObjectLineWidth](#) ([EmbReal](#) lineWeight)
- void [setObjectPath](#) (const QPainterPath &p)
- void [setObjectRubberMode](#) (int mode)
- void [setObjectRubberPoint](#) (const QString &key, const QPointF &point)
- void [setObjectRubberText](#) (const QString &key, const QString &txt)
- virtual QRectF [boundingRect](#) () const
- virtual QPainterPath [shape](#) () const
- void [drawRubberLine](#) (const QLineF &rubLine, QPainter *painter=0, const char *colorFromScene=0)
- virtual void [vulcanize](#) ()=0
- virtual QPointF [mouseSnapPoint](#) (const QPointF &mousePoint)=0
- virtual QList< QPointF > [allGripPoints](#) ()=0
- virtual void [gripEdit](#) (const QPointF &before, const QPointF &after)=0

Protected Member Functions

- void [paint](#) (Painter *, const QStyleOptionGraphicsItem *, QWidget *)

Protected Member Functions inherited from [BaseObject](#)

- QPen [lineWeightPen](#) () const
- void [realRender](#) (Painter *painter, const QPainterPath &renderPath)

Additional Inherited Members

Public Attributes inherited from [BaseObject](#)

- QPen [objPen](#)
- QPen [lwtPen](#)
- QLineF [objLine](#)
- int [objRubberMode](#)
- QHash<QString, QPointF> [objRubberPoints](#)
- QHash<QString, QString> [objRubberTexts](#)
- qint64 [objID](#)

17.20.1 Member Enumeration Documentation

17.20.1.1 anonymous enum [anonymous enum](#)

Enumerator

| | |
|------|---------------------------------|
| Type | <input type="button" value=""/> |
|------|---------------------------------|

17.20.2 Constructor & Destructor Documentation

17.20.2.1 [EllipseObject\(\)](#) [1/2] [EllipseObject](#) (

```
    EmbReal centerX,
    EmbReal centerY,
    EmbReal width,
    EmbReal height,
    QRgb rgb,
    QGraphicsItem * parent = 0 )
```

17.20.2.2 [EllipseObject\(\)](#) [2/2] [EllipseObject](#) (

```
    EllipseObject * obj,
    QGraphicsItem * parent = 0 )
```

17.20.2.3 ~[EllipseObject\(\)](#) ~[EllipseObject](#) ()

17.20.3 Member Function Documentation

17.20.3.1 allGripPoints() `QList< QPointF > allGripPoints () [virtual]`

Implements [BaseObject](#).

17.20.3.2 gripEdit() `void gripEdit (const QPointF & before, const QPointF & after) [virtual]`

Todo `gripEdit()` for [EllipseObject](#)

Implements [BaseObject](#).

17.20.3.3 init() `void init (EmbReal centerX, EmbReal centerY, EmbReal width, EmbReal height, QRgb rgb, Qt::PenStyle lineType)`

Warning

DO NOT enable `QGraphicsItem::ItemIsMovable`. If it is enabled, and the item is double clicked, the scene will erratically move the item while zooming. All movement has to be handled explicitly by us, not by the scene.

17.20.3.4 mouseSnapPoint() `QPointF mouseSnapPoint (const QPointF & mousePoint) [virtual]`

Returns the closest snap point to the mouse point.

Implements [BaseObject](#).

17.20.3.5 objectDiameterMajor() `EmbReal objectDiameterMajor () const [inline]`

17.20.3.6 objectDiameterMinor() `EmbReal objectDiameterMinor () const [inline]`

17.20.3.7 objectHeight() `EmbReal objectHeight () const [inline]`

17.20.3.8 objectQuadrant0() `QPointF objectQuadrant0 () const`

17.20.3.9 objectQuadrant180() `QPointF objectQuadrant180 () const`

17.20.3.10 objectQuadrant270() `QPointF objectQuadrant270 () const`

17.20.3.11 objectQuadrant90() `QPointF objectQuadrant90 () const`

17.20.3.12 objectRadiusMajor() `EmbReal objectRadiusMajor () const [inline]`

17.20.3.13 objectRadiusMinor() `EmbReal objectRadiusMinor () const [inline]`

17.20.3.14 objectSavePath() `QPainterPath objectSavePath () const`

17.20.3.15 objectWidth() `EmbReal objectWidth () const [inline]`

17.20.3.16 paint() `void paint (`
 `QPainter * painter,`
 `const QStyleOptionGraphicsItem * option,`
 `QWidget *) [protected]`

17.20.3.17 `setObjectDiameterMajor()` void setObjectDiameterMajor (EmbReal diameter)

17.20.3.18 `setObjectDiameterMinor()` void setObjectDiameterMinor (EmbReal diameter)

17.20.3.19 `setObjectRadiusMajor()` void setObjectRadiusMajor (EmbReal radius)

17.20.3.20 `setObjectRadiusMinor()` void setObjectRadiusMinor (EmbReal radius)

17.20.3.21 `setObjectSize()` void setObjectSize (EmbReal width, EmbReal height)

17.20.3.22 `type()` virtual int type () const [inline], [virtual]

Reimplemented from [BaseObject](#).

17.20.3.23 `updatePath()` void updatePath ()

17.20.3.24 `updateRubber()` void updateRubber (QPainter * painter = 0)

17.20.3.25 `vulcanize()` void vulcanize () [virtual]

Implements [BaseObject](#).

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/object-ellipse.cpp](#)

17.21 EmbAlignedDim_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- EmbVector position

17.21.1 Member Data Documentation

17.21.1.1 position EmbVector position

The documentation for this struct was generated from the following file:

- extern/libembroidery/src/embroidery.h

17.22 EmbAngularDim_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- EmbVector position

17.22.1 Member Data Documentation

17.22.1.1 position EmbVector position

The documentation for this struct was generated from the following file:

- extern/libembroidery/src/embroidery.h

17.23 EmbArc_ Struct Reference

absolute position (not relative)

```
#include <embroidery.h>
```

Public Attributes

- EmbVector start
- EmbVector mid
- EmbVector end

17.23.1 Detailed Description

absolute position (not relative)

17.23.2 Member Data Documentation**17.23.2.1 end EmbVector end****17.23.2.2 mid EmbVector mid****17.23.2.3 start EmbVector start**

The documentation for this struct was generated from the following file:

- extern/libembroidery/src/embroidery.h

17.24 EmbArcLengthDim_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- EmbVector position

17.24.1 Member Data Documentation**17.24.1.1 position EmbVector position**

The documentation for this struct was generated from the following file:

- extern/libembroidery/src/embroidery.h

17.25 EmbArray_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- `EmbGeometry * geometry`
- `EmbStitch * stitch`
- `EmbThread * thread`
- `int count`
- `int length`
- `int type`

17.25.1 Member Data Documentation

17.25.1.1 count `int count`

17.25.1.2 geometry `EmbGeometry* geometry`

17.25.1.3 length `int length`

17.25.1.4 stitch `EmbStitch* stitch`

17.25.1.5 thread `EmbThread* thread`

17.25.1.6 type `int type`

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery.h`

17.26 EmbBezier_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- EmbVector start
- EmbVector control1
- EmbVector control2
- EmbVector end

17.26.1 Member Data Documentation

17.26.1.1 control1 EmbVector control1

17.26.1.2 control2 EmbVector control2

17.26.1.3 end EmbVector end

17.26.1.4 start EmbVector start

The documentation for this struct was generated from the following file:

- extern/libembroidery/src/embroidery.h

17.27 EmbBlock_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- EmbVector position

17.27.1 Member Data Documentation

17.27.1.1 position [EmbVector](#) position

The documentation for this struct was generated from the following file:

- [extern/libembroidery/src/embroidery.h](#)

17.28 EmbCircle_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- [EmbVector center](#)
- [EmbReal radius](#)

17.28.1 Member Data Documentation

17.28.1.1 center [EmbVector](#) center

17.28.1.2 radius [EmbReal](#) radius

The documentation for this struct was generated from the following file:

- [extern/libembroidery/src/embroidery.h](#)

17.29 EmbColor_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- [unsigned char r](#)
- [unsigned char g](#)
- [unsigned char b](#)

17.29.1 Detailed Description

EmbColor uses the light primaries: red, green, blue in that order.

17.29.2 Member Data Documentation

17.29.2.1 b unsigned char b

17.29.2.2 g unsigned char g

17.29.2.3 r unsigned char r

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery.h`

17.30 EmbDetailsDialog Class Reference

```
#include <embroidermodder.h>
```

Public Member Functions

- `EmbDetailsDialog (QGraphicsScene *theScene, QWidget *parent=0)`
- `~EmbDetailsDialog ()`
- `void getInfo ()`
- `QWidget * createMainWidget ()`
- `QWidget * createHistogram ()`

Public Attributes

- `QWidget * mainWidget`
- `QDialogButtonBox * buttonBox`
- `quint32 stitchesTotal`
- `quint32 stitchesReal`
- `quint32 stitchesJump`
- `quint32 stitchesTrim`
- `quint32 colorTotal`
- `quint32 colorChanges`
- `QRectF boundingRect`

17.30.1 Detailed Description

17.30.2 Constructor & Destructor Documentation

17.30.2.1 EmbDetailsDialog() `EmbDetailsDialog (`
 `QGraphicsScene * theScene,`
 `QWidget * parent = 0)`

17.30.2.2 ~EmbDetailsDialog() `~EmbDetailsDialog ()`

17.30.3 Member Function Documentation

17.30.3.1 createHistogram() `QWidget * createHistogram ()`

17.30.3.2 createMainWidget() `QWidget * createMainWidget ()`

17.30.3.3 getInfo() `void getInfo ()`

17.30.4 Member Data Documentation

17.30.4.1 boundingRect `QRectF boundingRect`

17.30.4.2 buttonBox `QDialogButtonBox* buttonBox`

17.30.4.3 colorChanges `quint32 colorChanges`

17.30.4.4 colorTotal `quint32 colorTotal`

17.30.4.5 mainWidget `QWidget* mainWidget`

17.30.4.6 stitchesJump `quint32 stitchesJump`

17.30.4.7 stitchesReal `quint32 stitchesReal`

17.30.4.8 stitchesTotal `quint32 stitchesTotal`

17.30.4.9 stitchesTrim `quint32 stitchesTrim`

The documentation for this class was generated from the following files:

- `embroidermodder2/embroidermodder.h`
- `embroidermodder2/embdetails-dialog.cpp`

17.31 EmbDiameterDim_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- `EmbVector position`

17.31.1 Member Data Documentation

17.31.1.1 position `EmbVector position`

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery.h`

17.32 EmbEllipse_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- EmbVector center
- EmbVector radius
- EmbReal rotation

17.32.1 Member Data Documentation**17.32.1.1 center EmbVector center****17.32.1.2 radius EmbVector radius****17.32.1.3 rotation EmbReal rotation**

The documentation for this struct was generated from the following file:

- extern/libembroidery/src/embroidery.h

17.33 EmbFormatList_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- char extension [2+EMBFORMAT_MAXEXT]
- char description [EMBFORMAT_MAXDESC]
- char reader_state
- char writer_state
- int type
- int color_only
- int check_for_color_file
- int write_external_color_file

17.33.1 Member Data Documentation**17.33.1.1 check_for_color_file int check_for_color_file**

17.33.1.2 color_only int color_only

17.33.1.3 description char description[EMBFORMAT_MAXDESC]

17.33.1.4 extension char extension[2+EMBFORMAT_MAXEXT]

17.33.1.5 reader_state char reader_state

17.33.1.6 type int type

17.33.1.7 write_external_color_file int write_external_color_file

17.33.1.8 writer_state char writer_state

The documentation for this struct was generated from the following file:

- [extern/libembroidery/src/embroidery.h](#)

17.34 EmbGeometry_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- union {
 - EmbArc arc
 - EmbCircle circle
 - EmbColor color
 - EmbEllipse ellipse
 - EmbLine line
 - EmbPath path
 - EmbPoint point
 - EmbPolygon polygon
 - EmbPolyline polyline
 - EmbRect rect
 - EmbSpline spline
 - EmbVector vector}
- EmbStitch stitch
- EmbThread thread
- int flag
- int type
- int lineType

17.34.1 Member Data Documentation

17.34.1.1 arc [EmbArc](#) arc

17.34.1.2 circle [EmbCircle](#) circle

17.34.1.3 color [EmbColor](#) color

17.34.1.4 ellipse [EmbEllipse](#) ellipse

17.34.1.5 flag int flag

17.34.1.6 line [EmbLine](#) line

17.34.1.7 lineType int lineType

17.34.1.8 union { ... } object

17.34.1.9 path [EmbPath](#) path

17.34.1.10 point [EmbPoint](#) point

17.34.1.11 polygon [EmbPolygon](#) polygon

17.34.1.12 polyline [EmbPolyline](#) polyline

17.34.1.13 rect [EmbRect](#) rect

17.34.1.14 spline [EmbSpline](#) spline

17.34.1.15 stitch [EmbStitch](#) stitch

17.34.1.16 thread [EmbThread](#) thread

17.34.1.17 type int type

17.34.1.18 vector [EmbVector](#) vector

The documentation for this struct was generated from the following file:

- [extern/libembroidery/src/embroidery.h](#)

17.35 EmbImage_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- [EmbVector position](#)
- [EmbVector dimensions](#)
- [unsigned char * data](#)
- [int width](#)
- [int height](#)
- [char path \[200\]](#)
- [char name \[200\]](#)

17.35.1 Member Data Documentation

17.35.1.1 data `unsigned char* data`

17.35.1.2 dimensions `EmbVector dimensions`

17.35.1.3 height `int height`

17.35.1.4 name `char name[200]`

17.35.1.5 path `char path[200]`

17.35.1.6 position `EmbVector position`

17.35.1.7 width `int width`

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery.h`

17.36 EmbInfiniteLine_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- `EmbVector position`

17.36.1 Member Data Documentation

17.36.1.1 position `EmbVector` position

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery.h`

17.37 EmbLayer_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- `char name [100]`
- `EmbArray * geometry`

17.37.1 Member Data Documentation

17.37.1.1 geometry `EmbArray*` geometry

17.37.1.2 name `char name[100]`

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery.h`

17.38 EmbLeaderDim_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- `EmbVector position`

17.38.1 Member Data Documentation

17.38.1.1 position `EmbVector` position

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery.h`

17.39 EmbLine_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- `EmbVector start`
- `EmbVector end`
- `int lineType`
- `EmbColor color`

17.39.1 Member Data Documentation

17.39.1.1 color `EmbColor` color

17.39.1.2 end `EmbVector` end

17.39.1.3 lineType `int` lineType

17.39.1.4 start `EmbVector` start

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery.h`

17.40 EmbLinearDim_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- EmbVector position

17.40.1 Member Data Documentation

17.40.1.1 position EmbVector position

The documentation for this struct was generated from the following file:

- extern/libembroidery/src/embroidery.h

17.41 EmbOrdinateDim_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- EmbVector position

17.41.1 Member Data Documentation

17.41.1.1 position EmbVector position

The documentation for this struct was generated from the following file:

- extern/libembroidery/src/embroidery.h

17.42 EmbPath_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- `EmbArray * pointList`
- `EmbArray * flagList`
- `int lineType`
- `EmbColor color`

17.42.1 Member Data Documentation**17.42.1.1 color** `EmbColor color`**17.42.1.2 flagList** `EmbArray* flagList`**17.42.1.3 lineType** `int lineType`**17.42.1.4 pointList** `EmbArray* pointList`

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery.h`

17.43 EmbPattern_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- `unsigned int dstJumpsPerTrim`
- `EmbVector home`
- `EmbReal hoop_width`
- `EmbReal hoop_height`
- `EmbArray * thread_list`
- `EmbArray * stitch_list`
- `EmbArray * geometry`
- `EmbLayer layer [EMB_MAX_LAYERS]`
- `int currentColorIndex`

17.43.1 Member Data Documentation

17.43.1.1 currentColorIndex int currentColorIndex

17.43.1.2 dstJumpsPerTrim unsigned int dstJumpsPerTrim

17.43.1.3 geometry EmbArray* geometry

17.43.1.4 home EmbVector home

17.43.1.5 hoop_height EmbReal hoop_height

17.43.1.6 hoop_width EmbReal hoop_width

17.43.1.7 layer EmbLayer layer[EMB_MAX_LAYERS]

17.43.1.8 stitch_list EmbArray* stitch_list

17.43.1.9 thread_list EmbArray* thread_list

The documentation for this struct was generated from the following file:

- extern/libembroidery/src/embroidery.h

17.44 EmbPoint_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- [EmbVector position](#)
- int [lineType](#)
- [EmbColor color](#)

17.44.1 Member Data Documentation**17.44.1.1 color** [EmbColor](#) `color`**17.44.1.2 lineType** int `lineType`**17.44.1.3 position** [EmbVector](#) `position`

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery.h`

17.45 EmbRadiusDim_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- [EmbVector position](#)

17.45.1 Member Data Documentation**17.45.1.1 position** [EmbVector](#) `position`

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery.h`

17.46 EmbRay_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- [EmbVector position](#)

17.46.1 Member Data Documentation

17.46.1.1 **position** [EmbVector](#) position

The documentation for this struct was generated from the following file:

- [extern/libembroidery/src/embroidery.h](#)

17.47 EmbRect_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- [EmbReal top](#)
- [EmbReal left](#)
- [EmbReal bottom](#)
- [EmbReal right](#)
- [EmbReal rotation](#)
- [EmbReal radius](#)

17.47.1 Member Data Documentation

17.47.1.1 **bottom** [EmbReal](#) bottom

17.47.1.2 **left** [EmbReal](#) left

17.47.1.3 radius `EmbReal` `radius`

17.47.1.4 right `EmbReal` `right`

17.47.1.5 rotation `EmbReal` `rotation`

17.47.1.6 top `EmbReal` `top`

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery.h`

17.48 EmbSatinOutline_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- `int length`
- `EmbArray * side1`
- `EmbArray * side2`

17.48.1 Member Data Documentation

17.48.1.1 length `int` `length`

17.48.1.2 side1 `EmbArray*` `side1`

17.48.1.3 side2 `EmbArray*` `side2`

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery.h`

17.49 EmbSpline_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- `EmbArray * beziers`

17.49.1 Member Data Documentation

17.49.1.1 **beziers** `EmbArray* beziers`

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery.h`

17.50 EmbStitch_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- `int flags`
- `EmbReal x`
- `EmbReal y`
- `int color`

17.50.1 Member Data Documentation

17.50.1.1 **color** `int color`

positive is up, units are in mm

17.50.1.2 **flags** `int flags`

17.50.1.3 x EmbReal x

uses codes defined above

17.50.1.4 y EmbReal y

absolute position (not relative)

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery.h`

17.51 EmbTextMulti_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- `EmbVector position`
- `char text [200]`

17.51.1 Member Data Documentation

17.51.1.1 position `EmbVector position`

17.51.1.2 text `char text[200]`

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery.h`

17.52 EmbTextSingle_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- `EmbVector position`
- `char text [200]`

17.52.1 Member Data Documentation

17.52.1.1 position `EmbVector position`

17.52.1.2 text `char text[200]`

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery.h`

17.53 EmbThread_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- `EmbColor color`
- `char description [50]`
- `char catalogNumber [30]`

17.53.1 Member Data Documentation

17.53.1.1 catalogNumber `char catalogNumber[30]`

17.53.1.2 color `EmbColor color`

17.53.1.3 description `char description[50]`

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery.h`

17.54 EmbTime_Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- unsigned int **year**
- unsigned int **month**
- unsigned int **day**
- unsigned int **hour**
- unsigned int **minute**
- unsigned int **second**

17.54.1 Member Data Documentation

17.54.1.1 day unsigned int day

17.54.1.2 hour unsigned int hour

17.54.1.3 minute unsigned int minute

17.54.1.4 month unsigned int month

17.54.1.5 second unsigned int second

17.54.1.6 year unsigned int year

The documentation for this struct was generated from the following file:

- extern/libembroidery/src/[embroidery.h](#)

17.55 EmbVector_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- EmbReal x
- EmbReal y

17.55.1 Detailed Description

The basic type to represent points absolutely or represent directions.

Positive y is up, units are in mm.

17.55.2 Member Data Documentation

17.55.2.1 x EmbReal x

17.55.2.2 y EmbReal y

The documentation for this struct was generated from the following file:

- extern/libembroidery/src/[embroidery.h](#)

17.56 EmbView_ Struct Reference

```
#include <embroidermodder.h>
```

Public Attributes

- `EmbPattern * pattern`
- `EmbVector origin`
- `EmbReal scale`
- `char grid_type [200]`
- `int ui_mode`
- `bool snap_mode`
- `bool grid_mode`
- `bool ruler_mode`
- `bool ortho_mode`
- `bool polar_mode`
- `bool qsnap_mode`
- `bool qtrack_mode`
- `bool lwt_mode`
- `bool real_render`
- `bool metric`
- `bool simulate`
- `clock_t simulation_start`
- `char text_font [200]`
- `EmbReal text_size`
- `EmbReal text_angle`
- `bool text_style_bold`
- `bool text_style_italic`
- `bool text_style_underline`
- `bool text_style_overline`
- `bool text_style_strikeout`
- `char filename [200]`
- `UndoHistory undo_history`
- `int selected [100]`
- `int n_selected`
- `int rubber_mode`

17.56.1 Detailed Description

17.56.2 EmbViews

The EmbView describes how the render is displayed.

17.56.3 Member Data Documentation

17.56.3.1 `filename` `char filename[200]`

17.56.3.2 `grid_mode` `bool grid_mode`

17.56.3.3 `grid_type` `char grid_type[200]`

17.56.3.4 `lwt_mode` `bool lwt_mode`

17.56.3.5 `metric` `bool metric`

17.56.3.6 `n_selected` `int n_selected`

17.56.3.7 `origin` `EmbVector origin`

17.56.3.8 `ortho_mode` `bool ortho_mode`

17.56.3.9 `pattern` `EmbPattern* pattern`

17.56.3.10 `polar_mode` `bool polar_mode`

17.56.3.11 `qsnap_mode` `bool qsnap_mode`

17.56.3.12 `qtrack_mode` `bool qtrack_mode`

17.56.3.13 `real_render` `bool real_render`

17.56.3.14 rubber_mode int rubber_mode

17.56.3.15 ruler_mode bool ruler_mode

17.56.3.16 scale EmbReal scale

17.56.3.17 selected int selected[100]

17.56.3.18 simulate bool simulate

17.56.3.19 simulation_start clock_t simulation_start

17.56.3.20 snap_mode bool snap_mode

17.56.3.21 text_angle EmbReal text_angle

17.56.3.22 text_font char text_font[200]

17.56.3.23 text_size EmbReal text_size

17.56.3.24 text_style_bold bool text_style_bold

17.56.3.25 `text_style_italic` bool `text_style_italic`

17.56.3.26 `text_style_overline` bool `text_style_overline`

17.56.3.27 `text_style_strikeout` bool `text_style_strikeout`

17.56.3.28 `text_style_underline` bool `text_style_underline`

17.56.3.29 `ui_mode` int `ui_mode`

17.56.3.30 `undo_history` [UndoHistory](#) `undo_history`

The documentation for this struct was generated from the following file:

- `embroidermodder2/embroidermodder.h`

17.57 `hoop_padding` Struct Reference

Public Attributes

- int `left`
- int `right`
- int `top`
- int `bottom`

17.57.1 Member Data Documentation

17.57.1.1 `bottom` int `bottom`

17.57.1.2 `left` int `left`

17.57.1.3 right int right

17.57.1.4 top int top

The documentation for this struct was generated from the following file:

- [extern/libembroidery/src/formats/format_jef.c](#)

17.58 Huffman Struct Reference

```
#include <embroidery_internal.h>
```

Public Attributes

- int [default_value](#)
- int [lengths](#) [1000]
- int [nlengths](#)
- int [table](#) [1000]
- int [table_width](#)
- int [ntable](#)

17.58.1 Member Data Documentation

17.58.1.1 default_value int default_value

17.58.1.2 lengths int lengths[1000]

17.58.1.3 nlengths int nlengths

17.58.1.4 ntable int ntable

17.58.1.5 table int table[1000]

17.58.1.6 table_width int table_width

The documentation for this struct was generated from the following file:

- [extern/libembroidery/src/embroidery_internal.h](#)

17.59 ImageObject Class Reference

```
#include <embroidermodder.h>
```

Public Types

- enum { [Type](#) = OBJ_TYPE_IMAGE }

Public Types inherited from [BaseObject](#)

- enum { [Type](#) = OBJ_TYPE_BASE }

Public Member Functions

- [ImageObject \(EmbReal x, EmbReal y, EmbReal w, EmbReal h, QRgb rgb, QGraphicsItem *parent=0\)](#)
- [ImageObject \(ImageObject *obj, QGraphicsItem *parent=0\)](#)
- [~ImageObject \(\)](#)
- void [init \(EmbReal x, EmbReal y, EmbReal w, EmbReal h, QRgb rgb, Qt::PenStyle lineType\)](#)
- void [updatePath \(\)](#)
- virtual int [type \(\) const](#)
- QPointF [objectTopLeft \(\) const](#)
- QPointF [objectTopRight \(\) const](#)
- QPointF [objectBottomLeft \(\) const](#)
- QPointF [objectBottomRight \(\) const](#)
- EmbReal [objectWidth \(\) const](#)
- EmbReal [objectHeight \(\) const](#)
- EmbReal [objectArea \(\) const](#)
- void [setObjectRect \(EmbReal x, EmbReal y, EmbReal w, EmbReal h\)](#)
- void [updateRubber \(QPainter *painter=0\)](#)
- virtual void [vulcanize \(\)](#)
- virtual QPointF [mouseSnapPoint \(const QPointF &mousePoint\)](#)
- virtual QList< QPointF > [allGripPoints \(\)](#)
- virtual void [gripEdit \(const QPointF &before, const QPointF &after\)](#)

Public Member Functions inherited from BaseObject

- `BaseObject` (QGraphicsItem *parent=0)
- virtual `~BaseObject` ()
- virtual int `type` () const
- qint64 `objectID` () const
- QPen `objectPen` () const
- QColor `objectColor` () const
- QRgb `objectColorRGB` () const
- Qt::PenStyle `objectLineType` () const
- EmbReal `objectLineWidth` () const
- QPainterPath `objectPath` () const
- int `objectRubberMode` () const
- QPointF `objectRubberPoint` (const QString &key) const
- QString `objectRubberText` (const QString &key) const
- QPointF `objectCenter` () const
- EmbReal `objectCenterX` () const
- EmbReal `objectCenterY` () const
- void `setObjectCenter` (EmbVector center)
- void `setObjectCenterX` (EmbReal centerX)
- void `setObjectCenterY` (EmbReal centerY)
- QRectF `rect` () const
- void `setRect` (const QRectF &r)
- void `setRect` (EmbReal x, EmbReal y, EmbReal w, EmbReal h)
- QLineF `line` () const
- void `setLine` (const QLineF &li)
- void `setLine` (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2)
- void `setObjectColor` (const QColor &color)
- void `setObjectColorRGB` (QRgb rgb)
- void `setObjectLineType` (Qt::PenStyle lineType)
- void `setObjectLineWidth` (EmbReal lineWidth)
- void `setObjectPath` (const QPainterPath &p)
- void `setObjectRubberMode` (int mode)
- void `setObjectRubberPoint` (const QString &key, const QPointF &point)
- void `setObjectRubberText` (const QString &key, const QString &txt)
- virtual QRectF `boundingRect` () const
- virtual QPainterPath `shape` () const
- void `drawRubberLine` (const QLineF &rubLine, QPainter *painter=0, const char *colorFromScene=0)
- virtual void `vulcanize` ()=0
- virtual QPointF `mouseSnapPoint` (const QPointF &mousePoint)=0
- virtual QList< QPointF > `allGripPoints` ()=0
- virtual void `gripEdit` (const QPointF &before, const QPointF &after)=0

Protected Member Functions

- void `paint` (Painter *, const QStyleOptionGraphicsItem *, QWidget *)

Protected Member Functions inherited from BaseObject

- QPen `lineWeightPen` () const
- void `realRender` (Painter *painter, const QPainterPath &renderPath)

Additional Inherited Members

Public Attributes inherited from [BaseObject](#)

- QPen [objPen](#)
- QPen [lwtPen](#)
- QLineF [objLine](#)
- int [objRubberMode](#)
- QHash<QString, QPointF> [objRubberPoints](#)
- QHash<QString, QString> [objRubberTexts](#)
- qint64 [objID](#)

17.59.1 Member Enumeration Documentation

17.59.1.1 anonymous enum [anonymous enum](#)

Enumerator

| | |
|------|---------------------------------|
| Type | <input type="button" value=""/> |
|------|---------------------------------|

17.59.2 Constructor & Destructor Documentation

17.59.2.1 [ImageObject\(\)](#) [1/2] [ImageObject](#) (

```
EmbReal x,  
EmbReal y,  
EmbReal w,  
EmbReal h,  
QRgb rgb,  
QGraphicsItem * parent = 0 )
```

17.59.2.2 [ImageObject\(\)](#) [2/2] [ImageObject](#) (

```
ImageObject * obj,  
QGraphicsItem * parent = 0 )
```

17.59.2.3 [~ImageObject\(\)](#) [~ImageObject](#) ()

17.59.3 Member Function Documentation

17.59.3.1 allGripPoints() `QList< QPointF > allGripPoints () [virtual]`

Implements [BaseObject](#).

17.59.3.2 gripEdit() `void gripEdit (const QPointF & before, const QPointF & after) [virtual]`

Implements [BaseObject](#).

17.59.3.3 init() `void init (EmbReal x, EmbReal y, EmbReal w, EmbReal h, QRgb rgb, Qt::PenStyle lineType)`

17.59.3.4 mouseSnapPoint() `QPointF mouseSnapPoint (const QPointF & mousePoint) [virtual]`

Implements [BaseObject](#).

17.59.3.5 objectArea() `EmbReal objectArea () const [inline]`

17.59.3.6 objectBottomLeft() `QPointF objectBottomLeft () const`

17.59.3.7 objectBottomRight() `QPointF objectBottomRight () const`

17.59.3.8 objectHeight() `EmbReal objectHeight () const [inline]`

17.59.3.9 objectTopLeft() `QPointF objectTopLeft () const`

17.59.3.10 objectTopRight() `QPointF objectTopRight () const`

17.59.3.11 objectWidth() `EmbReal objectWidth () const [inline]`

17.59.3.12 paint() `void paint (`
 `QPainter * painter,`
 `const QStyleOptionGraphicsItem * option,`
 `QWidget *) [protected]`

17.59.3.13 setObjectRect() `void setObjectRect (`
 `EmbReal x,`
 `EmbReal y,`
 `EmbReal w,`
 `EmbReal h)`

17.59.3.14 type() `virtual int type () const [inline], [virtual]`

Reimplemented from [BaseObject](#).

17.59.3.15 updatePath() `void updatePath ()`

17.59.3.16 updateRubber() `void updateRubber (`
 `QPainter * painter = 0)`

17.59.3.17 `vulcanize()` `void vulcanize () [virtual]`

Implements [BaseObject](#).

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/object-image.cpp](#)

17.60 ImageWidget Class Reference

```
#include <embroidermodder.h>
```

Public Member Functions

- `ImageWidget (const QString &filename, QWidget *parent=0)`
ImageWidget::ImageWidget.
- `~ImageWidget ()`
ImageWidget::~ImageWidget.
- `bool load (const QString &fileName)`
ImageWidget::load.
- `bool save (const QString &fileName)`
ImageWidget::save.

Public Attributes

- `QImage img`

Protected Member Functions

- `void paintEvent (QPaintEvent *event)`
ImageWidget::paintEvent.

17.60.1 Detailed Description

17.60.2 Constructor & Destructor Documentation

17.60.2.1 `ImageWidget()` `ImageWidget (`
 `const QString & filename,`
 `QWidget * parent = 0)`

ImageWidget::ImageWidget.

Parameters

| | |
|-----------------|----------------------|
| <i>filename</i> | <input type="text"/> |
| <i>parent</i> | <input type="text"/> |

17.60.2.2 ~ImageWidget() ~ImageWidget ()

ImageWidget::~ImageWidget.

17.60.3 Member Function Documentation**17.60.3.1 load()** bool load (const QString & *fileName*)

ImageWidget::load.

Parameters

| | |
|-----------------|----------------------|
| <i>fileName</i> | <input type="text"/> |
|-----------------|----------------------|

Returns**17.60.3.2 paintEvent()** void paintEvent (QPaintEvent * *event*) [protected]

ImageWidget::paintEvent.

17.60.3.3 save() bool save (const QString & *fileName*)

ImageWidget::save.

Parameters

| | |
|-----------------|----------------------|
| <i>fileName</i> | <input type="text"/> |
|-----------------|----------------------|

Returns

17.60.4 Member Data Documentation

17.60.4.1 img QImage img

The documentation for this class was generated from the following files:

- embroidermodder2/[embroidermodder.h](#)
- embroidermodder2/[imagewidget.cpp](#)

17.61 LayerManager Class Reference

```
#include <embroidermodder.h>
```

Public Member Functions

- [LayerManager \(MainWindow *mw, QWidget *parent=0\)](#)
LayerManager::LayerManager.
- [~LayerManager \(\)](#)
LayerManager::~LayerManager.
- void [addLayer \(const QString &name, const bool visible, const bool frozen, const EmbReal zValue, const QRgb color, const QString &lineType, const QString &lineWeight, const bool print\)](#)
LayerManager::addLayer.

Public Attributes

- QStandardItemModel * [layerModel](#)
- QSortFilterProxyModel * [layerModelSorted](#)
- QTreeView * [treeView](#)

17.61.1 Detailed Description

17.61.2 Constructor & Destructor Documentation

17.61.2.1 LayerManager() [LayerManager \(](#) [MainWindow * mw,](#) [QWidget * parent = 0 \)](#)

LayerManager::LayerManager.

Parameters

| | |
|---------------|--|
| <i>mw</i> | |
| <i>parent</i> | |

17.61.2.2 ~LayerManager() ~[LayerManager](#) ()

[LayerManager](#)::~[LayerManager](#).

17.61.3 Member Function Documentation

17.61.3.1 addLayer() void addLayer (

```
    const QString & name,
    const bool visible,
    const bool frozen,
    const EmbReal zValue,
    const QRgb color,
    const QString & lineType,
    const QString & lineWeight,
    const bool print )
```

[LayerManager](#)::addLayer.

Parameters

| | |
|-------------------|--|
| <i>name</i> | |
| <i>visible</i> | |
| <i>frozen</i> | |
| <i>zValue</i> | |
| <i>color</i> | |
| <i>lineType</i> | |
| <i>lineWeight</i> | |
| <i>print</i> | |

17.61.4 Member Data Documentation**17.61.4.1 layerModel** QStandardItemModel* layerModel

17.61.4.2 layerModelSorted QSortFilterProxyModel* layerModelSorted

17.61.4.3 treeView QTreeView* treeView

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/layer-manager.cpp](#)

17.62 LineObject Class Reference

```
#include <embroidermodder.h>
```

Public Types

- enum { [Type](#) = OBJ_TYPE_LINE }

Public Types inherited from [BaseObject](#)

- enum { [Type](#) = OBJ_TYPE_BASE }

Public Member Functions

- [LineObject \(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, QRgb rgb, QGraphicsItem *parent=0\)](#)
- [LineObject \(LineObject *obj, QGraphicsItem *parent=0\)](#)
- [~LineObject \(\)](#)
- void [init \(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, QRgb rgb, Qt::PenStyle lineType\)](#)
- virtual int [type \(\) const](#)
- QPainterPath [objectSavePath \(\) const](#)
- QPointF [objectEndPoint1 \(\) const](#)
- QPointF [objectEndPoint2 \(\) const](#)
- QPointF [objectMidPoint \(\) const](#)
- EmbReal [objectX1 \(\) const](#)
- EmbReal [objectY1 \(\) const](#)
- EmbReal [objectX2 \(\) const](#)
- EmbReal [objectY2 \(\) const](#)
- EmbReal [objectDeltaX \(\) const](#)
- EmbReal [objectDeltaY \(\) const](#)
- EmbReal [objectAngle \(\) const](#)
- EmbReal [objectLength \(\) const](#)
- void [setObjectEndPoint1 \(const QPointF &endPt1\)](#)
- void [setObjectEndPoint1 \(EmbReal x1, EmbReal y1\)](#)
- void [setObjectEndPoint2 \(const QPointF &endPt2\)](#)
- void [setObjectEndPoint2 \(EmbReal x2, EmbReal y2\)](#)
- void [setObjectX1 \(EmbReal x\)](#)
- void [setObjectY1 \(EmbReal y\)](#)
- void [setObjectX2 \(EmbReal x\)](#)
- void [setObjectY2 \(EmbReal y\)](#)
- void [updateRubber \(QPainter *painter=0\)](#)
- virtual void [vulcanize \(\)](#)
- virtual QPointF [mouseSnapPoint \(const QPointF &mousePoint\)](#)
- virtual QList< QPointF > [allGripPoints \(\)](#)
- virtual void [gripEdit \(const QPointF &before, const QPointF &after\)](#)

Public Member Functions inherited from [BaseObject](#)

- [BaseObject](#) (QGraphicsItem *parent=0)
- virtual [~BaseObject](#) ()
- virtual int [type](#) () const
- qint64 [objectID](#) () const
- QPen [objectPen](#) () const
- QColor [objectColor](#) () const
- QRgb [objectColorRGB](#) () const
- Qt::PenStyle [objectLineType](#) () const
- EmbReal [objectLineWidth](#) () const
- QPainterPath [objectPath](#) () const
- int [objectRubberMode](#) () const
- QPointF [objectRubberPoint](#) (const QString &key) const
- QString [objectRubberText](#) (const QString &key) const
- QPointF [objectCenter](#) () const
- EmbReal [objectCenterX](#) () const
- EmbReal [objectCenterY](#) () const
- void [setObjectCenter](#) (EmbVector center)
- void [setObjectCenterX](#) (EmbReal centerX)
- void [setObjectCenterY](#) (EmbReal centerY)
- QRectF [rect](#) () const
- void [setRect](#) (const QRectF &r)
- void [setRect](#) (EmbReal x, EmbReal y, EmbReal w, EmbReal h)
- QLineF [line](#) () const
- void [setLine](#) (const QLineF &l)
- void [setLine](#) (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2)
- void [setObjectColor](#) (const QColor &color)
- void [setObjectColorRGB](#) (QRgb rgb)
- void [setObjectLineType](#) (Qt::PenStyle lineType)
- void [setObjectLineWidth](#) (EmbReal lineWidth)
- void [setObjectPath](#) (const QPainterPath &p)
- void [setObjectRubberMode](#) (int mode)
- void [setObjectRubberPoint](#) (const QString &key, const QPointF &point)
- void [setObjectRubberText](#) (const QString &key, const QString &txt)
- virtual QRectF [boundingRect](#) () const
- virtual QPainterPath [shape](#) () const
- void [drawRubberLine](#) (const QLineF &rubLine, QPainter *painter=0, const char *colorFromScene=0)
- virtual void [vulcanize](#) ()=0
- virtual QPointF [mouseSnapPoint](#) (const QPointF &mousePoint)=0
- virtual QList< QPointF > [allGripPoints](#) ()=0
- virtual void [gripEdit](#) (const QPointF &before, const QPointF &after)=0

Protected Member Functions

- void [paint](#) (QPainter *, const QStyleOptionGraphicsItem *, QWidget *)

Protected Member Functions inherited from [BaseObject](#)

- QPen [lineWeightPen](#) () const
- void [realRender](#) (QPainter *painter, const QPainterPath &renderPath)

Additional Inherited Members

Public Attributes inherited from [BaseObject](#)

- QPen [objPen](#)
- QPen [lwtPen](#)
- QLineF [objLine](#)
- int [objRubberMode](#)
- QHash<QString, QPointF> [objRubberPoints](#)
- QHash<QString, QString> [objRubberTexts](#)
- qint64 [objID](#)

17.62.1 Member Enumeration Documentation

17.62.1.1 anonymous enum [anonymous enum](#)

Enumerator

| | |
|------|---------------------------------|
| Type | <input type="button" value=""/> |
|------|---------------------------------|

17.62.2 Constructor & Destructor Documentation

17.62.2.1 [LineObject\(\)](#) [1/2] [LineObject](#) (

```
    EmbReal x1,  
    EmbReal y1,  
    EmbReal x2,  
    EmbReal y2,  
    QRgb rgb,  
    QGraphicsItem * parent = 0 )
```

17.62.2.2 [LineObject\(\)](#) [2/2] [LineObject](#) (

```
    LineObject * obj,  
    QGraphicsItem * parent = 0 )
```

17.62.2.3 [~LineObject\(\)](#) [~LineObject](#) ()

17.62.3 Member Function Documentation

17.62.3.1 allGripPoints() `QList< QPointF > allGripPoints () [virtual]`

Implements [BaseObject](#).

17.62.3.2 gripEdit() `void gripEdit (const QPointF & before, const QPointF & after) [virtual]`

Implements [BaseObject](#).

17.62.3.3 init() `void init (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, QRgb rgb, Qt::PenStyle lineType)`

17.62.3.4 mouseSnapPoint() `QPointF mouseSnapPoint (const QPointF & mousePoint) [virtual]`

Implements [BaseObject](#).

17.62.3.5 objectAngle() `EmbReal objectAngle () const`

17.62.3.6 objectDeltaX() `EmbReal objectDeltaX () const [inline]`

17.62.3.7 objectDeltaY() `EmbReal objectDeltaY () const [inline]`

17.62.3.8 objectEndPoint1() QPointF objectEndPoint1 () const [inline]

17.62.3.9 objectEndPoint2() QPointF objectEndPoint2 () const

17.62.3.10 objectLength() EmbReal objectLength () const [inline]

17.62.3.11 objectMidPoint() QPointF objectMidPoint () const

17.62.3.12 objectSavePath() QPainterPath objectSavePath () const

17.62.3.13 objectX1() EmbReal objectX1 () const [inline]

17.62.3.14 objectX2() EmbReal objectX2 () const [inline]

17.62.3.15 objectY1() EmbReal objectY1 () const [inline]

17.62.3.16 objectY2() EmbReal objectY2 () const [inline]

17.62.3.17 paint() void paint (
 QPainter * painter,
 const QStyleOptionGraphicsItem * option,
 QWidget *) [protected]

17.62.3.18 setObjectEndPoint1() [1/2] void setObjectEndPoint1 (
 const QPointF & endPtl)

17.62.3.19 setObjectEndPoint1() [2/2] void setObjectEndPoint1 (EmbReal x1, EmbReal y1)

17.62.3.20 setObjectEndPoint2() [1/2] void setObjectEndPoint2 (const QPointF & endPt2)

17.62.3.21 setObjectEndPoint2() [2/2] void setObjectEndPoint2 (EmbReal x2, EmbReal y2)

17.62.3.22 setObjectX1() void setObjectX1 (EmbReal x) [inline]

17.62.3.23 setObjectX2() void setObjectX2 (EmbReal x) [inline]

17.62.3.24 setObjectY1() void setObjectY1 (EmbReal y) [inline]

17.62.3.25 setObjectY2() void setObjectY2 (EmbReal y) [inline]

17.62.3.26 type() virtual int type () const [inline], [virtual]

Reimplemented from [BaseObject](#).

17.62.3.27 updateRubber() void updateRubber (QPainter * painter = 0)

17.62.3.28 vulcanize() void vulcanize () [virtual]

Implements [BaseObject](#).

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/object-line.cpp](#)

17.63 LSYSTEM Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- char **axiom**
- char * **alphabet**
- char * **constants**
- char ** **rules**

17.63.1 Member Data Documentation

17.63.1.1 **alphabet** char* alphabet

17.63.1.2 **axiom** char axiom

17.63.1.3 **constants** char* constants

17.63.1.4 **rules** char** rules

The documentation for this struct was generated from the following file:

- [extern/libembroidery/src/embroidery.h](#)

17.64 MainWindow Class Reference

The [MainWindow](#) class.

```
#include <embroidermodder.h>
```

Public Slots

- void `enablePromptRapidFire ()`
- void `disablePromptRapidFire ()`
- void `enableMoveRapidFire ()`
- void `disableMoveRapidFire ()`
- void `onCloseWindow ()`
`MainWindow::onCloseWindow.`
- virtual void `onCloseMdiWin (MdiWindow *)`
`MainWindow::onCloseMdiWin.`
- void `recentMenuAboutToShow ()`
`MainWindow::recentMenuAboutToShow.`
- void `onWindowActivated (QMdiSubWindow *w)`
`MainWindow::onWindowActivated.`
- void `windowMenuAboutToShow ()`
`MainWindow::windowMenuAboutToShow.`
- void `windowMenuActivated (bool checked)`
`MainWindow::windowMenuActivated.`
- QAction * `getAction (int actionEnum)`
`MainWindow::getAction.`
- void `updateAllViewScrollBars (bool val)`
- void `updateAllViewCrossHairColors (QRgb color)`
- void `updateAllViewBackgroundColors (QRgb color)`
- void `updateAllViewSelectBoxColors (QRgb colorL, QRgb fillL, QRgb colorR, QRgb fillR, int alpha)`
- void `updateAllViewGridColors (QRgb color)`
- void `updateAllViewRulerColors (QRgb color)`
- void `updatePickAddMode (bool val)`
- void `pickAddModeToggled ()`
- void `settingsPrompt ()`
- void `settingsDialog (const QString &showTab=QString())`
- void `readSettings ()`
`MainWindow::readSettings.`
- void `writeSettings ()`
`MainWindow::writeSettings.`
- static bool `validFileFormat (const QString &fileName)`
`MainWindow::validFileFormat.`
- void `stub_implement (QString txt)`
`MainWindow::stub_implement txt.`
- void `stub_testing ()`
`MainWindow::stub_testing.`
- void `promptHistoryAppended (const QString &txt)`
- void `logPromptInput (const QString &txt)`
- void `promptInputPrevious ()`
- void `promptInputNext ()`
- void `runCommand ()`
- void `runCommandMain (const QString &cmd)`
- void `runCommandClick (const QString &cmd, EmbReal x, EmbReal y)`
- void `runCommandMove (const QString &cmd, EmbReal x, EmbReal y)`
- void `runCommandContext (const QString &cmd, const QString &str)`
- void `runCommandPrompt (const QString &cmd, const QString &str)`
- void `newFile ()`
`MainWindow::newFile.`
- void `openFile (bool recent=false, const QString &recentFile="")`

- void **openFile** (*MainWindow::openFile*)
- void **openFilesSelected** (const QStringList &)
- MainWindow::openFilesSelected*.
- void **openrecentfile** ()
- MainWindow::openrecentfile*.
- void **savefile** ()
- MainWindow::savefile*.
- void **saveasfile** ()
- MainWindow::saveasfile*.
- void **print** ()
- void **designDetails** ()
- void **exit** ()
- MainWindow::exit*.
- void **quit** ()
- MainWindow::quit*.
- void **checkForUpdates** ()
- void **tipOfTheDay** ()
- void **buttonTipOfTheDayClicked** (int)
- void **checkBoxTipOfTheDayStateChanged** (int)
- void **help** ()
- void **changelog** ()
- void **about** ()
- void **whatsThisContextHelp** ()
- void **cut** ()
- void **copy** ()
- void **paste** ()
- void **selectAll** ()
- void **closeToolBar** (QAction *)
- MainWindow::closeToolBar*.
- void **floatingChangedToolBar** (bool)
- MainWindow::floatingChangedToolBar*.
- void **toggleGrid** ()
- void **toggleRuler** ()
- void **toggleLwt** ()
- void **iconResize** (int iconSize)
- void **layerSelectorIndexChanged** (int index)
- void **colorSelectorIndexChanged** (int index)
- void **linetypeSelectorIndexChanged** (int index)
- void **lineweightSelectorIndexChanged** (int index)
- void **textFontSelectorCurrentFontChanged** (const QFont &font)
- void **textSizeSelectorIndexChanged** (int index)
- QString **textFont** ()
- EmbReal **textSize** ()
- EmbReal **textAngle** ()
- bool **textBold** ()
- bool **textItalic** ()
- bool **textUnderline** ()
- bool **textStrikeOut** ()
- bool **textOverline** ()
- void **setTextFont** (const QString &str)
- void **setTextSize** (EmbReal num)
- void **setTextAngle** (EmbReal num)
- void **setTextBold** (bool val)

- void `setTextItalic` (bool val)
 - void `setTextUnderline` (bool val)
 - void `setTextStrikeOut` (bool val)
 - void `setTextOverline` (bool val)
 - QString `getCurrentLayer` ()
 - QRgb `getCurrentColor` ()
 - QString `getCurrentLineType` ()
 - QString `getCurrentLineWidth` ()
 - void `undo` ()
 - void `redo` ()
 - bool `isShiftPressed` ()
 - void `setShiftPressed` ()
 - void `setShiftReleased` ()
 - void `deletePressed` ()
 - void `escapePressed` ()
 - void `makeLayerActive` ()
 - void `layerManager` ()
 - void `layerPrevious` ()
 - void `zoomRealtime` ()
 - void `zoomPrevious` ()
 - void `zoomWindow` ()
 - void `zoomDynamic` ()
 - void `zoomScale` ()
 - void `zoomCenter` ()
 - void `zoomIn` ()
 - void `zoomOut` ()
 - void `zoomSelected` ()
 - void `zoomAll` ()
 - void `zoomExtents` ()
 - void `panrealtime` ()
 - void `panpoint` ()
 - void `panLeft` ()
 - void `panRight` ()
 - void `panUp` ()
 - void `panDown` ()
- MainWindow::panDown.*
- void `dayVision` ()
- MainWindow::dayVision.*
- void `nightVision` ()
- MainWindow::nightVision.*
- void `doNothing` ()

Public Member Functions

- `MainWindow` ()
MainWindow::MainWindow.
- `~MainWindow` ()
MainWindow::~MainWindow.
- `MdiArea *` `getMdiArea` ()
MainWindow::getMdiArea.
- `MainWindow *` `getApplication` ()
MainWindow::getApplication.
- `MdiWindow *` `activeMdiWindow` ()

- `View * activeView ()`
- `QGraphicsScene * activeScene ()`
- `QUndoStack * activeUndoStack ()`
- `void setUndoCleanIcon (bool opened)`
- `virtual void updateMenuToolbarStatusbar ()`
MainWindow::updateMenuToolbarStatusbar.
- `std::string actuator (std::string command)`
MainWindow::actuator.
- `std::string run_script_file (std::string fname)`
MainWindow::run_script_file.
- `std::string run_script (std::vector< std::string > script)`
A basic line-by-line script processor to allow for extensions to the program.
- `void LoadCommand (QString cmdName)`
- `bool isCommandActive ()`
- `QString activeCommand ()`
- `QIcon create_icon (QString stub)`
- `void create_toolbar (QToolBar *toolbar, std::string label, std::vector< std::string > entries)`
- `QString platformString ()`
- `void nativeAlert (const QString &txt)`
- `void nativeBlinkPrompt ()`
- `void nativeSetPromptPrefix (const QString &txt)`
- `void nativeAppendPromptHistory (const QString &txt)`
- `void nativeEnablePromptRapidFire ()`
- `void nativeDisablePromptRapidFire ()`
- `void nativeInitCommand ()`
- `void nativeEndCommand ()`
- `void nativeEnableMoveRapidFire ()`
- `void nativeDisableMoveRapidFire ()`
- `void nativeNewFile ()`
- `void nativeOpenFile ()`
- `void nativeExit ()`
- `void nativeTipOfDay ()`
- `void nativeWindowCascade ()`
- `void nativeWindowTile ()`
- `void nativeWindowClose ()`
- `void nativeWindowCloseAll ()`
- `void nativeWindowNext ()`
- `void nativeWindowPrevious ()`
- `void nativeMessageBox (const QString &type, const QString &title, const QString &text)`
MainWindow::nativeMessageBox type title text.
- `void nativePrintArea (EmbReal x, EmbReal y, EmbReal w, EmbReal h)`
MainWindow::nativePrintArea x y w h.
- `void nativeSetBackgroundColor (uint8_t r, uint8_t g, uint8_t b)`
- `void nativeSetCrossHairColor (uint8_t r, uint8_t g, uint8_t b)`
- `void nativeSetGridColor (uint8_t r, uint8_t g, uint8_t b)`
- `QString nativeTextFont ()`
- `EmbReal nativeTextSize ()`
- `EmbReal nativeTextAngle ()`
- `bool nativeTextBold ()`
- `bool nativeTextItalic ()`
- `bool nativeTextUnderline ()`
- `bool nativeTextStrikeOut ()`
- `bool nativeTextOverline ()`
- `void nativePreviewOn (int clone, int mode, EmbReal x, EmbReal y, EmbReal data)`

- void nativePreviewOff ()
- void nativeVulcanize ()
- void nativeClearRubber ()
- bool nativeAllowRubber ()
- void nativeSpareRubber (qint64 id)
- void nativeSetRubberMode (int mode)
- void nativeSetRubberPoint (const QString &key, EmbReal x, EmbReal y)
- void nativeSetRubberText (const QString &key, const QString &txt)
- void nativeAddTextMulti (const QString &str, EmbReal x, EmbReal y, EmbReal rot, bool fill, int rubberMode)
- void nativeAddTextSingle (const QString &str, EmbReal x, EmbReal y, EmbReal rot, bool fill, int rubberMode)
- void nativeAddInfiniteLine (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal rot)
- void nativeAddRay (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal rot)
- void nativeAddLine (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal rot, int rubberMode)
- void nativeAddTriangle (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal x3, EmbReal y3, EmbReal rot, bool fill)
- void nativeAddRectangle (EmbReal x, EmbReal y, EmbReal w, EmbReal h, EmbReal rot, bool fill, int rubberMode)
- void nativeAddRoundedRectangle (EmbReal x, EmbReal y, EmbReal w, EmbReal h, EmbReal rad, EmbReal rot, bool fill)
- void nativeAddArc (EmbReal startX, EmbReal startY, EmbReal midX, EmbReal midY, EmbReal endX, EmbReal endY, int rubberMode)
- void nativeAddCircle (EmbReal centerX, EmbReal centerY, EmbReal radius, bool fill, int rubberMode)
- void nativeAddSlot (EmbReal centerX, EmbReal centerY, EmbReal diameter, EmbReal length, EmbReal rot, bool fill, int rubberMode)
- void nativeAddEllipse (EmbReal centerX, EmbReal centerY, EmbReal width, EmbReal height, EmbReal rot, bool fill, int rubberMode)
- void nativeAddPoint (EmbReal x, EmbReal y)
- void nativeAddRegularPolygon (EmbReal centerX, EmbReal centerY, quint16 sides, uint8_t mode, EmbReal rad, EmbReal rot, bool fill, int rubberMode)
- void nativeAddPolygon (EmbReal startX, EmbReal startY, const QPainterPath &p, int rubberMode)
- void nativeAddPolyline (EmbReal startX, EmbReal startY, const QPainterPath &p, int rubberMode)
- void nativeAddPath (EmbReal startX, EmbReal startY, const QPainterPath &p, int rubberMode)
- void nativeAddHorizontalDimension (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal legHeight)
- void nativeAddVerticalDimension (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal legHeight)
- void nativeAddImage (const QString &img, EmbReal x, EmbReal y, EmbReal w, EmbReal h, EmbReal rot)
- void nativeAddDimLeader (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal rot, int rubberMode)
- void nativeSetCursorShape (const QString &str)
- EmbReal nativeCalculateAngle (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2)
- EmbReal nativeCalculateDistance (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2)
- EmbReal nativePerpendicularDistance (EmbReal px, EmbReal py, EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2)
- int nativeNumSelected ()
- void nativeSelectAll ()
- void nativeAddToSelection (const QPainterPath path, Qt::ItemSelectionMode mode)
- void nativeClearSelection ()
- void nativeDeleteSelected ()
 - MainWindow::nativeDeleteSelected.*
- void nativeCutSelected (EmbReal x, EmbReal y)
 - MainWindow::nativeCutSelected x y.*
- void nativeCopySelected (EmbReal x, EmbReal y)
 - MainWindow::nativeCopySelected x y.*
- void nativePasteSelected (EmbReal x, EmbReal y)
 - MainWindow::nativePasteSelected x y.*

- void `nativeMoveSelected (EmbReal dx, EmbReal dy)`
MainWindow::nativeMoveSelected dx dy.
- void `nativeScaleSelected (EmbReal x, EmbReal y, EmbReal factor)`
MainWindow::nativeScaleSelected x y factor.
- void `nativeRotateSelected (EmbReal x, EmbReal y, EmbReal rot)`
MainWindow::nativeRotateSelected x y rot.
- void `nativeMirrorSelected (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2)`
MainWindow::nativeMirrorSelected x1 y1 x2 y2.
- `EmbReal nativeQSnapX ()`
MainWindow::nativeQSnapX.
- `EmbReal nativeQSnapY ()`
MainWindow::nativeQSnapY.
- `EmbReal nativeMouseX ()`
MainWindow::nativeMouseX.
- `EmbReal nativeMouseY ()`
MainWindow::nativeMouseY.

Public Attributes

- `QString settings_general_language`
- `QString settings_general_icon_theme`
- `int settings_general_icon_size`
- `bool settings_general_mdi_bg_use_logo`
- `bool settings_general_mdi_bg_use_texture`
- `bool settings_general_mdi_bg_use_color`
- `QString settings_general_mdi_bg_logo`
- `QString settings_general_mdi_bg_texture`
- `QRgb settings_general_mdi_bg_color`
- `bool settings_general_tip_of_the_day`
- `quint16 settings_general_current_tip`
- `bool settings_general_system_help_browser`
- `bool settings_general_check_for_updates`
- `bool settings_display_use_opengl`
- `bool settings_display_renderhint_aa`
- `bool settings_display_renderhint_text_aa`
- `bool settings_display_renderhint_smooth_pix`
- `bool settings_display_renderhint_high_aa`
- `bool settings_display_renderhint_noncosmetic`
- `bool settings_display_show_scrollbars`
- `int settings_display_scrollbar_widget_num`
- `QRgb settings_display_crosshair_color`
- `QRgb settings_display_bg_color`
- `QRgb settings_display_selectbox_left_color`
- `QRgb settings_display_selectbox_left_fill`
- `QRgb settings_display_selectbox_right_color`
- `QRgb settings_display_selectbox_right_fill`
- `uint8_t settings_display_selectbox_alpha`
- `EmbReal settings_display_zoomscale_in`
- `EmbReal settings_display_zoomscale_out`
- `uint8_t settings_display_crosshair_percent`
- `QString settings_display_units`
- `QRgb settings_prompt_text_color`

- QRgb `settings_prompt_bg_color`
- QString `settings_prompt_font_family`
- QString `settings_prompt_font_style`
- uint8_t `settings_prompt_font_size`
- bool `settings_prompt_save_history`
- bool `settings_prompt_save_history_as_html`
- QString `settings_prompt_save_history_filename`
- QString `settings_opensave_custom_filter`
- QString `settings_opensave_open_format`
- bool `settings_opensave_open_thumbnail`
- QString `settings_opensave_save_format`
- bool `settings_opensave_save_thumbnail`
- uint8_t `settings_opensave_recent_max_files`
- QStringList `settings_opensave_recent_list_of_files`
- QString `settings_opensave_recent_directory`
- uint8_t `settings_opensave_trim_dst_num_jumps`
- QString `settings_printing_default_device`
- bool `settings_printing_use_last_device`
- bool `settings_printing_disable_bg`
- bool `settings_grid_show_on_load`
- bool `settings_grid_show_origin`
- bool `settings_grid_color_match_crosshair`
- QRgb `settings_grid_color`
- bool `settings_grid_load_from_file`
- QString `settings_grid_type`
- bool `settings_grid_center_on_origin`
- EmbReal `settings_grid_center_x`
- EmbReal `settings_grid_center_y`
- EmbReal `settings_grid_size_x`
- EmbReal `settings_grid_size_y`
- EmbReal `settings_grid_spacing_x`
- EmbReal `settings_grid_spacing_y`
- EmbReal `settings_grid_size_radius`
- EmbReal `settings_grid_spacing_radius`
- EmbReal `settings_grid_spacing_angle`
- bool `settings_ruler_show_on_load`
- bool `settings_ruler_metric`
- QRgb `settings_ruler_color`
- uint8_t `settings_ruler_pixel_size`
- bool `settings_qsnap_enabled`
- QRgb `settings_qsnap_locator_color`
- uint8_t `settings_qsnap_locator_size`
- uint8_t `settings_qsnap_aperture_size`
- bool `settings_qsnap_endpoint`
- bool `settings_qsnap_midpoint`
- bool `settings_qsnap_center`
- bool `settings_qsnap_node`
- bool `settings_qsnap_quadrant`
- bool `settings_qsnap_intersection`
- bool `settings_qsnap_extension`
- bool `settings_qsnap_insertion`
- bool `settings_qsnap_perpendicular`
- bool `settings_qsnap_tangent`
- bool `settings_qsnap_nearest`
- bool `settings_qsnap_apparent`

- bool `settings_qsnap_parallel`
- bool `settings_lwt_show_lwt`
- bool `settings_lwt_real_render`
- EmbReal `settings_lwt_default_lwt`
- bool `settings_selection_mode_pickfirst`
- bool `settings_selection_mode_pickadd`
- bool `settings_selection_mode_pickdrag`
- QRgb `settings_selection_coolgrip_color`
- QRgb `settings_selection_hotgrip_color`
- uint8_t `settings_selection_grip_size`
- uint8_t `settings_selection_pickbox_size`
- QString `settings_text_font`
- EmbReal `settings_text_size`
- EmbReal `settings_text_angle`
- bool `settings_text_style_bold`
- bool `settings_text_style_italic`
- bool `settings_text_style_underline`
- bool `settings_text_style_overline`
- bool `settings_text_style_strikeout`
- MainWindow * `mainWin`
- MdiArea * `mdiArea`
- CmdPrompt * `prompt`
- PropertyEditor * `dockPropEdit`
- UndoEditor * `dockUndoEdit`
- StatusBar * `statusbar`
- QList< QGraphicsItem * > `cutCopyObjectList`
- QAction * `actionHash` [200]
- QHash< QString, QToolBar * > `toolbarHash`
- QHash< QString, QMenu * > `menuHash`
- QString `formatFilterOpen`
- QString `formatFilterSave`

Protected Member Functions

- virtual void `resizeEvent` (QResizeEvent *)
MainWindow::resizeEvent.
- void `closeEvent` (QCloseEvent *event)
MainWindow::closeEvent.
- QAction * `getFileSeparator` ()
MainWindow::getFileSeparator.
- void `loadFormats` ()
MainWindow::loadFormats.
- QMdiSubWindow * `findMdiWindow` (const QString &fileName)
MainWindow::findMdiWindow.
- void `createAllActions` ()
MainWindow::createAllActions.
- void `createAllToolbars` ()
MainWindow::createAllToolbars.
- void `createPanToolbar` ()
- void `createIconToolbar` ()
- void `createHelpToolbar` ()
- void `createLayerToolbar` ()

```
    MainWindow::createLayerToolbar.  
• void createPropertiesToolbar ()  
    MainWindow::createPropertiesToolbar.  
• void createTextToolbar ()  
    MainWindow::createTextToolbar.  
• void createPromptToolbar ()  
    MainWindow::createPromptToolbar.  
• void createAllMenus ()  
• void createFileMenu ()  
    MainWindow::createFileMenu.  
• void createEditMenu ()  
• void createViewMenu ()  
• void createSettingsMenu ()  
• void createWindowMenu ()  
• void createHelpMenu ()
```

Protected Attributes

- bool shiftKeyPressedState
- QByteArray layoutState
- int numOfDocs
- int docIndex
- QList< MdiWindow * > listMdiWin
- QString openFilePath
- QAction * myFileSeparator
- QWizard * wizardTipOfTheDay
- QLabel * labelTipOfTheDay
- QCheckBox * checkBoxTipOfTheDay
- QStringList listTipOfTheDay
- QToolBar * toolbarFile
- QToolBar * toolbarEdit
- QToolBar * toolbarView
- QToolBar * toolbarZoom
- QToolBar * toolbarPan
- QToolBar * toolbarIcon
- QToolBar * toolbarHelp
- QToolBar * toolbarLayer
- QToolBar * toolbarText
- QToolBar * toolbarProperties
- QToolBar * toolbarPrompt
- QComboBox * layerSelector
- QComboBox * colorSelector
- QComboBox * linetypeSelector
- QComboBox * linewidthSelector
- QFontComboBox * textFontSelector
- QComboBox * textSizeSelector
- QMenu * fileMenu
- QMenu * editMenu
- QMenu * viewMenu
- QMenu * settingsMenu
- QMenu * windowMenu
- QMenu * helpMenu
- QMenu * recentMenu
- QMenu * zoomMenu
- QMenu * panMenu

Private Slots

- void [hideUnimplemented \(\)](#)
MainWindow::hideUnimplemented.

17.64.1 Detailed Description

The [MainWindow](#) class.

17.64.2 Constructor & Destructor Documentation**17.64.2.1 MainWindow() [MainWindow \(\)](#)**

[MainWindow::MainWindow.](#)

17.64.2.2 ~MainWindow() [~MainWindow \(\)](#)

[MainWindow::~MainWindow.](#)

17.64.3 Member Function Documentation**17.64.3.1 about void about () [slot]****17.64.3.2 activeCommand() [QString activeCommand \(\) \[inline\]](#)****17.64.3.3 activeMdiWindow() [MdiWindow * activeMdiWindow \(\)](#)****17.64.3.4 activeScene() [QGraphicsScene * activeScene \(\)](#)****17.64.3.5 activeUndoStack() [QUndoStack * activeUndoStack \(\)](#)****17.64.3.6 activeView() [View * activeView \(\)](#)****17.64.3.7 actuator() [std::string actuator \(std::string command \)](#)**

[MainWindow::actuator.](#)

Parameters

| | |
|----------------|---------------------------------|
| <i>command</i> | <input type="button" value=""/> |
|----------------|---------------------------------|

17.64.3.8 buttonTipOfTheDayClicked void buttonTipOfTheDayClicked (int *button*) [slot]

17.64.3.9 changelog void changelog () [slot]

17.64.3.10 checkBoxTipOfTheDayStateChanged void checkBoxTipOfTheDayStateChanged (int *checked*) [slot]

17.64.3.11 checkForUpdates void checkForUpdates () [slot]

17.64.3.12 closeEvent() void closeEvent (QCloseEvent * *event*) [protected]

MainWindow::closeEvent.

Parameters

| | |
|--------------|---------------------------------|
| <i>event</i> | <input type="button" value=""/> |
|--------------|---------------------------------|

17.64.3.13 closeToolBar void closeToolBar (QAction * *action*) [slot]

MainWindow::closeToolBar.

Parameters

| | |
|---------------|---------------------------------|
| <i>action</i> | <input type="button" value=""/> |
|---------------|---------------------------------|

```
17.64.3.14 colorSelectorIndexChanged void colorSelectorIndexChanged (
    int index ) [slot]
```

```
17.64.3.15 copy void copy ( ) [slot]
```

```
17.64.3.16 create_icon() QIcon create_icon (
    QString stub )
```

```
17.64.3.17 create_toolbar() void create_toolbar (
    QToolBar * toolbar,
    std::string label,
    std::vector< std::string > entries )
```

```
17.64.3.18 createAllActions() void createAllActions ( ) [protected]
```

MainWindow::createAllActions.

Todo Set What's This Context Help to statusTip for now so there is some infos there. Make custom whats this context help popup with more descriptive help than just the status bar/tip one liner(short but not real long) with a hyperlink in the custom popup at the bottom to open full help file description. Ex: like wxPython AGW's SuperToolTip. ACTION->setWhatsThis(statusTip);

Finish All Commands ... <.<

```
17.64.3.19 createAllMenus() void createAllMenus ( ) [protected]
```

```
17.64.3.20 createAllToolbars() void createAllToolbars ( ) [protected]
```

MainWindow::createAllToolbars.

```
17.64.3.21 createEditMenu() void createEditMenu ( ) [protected]
```

17.64.3.22 `createFileMenu()` void createFileMenu () [protected]

[MainWindow::createFileMenu](#).

17.64.3.23 `createHelpMenu()` void createHelpMenu () [protected]

17.64.3.24 `createHelpToolbar()` void createHelpToolbar () [protected]

17.64.3.25 `createIconToolbar()` void createIconToolbar () [protected]

17.64.3.26 `createLayerToolbar()` void createLayerToolbar () [protected]

[MainWindow::createLayerToolbar](#).

17.64.3.27 `createPanToolbar()` void createPanToolbar () [protected]

17.64.3.28 `createPromptToolbar()` void createPromptToolbar () [protected]

[MainWindow::createPromptToolbar](#).

17.64.3.29 `createPropertiesToolbar()` void createPropertiesToolbar () [protected]

[MainWindow::createPropertiesToolbar](#).

17.64.3.30 `createSettingsMenu()` void createSettingsMenu () [protected]

17.64.3.31 `createTextToolbar()` void createTextToolbar () [protected]

[MainWindow::createTextToolbar](#).

17.64.3.32 `createViewMenu()` void createViewMenu () [protected]

17.64.3.33 `createWindowMenu()` void createWindowMenu () [protected]

17.64.3.34 `cut` void cut () [slot]

17.64.3.35 `dayVision` void dayVision () [slot]

[MainWindow::dayVision](#).

17.64.3.36 `deletePressed` void deletePressed () [slot]

17.64.3.37 `designDetails` void designDetails () [slot]

17.64.3.38 `disableMoveRapidFire` void disableMoveRapidFire () [slot]

17.64.3.39 `disablePromptRapidFire` void disablePromptRapidFire () [slot]

17.64.3.40 `doNothing` void doNothing () [slot]

17.64.3.41 `enableMoveRapidFire` void enableMoveRapidFire () [slot]

17.64.3.42 `enablePromptRapidFire` void enablePromptRapidFire () [slot]

17.64.3.43 `escapePressed` void escapePressed () [slot]

17.64.3.44 `exit` void exit () [slot]

[MainWindow::exit](#).

17.64.3.45 `findMdiWindow()` QMdiSubWindow * findMdiWindow (const QString & fileName) [protected]

[MainWindow::findMdiWindow](#).

Parameters

| | |
|-----------------|----------------------|
| <i>fileName</i> | <input type="text"/> |
|-----------------|----------------------|

Returns

17.64.3.46 floatingChangedToolBar void floatingChangedToolBar (bool *isFloating*) [slot]

[MainWindow::floatingChangedToolBar.](#)

Parameters

| | |
|-------------------|----------------------|
| <i>isFloating</i> | <input type="text"/> |
|-------------------|----------------------|

17.64.3.47 getAction QAction * getAction (int *actionEnum*) [slot]

[MainWindow::getAction.](#)

Parameters

| | |
|-------------------|----------------------|
| <i>actionEnum</i> | <input type="text"/> |
|-------------------|----------------------|

Returns

17.64.3.48 getApplication() MainWindow * getApplication ()

[MainWindow::getApplication.](#)

Returns

17.64.3.49 getCurrentColor QRgb getCurrentColor () [slot]

17.64.3.50 getCurrentLayer `QString getCurrentLayer () [slot]`

17.64.3.51 getCurrentLineType `QString getCurrentLineType () [slot]`

17.64.3.52 getCurrentLineWeight `QString getCurrentLineWeight () [slot]`

17.64.3.53 getFileSeparator() `QAction * getFileSeparator () [protected]`

[MainWindow::getFileSeparator](#).

Returns

17.64.3.54 getMdiArea() `MdiArea * getMdiArea ()`

[MainWindow::getMdiArea](#).

Returns

17.64.3.55 help `void help () [slot]`

17.64.3.56 hideUnimplemented `void hideUnimplemented () [private], [slot]`

[MainWindow::hideUnimplemented](#).

17.64.3.57 iconResize `void iconResize (int iconSize) [slot]`

17.64.3.58 isCommandActive() bool isCommandActive () [inline]

17.64.3.59 isShiftPressed bool isShiftPressed () [slot]

17.64.3.60 layerManager void layerManager () [slot]

17.64.3.61 layerPrevious void layerPrevious () [slot]

17.64.3.62 layerSelectorIndexChanged void layerSelectorIndexChanged (int *index*) [slot]

17.64.3.63 linetypeSelectorIndexChanged void linetypeSelectorIndexChanged (int *index*) [slot]

17.64.3.64 linewidthSelectorIndexChanged void linewidthSelectorIndexChanged (int *index*) [slot]

17.64.3.65 LoadCommand() void LoadCommand (QString *cmdName*)

NOTE: Every QScriptProgram must have a unique function name to call. If every function was called [main\(\)](#), then the QScriptEngine would only call the last script evaluated (which happens to be [main\(\)](#) in another script). Thus, by adding the cmdName before [main\(\)](#), it becomes line_main(), circle_main(), etc... Do not change this code unless you really know what you are doing. I mean it.

17.64.3.66 loadFormats() void loadFormats () [protected]

[MainWindow::loadFormats](#).

17.64.3.67 logPromptInput void logPromptInput (const QString & *txt*) [slot]

17.64.3.68 makeLayerActive void makeLayerActive () [slot]

17.64.3.69 nativeAddArc() void nativeAddArc (

```
EmbReal startX,
EmbReal startY,
EmbReal midX,
EmbReal midY,
EmbReal endX,
EmbReal endY,
int rubberMode )
```

17.64.3.70 nativeAddCircle() void nativeAddCircle (

```
EmbReal centerX,
EmbReal centerY,
EmbReal radius,
bool fill,
int rubberMode )
```

17.64.3.71 nativeAddDimLeader() void nativeAddDimLeader (

```
EmbReal x1,
EmbReal y1,
EmbReal x2,
EmbReal y2,
EmbReal rot,
int rubberMode )
```

17.64.3.72 nativeAddEllipse() void nativeAddEllipse (

```
EmbReal centerX,
EmbReal centerY,
EmbReal width,
EmbReal height,
EmbReal rot,
bool fill,
int rubberMode )
```

17.64.3.73 nativeAddHorizontalDimension() void nativeAddHorizontalDimension (

```
EmbReal x1,
EmbReal y1,
EmbReal x2,
EmbReal y2,
EmbReal legHeight )
```

```
17.64.3.74 nativeAddImage() void nativeAddImage (
    const QString & img,
    EmbReal x,
    EmbReal y,
    EmbReal w,
    EmbReal h,
    EmbReal rot )
```

```
17.64.3.75 nativeAddInfiniteLine() void nativeAddInfiniteLine (
    EmbReal x1,
    EmbReal y1,
    EmbReal x2,
    EmbReal y2,
    EmbReal rot )
```

```
17.64.3.76 nativeAddLine() void nativeAddLine (
    EmbReal x1,
    EmbReal y1,
    EmbReal x2,
    EmbReal y2,
    EmbReal rot,
    int rubberMode )
```

```
17.64.3.77 nativeAddPath() void nativeAddPath (
    EmbReal startX,
    EmbReal startY,
    const QPainterPath & p,
    int rubberMode )
```

```
17.64.3.78 nativeAddPoint() void nativeAddPoint (
    EmbReal x,
    EmbReal y )
```

```
17.64.3.79 nativeAddPolygon() void nativeAddPolygon (
    EmbReal startX,
    EmbReal startY,
    const QPainterPath & p,
    int rubberMode )
```

```
17.64.3.80 nativeAddPolyline() void nativeAddPolyline (
    EmbReal startX,
    EmbReal startY,
    const QPainterPath & p,
    int rubberMode )
```

```
17.64.3.81 nativeAddRay() void nativeAddRay (
    EmbReal x1,
    EmbReal y1,
    EmbReal x2,
    EmbReal y2,
    EmbReal rot )
```

```
17.64.3.82 nativeAddRectangle() void nativeAddRectangle (
    EmbReal x,
    EmbReal y,
    EmbReal w,
    EmbReal h,
    EmbReal rot,
    bool fill,
    int rubberMode )
```

```
17.64.3.83 nativeAddRegularPolygon() void nativeAddRegularPolygon (
    EmbReal centerX,
    EmbReal centerY,
    quint16 sides,
    uint8_t mode,
    EmbReal rad,
    EmbReal rot,
    bool fill )
```

```
17.64.3.84 nativeAddRoundedRectangle() void nativeAddRoundedRectangle (
    EmbReal x,
    EmbReal y,
    EmbReal w,
    EmbReal h,
    EmbReal rad,
    EmbReal rot,
    bool fill )
```

```
17.64.3.85 nativeAddSlot() void nativeAddSlot (
```

```
    EmbReal centerX,
```

```
    EmbReal centerY,
```

```
    EmbReal diameter,
```

```
    EmbReal length,
```

```
    EmbReal rot,
```

```
    bool fill,
```

```
    int rubberMode )
```

```
17.64.3.86 nativeAddTextMulti() void nativeAddTextMulti (
```

```
    const QString & str,
```

```
    EmbReal x,
```

```
    EmbReal y,
```

```
    EmbReal rot,
```

```
    bool fill,
```

```
    int rubberMode )
```

```
17.64.3.87 nativeAddTextSingle() void nativeAddTextSingle (
```

```
    const QString & str,
```

```
    EmbReal x,
```

```
    EmbReal y,
```

```
    EmbReal rot,
```

```
    bool fill,
```

```
    int rubberMode )
```

```
17.64.3.88 nativeAddToSelection() void nativeAddToSelection (
```

```
    const QPainterPath path,
```

```
    Qt::ItemSelectionMode mode )
```

```
17.64.3.89 nativeAddTriangle() void nativeAddTriangle (
```

```
    EmbReal x1,
```

```
    EmbReal y1,
```

```
    EmbReal x2,
```

```
    EmbReal y2,
```

```
    EmbReal x3,
```

```
    EmbReal y3,
```

```
    EmbReal rot,
```

```
    bool fill )
```

17.64.3.90 nativeAddVerticalDimension() void nativeAddVerticalDimension (

```
EmbReal x1,  
EmbReal y1,  
EmbReal x2,  
EmbReal y2,  
EmbReal legHeight )
```

17.64.3.91 nativeAlert() void nativeAlert (

```
const QString & txt )
```

17.64.3.92 nativeAllowRubber() bool nativeAllowRubber ()

17.64.3.93 nativeAppendPromptHistory() void nativeAppendPromptHistory (

```
const QString & txt )
```

17.64.3.94 nativeBlinkPrompt() void nativeBlinkPrompt ()

17.64.3.95 nativeCalculateAngle() EmbReal nativeCalculateAngle (

```
EmbReal x1,  
EmbReal y1,  
EmbReal x2,  
EmbReal y2 )
```

17.64.3.96 nativeCalculateDistance() EmbReal nativeCalculateDistance (

```
EmbReal x1,  
EmbReal y1,  
EmbReal x2,  
EmbReal y2 )
```

17.64.3.97 nativeClearRubber() void nativeClearRubber ()

17.64.3.98 nativeClearSelection() void nativeClearSelection ()

17.64.3.99 nativeCopySelected() void nativeCopySelected (

```
EmbReal x,  
EmbReal y )
```

MainWindow::nativeCopySelected x y.

17.64.3.100 nativeCutSelected() void nativeCutSelected (

```
EmbReal x,  
EmbReal y )
```

MainWindow::nativeCutSelected x y.

17.64.3.101 nativeDeleteSelected() void nativeDeleteSelected ()

MainWindow::nativeDeleteSelected.

17.64.3.102 nativeDisableMoveRapidFire() void nativeDisableMoveRapidFire ()

17.64.3.103 nativeDisablePromptRapidFire() void nativeDisablePromptRapidFire ()

17.64.3.104 nativeEnableMoveRapidFire() void nativeEnableMoveRapidFire ()

17.64.3.105 nativeEnablePromptRapidFire() void nativeEnablePromptRapidFire ()

17.64.3.106 nativeEndCommand() void nativeEndCommand ()

17.64.3.107 nativeExit() void nativeExit ()

17.64.3.108 nativeInitCommand() void nativeInitCommand ()

17.64.3.109 nativeMessageBox() void nativeMessageBox (
 const QString & type,
 const QString & title,
 const QString & text)

MainWindow::nativeMessageBox *type title text.*

17.64.3.110 nativeMirrorSelected() void nativeMirrorSelected (
 EmbReal x1,
 EmbReal y1,
 EmbReal x2,
 EmbReal y2)

MainWindow::nativeMirrorSelected *x1 y1 x2 y2.*

17.64.3.111 nativeMouseX() EmbReal nativeMouseX ()

MainWindow::nativeMouseX.

Returns

17.64.3.112 nativeMouseY() EmbReal nativeMouseY ()

MainWindow::nativeMouseY.

Returns

17.64.3.113 nativeMoveSelected() void nativeMoveSelected (
 EmbReal dx,
 EmbReal dy)

MainWindow::nativeMoveSelected *dx dy.*

17.64.3.114 nativeNewFile() void nativeNewFile ()

17.64.3.115 nativeNumSelected() int nativeNumSelected ()

17.64.3.116 nativeOpenFile() void nativeOpenFile ()

17.64.3.117 nativePasteSelected() void nativePasteSelected (EmbReal x,
EmbReal y)

MainWindow::nativePasteSelected x.y.

17.64.3.118 nativePerpendicularDistance() EmbReal nativePerpendicularDistance (EmbReal px,
EmbReal py,
EmbReal x1,
EmbReal y1,
EmbReal x2,
EmbReal y2)

17.64.3.119 nativePreviewOff() void nativePreviewOff ()

17.64.3.120 nativePreviewOn() void nativePreviewOn (int clone,
int mode,
EmbReal x,
EmbReal y,
EmbReal data)

17.64.3.121 nativePrintArea() void nativePrintArea (EmbReal x,
EmbReal y,
EmbReal w,
EmbReal h)

MainWindow::nativePrintArea x y w h.

17.64.3.122 nativeQSnapX() `EmbReal nativeQSnapX ()`

`MainWindow::nativeQSnapX.`

Returns

17.64.3.123 nativeQSnapY() `EmbReal nativeQSnapY ()`

`MainWindow::nativeQSnapY.`

Returns

17.64.3.124 nativeRotateSelected() `void nativeRotateSelected (`

```
    EmbReal x,  
    EmbReal y,  
    EmbReal rot )
```

`MainWindow::nativeRotateSelected x y rot.`

17.64.3.125 nativeScaleSelected() `void nativeScaleSelected (`

```
    EmbReal x,  
    EmbReal y,  
    EmbReal factor )
```

`MainWindow::nativeScaleSelected x y factor.`

17.64.3.126 nativeSelectAll() `void nativeSelectAll ()`**17.64.3.127 nativeSetBackgroundColor()** `void nativeSetBackgroundColor (`

```
    uint8_t r,  
    uint8_t g,  
    uint8_t b )
```

17.64.3.128 nativeSetCrossHairColor() void nativeSetCrossHairColor (uint8_t r, uint8_t g, uint8_t b)

17.64.3.129 nativeSetCursorShape() void nativeSetCursorShape (const QString & str)

17.64.3.130 nativeSetGridColor() void nativeSetGridColor (uint8_t r, uint8_t g, uint8_t b)

17.64.3.131 nativeSetPromptPrefix() void nativeSetPromptPrefix (const QString & txt)

17.64.3.132 nativeSetRubberMode() void nativeSetRubberMode (int mode)

17.64.3.133 nativeSetRubberPoint() void nativeSetRubberPoint (const QString & key, EmbReal x, EmbReal y)

17.64.3.134 nativeSetRubberText() void nativeSetRubberText (const QString & key, const QString & txt)

17.64.3.135 nativeSpareRubber() void nativeSpareRubber (qint64 id)

17.64.3.136 nativeTextAngle() EmbReal nativeTextAngle ()

17.64.3.137 nativeTextBold() bool nativeTextBold ()

17.64.3.138 nativeTextFont() QString nativeTextFont ()

17.64.3.139 nativeTextItalic() bool nativeTextItalic ()

17.64.3.140 nativeTextOverline() bool nativeTextOverline ()

17.64.3.141 nativeTextSize() EmbReal nativeTextSize ()

17.64.3.142 nativeTextStrikeOut() bool nativeTextStrikeOut ()

17.64.3.143 nativeTextUnderline() bool nativeTextUnderline ()

17.64.3.144 nativeTipOfDay() void nativeTipOfDay ()

17.64.3.145 nativeVulcanize() void nativeVulcanize ()

17.64.3.146 nativeWindowCascade() void nativeWindowCascade ()

17.64.3.147 nativeWindowClose() void nativeWindowClose ()

17.64.3.148 nativeWindowCloseAll() void nativeWindowCloseAll ()

17.64.3.149 nativeWindowNext() void nativeWindowNext ()

17.64.3.150 nativeWindowPrevious() void nativeWindowPrevious ()

17.64.3.151 nativeWindowTile() void nativeWindowTile ()

17.64.3.152 newFile void newFile () [slot]

[MainWindow::newFile](#).

17.64.3.153 nightVision void nightVision () [slot]

[MainWindow::nightVision](#).

17.64.3.154 onCloseMdiWin void onCloseMdiWin (
 [MdiWindow](#) * theMdiWin) [virtual], [slot]

[MainWindow::onCloseMdiWin](#).

Parameters

[theMdiWin](#)

17.64.3.155 onCloseWindow void onCloseWindow () [slot]

[MainWindow::onCloseWindow](#).

17.64.3.156 onWindowActivated void onWindowActivated (
 [QMdiSubWindow](#) * w) [slot]

[MainWindow::onWindowActivated](#).

Parameters

| | |
|----------|----------------------|
| <i>w</i> | <input type="text"/> |
|----------|----------------------|

17.64.3.157 openFile void openFile (bool recent = false, const QString & recentFile = "") [slot]

[MainWindow::openFile](#).

Parameters

| | |
|-------------------|----------------------|
| <i>recent</i> | <input type="text"/> |
| <i>recentFile</i> | <input type="text"/> |

17.64.3.158 openFilesSelected void openFilesSelected (const QStringList & filesToOpen) [slot]

[MainWindow::openFilesSelected](#).

Parameters

| | |
|--------------------|----------------------|
| <i>filesToOpen</i> | <input type="text"/> |
|--------------------|----------------------|

17.64.3.159 openrecentfile void openrecentfile () [slot]

[MainWindow::openrecentfile](#).

17.64.3.160 panDown void panDown () [slot]

[MainWindow::panDown](#).

17.64.3.161 panLeft void panLeft () [slot]

17.64.3.162 panpoint void panpoint () [slot]

17.64.3.163 panrealtime void panrealtime () [slot]

17.64.3.164 panRight void panRight () [slot]

17.64.3.165 panUp void panUp () [slot]

17.64.3.166 paste void paste () [slot]

17.64.3.167 pickAddModeToggled void pickAddModeToggled () [slot]

17.64.3.168 platformString() QString platformString ()

17.64.3.169 print void print () [slot]

17.64.3.170 promptHistoryAppended void promptHistoryAppended (const QString & txt) [slot]

17.64.3.171 promptInputNext void promptInputNext () [slot]

17.64.3.172 promptInputPrevious void promptInputPrevious () [slot]

17.64.3.173 quit void quit () [slot]

[MainWindow::quit](#).

17.64.3.174 readSettings void readSettings () [slot]

[MainWindow::readSettings](#).

17.64.3.175 recentMenuAboutToShow void recentMenuAboutToShow () [slot]

[MainWindow::recentMenuAboutToShow](#).

17.64.3.176 redo void redo () [slot]

17.64.3.177 resizeEvent() void resizeEvent (QResizeEvent * e) [protected], [virtual]

[MainWindow::resizeEvent](#).

Parameters

| | |
|---|--|
| e | |
|---|--|

17.64.3.178 run_script() std::string run_script (std::vector< std::string > script)

A basic line-by-line script processor to allow for extensions to the program.

Since the actuator uses command line style parsing, a script is just a text file with each line a compatible command.

It should be stressed that this has no control flow or purpose. We don't want this to be hacked into a full scripting language that could cause havoc on the user's system.

However, it may be useful to set and get variables and define macros: neither of these will allow for endless loops, stack overflow or other problems that third-party scripts could introduce.

example.sh

```
-----  
# Save characters by defining functions.  
# The syntax features  
# Semi-colon ';' separates out lines like in bash.
```

```
# The line ending is the end of the function, but the style
# is a shell function, so we need to write the end brace.

donut() { circle $1 $2 $3 $5 ; circle $1 $2 $4 $5 }

donut 10 20 20 black
donut 20 40 20 black
-----
```

17.64.3.179 `run_script_file()` `std::string run_script_file (`
`std::string fname)`

[MainWindow::run_script_file](#).

Parameters

| | |
|--------------------|--------------------------------|
| <code>fname</code> | The path of the script to run. |
|--------------------|--------------------------------|

17.64.3.180 `runCommand` `void runCommand () [slot]`

17.64.3.181 `runCommandClick` `void runCommandClick (`
`const QString & cmd,`
`EmbrReal x,`
`EmbrReal y) [slot]`

17.64.3.182 `runCommandContext` `void runCommandContext (`
`const QString & cmd,`
`const QString & str) [slot]`

17.64.3.183 `runCommandMain` `void runCommandMain (`
`const QString & cmd) [slot]`

17.64.3.184 `runCommandMove` `void runCommandMove (`
`const QString & cmd,`
`EmbrReal x,`
`EmbrReal y) [slot]`

17.64.3.185 runCommandPrompt void runCommandPrompt (const QString & cmd, const QString & str) [slot]

17.64.3.186 saveasfile void saveasfile () [slot]

[MainWindow::saveasfile](#).

17.64.3.187 savefile void savefile () [slot]

[MainWindow::savefile](#).

17.64.3.188 selectAll void selectAll () [slot]

17.64.3.189 setShiftPressed void setShiftPressed () [slot]

17.64.3.190 setShiftReleased void setShiftReleased () [slot]

17.64.3.191 setTextAngle void setTextAngle (EmbReal num) [slot]

17.64.3.192 setTextBold void setTextBold (bool val) [slot]

17.64.3.193 setTextFont void setTextFont (const QString & str) [slot]

17.64.3.194 setTextItalic void setTextItalic (bool val) [slot]

17.64.3.195 `setTextOverline` void setTextOverline (bool val) [slot]

17.64.3.196 `setTextSize` void setTextSize (EmbReal num) [slot]

17.64.3.197 `setTextStrikeOut` void setTextStrikeOut (bool val) [slot]

17.64.3.198 `setTextUnderline` void setTextUnderline (bool val) [slot]

17.64.3.199 `settingsDialog` void settingsDialog (const QString & showTab = QString()) [slot]

17.64.3.200 `settingsPrompt` void settingsPrompt () [slot]

17.64.3.201 `setUndoCleanIcon()` void setUndoCleanIcon (bool opened)

17.64.3.202 `stub_implement` void stub_implement (QString txt) [slot]

MainWindow::stub_implement *txt*.

17.64.3.203 `stub_testing` void stub_testing () [slot]

MainWindow::stub_testing.

17.64.3.204 `textAngle` `EmbReal textAngle () [slot]`

17.64.3.205 `textBold` `bool textBold () [slot]`

17.64.3.206 `textFont` `QString textFont () [slot]`

17.64.3.207 `textFontSelectorCurrentFontChanged` `void textFontSelectorCurrentFontChanged (const QFont & font) [slot]`

17.64.3.208 `textItalic` `bool textItalic () [slot]`

17.64.3.209 `textOverline` `bool textOverline () [slot]`

17.64.3.210 `textSize` `EmbReal textSize () [slot]`

17.64.3.211 `textSizeSelectorIndexChanged` `void textSizeSelectorIndexChanged (int index) [slot]`

17.64.3.212 `textStrikeOut` `bool textStrikeOut () [slot]`

17.64.3.213 `textUnderline` `bool textUnderline () [slot]`

17.64.3.214 `tipOfDay` `void tipOfDay () [slot]`

17.64.3.215 toggleGrid void toggleGrid () [slot]

17.64.3.216 toggleLwt void toggleLwt () [slot]

17.64.3.217 toggleRuler void toggleRuler () [slot]

17.64.3.218 undo void undo () [slot]

17.64.3.219 updateAllViewBackgroundColors void updateAllViewBackgroundColors (QRgb color) [slot]

17.64.3.220 updateAllViewCrossHairColors void updateAllViewCrossHairColors (QRgb color) [slot]

17.64.3.221 updateAllViewGridColors void updateAllViewGridColors (QRgb color) [slot]

17.64.3.222 updateAllViewRulerColors void updateAllViewRulerColors (QRgb color) [slot]

17.64.3.223 updateAllViewScrollBars void updateAllViewScrollBars (bool val) [slot]

17.64.3.224 updateAllViewSelectBoxColors void updateAllViewSelectBoxColors (QRgb colorL,
QRgb fillL,
QRgb colorR,
QRgb fillR,
int alpha) [slot]

17.64.3.225 updateMenuToolbarStatusbar() void updateMenuToolbarStatusbar () [virtual]

[MainWindow::updateMenuToolbarStatusbar](#).

17.64.3.226 updatePickAddMode void updatePickAddMode (bool val) [slot]

17.64.3.227 validFileFormat bool validFileFormat (const QString & *fileName*) [static], [slot]

[MainWindow::validFileFormat](#).

Parameters

| | |
|-----------------|----------------------|
| <i>fileName</i> | <input type="text"/> |
|-----------------|----------------------|

Returns

Todo check the file exists on the system, rename to validFile?

17.64.3.228 whatsThisContextHelp void whatsThisContextHelp () [slot]

17.64.3.229 windowMenuAboutToShow void windowMenuAboutToShow () [slot]

[MainWindow::windowMenuAboutToShow](#).

17.64.3.230 windowMenuActivated void windowMenuActivated (bool *checked*) [slot]

[MainWindow::windowMenuActivated](#).

Parameters

| | |
|----------------|--------------------------|
| <i>checked</i> | <input type="checkbox"/> |
|----------------|--------------------------|

17.64.3.231 writeSettings void writeSettings () [slot]

[MainWindow::writeSettings](#).

This file needs to be read from the users home directory to ensure it is writable

17.64.3.232 zoomAll void zoomAll () [slot]

17.64.3.233 zoomCenter void zoomCenter () [slot]

17.64.3.234 zoomDynamic void zoomDynamic () [slot]

17.64.3.235 zoomExtents void zoomExtents () [slot]

17.64.3.236 zoomIn void zoomIn () [slot]

17.64.3.237 zoomOut void zoomOut () [slot]

17.64.3.238 zoomPrevious void zoomPrevious () [slot]

17.64.3.239 zoomRealtime void zoomRealtime () [slot]

17.64.3.240 zoomScale void zoomScale () [slot]

17.64.3.241 zoomSelected void zoomSelected () [slot]

17.64.3.242 zoomWindow void zoomWindow () [slot]

17.64.4 Member Data Documentation

17.64.4.1 actionHash QAction* actionHash[200]

17.64.4.2 checkBoxTipOfDay QCheckBox* checkBoxTipOfDay [protected]

17.64.4.3 colorSelector QComboBox* colorSelector [protected]

17.64.4.4 cutCopyObjectList QList<QGraphicsItem*> cutCopyObjectList

17.64.4.5 docIndex int docIndex [protected]

17.64.4.6 dockPropEdit [PropertyEditor*](#) dockPropEdit

17.64.4.7 dockUndoEdit [UndoEditor*](#) dockUndoEdit

17.64.4.8 editMenu QMenu* editMenu [protected]

17.64.4.9 fileMenu QMenu* fileMenu [protected]

17.64.4.10 formatFilterOpen QString formatFilterOpen

17.64.4.11 formatFilterSave `QString formatFilterSave`

17.64.4.12 helpMenu `QMenu* helpMenu [protected]`

17.64.4.13 labelTipOfDay `QLabel* labelTipOfDay [protected]`

17.64.4.14 layerSelector `QComboBox* layerSelector [protected]`

17.64.4.15 layoutState `QByteArray layoutState [protected]`

17.64.4.16 linetypeSelector `QComboBox* linetypeSelector [protected]`

17.64.4.17 linewidthSelector `QComboBox* linewidthSelector [protected]`

17.64.4.18 listMdiWin `QList<MdiWindow*> listMdiWin [protected]`

17.64.4.19 listTipOfDay `QStringList listTipOfDay [protected]`

17.64.4.20 mainWin `MainWindow* mainWin`

17.64.4.21 mdiArea `MdiArea* mdiArea`

17.64.4.22 menuHash `QHash<QString, QMenu*> menuHash`

17.64.4.23 myFileSeparator `QAction* myFileSeparator [protected]`

17.64.4.24 numOfDocs `int numOfDocs [protected]`

17.64.4.25 openFilesPath `QString openFilesPath [protected]`

17.64.4.26 panMenu `QMenu* panMenu [protected]`

17.64.4.27 prompt `CmdPrompt* prompt`

17.64.4.28 recentMenu `QMenu* recentMenu [protected]`

17.64.4.29 settings_display_bg_color `QRgb settings_display_bg_color`

17.64.4.30 settings_display_crosshair_color `QRgb settings_display_crosshair_color`

17.64.4.31 settings_display_crosshair_percent `uint8_t settings_display_crosshair_percent`

17.64.4.32 settings_display_renderhint_aa `bool settings_display_renderhint_aa`

17.64.4.33 **settings_display_renderhint_high_aa** bool settings_display_renderhint_high_aa

17.64.4.34 **settings_display_renderhint_noncosmetic** bool settings_display_renderhint_noncosmetic

17.64.4.35 **settings_display_renderhint_smooth_pix** bool settings_display_renderhint_smooth_pix

17.64.4.36 **settings_display_renderhint_text_aa** bool settings_display_renderhint_text_aa

17.64.4.37 **settings_display_scrollbar_widget_num** int settings_display_scrollbar_widget_num

17.64.4.38 **settings_display_selectbox_alpha** uint8_t settings_display_selectbox_alpha

17.64.4.39 **settings_display_selectbox_left_color** QRgb settings_display_selectbox_left_color

17.64.4.40 **settings_display_selectbox_left_fill** QRgb settings_display_selectbox_left_fill

17.64.4.41 **settings_display_selectbox_right_color** QRgb settings_display_selectbox_right_color

17.64.4.42 **settings_display_selectbox_right_fill** QRgb settings_display_selectbox_right_fill

17.64.4.43 **settings_display_show_scrollbars** bool settings_display_show_scrollbars

17.64.4.44 settings_display_units `QString settings_display_units`

17.64.4.45 settings_display_use_opengl `bool settings_display_use_opengl`

17.64.4.46 settings_display_zoomscale_in `EmbReal settings_display_zoomscale_in`

17.64.4.47 settings_display_zoomscale_out `EmbReal settings_display_zoomscale_out`

17.64.4.48 settings_general_check_for_updates `bool settings_general_check_for_updates`

17.64.4.49 settings_general_current_tip `quint16 settings_general_current_tip`

17.64.4.50 settings_general_icon_size `int settings_general_icon_size`

17.64.4.51 settings_general_icon_theme `QString settings_general_icon_theme`

17.64.4.52 settings_general_language `QString settings_general_language`

17.64.4.53 settings_general_mdi_bg_color `QRgb settings_general_mdi_bg_color`

17.64.4.54 settings_general_mdi_bg_logo `QString settings_general_mdi_bg_logo`

17.64.4.55 **settings_general_mdi_bg_texture** `QString settings_general_mdi_bg_texture`

17.64.4.56 **settings_general_mdi_bg_use_color** `bool settings_general_mdi_bg_use_color`

17.64.4.57 **settings_general_mdi_bg_use_logo** `bool settings_general_mdi_bg_use_logo`

17.64.4.58 **settings_general_mdi_bg_use_texture** `bool settings_general_mdi_bg_use_texture`

17.64.4.59 **settings_general_system_help_browser** `bool settings_general_system_help_browser`

17.64.4.60 **settings_general_tip_of_the_day** `bool settings_general_tip_of_the_day`

17.64.4.61 **settings_grid_center_on_origin** `bool settings_grid_center_on_origin`

17.64.4.62 **settings_grid_center_x** `EmbReal settings_grid_center_x`

17.64.4.63 **settings_grid_center_y** `EmbReal settings_grid_center_y`

17.64.4.64 **settings_grid_color** `QRgb settings_grid_color`

17.64.4.65 **settings_grid_color_match_crosshair** `bool settings_grid_color_match_crosshair`

17.64.4.66 settings_grid_load_from_file bool settings_grid_load_from_file

17.64.4.67 settings_grid_show_on_load bool settings_grid_show_on_load

17.64.4.68 settings_grid_show_origin bool settings_grid_show_origin

17.64.4.69 settings_grid_size_radius EmbReal settings_grid_size_radius

17.64.4.70 settings_grid_size_x EmbReal settings_grid_size_x

17.64.4.71 settings_grid_size_y EmbReal settings_grid_size_y

17.64.4.72 settings_grid_spacing_angle EmbReal settings_grid_spacing_angle

17.64.4.73 settings_grid_spacing_radius EmbReal settings_grid_spacing_radius

17.64.4.74 settings_grid_spacing_x EmbReal settings_grid_spacing_x

17.64.4.75 settings_grid_spacing_y EmbReal settings_grid_spacing_y

17.64.4.76 settings_grid_type QString settings_grid_type

17.64.4.77 settings_lwt_default_lwt `EmbReal settings_lwt_default_lwt`

17.64.4.78 settings_lwt_real_render `bool settings_lwt_real_render`

17.64.4.79 settings_lwt_show_lwt `bool settings_lwt_show_lwt`

17.64.4.80 settings_opensave_custom_filter `QString settings_opensave_custom_filter`

17.64.4.81 settings_opensave_open_format `QString settings_opensave_open_format`

17.64.4.82 settings_opensave_open_thumbnail `bool settings_opensave_open_thumbnail`

17.64.4.83 settings_opensave_recent_directory `QString settings_opensave_recent_directory`

17.64.4.84 settings_opensave_recent_list_of_files `QStringList settings_opensave_recent_list_of_files`

17.64.4.85 settings_opensave_recent_max_files `uint8_t settings_opensave_recent_max_files`

17.64.4.86 settings_opensave_save_format `QString settings_opensave_save_format`

17.64.4.87 settings_opensave_save_thumbnail `bool settings_opensave_save_thumbnail`

17.64.4.88 settings_opensave_trim_dst_num_jumps uint8_t settings_opensave_trim_dst_num_jumps

17.64.4.89 settings_printing_default_device QString settings_printing_default_device

17.64.4.90 settings_printing_disable_bg bool settings_printing_disable_bg

17.64.4.91 settings_printing_use_last_device bool settings_printing_use_last_device

17.64.4.92 settings_prompt_bg_color QRgb settings_prompt_bg_color

17.64.4.93 settings_prompt_font_family QString settings_prompt_font_family

17.64.4.94 settings_prompt_font_size uint8_t settings_prompt_font_size

17.64.4.95 settings_prompt_font_style QString settings_prompt_font_style

17.64.4.96 settings_prompt_save_history bool settings_prompt_save_history

17.64.4.97 settings_prompt_save_history_as_html bool settings_prompt_save_history_as_html

17.64.4.98 settings_prompt_save_history_filename QString settings_prompt_save_history_filename

17.64.4.99 **settings_prompt_text_color** QRgb settings_prompt_text_color

17.64.4.100 **settings_qsnap_aperture_size** uint8_t settings_qsnap_aperture_size

17.64.4.101 **settings_qsnap_apparent** bool settings_qsnap_apparent

17.64.4.102 **settings_qsnap_center** bool settings_qsnap_center

17.64.4.103 **settings_qsnap_enabled** bool settings_qsnap_enabled

17.64.4.104 **settings_qsnap_endpoint** bool settings_qsnap_endpoint

17.64.4.105 **settings_qsnap_extension** bool settings_qsnap_extension

17.64.4.106 **settings_qsnap_insertion** bool settings_qsnap_insertion

17.64.4.107 **settings_qsnap_intersection** bool settings_qsnap_intersection

17.64.4.108 **settings_qsnap_locator_color** QRgb settings_qsnap_locator_color

17.64.4.109 **settings_qsnap_locator_size** uint8_t settings_qsnap_locator_size

17.64.4.110 **settings_qsnap_midpoint** bool settings_qsnap_midpoint

17.64.4.111 **settings_qsnap_nearest** bool settings_qsnap_nearest

17.64.4.112 **settings_qsnap_node** bool settings_qsnap_node

17.64.4.113 **settings_qsnap_parallel** bool settings_qsnap_parallel

17.64.4.114 **settings_qsnap_perpendicular** bool settings_qsnap_perpendicular

17.64.4.115 **settings_qsnap_quadrant** bool settings_qsnap_quadrant

17.64.4.116 **settings_qsnap_tangent** bool settings_qsnap_tangent

17.64.4.117 **settings_ruler_color** QRgb settings_ruler_color

17.64.4.118 **settings_ruler_metric** bool settings_ruler_metric

17.64.4.119 **settings_ruler_pixel_size** uint8_t settings_ruler_pixel_size

17.64.4.120 **settings_ruler_show_on_load** bool settings_ruler_show_on_load

17.64.4.121 **settings_selection_coolgrip_color** QRgb settings_selection_coolgrip_color

17.64.4.122 **settings_selection_grip_size** uint8_t settings_selection_grip_size

17.64.4.123 **settings_selection_hotgrip_color** QRgb settings_selection_hotgrip_color

17.64.4.124 **settings_selection_mode_pickadd** bool settings_selection_mode_pickadd

17.64.4.125 **settings_selection_mode_pickdrag** bool settings_selection_mode_pickdrag

17.64.4.126 **settings_selection_mode_pickfirst** bool settings_selection_mode_pickfirst

17.64.4.127 **settings_selection_pickbox_size** uint8_t settings_selection_pickbox_size

17.64.4.128 **settings_text_angle** EmbReal settings_text_angle

17.64.4.129 **settings_text_font** QString settings_text_font

17.64.4.130 **settings_text_size** EmbReal settings_text_size

17.64.4.131 **settings_text_style_bold** bool settings_text_style_bold

17.64.4.132 settings_text_style_italic bool settings_text_style_italic

17.64.4.133 settings_text_style_overline bool settings_text_style_overline

17.64.4.134 settings_text_style_strikeout bool settings_text_style_strikeout

17.64.4.135 settings_text_style_underline bool settings_text_style_underline

17.64.4.136 settingsMenu QMenu* settingsMenu [protected]

17.64.4.137 shiftKeyPressedState bool shiftKeyPressedState [protected]

17.64.4.138 statusbar StatusBar* statusbar

17.64.4.139 textFontSelector QFontComboBox* textFontSelector [protected]

17.64.4.140 textSizeSelector QComboBox* textSizeSelector [protected]

17.64.4.141 toolbarEdit QToolBar* toolbarEdit [protected]

17.64.4.142 toolbarFile QToolBar* toolbarFile [protected]

17.64.4.143 toolbarHash QHash<QString, QToolBar*> toolbarHash

17.64.4.144 toolbarHelp QToolBar* toolbarHelp [protected]

17.64.4.145 toolbarIcon QToolBar* toolbarIcon [protected]

17.64.4.146 toolbarLayer QToolBar* toolbarLayer [protected]

17.64.4.147 toolbarPan QToolBar* toolbarPan [protected]

17.64.4.148 toolbarPrompt QToolBar* toolbarPrompt [protected]

17.64.4.149 toolbarProperties QToolBar* toolbarProperties [protected]

17.64.4.150 toolbarText QToolBar* toolbarText [protected]

17.64.4.151 toolbarView QToolBar* toolbarView [protected]

17.64.4.152 toolbarZoom QToolBar* toolbarZoom [protected]

17.64.4.153 viewMenu QMenu* viewMenu [protected]

17.64.4.154 windowMenu QMenu* windowMenu [protected]

17.64.4.155 wizardTipOfDay QWizard* wizardTipOfDay [protected]

17.64.4.156 zoomMenu QMenu* zoomMenu [protected]

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/mainwindow-commands.cpp](#)
- [embroidermodder2/mainwindow-menus.cpp](#)
- [embroidermodder2/mainwindow-settings.cpp](#)
- [embroidermodder2/mainwindow-toolbars.cpp](#)
- [embroidermodder2/mainwindow.cpp](#)

17.65 MdiArea Class Reference

```
#include <embroidermodder.h>
```

Public Slots

- void **cascade** ()
MdiArea::cascade.
- void **tile** ()
MdiArea::tile.

Public Member Functions

- void **zoomExtentsAllSubWindows** ()
MdiArea::zoomExtentsAllSubWindows.
- void **forceRepaint** ()
MdiArea::forceRepaint.
- **MdiArea** (**MainWindow** *mw, **QWidget** *parent=0)
MdiArea::MdiArea.
- **~MdiArea** ()
MdiArea::~MdiArea.
- void **useBackgroundLogo** (bool use)
MdiArea::useBackgroundLogo.
- void **useBackgroundTexture** (bool use)
MdiArea::useBackgroundTexture.
- void **useBackgroundColor** (bool use)
- void **setBackgroundLogo** (const **QString** &fileName)
MdiArea::setBackgroundLogo.
- void **setBackgroundTexture** (const **QString** &fileName)
MdiArea::setBackgroundTexture.
- void **setBackgroundColor** (const **QColor** &color)
MdiArea::setBackgroundColor.

Public Attributes

- `MainWindow * mainWin`
- `bool useLogo`
- `bool useTexture`
- `bool useColor`
- `QPixmap bgLogo`
- `QPixmap bgTexture`
- `QColor bgColor`

Protected Member Functions

- `virtual void mouseDoubleClickEvent (QMouseEvent *e)`
MdiArea::mouseDoubleClickEvent.
- `virtual void paintEvent (QPaintEvent *e)`
MdiArea::paintEvent.

17.65.1 Constructor & Destructor Documentation

17.65.1.1 MdiArea() `MdiArea (`
 `MainWindow * mw,`
 `QWidget * parent = 0)`

`MdiArea::MdiArea.`

Parameters

| | |
|---------------------|--|
| <code>mw</code> | |
| <code>parent</code> | |

17.65.1.2 ~MdiArea() `~MdiArea ()`

`MdiArea::~MdiArea.`

17.65.2 Member Function Documentation

17.65.2.1 cascade `void cascade () [slot]`

`MdiArea::cascade.`

17.65.2.2 forceRepaint() void forceRepaint ()

MdiArea::forceRepaint.

17.65.2.3 mouseDoubleClickEvent() void mouseDoubleClickEvent (QMouseEvent * e) [protected], [virtual]

MdiArea::mouseDoubleClickEvent.

17.65.2.4 paintEvent() void paintEvent (QPaintEvent * e) [protected], [virtual]

MdiArea::paintEvent.

17.65.2.5 setBackgroundColor() void setBackgroundColor (const QColor & color)

MdiArea::setBackgroundColor.

Parameters

| | |
|-------|----------------------|
| color | <input type="text"/> |
|-------|----------------------|

17.65.2.6 setBackgroundLogo() void setBackgroundLogo (const QString & fileName)

MdiArea::setBackgroundLogo.

Parameters

| | |
|----------|----------------------|
| fileName | <input type="text"/> |
|----------|----------------------|

17.65.2.7 setBackgroundTexture() void setBackgroundTexture (const QString & fileName)

MdiArea::setBackgroundTexture.

Parameters

| | |
|-----------------|----------------------|
| <i>fileName</i> | <input type="text"/> |
|-----------------|----------------------|

17.65.2.8 tile void tile () [slot][MdiArea::tile.](#)**17.65.2.9 useBackgroundColor()** void useBackgroundColor (bool use)**Parameters**

| | |
|------------|----------------------|
| <i>use</i> | <input type="text"/> |
|------------|----------------------|

17.65.2.10 useBackgroundLogo() void useBackgroundLogo (bool use)[MdiArea::useBackgroundLogo.](#)**Parameters**

| | |
|------------|----------------------|
| <i>use</i> | <input type="text"/> |
|------------|----------------------|

17.65.2.11 useBackgroundTexture() void useBackgroundTexture (bool use)[MdiArea::useBackgroundTexture.](#)**Parameters**

| | |
|------------|----------------------|
| <i>use</i> | <input type="text"/> |
|------------|----------------------|

17.65.2.12 zoomExtentsAllSubWindows() void zoomExtentsAllSubWindows ()[MdiArea::zoomExtentsAllSubWindows.](#)

17.65.3 Member Data Documentation

17.65.3.1 `bgColor` QColor bgColor

17.65.3.2 `bgLogo` QPixmap bgLogo

17.65.3.3 `bgTexture` QPixmap bgTexture

17.65.3.4 `mainWin` MainWindow* mainWin

17.65.3.5 `useColor` bool useColor

17.65.3.6 `useLogo` bool useLogo

17.65.3.7 `useTexture` bool useTexture

The documentation for this class was generated from the following files:

- embroidermodder2/[embroidermodder.h](#)
- embroidermodder2/[mdiarea.cpp](#)

17.66 MdiWindow Class Reference

```
#include <embroidermodder.h>
```

Public Slots

- void `closeEvent` (QCloseEvent *e)
`MdiWindow::closeEvent.`
- void `onWindowActivated` ()
`MdiWindow::onWindowActivated.`
- void `currentLayerChanged` (const QString &layer)
`MdiWindow::currentLayerChanged.`
- void `currentColorChanged` (const QRgb &color)
`MdiWindow::currentColorChanged.`
- void `currentLinetypeChanged` (const QString &type)
`MdiWindow::currentLinetypeChanged.`
- void `currentLineweightChanged` (const QString &weight)
`MdiWindow::currentLineweightChanged.`
- void `updateColorLinetypeLinewidth` ()
- void `deletePressed` ()
- void `escapePressed` ()
- void `showViewScrollBars` (bool val)
- void `setViewCrossHairColor` (QRgb color)
- void `setViewBackgroundColor` (QRgb color)
- void `setViewSelectBoxColors` (QRgb colorL, QRgb fillL, QRgb colorR, QRgb fillR, int alpha)
- void `setViewGridColor` (QRgb color)
- void `setViewRulerColor` (QRgb color)
- void `print` ()
`MdiWindow::print.`
- void `saveBMC` ()
`MdiWindow::saveBMC.`
- void `promptHistoryAppended` (const QString &txt)
- void `logPromptInput` (const QString &txt)
- void `promptInputPrevious` ()
- void `promptInputNext` ()
`MdiWindow::promptInputNext.`

Signals

- void `sendCloseMdiWin` (MdiWindow *)

Public Member Functions

- MdiWindow (const int theIndex, MainWindow *mw, QMdiArea *parent, Qt::WindowFlags wflags)
- ~MdiWindow ()
`MdiWindow::~MdiWindow.`
- virtual QSize `sizeHint` () const
`MdiWindow::sizeHint.`
- QString `getCurrentFile` ()
- QString `getShortCurrentFile` ()
`MdiWindow::getShortCurrentFile.`
- View * `getView` ()
- QGraphicsScene * `getScene` ()
- QString `getCurrentLayer` ()
- QRgb `getCurrentColor` ()

- `QString getCurrentLineType ()`
- `QString getCurrentLineWeight ()`
- `void setCurrentLayer (const QString &layer)`
- `void setCurrentColor (const QRgb &color)`
- `void setCurrentLineType (const QString &lineType)`
- `void setCurrentLineWeight (const QString &lineWeight)`
- `void designDetails ()`
- `bool loadFile (const QString &fileName)`
MdiWindow::loadFile.
- `bool saveFile (const QString &fileName)`
MdiWindow::saveFile.

Private Member Functions

- `void setCurrentFile (const QString &fileName)`
MdiWindow::setCurrentFile.
- `QString fileExtension (const QString &fileName)`
MdiWindow::fileExtension.
- `void promptInputPrevNext (bool prev)`
MdiWindow::promptInputPrevNext.

Private Attributes

- `MainWindow * mainWin`
- `QMdiArea * mdiArea`
- `QGraphicsScene * gscene`
- `View * gview`
- `bool fileWasLoaded`
- `QString promptHistory`
- `QList<QString> promptInputList`
- `int promptInputNum`
- `QPrinter printer`
- `QString curFile`
- `int myIndex`
- `QString curLayer`
- `QRgb curColor`
- `QString curLineType`
- `QString curLineWeight`

17.66.1 Constructor & Destructor Documentation

```
17.66.1.1 MdiWindow() MdiWindow (
    const int theIndex,
    MainWindow * mw,
    QMdiArea * parent,
    Qt::WindowFlags wflags )
```

17.66.1.2 ~MdiWindow() ~MdiWindow()

MdiWindow::~MdiWindow.

17.66.2 Member Function Documentation**17.66.2.1 closeEvent void closeEvent (**
 QCLOSEEvent * e) [slot]

MdiWindow::closeEvent.

17.66.2.2 currentColorChanged void currentColorChanged (
 const QRgb & color) [slot]

MdiWindow::currentColorChanged.

Parameters

| | |
|--------------|----------------------|
| <i>color</i> | <input type="text"/> |
|--------------|----------------------|

17.66.2.3 currentLayerChanged void currentLayerChanged (
 const QString & layer) [slot]

MdiWindow::currentLayerChanged.

Parameters

| | |
|--------------|----------------------|
| <i>layer</i> | <input type="text"/> |
|--------------|----------------------|

17.66.2.4 currentLinetypeChanged void currentLinetypeChanged (
 const QString & type) [slot]

MdiWindow::currentLinetypeChanged.

Parameters

| | |
|-------------|----------------------|
| <i>type</i> | <input type="text"/> |
|-------------|----------------------|

17.66.2.5 currentLineweightChanged void currentLineweightChanged (const QString & weight) [slot]

MdiWindow::currentLineweightChanged.

Parameters

| | |
|---------------|--|
| <i>weight</i> | |
|---------------|--|

17.66.2.6 deletePressed void deletePressed () [slot]

17.66.2.7 designDetails() void designDetails ()

17.66.2.8 escapePressed void escapePressed () [slot]

17.66.2.9 fileExtension() QString fileExtension (const QString & fileName) [private]

MdiWindow::fileExtension.

Parameters

| | |
|-----------------|--|
| <i>fileName</i> | |
|-----------------|--|

Returns

17.66.2.10 getCurrentColor() QRgb getCurrentColor () [inline]

17.66.2.11 getCurrentFile() QString getCurrentFile () [inline]

17.66.2.12 `getCurrentLayer()` `QString getCurrentLayer () [inline]`

17.66.2.13 `getCurrentLineType()` `QString getCurrentLineType () [inline]`

17.66.2.14 `getCurrentLineWeight()` `QString getCurrentLineWeight () [inline]`

17.66.2.15 `getScene()` `QGraphicsScene * getScene () [inline]`

17.66.2.16 `getShortCurrentFile()` `QString getShortCurrentFile ()`

[MdiWindow::getShortCurrentFile.](#)

Returns

17.66.2.17 `getView()` `View * getView () [inline]`

17.66.2.18 `loadFile()` `bool loadFile (const QString & fileName)`

[MdiWindow::loadFile.](#)

Parameters

| |
|-----------------------|
| <code>fileName</code> |
|-----------------------|

Returns

Todo reincorporate embPattern_moveStitchListToPolylines(p); //TODO: Test more

17.66.2.19 logPromptInput void logPromptInput (const QString & txt) [slot]

17.66.2.20 onWindowActivated void onWindowActivated () [slot]

[MdiWindow::onWindowActivated.](#)

17.66.2.21 print void print () [slot]

[MdiWindow::print.](#)

17.66.2.22 promptHistoryAppended void promptHistoryAppended (const QString & txt) [slot]

17.66.2.23 promptInputNext void promptInputNext () [slot]

[MdiWindow::promptInputNext.](#)

17.66.2.24 promptInputPrevious void promptInputPrevious () [slot]

17.66.2.25 promptInputPrevNext() void promptInputPrevNext (bool prev) [private]

[MdiWindow::promptInputPrevNext.](#)

Parameters

| | |
|-------------|--------------------------|
| <i>prev</i> | <input type="checkbox"/> |
|-------------|--------------------------|

17.66.2.26 saveBMC void saveBMC () [slot]

[MdiWindow::saveBMC.](#)

Todo Save a Brother PEL image (An 8bpp, 130x113 pixel monochromatic? bitmap image) Why 8bpp when only 1bpp is needed?

Todo Should BMC be limited to ~32KB or is this a mix up with Bitmap Cache?

Is there/should there be other embedded data in the bitmap besides the image itself?

Note

Can save a Singer BMC image (An 8bpp, 130x113 pixel colored bitmap image)

17.66.2.27 `saveFile()` `bool saveFile (`
`const QString & fileName)`

[MdiWindow::saveFile.](#)

Parameters

| | |
|-----------------------|----------------------|
| <code>fileName</code> | <input type="text"/> |
|-----------------------|----------------------|

Returns

17.66.2.28 `sendCloseMdiWin` `void sendCloseMdiWin (`
`MdiWindow *) [signal]`

17.66.2.29 `setCurrentColor()` `void setCurrentColor (`
`const QRgb & color) [inline]`

17.66.2.30 `setCurrentFile()` `void setCurrentFile (`
`const QString & fileName) [private]`

[MdiWindow::setCurrentFile.](#)

Parameters

| | |
|-----------------------|----------------------|
| <code>fileName</code> | <input type="text"/> |
|-----------------------|----------------------|

17.66.2.31 setCurrentLayer() void setCurrentLayer (const QString & *layer*) [inline]

17.66.2.32 setCurrentLineType() void setCurrentLineType (const QString & *lineType*) [inline]

17.66.2.33 setCurrentLineWeight() void setCurrentLineWeight (const QString & *lineWeight*) [inline]

17.66.2.34 setViewBackgroundColor void setViewBackgroundColor (QRgb *color*) [slot]

17.66.2.35 setViewCrossHairColor void setViewCrossHairColor (QRgb *color*) [slot]

17.66.2.36 setViewGridColor void setViewGridColor (QRgb *color*) [slot]

17.66.2.37 setViewRulerColor void setViewRulerColor (QRgb *color*) [slot]

17.66.2.38 setViewSelectBoxColors void setViewSelectBoxColors (QRgb *colorL*, QRgb *fillL*, QRgb *colorR*, QRgb *fillR*, int *alpha*) [slot]

17.66.2.39 showViewScrollBars void showViewScrollBars (bool *val*) [slot]

17.66.2.40 sizeHint() QSize sizeHint () const [virtual]

MdiWindow::sizeHint.

Returns

17.66.2.41 updateColorLinetypeLinewidth void updateColorLinetypeLinewidth () [slot]

17.66.3 Member Data Documentation

17.66.3.1 curColor QRgb curColor [private]

17.66.3.2 curFile QString curFile [private]

17.66.3.3 curLayer QString curLayer [private]

17.66.3.4 curLineType QString curLineType [private]

17.66.3.5 curLineWeight QString curLineWeight [private]

17.66.3.6 fileWasLoaded bool fileWasLoaded [private]

17.66.3.7 gscene QGraphicsScene* gscene [private]

17.66.3.8 gview `View* gview [private]`

17.66.3.9 mainWin `MainWindow* mainWin [private]`

17.66.3.10 mdiArea `QMdiArea* mdiArea [private]`

17.66.3.11 myIndex `int myIndex [private]`

17.66.3.12 printer `QPrinter printer [private]`

17.66.3.13 promptHistory `QString promptHistory [private]`

17.66.3.14 promptInputList `QList<QString> promptInputList [private]`

17.66.3.15 promptInputNum `int promptInputNum [private]`

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/mdiwindow.cpp](#)

17.67 Parameter_Struct Reference

Public Attributes

- `std::string s_value`
- `EmbReal r_value`
- `int i_value`

17.67.1 Member Data Documentation

17.67.1.1 i_value int i_value

17.67.1.2 r_value EmbReal r_value

17.67.1.3 s_value std::string s_value

The documentation for this struct was generated from the following file:

- embroidermodder2/mainwindow.cpp

17.68 PathObject Class Reference

```
#include <embroidermodder.h>
```

Public Types

- enum { Type = OBJ_TYPE_PATH }

Public Types inherited from BaseObject

- enum { Type = OBJ_TYPE_BASE }

Public Member Functions

- PathObject (EmbReal x, EmbReal y, const QPainterPath p, QRgb rgb, QGraphicsItem *parent=0)
- PathObject (PathObject *obj, QGraphicsItem *parent=0)
- ~PathObject ()
- virtual int type () const
- void init (EmbReal x, EmbReal y, const QPainterPath &p, QRgb rgb, Qt::PenStyle lineType)
- void updatePath (const QPainterPath &p)
- QPainterPath objectCopyPath () const
- QPainterPath objectSavePath () const
- QPointF objectPos () const
- EmbReal objectX () const
- EmbReal objectY () const
- void setObjectPos (const QPointF &point)
- void setObjectPos (EmbReal x, EmbReal y)
- void setObjectX (EmbReal x)
- void setObjectY (EmbReal y)
- void updateRubber (Painter *painter=0)
- virtual void vulcanize ()
- virtual QPointF mouseSnapPoint (const QPointF &mousePoint)
- virtual QList< QPointF > allGripPoints ()
- virtual void gripEdit (const QPointF &before, const QPointF &after)

Public Member Functions inherited from BaseObject

- `BaseObject` (`QGraphicsItem *parent=0`)
- `virtual ~BaseObject ()`
- `virtual int type () const`
- `qint64 objectID () const`
- `QPen objectPen () const`
- `QColor objectColor () const`
- `QRgb objectColorRGB () const`
- `Qt::PenStyle objectLineType () const`
- `EmbReal objectLineWidth () const`
- `QPainterPath objectPath () const`
- `int objectRubberMode () const`
- `QPointF objectRubberPoint (const QString &key) const`
- `QString objectRubberText (const QString &key) const`
- `QPointF objectCenter () const`
- `EmbReal objectCenterX () const`
- `EmbReal objectCenterY () const`
- `void setObjectCenter (EmbVector center)`
- `void setObjectCenterX (EmbReal centerX)`
- `void setObjectCenterY (EmbReal centerY)`
- `QRectF rect () const`
- `void setRect (const QRectF &r)`
- `void setRect (EmbReal x, EmbReal y, EmbReal w, EmbReal h)`
- `QLineF line () const`
- `void setLine (const QLineF &li)`
- `void setLine (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2)`
- `void setObjectColor (const QColor &color)`
- `void setObjectColorRGB (QRgb rgb)`
- `void setObjectLineType (Qt::PenStyle lineType)`
- `void setObjectLineWidth (EmbReal lineWidth)`
- `void setObjectPath (const QPainterPath &p)`
- `void setObjectRubberMode (int mode)`
- `void setObjectRubberPoint (const QString &key, const QPointF &point)`
- `void setObjectRubberText (const QString &key, const QString &txt)`
- `virtual QRectF boundingRect () const`
- `virtual QPainterPath shape () const`
- `void drawRubberLine (const QLineF &rubLine, QPainter *painter=0, const char *colorFromScene=0)`
- `virtual void vulcanize ()=0`
- `virtual QPointF mouseSnapPoint (const QPointF &mousePoint)=0`
- `virtual QList<QPointF> allGripPoints ()=0`
- `virtual void gripEdit (const QPointF &before, const QPointF &after)=0`

Public Attributes

- `QPainterPath normalPath`

Public Attributes inherited from BaseObject

- `QPen objPen`
- `QPen lwtPen`
- `QLineF objLine`
- `int objRubberMode`
- `QHash<QString, QPointF> objRubberPoints`
- `QHash<QString, QString> objRubberTexts`
- `qint64 objID`

Protected Member Functions

- void `paint` (QPainter *, const QStyleOptionGraphicsItem *, QWidget *)

Protected Member Functions inherited from `BaseObject`

- QPen `lineWeightPen` () const
- void `realRender` (QPainter *painter, const QPainterPath &renderPath)

17.68.1 Member Enumeration Documentation

17.68.1.1 anonymous enum `anonymous enum`

Enumerator

| | |
|------|--|
| Type | |
|------|--|

17.68.2 Constructor & Destructor Documentation

17.68.2.1 `PathObject()` [1/2] `PathObject` (

```
    EmbReal x,
    EmbReal y,
    const QPainterPath p,
    QRgb rgb,
    QGraphicsItem * parent = 0 )
```

17.68.2.2 `PathObject()` [2/2] `PathObject` (

```
    PathObject * obj,
    QGraphicsItem * parent = 0 )
```

17.68.2.3 `~PathObject()` `~PathObject` ()

17.68.3 Member Function Documentation

17.68.3.1 allGripPoints() QList< QPointF > allGripPoints () [virtual]

Implements [BaseObject](#).

17.68.3.2 gripEdit() void gripEdit (const QPointF & before, const QPointF & after) [virtual]

Implements [BaseObject](#).

17.68.3.3 init() void init (EmbReal x, EmbReal y, const QPainterPath & p, QRgb rgb, Qt::PenStyle lineType)

17.68.3.4 mouseSnapPoint() QPointF mouseSnapPoint (const QPointF & mousePoint) [virtual]

Implements [BaseObject](#).

17.68.3.5 objectCopyPath() QPainterPath objectCopyPath () const

17.68.3.6 objectPos() QPointF objectPos () const [inline]

17.68.3.7 objectSavePath() QPainterPath objectSavePath () const

17.68.3.8 objectX() EmbReal objectX () const [inline]

17.68.3.9 objectY() EmbReal objectY () const [inline]

17.68.3.10 `paint()` void paint (QPainter * painter, const QStyleOptionGraphicsItem * option, QWidget *) [protected]

17.68.3.11 `setObjectPos()` [1/2] void setObjectPos (const QPointF & point) [inline]

17.68.3.12 `setObjectPos()` [2/2] void setObjectPos (EmbReal x, EmbReal y) [inline]

17.68.3.13 `setObjectX()` void setObjectX (EmbReal x) [inline]

17.68.3.14 `setObjectY()` void setObjectY (EmbReal y) [inline]

17.68.3.15 `type()` virtual int type () const [inline], [virtual]

Reimplemented from [BaseObject](#).

17.68.3.16 `updatePath()` void updatePath (const QPainterPath & p)

17.68.3.17 `updateRubber()` void updateRubber (QPainter * painter = 0)

17.68.3.18 `vulcanize()` void vulcanize () [virtual]

Implements [BaseObject](#).

17.68.4 Member Data Documentation

17.68.4.1 normalPath QPainterPath normalPath

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/object-path.cpp](#)

17.69 PointObject Class Reference

```
#include <embroidermodder.h>
```

Public Types

- enum { [Type](#) = OBJ_TYPE_POINT }

Public Types inherited from [BaseObject](#)

- enum { [Type](#) = OBJ_TYPE_BASE }

Public Member Functions

- [PointObject \(EmbReal x, EmbReal y, QRgb rgb, QGraphicsItem *parent=0\)](#)
- [PointObject \(PointObject *obj, QGraphicsItem *parent=0\)](#)
- [~PointObject \(\)](#)
- void [init \(EmbReal x, EmbReal y, QRgb rgb, Qt::PenStyle lineType\)](#)
- virtual int [type \(\) const](#)
- QPainterPath [objectSavePath \(\) const](#)
- QPointF [objectPos \(\) const](#)
- [EmbReal objectX \(\) const](#)
- [EmbReal objectY \(\) const](#)
- void [setObjectPos \(const QPointF &point\)](#)
- void [setObjectPos \(EmbReal x, EmbReal y\)](#)
- void [setObjectX \(EmbReal x\)](#)
- void [setObjectY \(EmbReal y\)](#)
- void [updateRubber \(Painter *painter=0\)](#)
- virtual void [vulcanize \(\)](#)
- virtual QPointF [mouseSnapPoint \(const QPointF &mousePoint\)](#)
- virtual QList<QPointF> [allGripPoints \(\)](#)
- virtual void [gripEdit \(const QPointF &before, const QPointF &after\)](#)

Public Member Functions inherited from [BaseObject](#)

- [BaseObject](#) (QGraphicsItem *parent=0)
- virtual [~BaseObject](#) ()
- virtual int [type](#) () const
- qint64 [objectID](#) () const
- QPen [objectPen](#) () const
- QColor [objectColor](#) () const
- QRgb [objectColorRGB](#) () const
- Qt::PenStyle [objectLineType](#) () const
- EmbReal [objectLineWidth](#) () const
- QPainterPath [objectPath](#) () const
- int [objectRubberMode](#) () const
- QPointF [objectRubberPoint](#) (const QString &key) const
- QString [objectRubberText](#) (const QString &key) const
- QPointF [objectCenter](#) () const
- EmbReal [objectCenterX](#) () const
- EmbReal [objectCenterY](#) () const
- void [setObjectCenter](#) (EmbVector center)
- void [setObjectCenterX](#) (EmbReal centerX)
- void [setObjectCenterY](#) (EmbReal centerY)
- QRectF [rect](#) () const
- void [setRect](#) (const QRectF &r)
- void [setRect](#) (EmbReal x, EmbReal y, EmbReal w, EmbReal h)
- QLineF [line](#) () const
- void [setLine](#) (const QLineF &li)
- void [setLine](#) (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2)
- void [setObjectColor](#) (const QColor &color)
- void [setObjectColorRGB](#) (QRgb rgb)
- void [setObjectLineType](#) (Qt::PenStyle lineType)
- void [setObjectLineWidth](#) (EmbReal lineWidth)
- void [setObjectPath](#) (const QPainterPath &p)
- void [setObjectRubberMode](#) (int mode)
- void [setObjectRubberPoint](#) (const QString &key, const QPointF &point)
- void [setObjectRubberText](#) (const QString &key, const QString &txt)
- virtual QRectF [boundingRect](#) () const
- virtual QPainterPath [shape](#) () const
- void [drawRubberLine](#) (const QLineF &rubLine, QPainter *painter=0, const char *colorFromScene=0)
- virtual void [vulcanize](#) ()=0
- virtual QPointF [mouseSnapPoint](#) (const QPointF &mousePoint)=0
- virtual QList< QPointF > [allGripPoints](#) ()=0
- virtual void [gripEdit](#) (const QPointF &before, const QPointF &after)=0

Protected Member Functions

- void [paint](#) (QPainter *, const QStyleOptionGraphicsItem *, QWidget *)

Protected Member Functions inherited from [BaseObject](#)

- QPen [lineWeightPen](#) () const
- void [realRender](#) (QPainter *painter, const QPainterPath &renderPath)

Additional Inherited Members

Public Attributes inherited from [BaseObject](#)

- QPen [objPen](#)
- QPen [lwtPen](#)
- QLineF [objLine](#)
- int [objRubberMode](#)
- QHash<QString, QPointF> [objRubberPoints](#)
- QHash<QString, QString> [objRubberTexts](#)
- qint64 [objID](#)

17.69.1 Member Enumeration Documentation

17.69.1.1 anonymous enum [anonymous enum](#)

Enumerator

| | |
|------|---------------------------------|
| Type | <input type="button" value=""/> |
|------|---------------------------------|

17.69.2 Constructor & Destructor Documentation

17.69.2.1 [PointObject\(\)](#) [1/2] [PointObject](#) (

```
    EmbReal x,  
    EmbReal y,  
    QRgb rgb,  
    QGraphicsItem * parent = 0 )
```

17.69.2.2 [PointObject\(\)](#) [2/2] [PointObject](#) (

```
    PointObject * obj,  
    QGraphicsItem * parent = 0 )
```

17.69.2.3 [~PointObject\(\)](#) [~PointObject](#) ()

17.69.3 Member Function Documentation

17.69.3.1 allGripPoints() QList< QPointF > allGripPoints () [virtual]

Implements [BaseObject](#).

17.69.3.2 gripEdit() void gripEdit (const QPointF & *before*, const QPointF & *after*) [virtual]

Implements [BaseObject](#).

17.69.3.3 init() void init (EmbReal *x*, EmbReal *y*, QRgb *rgb*, Qt::PenStyle *lineType*)

17.69.3.4 mouseSnapPoint() QPointF mouseSnapPoint (const QPointF & *mousePoint*) [virtual]

Implements [BaseObject](#).

17.69.3.5 objectPos() QPointF objectPos () const [inline]

17.69.3.6 objectSavePath() QPainterPath objectSavePath () const

17.69.3.7 objectX() EmbReal objectX () const [inline]

17.69.3.8 objectY() EmbReal objectY () const [inline]

17.69.3.9 paint() void paint (QPainter * *painter*, const QStyleOptionGraphicsItem * *option*, QWidget *) [protected]

17.69.3.10 setObjectPos() [1/2] void setObjectPos (const QPointF & point) [inline]

17.69.3.11 setObjectPos() [2/2] void setObjectPos (EmbReal x, EmbReal y) [inline]

17.69.3.12 setObjectX() void setObjectX (EmbReal x) [inline]

17.69.3.13 setObjectY() void setObjectY (EmbReal y) [inline]

17.69.3.14 type() virtual int type () const [inline], [virtual]

Reimplemented from [BaseObject](#).

17.69.3.15 updateRubber() void updateRubber (QPainter * painter = 0)

17.69.3.16 vulcanize() void vulcanize () [virtual]

Implements [BaseObject](#).

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/object-point.cpp](#)

17.70 PolygonObject Class Reference

```
#include <embroidermodder.h>
```

Public Types

- enum { [Type](#) = OBJ_TYPE_POLYGON }

Public Types inherited from [BaseObject](#)

- enum { [Type](#) = OBJ_TYPE_BASE }

Public Member Functions

- [PolygonObject \(EmbReal x, EmbReal y, const QPainterPath &p, QRgb rgb, QGraphicsItem *parent=0\)](#)
- [PolygonObject \(PolygonObject *obj, QGraphicsItem *parent=0\)](#)
- [~PolygonObject \(\)](#)
- virtual int [type \(\) const](#)
- void [init \(EmbReal x, EmbReal y, const QPainterPath &p, QRgb rgb, Qt::PenStyle lineType\)](#)
- void [updatePath \(const QPainterPath &p\)](#)
- int [findIndex \(const QPointF &point\)](#)
- QPainterPath [objectCopyPath \(\) const](#)
- QPainterPath [objectSavePath \(\) const](#)
- QPointF [objectPos \(\) const](#)
- EmbReal [objectX \(\) const](#)
- EmbReal [objectY \(\) const](#)
- void [setObjectPos \(const QPointF &point\)](#)
- void [setObjectPos \(EmbReal x, EmbReal y\)](#)
- void [setObjectX \(EmbReal x\)](#)
- void [setObjectY \(EmbReal y\)](#)
- void [updateRubber \(QPainter *painter=0\)](#)
- virtual void [vulcanize \(\)](#)
- virtual QPointF [mouseSnapPoint \(const QPointF &mousePoint\)](#)
- virtual QList< QPointF > [allGripPoints \(\)](#)
- virtual void [gripEdit \(const QPointF &before, const QPointF &after\)](#)

Public Member Functions inherited from [BaseObject](#)

- [BaseObject \(QGraphicsItem *parent=0\)](#)
- virtual [~BaseObject \(\)](#)
- virtual int [type \(\) const](#)
- qint64 [objectID \(\) const](#)
- QPen [objectPen \(\) const](#)
- QColor [objectColor \(\) const](#)
- QRgb [objectColorRGB \(\) const](#)
- Qt::PenStyle [objectLineType \(\) const](#)
- EmbReal [objectLineWidth \(\) const](#)
- QPainterPath [objectPath \(\) const](#)
- int [objectRubberMode \(\) const](#)
- QPointF [objectRubberPoint \(const QString &key\) const](#)
- QString [objectRubberText \(const QString &key\) const](#)
- QPointF [objectCenter \(\) const](#)
- EmbReal [objectCenterX \(\) const](#)
- EmbReal [objectCenterY \(\) const](#)
- void [setObjectCenter \(EmbVector center\)](#)
- void [setObjectCenterX \(EmbReal centerX\)](#)
- void [setObjectCenterY \(EmbReal centerY\)](#)
- QRectF [rect \(\) const](#)
- void [setRect \(const QRectF &r\)](#)
- void [setRect \(EmbReal x, EmbReal y, EmbReal w, EmbReal h\)](#)

- `QLineF line () const`
- `void setLine (const QLineF &li)`
- `void setLine (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2)`
- `void setObjectColor (const QColor &color)`
- `void setObjectColorRGB (QRgb rgb)`
- `void setObjectLineType (Qt::PenStyle lineType)`
- `void setObjectLineWeight (EmbReal lineWeight)`
- `void setObjectPath (const QPainterPath &p)`
- `void setObjectRubberMode (int mode)`
- `void setObjectRubberPoint (const QString &key, const QPointF &point)`
- `void setObjectRubberText (const QString &key, const QString &txt)`
- `virtual QRectF boundingRect () const`
- `virtual QPainterPath shape () const`
- `void drawRubberLine (const QLineF &rubLine, QPainter *painter=0, const char *colorFromScene=0)`
- `virtual void vulcanize ()=0`
- `virtual QPointF mouseSnapPoint (const QPointF &mousePoint)=0`
- `virtual QList< QPointF > allGripPoints ()=0`
- `virtual void gripEdit (const QPointF &before, const QPointF &after)=0`

Public Attributes

- `QPainterPath normalPath`
- `int gripIndex`

Public Attributes inherited from [BaseObject](#)

- `QPen objPen`
- `QPen lwtPen`
- `QLineF objLine`
- `int objRubberMode`
- `QHash<QString, QPointF> objRubberPoints`
- `QHash<QString, QString> objRubberTexts`
- `qint64 objID`

Protected Member Functions

- `void paint (QPainter *, const QStyleOptionGraphicsItem *, QWidget *)`

Protected Member Functions inherited from [BaseObject](#)

- `QPen lineWidthPen () const`
- `void realRender (QPainter *painter, const QPainterPath &renderPath)`

17.70.1 Member Enumeration Documentation

17.70.1.1 anonymous enum anonymous enum

Enumerator

| | |
|------|---------------------------------|
| Type | <input type="button" value=""/> |
|------|---------------------------------|

17.70.2 Constructor & Destructor Documentation**17.70.2.1 [PolygonObject\(\)](#) [1/2]** [PolygonObject](#) (

```
EmbReal x,  
EmbReal y,  
const QPainterPath & p,  
QRgb rgb,  
QGraphicsItem * parent = 0 )
```

17.70.2.2 [PolygonObject\(\)](#) [2/2] [PolygonObject](#) (

```
PolygonObject * obj,  
QGraphicsItem * parent = 0 )
```

17.70.2.3 [~PolygonObject\(\)](#) [~PolygonObject](#) ()**17.70.3 Member Function Documentation****17.70.3.1 [allGripPoints\(\)](#)** [QList< QPointF >](#) allGripPoints () [virtual]

Implements [BaseObject](#).

17.70.3.2 [findIndex\(\)](#) int findIndex (
 const QPointF & point)**17.70.3.3 [gripEdit\(\)](#)** void gripEdit (
 const QPointF & before,
 const QPointF & after) [virtual]

Implements [BaseObject](#).

```
17.70.3.4 init() void init (
    EmbReal x,
    EmbReal y,
    const QPainterPath & p,
    QRgb rgb,
    Qt::PenStyle lineType )
```

```
17.70.3.5 mouseSnapPoint() QPointF mouseSnapPoint (
    const QPointF & mousePoint ) [virtual]
```

Implements [BaseObject](#).

```
17.70.3.6 objectCopyPath() QPainterPath objectCopyPath () const
```

```
17.70.3.7 objectPos() QPointF objectPos () const [inline]
```

```
17.70.3.8 objectSavePath() QPainterPath objectSavePath () const
```

```
17.70.3.9 objectX() EmbReal objectX () const [inline]
```

```
17.70.3.10 objectY() EmbReal objectY () const [inline]
```

```
17.70.3.11 paint() void paint (
    QPainter * painter,
    const QStyleOptionGraphicsItem * option,
    QWidget * ) [protected]
```

```
17.70.3.12 setObjectPos() [1/2] void setObjectPos (
    const QPointF & point ) [inline]
```

17.70.3.13 `setObjectPos()` [2/2] void setObjectPos (EmbReal x, EmbReal y) [inline]

17.70.3.14 `setObjectX()` void setObjectX (EmbReal x) [inline]

17.70.3.15 `setObjectY()` void setObjectY (EmbReal y) [inline]

17.70.3.16 `type()` virtual int type () const [inline], [virtual]

Reimplemented from [BaseObject](#).

17.70.3.17 `updatePath()` void updatePath (const QPainterPath & p)

17.70.3.18 `updateRubber()` void updateRubber (QPainter * painter = 0)

17.70.3.19 `vulcanize()` void vulcanize () [virtual]

Implements [BaseObject](#).

17.70.4 Member Data Documentation

17.70.4.1 `gripIndex` int gripIndex

17.70.4.2 **normalPath** QPainterPath normalPath

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/object-polygon.cpp](#)

17.71 PolylineObject Class Reference

```
#include <embroidermodder.h>
```

Public Types

- enum { [Type](#) = OBJ_TYPE_POLYLINE }

Public Types inherited from [BaseObject](#)

- enum { [Type](#) = OBJ_TYPE_BASE }

Public Member Functions

- [PolylineObject \(EmbReal x, EmbReal y, const QPainterPath &p, QRgb rgb, QGraphicsItem *parent=0\)](#)
- [PolylineObject \(PolylineObject *obj, QGraphicsItem *parent=0\)](#)
- [~PolylineObject \(\)](#)
- virtual int [type \(\) const](#)
- void [init \(EmbReal x, EmbReal y, const QPainterPath &p, QRgb rgb, Qt::PenStyle lineType\)](#)
- void [updatePath \(const QPainterPath &p\)](#)
- int [findIndex \(const QPointF &point\)](#)
- QPainterPath [objectCopyPath \(\) const](#)
- QPainterPath [objectSavePath \(\) const](#)
- QPointF [objectPos \(\) const](#)
- EmbReal [objectX \(\) const](#)
- EmbReal [objectY \(\) const](#)
- void [setObjectPos \(const QPointF &point\)](#)
- void [setObjectPos \(EmbReal x, EmbReal y\)](#)
- void [setObjectX \(EmbReal x\)](#)
- void [setObjectY \(EmbReal y\)](#)
- void [updateRubber \(Painter *painter=0\)](#)
- virtual void [vulcanize \(\)](#)
- virtual QPointF [mouseSnapPoint \(const QPointF &mousePoint\)](#)
- virtual QList< QPointF > [allGripPoints \(\)](#)
- virtual void [gripEdit \(const QPointF &before, const QPointF &after\)](#)

Public Member Functions inherited from `BaseObject`

- `BaseObject` (QGraphicsItem *parent=0)
- virtual `~BaseObject` ()
- virtual int `type` () const
- qint64 `objectID` () const
- QPen `objectPen` () const
- QColor `objectColor` () const
- QRgb `objectColorRGB` () const
- Qt::PenStyle `objectLineType` () const
- EmbReal `objectLineWidth` () const
- QPainterPath `objectPath` () const
- int `objectRubberMode` () const
- QPointF `objectRubberPoint` (const QString &key) const
- QString `objectRubberText` (const QString &key) const
- QPointF `objectCenter` () const
- EmbReal `objectCenterX` () const
- EmbReal `objectCenterY` () const
- void `setObjectCenter` (EmbVector center)
- void `setObjectCenterX` (EmbReal centerX)
- void `setObjectCenterY` (EmbReal centerY)
- QRectF `rect` () const
- void `setRect` (const QRectF &r)
- void `setRect` (EmbReal x, EmbReal y, EmbReal w, EmbReal h)
- QLineF `line` () const
- void `setLine` (const QLineF &l)
- void `setLine` (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2)
- void `setObjectColor` (const QColor &color)
- void `setObjectColorRGB` (QRgb rgb)
- void `setObjectLineType` (Qt::PenStyle lineType)
- void `setObjectLineWidth` (EmbReal lineWidth)
- void `setObjectPath` (const QPainterPath &p)
- void `setObjectRubberMode` (int mode)
- void `setObjectRubberPoint` (const QString &key, const QPointF &point)
- void `setObjectRubberText` (const QString &key, const QString &txt)
- virtual QRectF `boundingRect` () const
- virtual QPainterPath `shape` () const
- void `drawRubberLine` (const QLineF &rubLine, QPainter *painter=0, const char *colorFromScene=0)
- virtual void `vulcanize` ()=0
- virtual QPointF `mouseSnapPoint` (const QPointF &mousePoint)=0
- virtual QList< QPointF > `allGripPoints` ()=0
- virtual void `gripEdit` (const QPointF &before, const QPointF &after)=0

Public Attributes

- QPainterPath `normalPath`
- int `gripIndex`

Public Attributes inherited from BaseObject

- QPen `objPen`
- QPen `lwtPen`
- QLineF `objLine`
- int `objRubberMode`
- QHash<QString, QPointF> `objRubberPoints`
- QHash<QString, QString> `objRubberTexts`
- qint64 `objID`

Protected Member Functions

- void `paint` (QPainter *, const QStyleOptionGraphicsItem *, QWidget *)

Protected Member Functions inherited from BaseObject

- QPen `lineWeightPen () const`
- void `realRender` (QPainter *painter, const QPainterPath &renderPath)

17.71.1 Member Enumeration Documentation**17.71.1.1 anonymous enum** anonymous enum**Enumerator**

| | |
|------|---------------------------------|
| Type | <input type="button" value=""/> |
|------|---------------------------------|

17.71.2 Constructor & Destructor Documentation**17.71.2.1 PolylineObject() [1/2]** `PolylineObject (`

```
    EmbReal x,
    EmbReal y,
    const QPainterPath & p,
    QRgb rgb,
    QGraphicsItem * parent = 0 )
```

17.71.2.2 PolylineObject() [2/2] `PolylineObject (`

```
    PolylineObject * obj,
    QGraphicsItem * parent = 0 )
```

17.71.2.3 ~PolylineObject() ~[PolylineObject](#) ()

17.71.3 Member Function Documentation

17.71.3.1 allGripPoints() QList< [QPointF](#) > allGripPoints () [virtual]

Implements [BaseObject](#).

17.71.3.2 findIndex() int findIndex (const [QPointF](#) & point)

17.71.3.3 gripEdit() void gripEdit (const [QPointF](#) & before, const [QPointF](#) & after) [virtual]

Implements [BaseObject](#).

17.71.3.4 init() void init ([EmbReal](#) x, [EmbReal](#) y, const [QPainterPath](#) & p, [QRgb](#) rgb, [Qt::PenStyle](#) lineType)

17.71.3.5 mouseSnapPoint() [QPointF](#) mouseSnapPoint (const [QPointF](#) & mousePoint) [virtual]

Implements [BaseObject](#).

17.71.3.6 objectCopyPath() [QPainterPath](#) objectCopyPath () const

17.71.3.7 objectPos() [QPointF](#) objectPos () const [inline]

17.71.3.8 objectSavePath() QPainterPath objectSavePath () const

17.71.3.9 objectX() EmbReal objectX () const [inline]

17.71.3.10 objectY() EmbReal objectY () const [inline]

17.71.3.11 paint() void paint (
 QPainter * painter,
 const QStyleOptionGraphicsItem * option,
 QWidget *) [protected]

17.71.3.12 setObjectPos() [1/2] void setObjectPos (
 const QPointF & point) [inline]

17.71.3.13 setObjectPos() [2/2] void setObjectPos (
 EmbReal x,
 EmbReal y) [inline]

17.71.3.14 setObjectX() void setObjectX (
 EmbReal x) [inline]

17.71.3.15 setObjectY() void setObjectY (
 EmbReal y) [inline]

17.71.3.16 type() virtual int type () const [inline], [virtual]

Reimplemented from [BaseObject](#).

```
17.71.3.17 updatePath() void updatePath (
    const QPainterPath & p )
```

```
17.71.3.18 updateRubber() void updateRubber (
    QPainter * painter = 0 )
```

```
17.71.3.19 vulcanize() void vulcanize ( ) [virtual]
```

Implements [BaseObject](#).

17.71.4 Member Data Documentation

```
17.71.4.1 gripIndex int gripIndex
```

```
17.71.4.2 normalPath QPainterPath normalPath
```

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/object-polyline.cpp](#)

17.72 PreviewDialog Class Reference

```
#include <embroidermodder.h>
```

Public Member Functions

- [PreviewDialog](#) (QWidget *parent=0, const QString &caption=QString(), const QString &directory=QString(), const QString &filter=QString())
- [~PreviewDialog](#) ()

Public Attributes

- [ImageWidget](#) * imgWidget

17.72.1 Constructor & Destructor Documentation

```
17.72.1.1 PreviewDialog() PreviewDialog ( QWidget * parent = 0,
    const QString & caption = QString(),
    const QString & directory = QString(),
    const QString & filter = QString() )
```

```
17.72.1.2 ~PreviewDialog() ~PreviewDialog ( )
```

17.72.2 Member Data Documentation

```
17.72.2.1 imgWidget ImageWidget* imgWidget
```

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/preview-dialog.cpp](#)

17.73 PropertyEditor Class Reference

```
#include <embroidermodder.h>
```

Public Slots

- void [setSelectedItems](#) (QList< QGraphicssItem * > itemList)
- void [updatePickAddModeButton](#) (bool pickAddMode)

Signals

- void [pickAddModeToggled](#) ()

Public Member Functions

- [PropertyEditor](#) (const QString &iconDirectory=QString(), bool pickAddMode=true, QWidget *widgetToFocus=0, QWidget *parent=0)
- [~PropertyEditor](#) ()

Protected Member Functions

- bool [eventFilter](#) (QObject *obj, QEvent *event)

Private Slots

- void `fieldEdited` (QObject *fieldObj)
- void `showGroups` (int objType)
- void `showOneType` (int index)
- void `hideAllGroups` ()
- void `clearAllFields` ()
- void `togglePickAddMode` ()

Private Member Functions

- QToolButton * `createToolButton` (const QString &iconName, const QString &txt)
- QLineEdit * `createLineEdit` (const QString &validatorType=QString(), bool readOnly=false)
- QComboBox * `createComboBox` (bool disable=false)
- QFontComboBox * `createFontComboBox` (bool disable=false)
- void `updateLineEditStrIfVaries` (QLineEdit *lineEdit, const QString &str)
- void `updateLineEditNumIfVaries` (QLineEdit *lineEdit, `EmbReal` num, bool useAnglePrecision)
- void `updateFontComboBoxStrIfVaries` (QFontComboBox *fontComboBox, const QString &str)
- void `updateComboBoxStrIfVaries` (QComboBox *comboBox, const QString &str, const QStringList &strList)
- void `updateComboBoxBoolIfVaries` (QComboBox *comboBox, bool val, bool yesOrNoText)
- void `mapSignal` (QObject *fieldObj, const QString &name, QVariant value)
- QComboBox * `createComboBoxSelected` ()
- QToolButton * `createToolButtonQSelect` ()
- QToolButton * `createToolButtonPickAdd` ()
- QGroupBox * `createGroupBoxGeneral` ()
- QGroupBox * `createGroupBoxGeometryArc` ()
- QGroupBox * `createGroupBoxMiscArc` ()
- QGroupBox * `createGroupBoxGeometryBlock` ()
- QGroupBox * `createGroupBoxGeometryCircle` ()
- QGroupBox * `createGroupBoxGeometryDimAligned` ()
- QGroupBox * `createGroupBoxGeometryDimAngular` ()
- QGroupBox * `createGroupBoxGeometryDimArcLength` ()
- QGroupBox * `createGroupBoxGeometryDimDiameter` ()
- QGroupBox * `createGroupBoxGeometryDimLeader` ()
- QGroupBox * `createGroupBoxGeometryDimLinear` ()
- QGroupBox * `createGroupBoxGeometryDimOrdinate` ()
- QGroupBox * `createGroupBoxGeometryDimRadius` ()
- QGroupBox * `createGroupBoxGeometryEllipse` ()
- QGroupBox * `createGroupBoxGeometryImage` ()
- QGroupBox * `createGroupBoxMiscImage` ()
- QGroupBox * `createGroupBoxGeometryInfiniteLine` ()
- QGroupBox * `createGroupBoxGeometryLine` ()
- QGroupBox * `createGroupBoxGeometryPath` ()
- QGroupBox * `createGroupBoxMiscPath` ()
- QGroupBox * `createGroupBoxGeometryPoint` ()
- QGroupBox * `createGroupBoxGeometryPolygon` ()
- QGroupBox * `createGroupBoxGeometryPolyline` ()
- QGroupBox * `createGroupBoxMiscPolyline` ()
- QGroupBox * `createGroupBoxGeometryRay` ()
- QGroupBox * `createGroupBoxGeometryRectangle` ()
- QGroupBox * `createGroupBoxGeometryTextMulti` ()
- QGroupBox * `createGroupBoxTextTextSingle` ()
- QGroupBox * `createGroupBoxGeometryTextSingle` ()
- QGroupBox * `createGroupBoxMiscTextSingle` ()

Private Attributes

- QWidget * `focusWidget`
- QString `iconDir`
- int `iconSize`
- Qt::ToolButtonStyle `propertyEditorButtonStyle`
- bool `pickAdd`
- QList<QGraphicsItem * > `selectedItemList`
- `ArcObject` * `tempArcObj`
- `BlockObject` * `tempBlockObj`
- `CircleObject` * `tempCircleObj`
- `DimAlignedObject` * `tempDimAlignedObj`
- `DimAngularObject` * `tempDimAngularObj`
- `DimArcLengthObject` * `tempDimArcLenObj`
- `DimDiameterObject` * `tempDimDiamObj`
- `DimLeaderObject` * `tempDimLeaderObj`
- `DimLinearObject` * `tempDimLinearObj`
- `DimOrdinateObject` * `tempDimOrdObj`
- `DimRadiusObject` * `tempDimRadiusObj`
- `EllipseObject` * `tempEllipseObj`
- `EllipseArcObject` * `tempEllipseArcObj`
- `HatchObject` * `tempHatchObj`
- `ImageObject` * `tempImageObj`
- `InfiniteLineObject` * `tempInfLineObj`
- `LineObject` * `tempLineObj`
- `PathObject` * `tempPathObj`
- `PointObject` * `tempPointObj`
- `PolygonObject` * `tempPolygonObj`
- `PolylineObject` * `tempPolylineObj`
- `RayObject` * `tempRayObj`
- `RectObject` * `tempRectObj`
- `SplineObject` * `tempSplineObj`
- `TextMultiObject` * `tempTextMultiObj`
- `TextSingleObject` * `tempTextSingleObj`
- int `precisionAngle`
- int `precisionLength`
- QString `fieldOldText`
- QString `fieldNewText`
- QString `fieldVariesText`
- QString `fieldYesText`
- QString `fieldNoText`
- QString `fieldOnText`
- QString `fieldOffText`
- QSignalMapper * `signalMapper`
- QComboBox * `comboBoxSelected`
- QToolButton * `toolButtonQSelect`
- QToolButton * `toolButtonPickAdd`

17.73.1 Constructor & Destructor Documentation

```
17.73.1.1 PropertyEditor() PropertyEditor (  
    const QString & iconDirectory = QString(),  
    bool pickAddMode = true,  
    QWidget * widgetToFocus = 0,  
    QWidget * parent = 0 )
```

17.73.1.2 ~PropertyEditor() *~PropertyEditor* ()

Todo document this

17.73.2 Member Function Documentation

17.73.2.1 clearAllFields void *clearAllFields* () [private], [slot]

17.73.2.2 createComboBox() QComboBox * *createComboBox* (
 bool *disable* = false) [private]

17.73.2.3 createComboBoxSelected() QComboBox * *createComboBoxSelected* () [private]

Todo document this

17.73.2.4 createFontComboBox() QFontComboBox * *createFontComboBox* (
 bool *disable* = false) [private]

17.73.2.5 createGroupBoxGeneral() QGroupBox * *createGroupBoxGeneral* () [private]

Todo use proper icons for toolButtons

17.73.2.6 `createGroupBoxGeometryArc()` `QGroupBox * createGroupBoxGeometryArc ()` [private]

Todo use proper icons for toolButtons

17.73.2.7 `createGroupBoxGeometryBlock()` `QGroupBox * createGroupBoxGeometryBlock ()` [private]

Todo use proper icons for toolButtons

mapSignal for blocks

17.73.2.8 `createGroupBoxGeometryCircle()` `QGroupBox * createGroupBoxGeometryCircle ()` [private]

Todo use proper icons for toolButtons

17.73.2.9 `createGroupBoxGeometryDimAligned()` `QGroupBox * createGroupBoxGeometryDimAligned ()` [private]

17.73.2.10 `createGroupBoxGeometryDimAngular()` `QGroupBox * createGroupBoxGeometryDimAngular ()` [private]

17.73.2.11 `createGroupBoxGeometryDimArcLength()` `QGroupBox * createGroupBoxGeometryDimArcLength ()` [private]

17.73.2.12 `createGroupBoxGeometryDimDiameter()` `QGroupBox * createGroupBoxGeometryDimDiameter ()` [private]

17.73.2.13 `createGroupBoxGeometryDimLeader()` `QGroupBox * createGroupBoxGeometryDimLeader ()` [private]

17.73.2.14 `createGroupBoxGeometryDimLinear()` `QGroupBox * createGroupBoxGeometryDimLinear ()`
[private]

17.73.2.15 `createGroupBoxGeometryDimOrdinate()` `QGroupBox * createGroupBoxGeometryDimOrdinate ()` [private]

17.73.2.16 `createGroupBoxGeometryDimRadius()` `QGroupBox * createGroupBoxGeometryDimRadius ()`
[private]

17.73.2.17 `createGroupBoxGeometryEllipse()` `QGroupBox * createGroupBoxGeometryEllipse ()` [private]

Todo use proper icons for toolButtons

17.73.2.18 `createGroupBoxGeometryImage()` `QGroupBox * createGroupBoxGeometryImage ()` [private]

Todo use proper icons for toolButtons
mapSignal for images

17.73.2.19 `createGroupBoxGeometryInfiniteLine()` `QGroupBox * createGroupBoxGeometryInfiniteLine ()` [private]

Todo use proper icons for toolButtons
mapSignal for infinite lines

17.73.2.20 `createGroupBoxGeometryLine()` `QGroupBox * createGroupBoxGeometryLine ()` [private]

17.73.2.21 `createGroupBoxGeometryPath()` `QGroupBox * createGroupBoxGeometryPath ()` [private]

17.73.2.22 `createGroupBoxGeometryPoint()` `QGroupBox * createGroupBoxGeometryPoint () [private]`

17.73.2.23 `createGroupBoxGeometryPolygon()` `QGroupBox * createGroupBoxGeometryPolygon () [private]`

17.73.2.24 `createGroupBoxGeometryPolyline()` `QGroupBox * createGroupBoxGeometryPolyline () [private]`

17.73.2.25 `createGroupBoxGeometryRay()` `QGroupBox * createGroupBoxGeometryRay () [private]`

17.73.2.26 `createGroupBoxGeometryRectangle()` `QGroupBox * createGroupBoxGeometryRectangle () [private]`

17.73.2.27 `createGroupBoxGeometryTextMulti()` `QGroupBox * createGroupBoxGeometryTextMulti () [private]`

17.73.2.28 `createGroupBoxGeometryTextSingle()` `QGroupBox * createGroupBoxGeometryTextSingle () [private]`

17.73.2.29 `createGroupBoxMiscArc()` `QGroupBox * createGroupBoxMiscArc () [private]`

Todo use proper icons for toolButtons

17.73.2.30 `createGroupBoxMiscImage()` `QGroupBox * createGroupBoxMiscImage () [private]`

Todo use proper icons for toolButtons

17.73.2.31 `createGroupBoxMiscPath()` `QGroupBox * createGroupBoxMiscPath () [private]`

17.73.2.32 `createGroupBoxMiscPolyline()` `QGroupBox * createGroupBoxMiscPolyline () [private]`

17.73.2.33 `createGroupBoxMiscTextSingle()` `QGroupBox * createGroupBoxMiscTextSingle () [private]`

17.73.2.34 `createGroupBoxTextTextSingle()` `QGroupBox * createGroupBoxTextTextSingle () [private]`

17.73.2.35 `createLineEdit()` `QLineEdit * createLineEdit (`
 `const QString & validatorType = QString(),`
 `bool readOnly = false) [private]`

17.73.2.36 `createToolButton()` `QToolButton * createToolButton (`
 `const QString & iconName,`
 `const QString & txt) [private]`

17.73.2.37 `createToolButtonPickAdd()` `QToolButton * createToolButtonPickAdd () [private]`

17.73.2.38 `createToolButtonQSelect()` `QToolButton * createToolButtonQSelect () [private]`

Todo document this

17.73.2.39 `eventFilter()` `bool eventFilter (`
 `QObject * obj,`
 `QEvent * event) [protected]`

Todo document this

17.73.2.40 fieldEdited void fieldEdited (QObject * *fieldObj*) [private], [slot]

17.73.2.41 hideAllGroups void hideAllGroups () [private], [slot]

17.73.2.42 mapSignal() void mapSignal (QObject * *fieldObj*, const QString & *name*, QVariant *value*) [private]

17.73.2.43 pickAddModeToggled void pickAddModeToggled () [signal]

17.73.2.44 setSelectedItems void setSelectedItems (QList< QGraphicsItem * > *itemList*) [slot]

17.73.2.45 showGroups void showGroups (int *objType*) [private], [slot]

17.73.2.46 showOneType void showOneType (int *index*) [private], [slot]

17.73.2.47 togglePickAddMode void togglePickAddMode () [private], [slot]

17.73.2.48 updateComboBoxBoolIfVaries() void updateComboBoxBoolIfVaries (QComboBox * *comboBox*, bool *val*, bool *yesOrNoText*) [private]

17.73.2.49 updateComboBoxStrIfVaries() void updateComboBoxStrIfVaries (QComboBox * *comboBox*, const QString & *str*, const QStringList & *strList*) [private]

17.73.2.50 updateFontComboBoxStrIfVaries() void updateFontComboBoxStrIfVaries (QFontComboBox * *fontComboBox*, const QString & *str*) [private]

17.73.2.51 updateLineEditNumIfVaries() void updateLineEditNumIfVaries (QLineEdit * *lineEdit*, EmbReal *num*, bool *useAnglePrecision*) [private]

17.73.2.52 updateLineEditStrIfVaries() void updateLineEditStrIfVaries (QLineEdit * *lineEdit*, const QString & *str*) [private]

17.73.2.53 updatePickAddModeButton void updatePickAddModeButton (bool *pickAddMode*) [slot]

17.73.3 Member Data Documentation

17.73.3.1 comboBoxSelected QComboBox* *comboBoxSelected* [private]

17.73.3.2 fieldNewText QString *fieldNewText* [private]

17.73.3.3 fieldNoText QString *fieldNoText* [private]

17.73.3.4 fieldOffText QString *fieldOffText* [private]

17.73.3.5 fieldOldText `QString fieldOldText [private]`

17.73.3.6 fieldOnText `QString fieldOnText [private]`

17.73.3.7 fieldVariesText `QString fieldVariesText [private]`

17.73.3.8 fieldYesText `QString fieldYesText [private]`

17.73.3.9 focusWidget `QWidget* focusWidget [private]`

17.73.3.10 iconDir `QString iconDir [private]`

17.73.3.11 iconSize `int iconSize [private]`

17.73.3.12 pickAdd `bool pickAdd [private]`

17.73.3.13 precisionAngle `int precisionAngle [private]`

17.73.3.14 precisionLength `int precisionLength [private]`

17.73.3.15 propertyEditorButtonStyle `Qt::ToolButtonStyle propertyEditorButtonStyle [private]`

17.73.3.16 selectedItemList QList<QGraphicsItem*> selectedItemList [private]

17.73.3.17 signalMapper QSignalMapper* signalMapper [private]

17.73.3.18 tempArcObj ArcObject* tempArcObj [private]

17.73.3.19 tempBlockObj BlockObject* tempBlockObj [private]

17.73.3.20 tempCircleObj CircleObject* tempCircleObj [private]

17.73.3.21 tempDimAlignedObj DimAlignedObject* tempDimAlignedObj [private]

17.73.3.22 tempDimAngularObj DimAngularObject* tempDimAngularObj [private]

17.73.3.23 tempDimArcLenObj DimArcLengthObject* tempDimArcLenObj [private]

17.73.3.24 tempDimDiamObj DimDiameterObject* tempDimDiamObj [private]

17.73.3.25 tempDimLeaderObj DimLeaderObject* tempDimLeaderObj [private]

17.73.3.26 tempDimLinearObj DimLinearObject* tempDimLinearObj [private]

17.73.3.27 tempDimOrdObj DimOrdinateObject* tempDimOrdObj [private]

17.73.3.28 tempDimRadiusObj DimRadiusObject* tempDimRadiusObj [private]

17.73.3.29 tempEllipseArcObj EllipseArcObject* tempEllipseArcObj [private]

17.73.3.30 tempEllipseObj EllipseObject* tempEllipseObj [private]

17.73.3.31 tempHatchObj HatchObject* tempHatchObj [private]

17.73.3.32 tempImageObj ImageObject* tempImageObj [private]

17.73.3.33 tempInfLineObj InfiniteLineObject* tempInfLineObj [private]

17.73.3.34 tempLineObj LineObject* tempLineObj [private]

17.73.3.35 tempPathObj PathObject* tempPathObj [private]

17.73.3.36 tempPointObj PointObject* tempPointObj [private]

17.73.3.37 tempPolygonObj PolygonObject* tempPolygonObj [private]

17.73.3.38 tempPolylineObj `PolylineObject*` `tempPolylineObj` [private]

17.73.3.39 tempRayObj `RayObject*` `tempRayObj` [private]

17.73.3.40 tempRectObj `RectObject*` `tempRectObj` [private]

17.73.3.41 tempSplineObj `SplineObject*` `tempSplineObj` [private]

17.73.3.42 tempTextMultiObj `TextMultiObject*` `tempTextMultiObj` [private]

17.73.3.43 tempTextSingleObj `TextSingleObject*` `tempTextSingleObj` [private]

17.73.3.44 toolButtonPickAdd `QToolButton*` `toolButtonPickAdd` [private]

17.73.3.45 toolButtonQSelect `QToolButton*` `toolButtonQSelect` [private]

The documentation for this class was generated from the following files:

- `embroidermodder2/embroidermodder.h`
- `embroidermodder2/property-editor.cpp`

17.74 RectObject Class Reference

```
#include <embroidermodder.h>
```

Public Types

- enum { `Type` = `OBJ_TYPE_RECTANGLE` }

Public Types inherited from BaseObject

- enum { `Type` = `OBJ_TYPE_BASE` }

Public Member Functions

- `RectObject (EmbReal x, EmbReal y, EmbReal w, EmbReal h, QRgb rgb, QGraphicsItem *parent=0)`
- `RectObject (RectObject *obj, QGraphicsItem *parent=0)`
- `~RectObject ()`
RectObject destructor.
- virtual int `type () const`
- `QPainterPath objectSavePath () const`
- void `init (EmbReal x, EmbReal y, EmbReal w, EmbReal h, QRgb rgb, Qt::PenStyle lineType)`
- void `updatePath ()`
- `QPointF objectPos () const`
- `QPointF objectTopLeft () const`
- `QPointF objectTopRight () const`
- `QPointF objectBottomLeft () const`
- `QPointF objectBottomRight () const`
- `EmbReal objectWidth () const`
- `EmbReal objectHeight () const`
- `EmbReal objectArea () const`
- void `setObjectRect (EmbReal x, EmbReal y, EmbReal w, EmbReal h)`
- void `updateRubber (QPainter *painter=0)`
- virtual void `vulcanize ()`
- virtual `QPointF mouseSnapPoint (const QPointF &mousePoint)`
- virtual `QList< QPointF > allGripPoints ()`
- virtual void `gripEdit (const QPointF &before, const QPointF &after)`

Public Member Functions inherited from BaseObject

- `BaseObject (QGraphicsItem *parent=0)`
- virtual `~BaseObject ()`
- virtual int `type () const`
- qint64 `objectID () const`
- `QPen objectPen () const`
- `QColor objectColor () const`
- `QRgb objectColorRGB () const`
- `Qt::PenStyle objectLineType () const`
- `EmbReal objectLineWidth () const`
- `QPainterPath objectPath () const`
- int `objectRubberMode () const`
- `QPointF objectRubberPoint (const QString &key) const`
- `QString objectRubberText (const QString &key) const`
- `QPointF objectCenter () const`
- `EmbReal objectCenterX () const`
- `EmbReal objectCenterY () const`
- void `setObjectCenter (EmbVector center)`
- void `setObjectCenterX (EmbReal centerX)`
- void `setObjectCenterY (EmbReal centerY)`
- `QRectF rect () const`
- void `setRect (const QRectF &r)`

- void `setRect` (EmbReal x, EmbReal y, EmbReal w, EmbReal h)
- QLineF `line` () const
- void `setLine` (const QLineF &li)
- void `setLine` (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2)
- void `setObjectColor` (const QColor &color)
- void `setObjectColorRGB` (QRgb rgb)
- void `setObjectLineType` (Qt::PenStyle lineType)
- void `setObjectLineWeight` (EmbReal lineWeight)
- void `setObjectPath` (const QPainterPath &p)
- void `setObjectRubberMode` (int mode)
- void `setObjectRubberPoint` (const QString &key, const QPointF &point)
- void `setObjectRubberText` (const QString &key, const QString &txt)
- virtual QRectF `boundingRect` () const
- virtual QPainterPath `shape` () const
- void `drawRubberLine` (const QLineF &rubLine, QPainter *painter=0, const char *colorFromScene=0)
- virtual void `vulcanize` ()=0
- virtual QPointF `mouseSnapPoint` (const QPointF &mousePoint)=0
- virtual QList< QPointF > `allGripPoints` ()=0
- virtual void `gripEdit` (const QPointF &before, const QPointF &after)=0

Protected Member Functions

- void `paint` (QPainter *, const QStyleOptionGraphicsItem *, QWidget *)

Protected Member Functions inherited from `BaseObject`

- QPen `lineWeightPen` () const
- void `realRender` (QPainter *painter, const QPainterPath &renderPath)

Additional Inherited Members

Public Attributes inherited from `BaseObject`

- QPen `objPen`
- QPen `lwtPen`
- QLineF `objLine`
- int `objRubberMode`
- QHash<QString, QPointF> `objRubberPoints`
- QHash<QString, QString> `objRubberTexts`
- qint64 `objID`

17.74.1 Member Enumeration Documentation

17.74.1.1 anonymous enum anonymous enum

Enumerator

| | |
|------|---------------------------------|
| Type | <input type="button" value=""/> |
|------|---------------------------------|

17.74.2 Constructor & Destructor Documentation

17.74.2.1 RectObject() [1/2] `RectObject (`

```
    EmbReal x,  
    EmbReal y,  
    EmbReal w,  
    EmbReal h,  
    QRgb rgb,  
    QGraphicsItem * parent = 0 )
```

17.74.2.2 RectObject() [2/2] `RectObject (`

```
    RectObject * obj,  
    QGraphicsItem * parent = 0 )
```

17.74.2.3 ~RectObject() `~RectObject ()`

`RectObject` destructor.

17.74.3 Member Function Documentation

17.74.3.1 allGripPoints() `QList< QPointF > allGripPoints () [virtual]`

Returns

A list of all grip points for the object.

Todo make return value a `std::vector<std::string>`

Implements `BaseObject`.

```
17.74.3.2 gripEdit() void gripEdit (
    const QPointF & before,
    const QPointF & after ) [virtual]
```

Implements [BaseObject](#).

```
17.74.3.3 init() void init (
    EmbReal x,
    EmbReal y,
    EmbReal w,
    EmbReal h,
    QRgb rgb,
    Qt::PenStyle lineType )
```

```
17.74.3.4 mouseSnapPoint() QPointF mouseSnapPoint (
    const QPointF & mousePoint ) [virtual]
```

Returns

The closest snap point to the mouse point.

Implements [BaseObject](#).

```
17.74.3.5 objectArea() EmbReal objectArea ( ) const [inline]
```

```
17.74.3.6 objectBottomLeft() QPointF objectBottomLeft ( ) const
```

```
17.74.3.7 objectBottomRight() QPointF objectBottomRight ( ) const
```

```
17.74.3.8 objectHeight() EmbReal objectHeight ( ) const [inline]
```

```
17.74.3.9 objectPos() QPointF objectPos ( ) const [inline]
```

17.74.3.10 objectSavePath() QPainterPath objectSavePath () const

17.74.3.11 objectTopLeft() QPointF objectTopLeft () const

Returns

The top left corner location as a QPointF.

17.74.3.12 objectTopRight() QPointF objectTopRight () const

17.74.3.13 objectWidth() EmbReal objectWidth () const [inline]

17.74.3.14 paint() void paint (
 QPainter * painter,
 const QStyleOptionGraphicsItem * option,
 QWidget *) [protected]

17.74.3.15 setObjectRect() void setObjectRect (
 EmbReal x,
 EmbReal y,
 EmbReal w,
 EmbReal h)

17.74.3.16 type() virtual int type () const [inline], [virtual]

Reimplemented from [BaseObject](#).

17.74.3.17 updatePath() void updatePath ()

17.74.3.18 updateRubber() void updateRubber (
 QPainter * painter = 0)

17.74.3.19 `vulcanize()` `void vulcanize () [virtual]`

Implements [BaseObject](#).

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/object-rect.cpp](#)

17.75 SaveObject Class Reference

```
#include <embroidermodder.h>
```

Public Member Functions

- `SaveObject (QGraphicsScene *theScene, QObject *parent=0)`
- `~SaveObject ()`
- `bool save (const QString &fileName)`
- `void addArc (EmbPattern *pattern, QGraphicsItem *item)`
SaveObject::addArc.
- `void addBlock (EmbPattern *pattern, QGraphicsItem *item)`
SaveObject::addBlock.
- `void addCircle (EmbPattern *pattern, QGraphicsItem *item)`
SaveObject::addCircle.
- `void addDimAligned (EmbPattern *pattern, QGraphicsItem *item)`
SaveObject::addDimAligned.
- `void addDimAngular (EmbPattern *pattern, QGraphicsItem *item)`
SaveObject::addDimAngular.
- `void addDimArcLength (EmbPattern *pattern, QGraphicsItem *item)`
SaveObject::addDimArcLength.
- `void addDimDiameter (EmbPattern *pattern, QGraphicsItem *item)`
SaveObject::addDimDiameter.
- `void addDimLeader (EmbPattern *pattern, QGraphicsItem *item)`
SaveObject::addDimLeader.
- `void addDimLinear (EmbPattern *pattern, QGraphicsItem *item)`
- `void addDimOrdinate (EmbPattern *pattern, QGraphicsItem *item)`
- `void addDimRadius (EmbPattern *pattern, QGraphicsItem *item)`
- `void addEllipse (EmbPattern *pattern, QGraphicsItem *item)`
- `void addEllipseArc (EmbPattern *pattern, QGraphicsItem *item)`
- `void addGrid (EmbPattern *pattern, QGraphicsItem *item)`
- `void addHatch (EmbPattern *pattern, QGraphicsItem *item)`
- `void addImage (EmbPattern *pattern, QGraphicsItem *item)`
- `void addInfiniteLine (EmbPattern *pattern, QGraphicsItem *item)`
- `void addLine (EmbPattern *pattern, QGraphicsItem *item)`
- `void addPath (EmbPattern *pattern, QGraphicsItem *item)`
- `void addPoint (EmbPattern *pattern, QGraphicsItem *item)`
- `void addPolygon (EmbPattern *pattern, QGraphicsItem *item)`
- `void addPolyline (EmbPattern *pattern, QGraphicsItem *item)`
- `void addRay (EmbPattern *pattern, QGraphicsItem *item)`
- `void addRectangle (EmbPattern *pattern, QGraphicsItem *item)`
- `void addSlot (EmbPattern *pattern, QGraphicsItem *item)`
- `void addSpline (EmbPattern *pattern, QGraphicsItem *item)`
- `void addTextMulti (EmbPattern *pattern, QGraphicsItem *item)`
- `void addTextSingle (EmbPattern *pattern, QGraphicsItem *item)`
- `void toPolyline (EmbPattern *pattern, const QPointF &objPos, const QPainterPath &objPath, const QString &layer, const QColor &color, const QString &lineType, const QString &lineWeight)`

Public Attributes

- `QGraphicsScene * gscene`
- `int formatType`

17.75.1 Constructor & Destructor Documentation

17.75.1.1 `SaveObject()` `SaveObject (`
 `QGraphicsScene * theScene,`
 `QObject * parent = 0)`

17.75.1.2 `~SaveObject()` `~SaveObject ()`

17.75.2 Member Function Documentation

17.75.2.1 `addArc()` `void addArc (`
 `EmbPattern * pattern,`
 `QGraphicsItem * item)`

`SaveObject::addArc.`

Parameters

| | |
|----------------------|--|
| <code>pattern</code> | |
| <code>item</code> | |

17.75.2.2 `addBlock()` `void addBlock (`
 `EmbPattern * pattern,`
 `QGraphicsItem * item)`

`SaveObject::addBlock.`

Parameters

| | |
|----------------------|--|
| <code>pattern</code> | |
| <code>item</code> | |

17.75.2.3 addCircle() void addCircle (

```
EmbPattern * pattern,
QGraphicsItem * item )
```

[SaveObject::addCircle](#).

Parameters

| | |
|----------------|---------------------------------|
| <i>pattern</i> | <input type="button" value=""/> |
| <i>item</i> | <input type="button" value=""/> |

17.75.2.4 addDimAligned() void addDimAligned (

```
EmbPattern * pattern,
QGraphicsItem * item )
```

[SaveObject::addDimAligned](#).

Parameters

| | |
|----------------|---------------------------------|
| <i>pattern</i> | <input type="button" value=""/> |
| <i>item</i> | <input type="button" value=""/> |

17.75.2.5 addDimAngular() void addDimAngular (

```
EmbPattern * pattern,
QGraphicsItem * item )
```

[SaveObject::addDimAngular](#).

Parameters

| | |
|----------------|---------------------------------|
| <i>pattern</i> | <input type="button" value=""/> |
| <i>item</i> | <input type="button" value=""/> |

17.75.2.6 addDimArcLength() void addDimArcLength (

```
EmbPattern * pattern,
QGraphicsItem * item )
```

[SaveObject::addDimArcLength](#).

Parameters

| | |
|----------------|---------------------------------|
| <i>pattern</i> | <input type="button" value=""/> |
| <i>item</i> | <input type="button" value=""/> |

```
17.75.2.7 addDimDiameter() void addDimDiameter (
    EmbPattern * pattern,
    QGraphicsItem * item )
```

[SaveObject::addDimDiameter](#).

Parameters

| | |
|----------------|----------------------|
| <i>pattern</i> | <input type="text"/> |
| <i>item</i> | <input type="text"/> |

```
17.75.2.8 addDimLeader() void addDimLeader (
    EmbPattern * pattern,
    QGraphicsItem * item )
```

[SaveObject::addDimLeader](#).

Parameters

| | |
|----------------|----------------------|
| <i>pattern</i> | <input type="text"/> |
| <i>item</i> | <input type="text"/> |

```
17.75.2.9 addDimLinear() void addDimLinear (
    EmbPattern * pattern,
    QGraphicsItem * item )
```

```
17.75.2.10 addDimOrdinate() void addDimOrdinate (
    EmbPattern * pattern,
    QGraphicsItem * item )
```

```
17.75.2.11 addDimRadius() void addDimRadius (
    EmbPattern * pattern,
    QGraphicsItem * item )
```

```
17.75.2.12 addEllipse() void addEllipse (
    EmbPattern * pattern,
    QGraphicsItem * item )
```

17.75.2.13 addEllipseArc() void addEllipseArc (

```
EmbPattern * pattern,
QGraphicsItem * item )
```

17.75.2.14 addGrid() void addGrid (

```
EmbPattern * pattern,
QGraphicsItem * item )
```

17.75.2.15 addHatch() void addHatch (

```
EmbPattern * pattern,
QGraphicsItem * item )
```

17.75.2.16 addImage() void addImage (

```
EmbPattern * pattern,
QGraphicsItem * item )
```

17.75.2.17 addInfiniteLine() void addInfiniteLine (

```
EmbPattern * pattern,
QGraphicsItem * item )
```

17.75.2.18 addLine() void addLine (

```
EmbPattern * pattern,
QGraphicsItem * item )
```

17.75.2.19 addPath() void addPath (

```
EmbPattern * pattern,
QGraphicsItem * item )
```

17.75.2.20 addPoint() void addPoint (

```
EmbPattern * pattern,
QGraphicsItem * item )
```

```
17.75.2.21 addPolygon() void addPolygon (
    EmbPattern * pattern,
    QGraphicsItem * item )
```

```
17.75.2.22 addPolyline() void addPolyline (
    EmbPattern * pattern,
    QGraphicsItem * item )
```

```
17.75.2.23 addRay() void addRay (
    EmbPattern * pattern,
    QGraphicsItem * item )
```

```
17.75.2.24 addRectangle() void addRectangle (
    EmbPattern * pattern,
    QGraphicsItem * item )
```

```
17.75.2.25 addSlot() void addSlot (
    EmbPattern * pattern,
    QGraphicsItem * item )
```

```
17.75.2.26 addSpline() void addSpline (
    EmbPattern * pattern,
    QGraphicsItem * item )
```

```
17.75.2.27 addTextMulti() void addTextMulti (
    EmbPattern * pattern,
    QGraphicsItem * item )
```

```
17.75.2.28 addTextSingle() void addTextSingle (
    EmbPattern * pattern,
    QGraphicsItem * item )
```

```
17.75.2.29 save() bool save (
    const QString & fileName )
```

Returns whether the save to file process was successful.

Todo Before saving to a stitch only format, Embroidermodder needs to calculate the optimal path to minimize jump stitches. Also based upon which layer needs to be stitched first, the path to the next object needs to be hidden beneath fills that will come later. When finding the optimal path, we need to take into account the color of the thread, as we do not want to try to hide dark colored stitches beneath light colored fills.

```
17.75.2.30 toPolyline() void toPolyline (
    EmbPattern * pattern,
    const QPointF & objPos,
    const QPainterPath & objPath,
    const QString & layer,
    const QColor & color,
    const QString & lineType,
    const QString & lineWeight )
```

Todo FIX EmbPolyline* polyObject = embPolyline_init(pointList, color_out, 1); //TODO: proper lineType emb←
Pattern_addPolylineAbs(pattern, polyObject);

17.75.3 Member Data Documentation

17.75.3.1 **formatType** int formatType

17.75.3.2 **gscene** QGraphicsScene* gscene

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/object-save.cpp](#)

17.76 SelectBox Class Reference

```
#include <embroidermodder.h>
```

Public Slots

- void [setDirection](#) (int dir)
- void [setColors](#) (const QColor &colorL, const QColor &fillL, const QColor &colorR, const QColor &fillR, int newAlpha)

Public Member Functions

- `SelectBox (Shape s, QWidget *parent=0)`
- `void forceRepaint ()`

Public Attributes

- `QColor leftBrushColor`
- `QColor rightBrushColor`
- `QColor leftPenColor`
- `QColor rightPenColor`
- `quint8 alpha`
- `QBrush dirBrush`
- `QBrush leftBrush`
- `QBrush rightBrush`
- `QPen dirPen`
- `QPen leftPen`
- `QPen rightPen`
- `bool boxDir`

Protected Member Functions

- `void paintEvent (QPaintEvent *)`

17.76.1 Constructor & Destructor Documentation

```
17.76.1.1 SelectBox() SelectBox (  
    Shape s,  
    QWidget * parent = 0 )
```

Embroidermodder 2,
Copyright 2013-2022 The Embroidermodder Team Embroidermodder 2 is Open Source Software. See LICENSE
for licensing terms.
Use Python's PEP7 style guide. <https://peps.python.org/pep-0007/>

17.76.2 Member Function Documentation

```
17.76.2.1 forceRepaint() void forceRepaint ( )
```

```
17.76.2.2 paintEvent() void paintEvent (  
    QPaintEvent * ) [protected]
```

```
17.76.2.3 setColors void setColors (
    const QColor & colorL,
    const QColor & fillL,
    const QColor & colorR,
    const QColor & fillR,
    int newAlpha ) [slot]
```

```
17.76.2.4 setDirection void setDirection (
    int dir ) [slot]
```

17.76.3 Member Data Documentation

17.76.3.1 alpha quint8 alpha

17.76.3.2 boxDir bool boxDir

17.76.3.3 dirBrush QBrush dirBrush

17.76.3.4 dirPen QPen dirPen

17.76.3.5 leftBrush QBrush leftBrush

17.76.3.6 leftBrushColor QColor leftBrushColor

17.76.3.7 leftPen QPen leftPen

17.76.3.8 leftPenColor QColor leftPenColor

17.76.3.9 rightBrush QBrush rightBrush

17.76.3.10 rightBrushColor QColor rightBrushColor

17.76.3.11 rightPen QPen rightPen

17.76.3.12 rightPenColor QColor rightPenColor

The documentation for this class was generated from the following files:

- embroidermodder2/[embroidermodder.h](#)
- embroidermodder2/[selectbox.cpp](#)

17.77 Settings_Struct Reference

Settings System.

```
#include <embroidermodder.h>
```

Public Attributes

- char `version` [200]
- bool `running`
- bool `testing`
- int `debug_mode`
- bool `show_about_dialog`
- bool `show_settings_editor`
- bool `show_editor`
- bool `show_details_dialog`
- bool `show_open_file_dialog`
- int `icon_size`
- char `icon_theme` [200]
- int `pattern_index`
- char `assets_dir` [200]
- bool `use_translation`
- char `language` [200]
- bool `mdi_bg_use_logo`
- bool `mdi_bg_use_texture`
- bool `mdi_bg_use_color`
- char `general_mdi_bg_logo` [200]
- char `general_mdi_bg_texture` [200]
- uint32_t `general_mdi_bg_color`
- bool `tip_of_the_day`
- uint32_t `general_current_tip`
- bool `general_system_help_browser`
- bool `general_check_for_updates`
- bool `display_use_opengl`
- bool `display_renderhint_aa`
- bool `display_renderhint_text_aa`
- bool `display_renderhint_smooth_pix`
- bool `display_renderhint_high_aa`
- bool `display_renderhint_noncosmetic`
- bool `display_show_scrollbars`
- int `display_scrollbar_widget_num`
- uint32_t `display_crosshair_color`
- uint32_t `display_bg_color`
- uint32_t `display_selectbox_left_color`
- uint32_t `display_selectbox_left_fill`
- uint32_t `display_selectbox_right_color`
- uint32_t `display_selectbox_right_fill`
- uint8_t `display_selectbox_alpha`
- EmbReal `display_zoomscale_in`
- EmbReal `display_zoomscale_out`
- uint8_t `display_crosshair_percent`
- std::string `display_units`
- std::string `opensave_custom_filter`
- std::string `opensave_open_format`
- bool `opensave_open_thumbnail`
- std::string `opensave_save_format`

- bool `opensave_save_thumbnail`
- uint8_t `opensave_recent_max_files`
- std::vector< std::string > `opensave_recent_list_of_files`
- std::string `opensave_recent_directory`
- uint8_t `opensave_trim_dst_num_jumps`
- std::string `printing_default_device`
- bool `printing_use_last_device`
- bool `printing_disable_bg`
- bool `grid_show_on_load`
- bool `grid_show_origin`
- bool `grid_color_match_crosshair`
- uint32_t `grid_color`
- bool `grid_load_from_file`
- std::string `grid_type`
- bool `grid_center_on_origin`
- EmbVector `grid_center`
- EmbVector `grid_size`
- EmbVector `grid_spacing`
- EmbReal `grid_size_radius`
- EmbReal `grid_spacing_radius`
- EmbReal `grid_spacing_angle`
- bool `ruler_show_on_load`
- bool `ruler_metric`
- uint32_t `ruler_color`
- uint8_t `ruler_pixel_size`
- bool `qsnap_enabled`
- uint32_t `qsnap_locator_color`
- uint8_t `qsnap_locator_size`
- uint8_t `qsnap_aperture_size`
- bool `qsnap_endpoint`
- bool `qsnap_midpoint`
- bool `qsnap_center`
- bool `qsnap_node`
- bool `qsnap_quadrant`
- bool `qsnap_intersection`
- bool `qsnap_extension`
- bool `qsnap_insertion`
- bool `qsnap_perpendicular`
- bool `qsnap_tangent`
- bool `qsnap_nearest`
- bool `qsnap_apparent`
- bool `qsnap_parallel`
- bool `lwt_show_lwt`
- bool `lwt_real_render`
- bool `shift_held`
- EmbReal `lwt_default_lwt`
- bool `selection_mode_pickfirst`
- bool `selection_mode_pickadd`
- bool `selection_mode_pickdrag`
- uint32_t `selection_coolgrip_color`
- uint32_t `selection_hotgrip_color`
- uint8_t `selection_grip_size`
- uint8_t `selection_pickbox_size`
- char `text_font` [200]
- EmbReal `text_size`

- `EmbReal text_angle`
- `bool text_style_bold`
- `bool text_style_italic`
- `bool text_style_underline`
- `bool text_style_overline`
- `bool text_style_strikeout`
- `Dictionary * texture_list`
- `uint32_t ticks_color`
- `uint32_t shine_color`
- `char to_open [200]`
- `char menu_action [200]`
- `char current_directory [200]`
- `EmbReal zoomInLimit`
- `EmbReal zoomOutLimit`
- `EmbReal ruler_width`
- `EmbReal tick_depth`
- `EmbReal major_tick_seperation`
- `EmbReal needle_speed`
- `EmbReal stitch_time`

17.77.1 Detailed Description

Settings System.

Rather than pollute the global namespace, we collect together all the global settings into a structure that stores them. This also allows us to create a complete copy of the settings for the purpose of restoring them if the user cancels out of the Settings Dialog.

Like all of our structs, it's C99 compliant.

17.77.2 Member Data Documentation

17.77.2.1 assets_dir `char assets_dir[200]`

17.77.2.2 current_directory `char current_directory[200]`

17.77.2.3 debug_mode `int debug_mode`

17.77.2.4 display_bg_color `uint32_t display_bg_color`

17.77.2.5 display_crosshair_color `uint32_t display_crosshair_color`

17.77.2.6 display_crosshair_percent `uint8_t display_crosshair_percent`

17.77.2.7 display_renderhint_aa `bool display_renderhint_aa`

17.77.2.8 display_renderhint_high_aa `bool display_renderhint_high_aa`

17.77.2.9 **display_renderhint_noncosmetic** bool display_renderhint_noncosmetic

17.77.2.10 **display_renderhint_smooth_pix** bool display_renderhint_smooth_pix

17.77.2.11 **display_renderhint_text_aa** bool display_renderhint_text_aa

17.77.2.12 **display_scrollbar_widget_num** int display_scrollbar_widget_num

17.77.2.13 **display_selectbox_alpha** uint8_t display_selectbox_alpha

17.77.2.14 **display_selectbox_left_color** uint32_t display_selectbox_left_color

17.77.2.15 **display_selectbox_left_fill** uint32_t display_selectbox_left_fill

17.77.2.16 **display_selectbox_right_color** uint32_t display_selectbox_right_color

17.77.2.17 **display_selectbox_right_fill** uint32_t display_selectbox_right_fill

17.77.2.18 **display_show_scrollbars** bool display_show_scrollbars

17.77.2.19 **display_units** std::string display_units

17.77.2.20 **display_use_opengl** bool display_use_opengl

17.77.2.21 **display_zoomscale_in** EmbReal display_zoomscale_in

17.77.2.22 **display_zoomscale_out** EmbReal display_zoomscale_out

17.77.2.23 **general_check_for_updates** bool general_check_for_updates

17.77.2.24 **general_current_tip** uint32_t general_current_tip

17.77.2.25 **general_mdi_bg_color** uint32_t general_mdi_bg_color

17.77.2.26 **general_mdi_bg_logo** char general_mdi_bg_logo[200]

17.77.2.27 **general_mdi_bg_texture** char general_mdi_bg_texture[200]

17.77.2.28 **general_system_help_browser** bool general_system_help_browser

17.77.2.29 **grid_center** EmbVector grid_center

17.77.2.30 **grid_center_on_origin** bool grid_center_on_origin

17.77.2.31 **grid_color** uint32_t grid_color

17.77.2.32 **grid_color_match_crosshair** bool grid_color_match_crosshair

17.77.2.33 **grid_load_from_file** bool grid_load_from_file

17.77.2.34 **grid_show_on_load** bool grid_show_on_load

17.77.2.35 **grid_show_origin** bool grid_show_origin

17.77.2.36 **grid_size** EmbVector grid_size

17.77.2.37 **grid_size_radius** EmbReal grid_size_radius

17.77.2.38 **grid_spacing** EmbVector grid_spacing

17.77.2.39 **grid_spacing_angle** EmbReal grid_spacing_angle

17.77.2.40 **grid_spacing_radius** EmbReal grid_spacing_radius

17.77.2.41 **grid_type** std::string grid_type

17.77.2.42 **icon_size** int icon_size

17.77.2.43 **icon_theme** char icon_theme[200]

17.77.2.44 **language** char language[200]

17.77.2.45 lwt_default_lwt `EmbReal lwt_default_lwt`

17.77.2.46 lwt_real_render `bool lwt_real_render`

17.77.2.47 lwt_show_lwt `bool lwt_show_lwt`

17.77.2.48 major_tick_seperation `EmbReal major_tick_seperation`

17.77.2.49 mdi_bg_use_color `bool mdi_bg_use_color`

17.77.2.50 mdi_bg_use_logo `bool mdi_bg_use_logo`

17.77.2.51 mdi_bg_use_texture `bool mdi_bg_use_texture`

17.77.2.52 menu_action `char menu_action[200]`

17.77.2.53 needle_speed `EmbReal needle_speed`

17.77.2.54 opensave_custom_filter `std::string opensave_custom_filter`

17.77.2.55 opensave_open_format `std::string opensave_open_format`

17.77.2.56 opensave_open_thumbnail `bool opensave_open_thumbnail`

17.77.2.57 opensave_recent_directory `std::string opensave_recent_directory`

17.77.2.58 opensave_recent_list_of_files `std::vector<std::string> opensave_recent_list_of_files`

17.77.2.59 opensave_recent_max_files `uint8_t opensave_recent_max_files`

17.77.2.60 opensave_save_format `std::string opensave_save_format`

17.77.2.61 opensave_save_thumbnail `bool opensave_save_thumbnail`

17.77.2.62 opensave_trim_dst_num_jumps `uint8_t opensave_trim_dst_num_jumps`

17.77.2.63 pattern_index int pattern_index

17.77.2.64 printing_default_device std::string printing_default_device

17.77.2.65 printing_disable_bg bool printing_disable_bg

17.77.2.66 printing_use_last_device bool printing_use_last_device

17.77.2.67 qsnap_aperture_size uint8_t qsnap_aperture_size

17.77.2.68 qsnap_apparent bool qsnap_apparent

17.77.2.69 qsnap_center bool qsnap_center

17.77.2.70 qsnap_enabled bool qsnap_enabled

17.77.2.71 qsnap_endpoint bool qsnap_endpoint

17.77.2.72 qsnap_extension bool qsnap_extension

17.77.2.73 qsnap_insertion bool qsnap_insertion

17.77.2.74 qsnap_intersection bool qsnap_intersection

17.77.2.75 qsnap_locator_color uint32_t qsnap_locator_color

17.77.2.76 qsnap_locator_size uint8_t qsnap_locator_size

17.77.2.77 qsnap_midpoint bool qsnap_midpoint

17.77.2.78 qsnap_nearest bool qsnap_nearest

17.77.2.79 qsnap_node bool qsnap_node

17.77.2.80 qsnap_parallel bool qsnap_parallel

17.77.2.81 `qsnap_perpendicular` `bool qsnap_perpendicular`

17.77.2.82 `qsnap_quadrant` `bool qsnap_quadrant`

17.77.2.83 `qsnap_tangent` `bool qsnap_tangent`

17.77.2.84 `ruler_color` `uint32_t ruler_color`

17.77.2.85 `ruler_metric` `bool ruler_metric`

17.77.2.86 `ruler_pixel_size` `uint8_t ruler_pixel_size`

17.77.2.87 `ruler_show_on_load` `bool ruler_show_on_load`

17.77.2.88 `ruler_width` `EmbReal ruler_width`

17.77.2.89 `running` `bool running`

17.77.2.90 `selection_coolgrip_color` `uint32_t selection_coolgrip_color`

17.77.2.91 `selection_grip_size` `uint8_t selection_grip_size`

17.77.2.92 `selection_hotgrip_color` `uint32_t selection_hotgrip_color`

17.77.2.93 `selection_mode_pickadd` `bool selection_mode_pickadd`

17.77.2.94 `selection_mode_pickdrag` `bool selection_mode_pickdrag`

17.77.2.95 `selection_mode_pickfirst` `bool selection_mode_pickfirst`

17.77.2.96 `selection_pickbox_size` `uint8_t selection_pickbox_size`

17.77.2.97 `shift_held` `bool shift_held`

17.77.2.98 `shine_color` `uint32_t shine_color`

17.77.2.99 show_about_dialog bool show_about_dialog

17.77.2.100 show_details_dialog bool show_details_dialog

17.77.2.101 show_editor bool show_editor

17.77.2.102 show_open_file_dialog bool show_open_file_dialog

17.77.2.103 show_settings_editor bool show_settings_editor

17.77.2.104 stitch_time EmbReal stitch_time

17.77.2.105 testing bool testing

17.77.2.106 text_angle EmbReal text_angle

17.77.2.107 text_font char text_font[200]

17.77.2.108 text_size EmbReal text_size

17.77.2.109 text_style_bold bool text_style_bold

17.77.2.110 text_style_italic bool text_style_italic

17.77.2.111 text_style_overline bool text_style_overline

17.77.2.112 text_style_strikeout bool text_style_strikeout

17.77.2.113 text_style_underline bool text_style_underline

17.77.2.114 texture_list Dictionary* texture_list

17.77.2.115 tick_depth EmbReal tick_depth

17.77.2.116 ticks_color uint32_t ticks_color

17.77.2.117 tip_of_the_day bool tip_of_the_day

17.77.2.118 to_open char to_open[200]

17.77.2.119 use_translation bool use_translation

17.77.2.120 version char version[200]

17.77.2.121 zoomInLimit EmbReal zoomInLimit

17.77.2.122 zoomOutLimit EmbReal zoomOutLimit

The documentation for this struct was generated from the following file:

- embroidermodder2/embroidermodder.h

17.78 Settings_Dialog Class Reference

#include <embroidermodder.h>

Signals

- void buttonCustomFilterSelectAll (bool)
- void buttonCustomFilterClearAll (bool)
- void buttonQSnapSelectAll (bool)
- void buttonQSnapClearAll (bool)

Public Member Functions

- [Settings_Dialog \(MainWindow *mw, const QString &showTab=QString\(\), QWidget *parent=0\)](#)
- [~Settings_Dialog \(\)](#)
- QWidget * [createTabGeneral \(\)](#)
- QWidget * [createTabFilePaths \(\)](#)
- QWidget * [createTabDisplay \(\)](#)
- QWidget * [createTabPrompt \(\)](#)
- QWidget * [createTabOpenSave \(\)](#)
- QWidget * [createTabPrinting \(\)](#)
- QWidget * [createTabSnap \(\)](#)
- QWidget * [createTabGridRuler \(\)](#)
- QWidget * [createTabOrthoPolar \(\)](#)
- QWidget * [createTabQuickSnap \(\)](#)
- QWidget * [createTabQuickTrack \(\)](#)
- QWidget * [createTabLineWeight \(\)](#)
- QWidget * [createTabSelection \(\)](#)
- void [addColorsToComboBox \(QComboBox *comboBox\)](#)

Public Attributes

- `MainWindow * mainWin`
- `QTabWidget * tabWidget`
- `QDialogButtonBox * buttonBox`
- `bool preview_general_mdi_bg_use_logo`
- `bool preview_general_mdi_bg_use_texture`
- `bool preview_general_mdi_bg_use_color`
- `QString accept_general_mdi_bg_logo`
- `QString accept_general_mdi_bg_texture`
- `QRgb preview_general_mdi_bg_color`
- `QRgb accept_general_mdi_bg_color`
- `bool preview_display_show_scrollbars`
- `QRgb preview_display_crosshair_color`
- `QRgb accept_display_crosshair_color`
- `QRgb preview_display_bg_color`
- `QRgb accept_display_bg_color`
- `QRgb preview_display_selectbox_left_color`
- `QRgb accept_display_selectbox_left_color`
- `QRgb preview_display_selectbox_left_fill`
- `QRgb accept_display_selectbox_left_fill`
- `QRgb preview_display_selectbox_right_color`
- `QRgb accept_display_selectbox_right_color`
- `QRgb preview_display_selectbox_right_fill`
- `QRgb accept_display_selectbox_right_fill`
- `quint8 preview_display_selectbox_alpha`
- `QRgb preview_prompt_text_color`
- `QRgb accept_prompt_text_color`
- `QRgb preview_prompt_bg_color`
- `QRgb accept_prompt_bg_color`
- `QString preview_prompt_font_family`
- `QString preview_prompt_font_style`
- `quint8 preview_prompt_font_size`
- `QRgb preview_grid_color`
- `QRgb accept_grid_color`
- `QRgb preview_ruler_color`
- `QRgb accept_ruler_color`
- `bool preview_lwt_show_lwt`
- `bool preview_lwt_real_render`
- `QString dialog_general_language`
- `QString dialog_general_icon_theme`
- `int dialog_general_icon_size`
- `bool dialog_general_mdi_bg_use_logo`
- `bool dialog_general_mdi_bg_use_texture`
- `bool dialog_general_mdi_bg_use_color`
- `QString dialog_general_mdi_bg_logo`
- `QString dialog_general_mdi_bg_texture`
- `QRgb dialog_general_mdi_bg_color`
- `bool dialog_general_tip_of_the_day`
- `bool dialog_general_system_help_browser`
- `bool dialog_display_use_opengl`
- `bool dialog_display_renderhint_aa`
- `bool dialog_display_renderhint_text_aa`
- `bool dialog_display_renderhint_smooth_pix`
- `bool dialog_display_renderhint_high_aa`

- bool `dialog_display_renderhint_noncosmetic`
- bool `dialog_display_show_scrollbars`
- int `dialog_display_scrollbar_widget_num`
- QRgb `dialog_display_crosshair_color`
- QRgb `dialog_display_bg_color`
- QRgb `dialog_display_selectbox_left_color`
- QRgb `dialog_display_selectbox_left_fill`
- QRgb `dialog_display_selectbox_right_color`
- QRgb `dialog_display_selectbox_right_fill`
- quint8 `dialog_display_selectbox_alpha`
- EmbReal `dialog_display_zoomscale_in`
- EmbReal `dialog_display_zoomscale_out`
- quint8 `dialog_display_crosshair_percent`
- QString `dialog_display_units`
- QRgb `dialog_prompt_text_color`
- QRgb `dialog_prompt_bg_color`
- QString `dialog_prompt_font_family`
- QString `dialog_prompt_font_style`
- quint8 `dialog_prompt_font_size`
- bool `dialog_prompt_save_history`
- bool `dialog_prompt_save_history_as_html`
- QString `dialog_prompt_save_history_filename`
- QString `dialog_opensave_custom_filter`
- QString `dialog_opensave_open_format`
- bool `dialog_opensave_open_thumbnail`
- QString `dialog_opensave_save_format`
- bool `dialog_opensave_save_thumbnail`
- quint8 `dialog_opensave_recent_max_files`
- quint8 `dialog_opensave_trim_dst_num_jumps`
- QString `dialog_printing_default_device`
- bool `dialog_printing_use_last_device`
- bool `dialog_printing_disable_bg`
- bool `dialog_grid_show_on_load`
- bool `dialog_grid_show_origin`
- bool `dialog_grid_color_match_crosshair`
- QRgb `dialog_grid_color`
- bool `dialog_grid_load_from_file`
- QString `dialog_grid_type`
- bool `dialog_grid_center_on_origin`
- EmbReal `dialog_grid_center_x`
- EmbReal `dialog_grid_center_y`
- EmbReal `dialog_grid_size_x`
- EmbReal `dialog_grid_size_y`
- EmbReal `dialog_grid_spacing_x`
- EmbReal `dialog_grid_spacing_y`
- EmbReal `dialog_grid_size_radius`
- EmbReal `dialog_grid_spacing_radius`
- EmbReal `dialog_grid_spacing_angle`
- bool `dialog_ruler_show_on_load`
- bool `dialog_ruler_metric`
- QRgb `dialog_ruler_color`
- quint8 `dialog_ruler_pixel_size`
- bool `dialog_qsnap_enabled`
- QRgb `dialog_qsnap_locator_color`
- quint8 `dialog_qsnap_locator_size`

- quint8 `dialog_qsnap_aperture_size`
- bool `dialog_qsnap_endpoint`
- bool `dialog_qsnap_midpoint`
- bool `dialog_qsnap_center`
- bool `dialog_qsnap_node`
- bool `dialog_qsnap_quadrant`
- bool `dialog_qsnap_intersection`
- bool `dialog_qsnap_extension`
- bool `dialog_qsnap_insertion`
- bool `dialog_qsnap_perpendicular`
- bool `dialog_qsnap_tangent`
- bool `dialog_qsnap_nearest`
- bool `dialog_qsnap_apparent`
- bool `dialog_qsnap_parallel`
- bool `dialog_lwt_show_lwt`
- bool `dialog_lwt_real_render`
- EmbReal `dialog_lwt_default_lwt`
- bool `dialog_selection_mode_pickfirst`
- bool `dialog_selection_mode_pickadd`
- bool `dialog_selection_mode_pickdrag`
- QRgb `dialog_selection_coolgrip_color`
- QRgb `dialog_selection_hotgrip_color`
- quint8 `dialog_selection_grip_size`
- quint8 `dialog_selection_pickbox_size`

Private Slots

- void `comboBoxLanguageCurrentIndexChanged` (const QString &)
- void `comboBoxIconThemeCurrentIndexChanged` (const QString &)
- void `comboBoxIconSizeCurrentIndexChanged` (int)
- void `checkBoxGeneralMdiBGUseLogoStateChanged` (int)
- void `chooseGeneralMdiBackgroundLogo` ()
- void `checkBoxGeneralMdiBGUseTextureStateChanged` (int)
- void `chooseGeneralMdiBackgroundTexture` ()
- void `checkBoxGeneralMdiBGUseColorStateChanged` (int)
- void `chooseGeneralMdiBackgroundColor` ()
- void `currentGeneralMdiBackgroundColorChanged` (const QColor &)
- void `checkBoxTipOfDayStateChanged` (int)
- void `checkBoxUseOpenGLStateChanged` (int)
- void `checkBoxRenderHintAAStateChanged` (int)
- void `checkBoxRenderHintTextAAStateChanged` (int)
- void `checkBoxRenderHintSmoothPixStateChanged` (int)
- void `checkBoxRenderHintHighAAStateChanged` (int)
- void `checkBoxRenderHintNonCosmeticStateChanged` (int)
- void `checkBoxShowScrollBarsStateChanged` (int)
- void `comboBoxScrollBarWidgetCurrentIndexChanged` (int)
- void `spinBoxZoomScaleInValueChanged` (double)
- void `spinBoxZoomScaleOutValueChanged` (double)
- void `checkBoxDisableBGStateChanged` (int)
- void `chooseDisplayCrossHairColor` ()
- void `currentDisplayCrossHairColorChanged` (const QColor &)
- void `chooseDisplayBackgroundColor` ()
- void `currentDisplayBackgroundColorChanged` (const QColor &)
- void `chooseDisplaySelectBoxLeftColor` ()
- void `currentDisplaySelectBoxLeftColorChanged` (const QColor &)

- void `chooseDisplaySelectBoxLeftFill ()`
- void `currentDisplaySelectBoxLeftFillChanged (const QColor &)`
- void `chooseDisplaySelectBoxRightColor ()`
- void `currentDisplaySelectBoxRightColorChanged (const QColor &)`
- void `chooseDisplaySelectBoxRightFill ()`
- void `currentDisplaySelectBoxRightFillChanged (const QColor &)`
- void `spinBoxDisplaySelectBoxAlphaValueChanged (int)`
- void `choosePromptTextColor ()`
- void `currentPromptTextColorChanged (const QColor &)`
- void `choosePromptBackgroundColor ()`
- void `currentPromptBackgroundColorChanged (const QColor &)`
- void `comboBoxPromptFontFamilyCurrentIndexChanged (const QString &)`
- void `comboBoxPromptFontStyleCurrentIndexChanged (const QString &)`
- void `spinBoxPromptFontSizeValueChanged (int)`
- void `checkBoxPromptSaveHistoryStateChanged (int)`
- void `checkBoxPromptSaveHistoryAsHtmlStateChanged (int)`
- void `checkBoxCustomFilterStateChanged (int)`
- void `buttonCustomFilterSelectAllClicked ()`
- void `buttonCustomFilterClearAllClicked ()`
- void `spinBoxRecentMaxFilesValueChanged (int)`
- void `spinBoxTrimDstNumJumpsValueChanged (int)`
- void `checkBoxGridShowOnLoadStateChanged (int)`
- void `checkBoxGridShowOriginStateChanged (int)`
- void `checkBoxGridColorMatchCrossHairStateChanged (int)`
- void `chooseGridColor ()`
- void `currentGridColorChanged (const QColor &)`
- void `checkBoxGridLoadFromFileStateChanged (int)`
- void `comboBoxGridTypeCurrentIndexChanged (const QString &)`
- void `checkBoxGridCenterOnOriginStateChanged (int)`
- void `spinBoxGridCenterXValueChanged (double)`
- void `spinBoxGridCenterYValueChanged (double)`
- void `spinBoxGridSizeXValueChanged (double)`
- void `spinBoxGridSizeYValueChanged (double)`
- void `spinBoxGridSpacingXValueChanged (double)`
- void `spinBoxGridSpacingYValueChanged (double)`
- void `spinBoxGridSizeRadiusValueChanged (double)`
- void `spinBoxGridSpacingRadiusValueChanged (double)`
- void `spinBoxGridSpacingAngleValueChanged (double)`
- void `checkBoxRulerShowOnLoadStateChanged (int)`
- void `comboBoxRulerMetricCurrentIndexChanged (int)`
- void `chooseRulerColor ()`
- void `currentRulerColorChanged (const QColor &)`
- void `spinBoxRulerPixelSizeValueChanged (double)`
- void `checkBoxQSnapEndPointStateChanged (int)`
- void `checkBoxQSnapMidPointStateChanged (int)`
- void `checkBoxQSnapCenterStateChanged (int)`
- void `checkBoxQSnapNodeStateChanged (int)`
- void `checkBoxQSnapQuadrantStateChanged (int)`
- void `checkBoxQSnapIntersectionStateChanged (int)`
- void `checkBoxQSnapExtensionStateChanged (int)`
- void `checkBoxQSnapInsertionStateChanged (int)`
- void `checkBoxQSnapPerpendicularStateChanged (int)`
- void `checkBoxQSnapTangentStateChanged (int)`
- void `checkBoxQSnapNearestStateChanged (int)`
- void `checkBoxQSnapApparentStateChanged (int)`

- void `checkBoxQSnapParallelStateChanged` (int)
- void `buttonQSnapSelectAllClicked` ()
- void `buttonQSnapClearAllClicked` ()
- void `comboBoxQSnapLocatorColorCurrentIndexChanged` (int)
- void `sliderQSnapLocatorSizeValueChanged` (int)
- void `sliderQSnapApertureSizeValueChanged` (int)
- void `checkBoxLwtShowLwtStateChanged` (int)
- void `checkBoxLwtRealRenderStateChanged` (int)
- void `checkBoxSelectionModePickFirstStateChanged` (int)
- void `checkBoxSelectionModePickAddStateChanged` (int)
- void `checkBoxSelectionModePickDragStateChanged` (int)
- void `comboBoxSelectionCoolGripColorCurrentIndexChanged` (int)
- void `comboBoxSelectionHotGripColorCurrentIndexChanged` (int)
- void `sliderSelectionGripSizeValueChanged` (int)
- void `sliderSelectionPickBoxSizeValueChanged` (int)
- void `acceptChanges` ()
- void `rejectChanges` ()

17.78.1 Constructor & Destructor Documentation

17.78.1.1 `Settings_Dialog()` `Settings_Dialog` (

```
    MainWindow * mw,
    const QString & showTab = QString(),
    QWidget * parent = 0 )
```

Embroidermodder 2.

Copyright 2013-2022 The Embroidermodder Team Embroidermodder 2 is Open Source Software. See LICENSE for licensing terms.

Use Python's PEP7 style guide. <https://peps.python.org/pep-0007/>

17.78.1.2 `~Settings_Dialog()` `~Settings_Dialog` ()

17.78.2 Member Function Documentation

17.78.2.1 `acceptChanges` void `acceptChanges` () [private], [slot]

17.78.2.2 `addColorsToComboBox()` void `addColorsToComboBox` (

```
    QComboBox * comboBox )
```

17.78.2.3 `buttonCustomFilterClearAll` void `buttonCustomFilterClearAll` (

```
    bool ) [signal]
```

17.78.2.4 `buttonCustomFilterClearAllClicked` void `buttonCustomFilterClearAllClicked` () [private], [slot]

17.78.2.5 `buttonCustomFilterSelectAll` void `buttonCustomFilterSelectAll` (

```
    bool ) [signal]
```

17.78.2.6 buttonCustomFilterSelectAllClicked void buttonCustomFilterSelectAllClicked () [private], [slot]

17.78.2.7 buttonQSnapClearAll void buttonQSnapClearAll (bool) [signal]

17.78.2.8 buttonQSnapClearAllClicked void buttonQSnapClearAllClicked () [private], [slot]

17.78.2.9 buttonQSnapSelectAll void buttonQSnapSelectAll (bool) [signal]

17.78.2.10 buttonQSnapSelectAllClicked void buttonQSnapSelectAllClicked () [private], [slot]

17.78.2.11 checkBoxCustomFilterStateChanged void checkBoxCustomFilterStateChanged (int checked) [private], [slot]

17.78.2.12 checkBoxDisableBGStateChanged void checkBoxDisableBGStateChanged (int checked) [private], [slot]

17.78.2.13 checkBoxGeneralMdiBGUseColorStateChanged void checkBoxGeneralMdiBGUseColorStateChanged (int checked) [private], [slot]

17.78.2.14 checkBoxGeneralMdiBGUseLogoStateChanged void checkBoxGeneralMdiBGUseLogoStateChanged (int checked) [private], [slot]

17.78.2.15 checkBoxGeneralMdiBGUseTextureStateChanged void checkBoxGeneralMdiBGUseTextureStateChanged (int checked) [private], [slot]

17.78.2.16 checkBoxGridCenterOnOriginStateChanged void checkBoxGridCenterOnOriginStateChanged (int checked) [private], [slot]

17.78.2.17 checkBoxGridColorMatchCrossHairStateChanged void checkBoxGridColorMatchCrossHairStateChanged (int checked) [private], [slot]

17.78.2.18 checkBoxGridLoadFromFileStateChanged void checkBoxGridLoadFromFileStateChanged (int checked) [private], [slot]

17.78.2.19 checkBoxGridShowOnLoadStateChanged void checkBoxGridShowOnLoadStateChanged (int *checked*) [private], [slot]

17.78.2.20 checkBoxGridShowOriginStateChanged void checkBoxGridShowOriginStateChanged (int *checked*) [private], [slot]

17.78.2.21 checkBoxLwtRealRenderStateChanged void checkBoxLwtRealRenderStateChanged (int *checked*) [private], [slot]

17.78.2.22 checkBoxLwtShowLwtStateChanged void checkBoxLwtShowLwtStateChanged (int *checked*) [private], [slot]

17.78.2.23 checkBoxPromptSaveHistoryAsHtmlStateChanged void checkBoxPromptSaveHistoryAsHtmlStateChanged (int *checked*) [private], [slot]

17.78.2.24 checkBoxPromptSaveHistoryStateChanged void checkBoxPromptSaveHistoryStateChanged (int *checked*) [private], [slot]

17.78.2.25 checkBoxQSnapApparentStateChanged void checkBoxQSnapApparentStateChanged (int *checked*) [private], [slot]

17.78.2.26 checkBoxQSnapCenterStateChanged void checkBoxQSnapCenterStateChanged (int *checked*) [private], [slot]

17.78.2.27 checkBoxQSnapEndPointStateChanged void checkBoxQSnapEndPointStateChanged (int *checked*) [private], [slot]

17.78.2.28 checkBoxQSnapExtensionStateChanged void checkBoxQSnapExtensionStateChanged (int *checked*) [private], [slot]

17.78.2.29 checkBoxQSnapInsertionStateChanged void checkBoxQSnapInsertionStateChanged (int *checked*) [private], [slot]

17.78.2.30 checkBoxQSnapIntersectionStateChanged void checkBoxQSnapIntersectionStateChanged (int *checked*) [private], [slot]

17.78.2.31 checkBoxQSnapMidPointStateChanged void checkBoxQSnapMidPointStateChanged (int *checked*) [private], [slot]

17.78.2.32 `checkBoxQSnapNearestStateChanged` void checkBoxQSnapNearestStateChanged (int checked) [private], [slot]

17.78.2.33 `checkBoxQSnapNodeStateChanged` void checkBoxQSnapNodeStateChanged (int checked) [private], [slot]

17.78.2.34 `checkBoxQSnapParallelStateChanged` void checkBoxQSnapParallelStateChanged (int checked) [private], [slot]

17.78.2.35 `checkBoxQSnapPerpendicularStateChanged` void checkBoxQSnapPerpendicularState Changed (int checked) [private], [slot]

17.78.2.36 `checkBoxQSnapQuadrantStateChanged` void checkBoxQSnapQuadrantStateChanged (int checked) [private], [slot]

17.78.2.37 `checkBoxQSnapTangentStateChanged` void checkBoxQSnapTangentStateChanged (int checked) [private], [slot]

17.78.2.38 `checkBoxRenderHintAAStateChanged` void checkBoxRenderHintAAStateChanged (int checked) [private], [slot]

17.78.2.39 `checkBoxRenderHintHighAAStateChanged` void checkBoxRenderHintHighAAStateChanged (int checked) [private], [slot]

17.78.2.40 `checkBoxRenderHintNonCosmeticStateChanged` void checkBoxRenderHintNonCosmetic StateChanged (int checked) [private], [slot]

17.78.2.41 `checkBoxRenderHintSmoothPixStateChanged` void checkBoxRenderHintSmoothPixState Changed (int checked) [private], [slot]

17.78.2.42 `checkBoxRenderHintTextAAStateChanged` void checkBoxRenderHintTextAAStateChanged (int checked) [private], [slot]

17.78.2.43 `checkBoxRulerShowOnLoadStateChanged` void checkBoxRulerShowOnLoadStateChanged (int checked) [private], [slot]

17.78.2.44 `checkBoxSelectionModePickAddStateChanged` void checkBoxSelectionModePickAddState Changed (int checked) [private], [slot]

17.78.2.45 checkBoxSelectionModePickDragStateChanged void checkBoxSelectionModePickDragStateChanged (int checked) [private], [slot]

17.78.2.46 checkBoxSelectionModePickFirstStateChanged void checkBoxSelectionModePickFirstStateChanged (int checked) [private], [slot]

17.78.2.47 checkBoxShowScrollBarsStateChanged void checkBoxShowScrollBarsStateChanged (int checked) [private], [slot]

17.78.2.48 checkBoxTipOfTheDayStateChanged void checkBoxTipOfTheDayStateChanged (int checked) [private], [slot]

17.78.2.49 checkBoxUseOpenGLStateChanged void checkBoxUseOpenGLStateChanged (int checked) [private], [slot]

17.78.2.50 chooseDisplayBackgroundColor void chooseDisplayBackgroundColor () [private], [slot]

17.78.2.51 chooseDisplayCrossHairColor void chooseDisplayCrossHairColor () [private], [slot]

17.78.2.52 chooseDisplaySelectBoxLeftColor void chooseDisplaySelectBoxLeftColor () [private], [slot]

17.78.2.53 chooseDisplaySelectBoxLeftFill void chooseDisplaySelectBoxLeftFill () [private], [slot]

17.78.2.54 chooseDisplaySelectBoxRightColor void chooseDisplaySelectBoxRightColor () [private], [slot]

17.78.2.55 chooseDisplaySelectBoxRightFill void chooseDisplaySelectBoxRightFill () [private], [slot]

17.78.2.56 chooseGeneralMdiBackgroundColor void chooseGeneralMdiBackgroundColor () [private], [slot]

17.78.2.57 chooseGeneralMdiBackgroundLogo void chooseGeneralMdiBackgroundLogo () [private], [slot]

```
17.78.2.58 chooseGeneralMdiBackgroundTexture void chooseGeneralMdiBackgroundTexture ( ) [private],  
[slot]  
  
17.78.2.59 chooseGridColor void chooseGridColor ( ) [private], [slot]  
  
17.78.2.60 choosePromptBackgroundColor void choosePromptBackgroundColor ( ) [private],  
[slot]  
  
17.78.2.61 choosePromptTextColor void choosePromptTextColor ( ) [private], [slot]  
  
17.78.2.62 chooseRulerColor void chooseRulerColor ( ) [private], [slot]  
  
17.78.2.63 comboBoxGridTypeCurrentIndexChanged void comboBoxGridTypeCurrentIndexChanged (const  
QString & type) [private], [slot]  
  
17.78.2.64 comboBoxIconSizeCurrentIndexChanged void comboBoxIconSizeCurrentIndexChanged (int  
index) [private], [slot]  
  
17.78.2.65 comboBoxIconThemeCurrentIndexChanged void comboBoxIconThemeCurrentIndexChanged (const  
QString & theme) [private], [slot]  
  
17.78.2.66 comboBoxLanguageCurrentIndexChanged void comboBoxLanguageCurrentIndexChanged (const  
QString & lang) [private], [slot]  
  
17.78.2.67 comboBoxPromptFontFamilyCurrentIndexChanged void comboBoxPromptFontFamilyCurrentIndexChanged (const  
QString & family) [private], [slot]  
  
17.78.2.68 comboBoxPromptFontStyleCurrentIndexChanged void comboBoxPromptFontStyleCurrentIndexChanged (const  
QString & style) [private], [slot]  
  
17.78.2.69 comboBoxQSnapLocatorColorCurrentIndexChanged void comboBoxQSnapLocatorColorCurrentIndexChanged (int  
index) [private], [slot]  
  
17.78.2.70 comboBoxRulerMetricCurrentIndexChanged void comboBoxRulerMetricCurrentIndexChanged (int  
index) [private], [slot]
```

- 17.78.2.71 comboBoxScrollBarWidgetCurrentIndexChanged** void comboBoxScrollBarWidgetCurrentIndexChanged (int index) [private], [slot]
- 17.78.2.72 comboBoxSelectionCoolGripColorCurrentIndexChanged** void comboBoxSelectionCoolGripColorCurrentIndexChanged (int index) [private], [slot]
- 17.78.2.73 comboBoxSelectionHotGripColorCurrentIndexChanged** void comboBoxSelectionHotGripColorCurrentIndexChanged (int index) [private], [slot]
- 17.78.2.74 createTabDisplay()** QWidget * createTabDisplay ()
- 17.78.2.75 createTabFilesPaths()** QWidget * createTabFilesPaths ()
- 17.78.2.76 createTabGeneral()** QWidget * createTabGeneral ()
- 17.78.2.77 createTabGridRuler()** QWidget * createTabGridRuler ()
- 17.78.2.78 createTabLineWeight()** QWidget * createTabLineWeight ()
- 17.78.2.79 createTabOpenSave()** QWidget * createTabOpenSave ()
- 17.78.2.80 createTabOrthoPolar()** QWidget * createTabOrthoPolar ()
- 17.78.2.81 createTabPrinting()** QWidget * createTabPrinting ()
- 17.78.2.82 createTabPrompt()** QWidget * createTabPrompt ()
- 17.78.2.83 createTabQuickSnap()** QWidget * createTabQuickSnap ()
- 17.78.2.84 createTabQuickTrack()** QWidget * createTabQuickTrack ()
- 17.78.2.85 createTabSelection()** QWidget * createTabSelection ()
- 17.78.2.86 createTabSnap()** QWidget * createTabSnap ()

17.78.2.87 currentDisplayBackgroundColorChanged void currentDisplayBackgroundColorChanged (const QColor & color) [private], [slot]

17.78.2.88 currentDisplayCrossHairColorChanged void currentDisplayCrossHairColorChanged (const QColor & color) [private], [slot]

17.78.2.89 currentDisplaySelectBoxLeftColorChanged void currentDisplaySelectBoxLeftColorChanged (const QColor & color) [private], [slot]

17.78.2.90 currentDisplaySelectBoxLeftFillChanged void currentDisplaySelectBoxLeftFillChanged (const QColor & color) [private], [slot]

17.78.2.91 currentDisplaySelectBoxRightColorChanged void currentDisplaySelectBoxRightColorChanged (const QColor & color) [private], [slot]

17.78.2.92 currentDisplaySelectBoxRightFillChanged void currentDisplaySelectBoxRightFillChanged (const QColor & color) [private], [slot]

17.78.2.93 currentGeneralMdiBackgroundColorChanged void currentGeneralMdiBackgroundColorChanged (const QColor & color) [private], [slot]

17.78.2.94 currentGridColorChanged void currentGridColorChanged (const QColor & color) [private], [slot]

17.78.2.95 currentPromptBackgroundColorChanged void currentPromptBackgroundColorChanged (const QColor & color) [private], [slot]

17.78.2.96 currentPromptTextColorChanged void currentPromptTextColorChanged (const QColor & color) [private], [slot]

17.78.2.97 currentRulerColorChanged void currentRulerColorChanged (const QColor & color) [private], [slot]

17.78.2.98 rejectChanges void rejectChanges () [private], [slot]

17.78.2.99 sliderQSnapApertureSizeValueChanged void sliderQSnapApertureSizeValueChanged (int value) [private], [slot]

- 17.78.2.100 **sliderQSnapLocatorSizeValueChanged** void sliderQSnapLocatorSizeValueChanged (int value) [private], [slot]
- 17.78.2.101 **sliderSelectionGripSizeValueChanged** void sliderSelectionGripSizeValueChanged (int value) [private], [slot]
- 17.78.2.102 **sliderSelectionPickBoxSizeValueChanged** void sliderSelectionPickBoxSizeValueChanged (int value) [private], [slot]
- 17.78.2.103 **spinBoxDisplaySelectBoxAlphaValueChanged** void spinBoxDisplaySelectBoxAlphaValueChanged (int value) [private], [slot]
- 17.78.2.104 **spinBoxGridCenterXValueChanged** void spinBoxGridCenterXValueChanged (double value) [private], [slot]
- 17.78.2.105 **spinBoxGridCenterYValueChanged** void spinBoxGridCenterYValueChanged (double value) [private], [slot]
- 17.78.2.106 **spinBoxGridSizeRadiusValueChanged** void spinBoxGridSizeRadiusValueChanged (double value) [private], [slot]
- 17.78.2.107 **spinBoxGridSizeXValueChanged** void spinBoxGridSizeXValueChanged (double value) [private], [slot]
- 17.78.2.108 **spinBoxGridSizeYValueChanged** void spinBoxGridSizeYValueChanged (double value) [private], [slot]
- 17.78.2.109 **spinBoxGridSpacingAngleValueChanged** void spinBoxGridSpacingAngleValueChanged (double value) [private], [slot]
- 17.78.2.110 **spinBoxGridSpacingRadiusValueChanged** void spinBoxGridSpacingRadiusValueChanged (double value) [private], [slot]
- 17.78.2.111 **spinBoxGridSpacingXValueChanged** void spinBoxGridSpacingXValueChanged (double value) [private], [slot]
- 17.78.2.112 **spinBoxGridSpacingYValueChanged** void spinBoxGridSpacingYValueChanged (double value) [private], [slot]

17.78.2.113 `spinBoxPromptFontSizeValueChanged` void spinBoxPromptFontSizeValueChanged (int value) [private], [slot]

17.78.2.114 `spinBoxRecentMaxFilesValueChanged` void spinBoxRecentMaxFilesValueChanged (int value) [private], [slot]

17.78.2.115 `spinBoxRulerPixelSizeValueChanged` void spinBoxRulerPixelSizeValueChanged (double value) [private], [slot]

17.78.2.116 `spinBoxTrimDstNumJumpsValueChanged` void spinBoxTrimDstNumJumpsValueChanged (int value) [private], [slot]

17.78.2.117 `spinBoxZoomScaleInValueChanged` void spinBoxZoomScaleInValueChanged (double value) [private], [slot]

17.78.2.118 `spinBoxZoomScaleOutValueChanged` void spinBoxZoomScaleOutValueChanged (double value) [private], [slot]

17.78.3 Member Data Documentation

17.78.3.1 `accept_display_bg_color` QRgb accept_display_bg_color

17.78.3.2 `accept_display_crosshair_color` QRgb accept_display_crosshair_color

17.78.3.3 `accept_display_selectbox_left_color` QRgb accept_display_selectbox_left_color

17.78.3.4 `accept_display_selectbox_left_fill` QRgb accept_display_selectbox_left_fill

17.78.3.5 `accept_display_selectbox_right_color` QRgb accept_display_selectbox_right_color

17.78.3.6 `accept_display_selectbox_right_fill` QRgb accept_display_selectbox_right_fill

17.78.3.7 `accept_general_mdi_bg_color` QRgb accept_general_mdi_bg_color

17.78.3.8 `accept_general_mdi_bg_logo` QString accept_general_mdi_bg_logo

17.78.3.9 `accept_general_mdi_bg_texture` QString accept_general_mdi_bg_texture

17.78.3.10 accept_grid_color QRgb accept_grid_color

17.78.3.11 accept_prompt_bg_color QRgb accept_prompt_bg_color

17.78.3.12 accept_prompt_text_color QRgb accept_prompt_text_color

17.78.3.13 accept_ruler_color QRgb accept_ruler_color

17.78.3.14 buttonBox QDialogButtonBox* buttonBox

17.78.3.15 dialog_display_bg_color QRgb dialog_display_bg_color

17.78.3.16 dialog_display_crosshair_color QRgb dialog_display_crosshair_color

17.78.3.17 dialog_display_crosshair_percent quint8 dialog_display_crosshair_percent

17.78.3.18 dialog_display_renderhint_aa bool dialog_display_renderhint_aa

17.78.3.19 dialog_display_renderhint_high_aa bool dialog_display_renderhint_high_aa

17.78.3.20 dialog_display_renderhint_noncosmetic bool dialog_display_renderhint_noncosmetic

17.78.3.21 dialog_display_renderhint_smooth_pix bool dialog_display_renderhint_smooth_pix

17.78.3.22 dialog_display_renderhint_text_aa bool dialog_display_renderhint_text_aa

17.78.3.23 dialog_display_scrollbar_widget_num int dialog_display_scrollbar_widget_num

17.78.3.24 dialog_display_selectbox_alpha quint8 dialog_display_selectbox_alpha

17.78.3.25 dialog_display_selectbox_left_color QRgb dialog_display_selectbox_left_color

17.78.3.26 dialog_display_selectbox_left_fill QRgb dialog_display_selectbox_left_fill

17.78.3.27 dialog_display_selectbox_right_color QRgb dialog_display_selectbox_right_color

17.78.3.28 **dialog_display_selectbox_right_fill** QRgb dialog_display_selectbox_right_fill

17.78.3.29 **dialog_display_show_scrollbars** bool dialog_display_show_scrollbars

17.78.3.30 **dialog_display_units** QString dialog_display_units

17.78.3.31 **dialog_display_use_opengl** bool dialog_display_use_opengl

17.78.3.32 **dialog_display_zoomscale_in** EmbReal dialog_display_zoomscale_in

17.78.3.33 **dialog_display_zoomscale_out** EmbReal dialog_display_zoomscale_out

17.78.3.34 **dialog_general_icon_size** int dialog_general_icon_size

17.78.3.35 **dialog_general_icon_theme** QString dialog_general_icon_theme

17.78.3.36 **dialog_general_language** QString dialog_general_language

17.78.3.37 **dialog_general_mdi_bg_color** QRgb dialog_general_mdi_bg_color

17.78.3.38 **dialog_general_mdi_bg_logo** QString dialog_general_mdi_bg_logo

17.78.3.39 **dialog_general_mdi_bg_texture** QString dialog_general_mdi_bg_texture

17.78.3.40 **dialog_general_mdi_bg_use_color** bool dialog_general_mdi_bg_use_color

17.78.3.41 **dialog_general_mdi_bg_use_logo** bool dialog_general_mdi_bg_use_logo

17.78.3.42 **dialog_general_mdi_bg_use_texture** bool dialog_general_mdi_bg_use_texture

17.78.3.43 **dialog_general_system_help_browser** bool dialog_general_system_help_browser

17.78.3.44 **dialog_general_tip_of_the_day** bool dialog_general_tip_of_the_day

17.78.3.45 **dialog_grid_center_on_origin** bool dialog_grid_center_on_origin

17.78.3.46 dialog_grid_center_x `EmbReal` `dialog_grid_center_x`

17.78.3.47 dialog_grid_center_y `EmbReal` `dialog_grid_center_y`

17.78.3.48 dialog_grid_color `QRgb` `dialog_grid_color`

17.78.3.49 dialog_grid_color_match_crosshair `bool` `dialog_grid_color_match_crosshair`

17.78.3.50 dialog_grid_load_from_file `bool` `dialog_grid_load_from_file`

17.78.3.51 dialog_grid_show_on_load `bool` `dialog_grid_show_on_load`

17.78.3.52 dialog_grid_show_origin `bool` `dialog_grid_show_origin`

17.78.3.53 dialog_grid_size_radius `EmbReal` `dialog_grid_size_radius`

17.78.3.54 dialog_grid_size_x `EmbReal` `dialog_grid_size_x`

17.78.3.55 dialog_grid_size_y `EmbReal` `dialog_grid_size_y`

17.78.3.56 dialog_grid_spacing_angle `EmbReal` `dialog_grid_spacing_angle`

17.78.3.57 dialog_grid_spacing_radius `EmbReal` `dialog_grid_spacing_radius`

17.78.3.58 dialog_grid_spacing_x `EmbReal` `dialog_grid_spacing_x`

17.78.3.59 dialog_grid_spacing_y `EmbReal` `dialog_grid_spacing_y`

17.78.3.60 dialog_grid_type `QString` `dialog_grid_type`

17.78.3.61 dialog_lwt_default_lwt `EmbReal` `dialog_lwt_default_lwt`

17.78.3.62 dialog_lwt_real_render `bool` `dialog_lwt_real_render`

17.78.3.63 dialog_lwt_show_lwt `bool` `dialog_lwt_show_lwt`

17.78.3.64 **dialog_opensave_custom_filter** `QString dialog_opensave_custom_filter`

17.78.3.65 **dialog_opensave_open_format** `QString dialog_opensave_open_format`

17.78.3.66 **dialog_opensave_open_thumbnail** `bool dialog_opensave_open_thumbnail`

17.78.3.67 **dialog_opensave_recent_max_files** `quint8 dialog_opensave_recent_max_files`

17.78.3.68 **dialog_opensave_save_format** `QString dialog_opensave_save_format`

17.78.3.69 **dialog_opensave_save_thumbnail** `bool dialog_opensave_save_thumbnail`

17.78.3.70 **dialog_opensave_trim_dst_num_jumps** `quint8 dialog_opensave_trim_dst_num_jumps`

17.78.3.71 **dialog_printing_default_device** `QString dialog_printing_default_device`

17.78.3.72 **dialog_printing_disable_bg** `bool dialog_printing_disable_bg`

17.78.3.73 **dialog_printing_use_last_device** `bool dialog_printing_use_last_device`

17.78.3.74 **dialog_prompt_bg_color** `QRgb dialog_prompt_bg_color`

17.78.3.75 **dialog_prompt_font_family** `QString dialog_prompt_font_family`

17.78.3.76 **dialog_prompt_font_size** `quint8 dialog_prompt_font_size`

17.78.3.77 **dialog_prompt_font_style** `QString dialog_prompt_font_style`

17.78.3.78 **dialog_prompt_save_history** `bool dialog_prompt_save_history`

17.78.3.79 **dialog_prompt_save_history_as_html** `bool dialog_prompt_save_history_as_html`

17.78.3.80 **dialog_prompt_save_history_filename** `QString dialog_prompt_save_history_filename`

17.78.3.81 **dialog_prompt_text_color** `QRgb dialog_prompt_text_color`

17.78.3.82 **dialog_qsnap_aperture_size** `quint8 dialog_qsnap_aperture_size`

17.78.3.83 **dialog_qsnap_apparent** `bool dialog_qsnap_apparent`

17.78.3.84 **dialog_qsnap_center** `bool dialog_qsnap_center`

17.78.3.85 **dialog_qsnap_enabled** `bool dialog_qsnap_enabled`

17.78.3.86 **dialog_qsnap_endpoint** `bool dialog_qsnap_endpoint`

17.78.3.87 **dialog_qsnap_extension** `bool dialog_qsnap_extension`

17.78.3.88 **dialog_qsnap_insertion** `bool dialog_qsnap_insertion`

17.78.3.89 **dialog_qsnap_intersection** `bool dialog_qsnap_intersection`

17.78.3.90 **dialog_qsnap_locator_color** `QRgb dialog_qsnap_locator_color`

17.78.3.91 **dialog_qsnap_locator_size** `quint8 dialog_qsnap_locator_size`

17.78.3.92 **dialog_qsnap_midpoint** `bool dialog_qsnap_midpoint`

17.78.3.93 **dialog_qsnap_nearest** `bool dialog_qsnap_nearest`

17.78.3.94 **dialog_qsnap_node** `bool dialog_qsnap_node`

17.78.3.95 **dialog_qsnap_parallel** `bool dialog_qsnap_parallel`

17.78.3.96 **dialog_qsnap_perpendicular** `bool dialog_qsnap_perpendicular`

17.78.3.97 **dialog_qsnap_quadrant** `bool dialog_qsnap_quadrant`

17.78.3.98 **dialog_qsnap_tangent** `bool dialog_qsnap_tangent`

17.78.3.99 **dialog_ruler_color** `QRgb dialog_ruler_color`

17.78.3.100 **dialog_ruler_metric** bool dialog_ruler_metric

17.78.3.101 **dialog_ruler_pixel_size** quint8 dialog_ruler_pixel_size

17.78.3.102 **dialog_ruler_show_on_load** bool dialog_ruler_show_on_load

17.78.3.103 **dialog_selection_coolgrip_color** QRgb dialog_selection_coolgrip_color

17.78.3.104 **dialog_selection_grip_size** quint8 dialog_selection_grip_size

17.78.3.105 **dialog_selection_hotgrip_color** QRgb dialog_selection_hotgrip_color

17.78.3.106 **dialog_selection_mode_pickadd** bool dialog_selection_mode_pickadd

17.78.3.107 **dialog_selection_mode_pickdrag** bool dialog_selection_mode_pickdrag

17.78.3.108 **dialog_selection_mode_pickfirst** bool dialog_selection_mode_pickfirst

17.78.3.109 **dialog_selection_pickbox_size** quint8 dialog_selection_pickbox_size

17.78.3.110 **mainWin** MainWindow* mainWin

17.78.3.111 **preview_display_bg_color** QRgb preview_display_bg_color

17.78.3.112 **preview_display_crosshair_color** QRgb preview_display_crosshair_color

17.78.3.113 **preview_display_selectbox_alpha** quint8 preview_display_selectbox_alpha

17.78.3.114 **preview_display_selectbox_left_color** QRgb preview_display_selectbox_left_color

17.78.3.115 **preview_display_selectbox_left_fill** QRgb preview_display_selectbox_left_fill

17.78.3.116 **preview_display_selectbox_right_color** QRgb preview_display_selectbox_right_color

17.78.3.117 **preview_display_selectbox_right_fill** QRgb preview_display_selectbox_right_fill

17.78.3.118 **preview_display_show_scrollbars** bool preview_display_show_scrollbars

17.78.3.119 **preview_general_mdi_bg_color** QRgb preview_general_mdi_bg_color

17.78.3.120 **preview_general_mdi_bg_use_color** bool preview_general_mdi_bg_use_color

17.78.3.121 **preview_general_mdi_bg_use_logo** bool preview_general_mdi_bg_use_logo

17.78.3.122 **preview_general_mdi_bg_use_texture** bool preview_general_mdi_bg_use_texture

17.78.3.123 **preview_grid_color** QRgb preview_grid_color

17.78.3.124 **preview_lwt_real_render** bool preview_lwt_real_render

17.78.3.125 **preview_lwt_show_lwt** bool preview_lwt_show_lwt

17.78.3.126 **preview_prompt_bg_color** QRgb preview_prompt_bg_color

17.78.3.127 **preview_prompt_font_family** QString preview_prompt_font_family

17.78.3.128 **preview_prompt_font_size** quint8 preview_prompt_font_size

17.78.3.129 **preview_prompt_font_style** QString preview_prompt_font_style

17.78.3.130 **preview_prompt_text_color** QRgb preview_prompt_text_color

17.78.3.131 **preview_ruler_color** QRgb preview_ruler_color

17.78.3.132 **tabWidget** QTabWidget* tabWidget

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/settings-dialog.cpp](#)

17.79 StatusBar Class Reference

```
#include <embroidermodder.h>
```

Public Member Functions

- [StatusBar \(MainWindow *mw, QWidget *parent=0\)](#)
- [void setMouseCoord \(EmbReal x, EmbReal y\)](#)

Public Attributes

- `StatusBarButton * statusBarSnapButton`
- `StatusBarButton * statusBarGridButton`
- `StatusBarButton * statusBarRulerButton`
- `StatusBarButton * statusBarOrthoButton`
- `StatusBarButton * statusBarPolarButton`
- `StatusBarButton * statusBarQSnapButton`
- `StatusBarButton * statusBarQTrackButton`
- `StatusBarButton * statusBarLwtButton`
- `QLabel * statusBarMouseCoord`

17.79.1 Constructor & Destructor Documentation

```
17.79.1.1 StatusBar() StatusBar (  
    MainWindow * mw,  
    QWidget * parent = 0 )
```

Embroidermodder 2

Copyright 2013-2022 The Embroidermodder Team Embroidermodder 2 is Open Source Software. See LICENSE for licensing terms.

Use Python's PEP7 style guide. <https://peps.python.org/pep-0007/>

17.79.2 Member Function Documentation

```
17.79.2.1 setMouseCoord() void setMouseCoord (  
    EmbReal x,  
    EmbReal y )
```

17.79.3 Member Data Documentation

```
17.79.3.1 statusBarGridButton StatusBarButton* statusBarGridButton
```

```
17.79.3.2 statusBarLwtButton StatusBarButton* statusBarLwtButton
```

```
17.79.3.3 statusBarMouseCoord QLabel* statusBarMouseCoord
```

```
17.79.3.4 statusBarOrthoButton StatusBarButton* statusBarOrthoButton
```

```
17.79.3.5 statusBarPolarButton StatusBarButton* statusBarPolarButton
```

```
17.79.3.6 statusBarQSnapButton StatusBarButton* statusBarQSnapButton
```

```
17.79.3.7 statusBarQTrackButton StatusBarButton* statusBarQTrackButton
```

17.79.3.8 statusBarRulerButton `StatusBarButton* statusBarRulerButton`**17.79.3.9 statusBarSnapButton** `StatusBarButton* statusBarSnapButton`

The documentation for this class was generated from the following files:

- `embroidermodder2/embroidermodder.h`
- `embroidermodder2/statusbar.cpp`

17.80 StatusBarButton Class Reference

```
#include <embroidermodder.h>
```

Public Slots

- `void enableLwt ()`
- `void disableLwt ()`
- `void enableReal ()`
- `void disableReal ()`

Public Member Functions

- `StatusBarButton (QString buttonText, MainWindow *mw, StatusBar *statbar, QWidget *parent=0)`

Public Attributes

- `MainWindow * mainWin`
- `StatusBar * statusbar`

Protected Member Functions

- `void contextMenuEvent (QContextMenuEvent *event=0)`

Private Slots

- `void settingsSnap ()`
- `void settingsGrid ()`
- `void settingsRuler ()`
- `void settingsOrtho ()`
- `void settingsPolar ()`
- `void settingsQSnap ()`
- `void settingsQTrack ()`
- `void settingsLwt ()`
- `void toggleSnap (bool on)`
- `void toggleGrid (bool on)`
- `void toggleRuler (bool on)`
- `void toggleOrtho (bool on)`
- `void togglePolar (bool on)`
- `void toggleQSnap (bool on)`
- `void toggleQTrack (bool on)`
- `void toggleLwt (bool on)`

17.80.1 Constructor & Destructor Documentation

```
17.80.1.1 StatusBarButton() statusBarButton (  
    QString buttonText,  
    MainWindow * mw,  
    StatusBar * statbar,  
    QWidget * parent = 0 )
```

Embroidermodder 2,

Copyright 2013-2022 The Embroidermodder Team Embroidermodder 2 is Open Source Software. See LICENSE for licensing terms.

Use Python's PEP7 style guide. <https://peps.python.org/pep-0007/>

17.80.2 Member Function Documentation

```
17.80.2.1 contextMenuEvent() void contextMenuEvent (  
    QContextMenuEvent * event = 0 ) [protected]
```

```
17.80.2.2 disableLwt void disableLwt () [slot]
```

```
17.80.2.3 disableReal void disableReal () [slot]
```

```
17.80.2.4 enableLwt void enableLwt () [slot]
```

```
17.80.2.5 enableReal void enableReal () [slot]
```

```
17.80.2.6 settingsGrid void settingsGrid () [private], [slot]
```

```
17.80.2.7 settingsLwt void settingsLwt () [private], [slot]
```

```
17.80.2.8 settingsOrtho void settingsOrtho () [private], [slot]
```

```
17.80.2.9 settingsPolar void settingsPolar () [private], [slot]
```

```
17.80.2.10 settingsQSnap void settingsQSnap () [private], [slot]
```

```
17.80.2.11 settingsQTrack void settingsQTrack () [private], [slot]
```

```
17.80.2.12 settingsRuler void settingsRuler () [private], [slot]
```

```
17.80.2.13 settingsSnap void settingsSnap () [private], [slot]
```

17.80.2.14 toggleGrid void toggleGrid (
 bool on) [private], [slot]

17.80.2.15 toggleLwt void toggleLwt (
 bool on) [private], [slot]

17.80.2.16 toggleOrtho void toggleOrtho (
 bool on) [private], [slot]

17.80.2.17 togglePolar void togglePolar (
 bool on) [private], [slot]

17.80.2.18 toggleQSnap void toggleQSnap (
 bool on) [private], [slot]

17.80.2.19 toggleQTrack void toggleQTrack (
 bool on) [private], [slot]

17.80.2.20 toggleRuler void toggleRuler (
 bool on) [private], [slot]

17.80.2.21 toggleSnap void toggleSnap (
 bool on) [private], [slot]

17.80.3 Member Data Documentation

17.80.3.1 mainWin [MainWindow*](#) mainWin

17.80.3.2 statusbar [StatusBar*](#) statusbar

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/statusbar-button.cpp](#)

17.81 StxThread_ Struct Reference

#include <embroidery_internal.h>

Public Attributes

- [char* colorCode](#)
- [char* colorName](#)
- [char* sectionName](#)
- [SubDescriptor* subDescriptors](#)
- [EmbColor stxColor](#)

17.81.1 Member Data Documentation

17.81.1.1 colorCode char* colorCode

17.81.1.2 colorName char* colorName

17.81.1.3 sectionName char* sectionName

17.81.1.4 stxColor EmbColor stxColor

17.81.1.5 subDescriptors SubDescriptor* subDescriptors

The documentation for this struct was generated from the following file:

- extern/libembroidery/src/embroidery_internal.h

17.82 SubDescriptor_ Struct Reference

```
#include <embroidery_internal.h>
```

Public Attributes

- int someNum
- int someInt
- int someOtherInt
- char * colorCode
- char * colorName

17.82.1 Member Data Documentation

17.82.1.1 colorCode char* colorCode

Todo better variable naming

17.82.1.2 colorName char* colorName

17.82.1.3 someInt int someInt

Todo better variable naming

17.82.1.4 someNum int someNum

17.82.1.5 someOtherInt int someOtherInt

Todo better variable naming

The documentation for this struct was generated from the following file:

- [extern/libembroidery/src/embroidery_internal.h](#)

17.83 SvgAttribute_ Struct Reference

```
#include <embroidery_internal.h>
```

Public Attributes

- `char * name`
- `char * value`

17.83.1 Member Data Documentation

17.83.1.1 name char* name

17.83.1.2 value char* value

The documentation for this struct was generated from the following file:

- [extern/libembroidery/src/embroidery_internal.h](#)

17.84 TextSingleObject Class Reference

```
#include <embroidermodder.h>
```

Public Types

- enum { `Type` = `OBJ_TYPE_TEXTSINGLE` }

Public Types inherited from `BaseObject`

- enum { `Type` = `OBJ_TYPE_BASE` }

Public Member Functions

- `TextSingleObject` (const `QString` &`str`, `EmbReal` `x`, `EmbReal` `y`, `QRgb` `rgb`, `QGraphicsItem` *`parent`=0)
- `TextSingleObject` (`TextSingleObject` *`obj`, `QGraphicsItem` *`parent`=0)
- `~TextSingleObject` ()
- virtual int `type` () const
- void `init` (const `QString` &`str`, `EmbReal` `x`, `EmbReal` `y`, `QRgb` `rgb`, `Qt::PenStyle` `lineType`)
- `QList< QPainterPath >` `objectSavePathList` () const
- `QList< QPainterPath >` `subPathList` () const
- `QPointF` `objectPos` () const
- `EmbReal` `objectX` () const
- `EmbReal` `objectY` () const
- `QStringList` `objectTextJustifyList` () const
- void `setObjectText` (const `QString` &`str`)
- void `setObjectTextFont` (const `QString` &`font`)
- void `setObjectTextJustify` (const `QString` &`justify`)
- void `setObjectTextSize` (`EmbReal` `size`)
- void `setObjectTextStyle` (bool `bold`, bool `italic`, bool `under`, bool `strike`, bool `over`)

- void `setObjectTextBold` (bool val)
- void `setObjectTextItalic` (bool val)
- void `setObjectTextUnderline` (bool val)
- void `setObjectTextStrikeOut` (bool val)
- void `setObjectTextOverline` (bool val)
- void `setObjectTextBackward` (bool val)
- void `setObjectTextUpsideDown` (bool val)
- void `setObjectPos` (const QPointF &point)
- void `setObjectPos` (EmbReal x, EmbReal y)
- void `setObjectX` (EmbReal x)
- void `setObjectY` (EmbReal y)
- void `updateRubber` (QPainter *painter=0)
- virtual void `vulcanize` ()
- virtual QPointF `mouseSnapPoint` (const QPointF &mousePoint)
- virtual QList< QPointF > `allGripPoints` ()
- virtual void `gripEdit` (const QPointF &before, const QPointF &after)

Public Member Functions inherited from `BaseObject`

- `BaseObject` (QGraphicsItem *parent=0)
- virtual ~`BaseObject` ()
- virtual int `type` () const
- qint64 `objectID` () const
- QPen `objectPen` () const
- QColor `objectColor` () const
- QRgb `objectColorRGB` () const
- Qt::PenStyle `objectLineType` () const
- EmbReal `objectLineWidth` () const
- QPainterPath `objectPath` () const
- int `objectRubberMode` () const
- QPointF `objectRubberPoint` (const QString &key) const
- QString `objectRubberText` (const QString &key) const
- QPointF `objectCenter` () const
- EmbReal `objectCenterX` () const
- EmbReal `objectCenterY` () const
- void `setObjectCenter` (EmbVector center)
- void `setObjectCenterX` (EmbReal centerX)
- void `setObjectCenterY` (EmbReal centerY)
- QRectF `rect` () const
- void `setRect` (const QRectF &r)
- void `setRect` (EmbReal x, EmbReal y, EmbReal w, EmbReal h)
- QLineF `line` () const
- void `setLine` (const QLineF &li)
- void `setLine` (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2)
- void `setObjectColor` (const QColor &color)
- void `setObjectColorRGB` (QRgb rgb)
- void `setObjectLineType` (Qt::PenStyle lineType)
- void `setObjectLineWidth` (EmbReal lineWidth)
- void `setObjectPath` (const QPainterPath &p)
- void `setObjectRubberMode` (int mode)
- void `setObjectRubberPoint` (const QString &key, const QPointF &point)
- void `setObjectRubberText` (const QString &key, const QString &txt)
- virtual QRectF `boundingRect` () const
- virtual QPainterPath `shape` () const
- void `drawRubberLine` (const QLineF &rubLine, QPainter *painter=0, const char *colorFromScene=0)
- virtual void `vulcanize` ()=0
- virtual QPointF `mouseSnapPoint` (const QPointF &mousePoint)=0
- virtual QList< QPointF > `allGripPoints` ()=0
- virtual void `gripEdit` (const QPointF &before, const QPointF &after)=0

Public Attributes

- `QString objText`
- `QString objTextFont`
- `QString objTextJustify`
- `EmbReal objTextSize`
- `bool objTextBold`
- `bool objTextItalic`
- `bool objTextUnderline`
- `bool objTextStrikeOut`
- `bool objTextOverline`
- `bool objTextBackward`
- `bool objTextUpsideDown`
- `QPainterPath objTextPath`

Public Attributes inherited from [BaseObject](#)

- `QPen objPen`
- `QPen lwtPen`
- `QLineF objLine`
- `int objRubberMode`
- `QHash<QString, QPointF> objRubberPoints`
- `QHash<QString, QString> objRubberTexts`
- `qint64 objID`

Protected Member Functions

- `void paint (QPainter *, const QStyleOptionGraphicsItem *, QWidget *)`

Protected Member Functions inherited from [BaseObject](#)

- `QPen lineWeightPen () const`
- `void realRender (QPainter *painter, const QPainterPath &renderPath)`

17.84.1 Member Enumeration Documentation

17.84.1.1 anonymous enum `anonymous enum`

Enumerator

| | |
|------|---------------------------------|
| Type | <input type="button" value=""/> |
|------|---------------------------------|

17.84.2 Constructor & Destructor Documentation

17.84.2.1 `TextSingleObject()` [1/2] `TextSingleObject (`

```
    const QString & str,
    EmbReal x,
    EmbReal y,
    QRgb rgb,
    QGraphicsItem * parent = 0 )
```

17.84.2.2 `TextSingleObject()` [2/2] `TextSingleObject (`

```
    TextSingleObject * obj,
    QGraphicsItem * parent = 0 )
```

17.84.2.3 `~TextSingleObject()` `~TextSingleObject ()`**17.84.3 Member Function Documentation****17.84.3.1 `allGripPoints()`** `QList< QPointF > allGripPoints () [virtual]`
Implements [BaseObject](#).**17.84.3.2 `gripEdit()`** `void gripEdit (`
 `const QPointF & before,`
 `const QPointF & after) [virtual]`Implements [BaseObject](#).**17.84.3.3 `init()`** `void init (`
 `const QString & str,`
 `EmbReal x,`
 `EmbReal y,`
 `QRgb rgb,`
 `Qt::PenStyle lineType)`**17.84.3.4 `mouseSnapPoint()`** `QPointF mouseSnapPoint (`
 `const QPointF & mousePoint) [virtual]`

Returns the closest snap point to the mouse point

Implements [BaseObject](#).**17.84.3.5 `objectPos()`** `QPointF objectPos () const [inline]`**17.84.3.6 `objectSavePathList()`** `QList< QPainterPath > objectSavePathList () const [inline]`**17.84.3.7 `objectTextJustifyList()`** `QStringList objectTextJustifyList () const`**17.84.3.8 `objectX()`** `EmbReal objectX () const [inline]`**17.84.3.9 `objectY()`** `EmbReal objectY () const [inline]`**17.84.3.10 `paint()`** `void paint (`
 `QPainter * painter,`
 `const QStyleOptionGraphicsItem * option,`
 `QWidget *) [protected]`

17.84.3.11 `setObjectPos()` [1/2] void setObjectPos (const QPointF & *point*) [inline]

17.84.3.12 `setObjectPos()` [2/2] void setObjectPos (EmbReal *x*, EmbReal *y*) [inline]

17.84.3.13 `setObjectText()` void setObjectText (const QString & *str*)

17.84.3.14 `setObjectTextBackward()` void setObjectTextBackward (bool *val*)

17.84.3.15 `setObjectTextBold()` void setObjectTextBold (bool *val*)

17.84.3.16 `setObjectTextFont()` void setObjectTextFont (const QString & *font*)

17.84.3.17 `setObjectTextItalic()` void setObjectTextItalic (bool *val*)

17.84.3.18 `setObjectTextJustify()` void setObjectTextJustify (const QString & *justify*)

17.84.3.19 `setObjectTextOverline()` void setObjectTextOverline (bool *val*)

17.84.3.20 `setObjectTextSize()` void setObjectTextSize (EmbReal *size*)

17.84.3.21 `setObjectTextStrikeOut()` void setObjectTextStrikeOut (bool *val*)

17.84.3.22 `setObjectTextStyle()` void setObjectTextStyle (bool *bold*, bool *italic*, bool *under*, bool *strike*, bool *over*)

17.84.3.23 `setObjectTextUnderline()` void setObjectTextUnderline (bool *val*)

17.84.3.24 `setObjectTextUpsideDown()` void setObjectTextUpsideDown (bool val)

17.84.3.25 `setObjectX()` void setObjectX (EmbReal x) [inline]

17.84.3.26 `setObjectY()` void setObjectY (EmbReal y) [inline]

17.84.3.27 `subPathList()` QList< QPainterPath > subPathList () const

17.84.3.28 `type()` virtual int type () const [inline], [virtual]
Reimplemented from [BaseObject](#).

17.84.3.29 `updateRubber()` void updateRubber (QPainter * painter = 0)

17.84.3.30 `vulcanize()` void vulcanize () [virtual]
Implements [BaseObject](#).

17.84.4 Member Data Documentation

17.84.4.1 `objText` QString objText

17.84.4.2 `objTextBackward` bool objTextBackward

17.84.4.3 `objTextBold` bool objTextBold

17.84.4.4 `objTextFont` QString objTextFont

17.84.4.5 `objTextItalic` bool objTextItalic

17.84.4.6 `objTextJustify` QString objTextJustify

17.84.4.7 `objTextOverline` bool objTextOverline

17.84.4.8 `objTextPath` QPainterPath objTextPath

17.84.4.9 objTextSize `EmbReal objTextSize`

17.84.4.10 objTextStrikeOut `bool objTextStrikeOut`

17.84.4.11 objTextUnderline `bool objTextUnderline`

17.84.4.12 objTextUpsideDown `bool objTextUpsideDown`

The documentation for this class was generated from the following files:

- `embroidermodder2/embroidermodder.h`
- `embroidermodder2/object-textsingle.cpp`

17.85 thread_color_ Struct Reference

```
#include <embroidery.h>
```

Public Attributes

- `char name [22]`
- `unsigned int hex_code`
- `int manufacturer_code`

17.85.1 Member Data Documentation

17.85.1.1 hex_code `unsigned int hex_code`

17.85.1.2 manufacturer_code `int manufacturer_code`

17.85.1.3 name `char name[22]`

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery.h`

17.86 ThredExtension_ Struct Reference

```
#include <embroidery_internal.h>
```

Public Attributes

- `float hoopX`
- `float hoopY`
- `float stitchGranularity`
- `char creatorName [50]`
- `char modifierName [50]`
- `char auxFormat`
- `char reserved [31]`

17.86.1 Member Data Documentation

17.86.1.1 auxFormat char auxFormat

17.86.1.2 creatorName char creatorName[50]

17.86.1.3 hoopX float hoopX

17.86.1.4 hoopY float hoopY

17.86.1.5 modifierName char modifierName[50]

17.86.1.6 reserved char reserved[31]

17.86.1.7 stitchGranularity float stitchGranularity

The documentation for this struct was generated from the following file:

- [extern/libembroidery/src/embroidery_internal.h](#)

17.87 ThredHeader_ Struct Reference

#include <embroidery_internal.h>

Public Attributes

- unsigned int [sigVersion](#)
- unsigned int [length](#)
- unsigned short [numStiches](#)
- unsigned short [hoopSize](#)
- unsigned short [reserved \[7\]](#)

17.87.1 Member Data Documentation

17.87.1.1 hoopSize unsigned short hoopSize

17.87.1.2 length unsigned int length

17.87.1.3 numStiches unsigned short numStiches

17.87.1.4 reserved unsigned short reserved[7]

17.87.1.5 sigVersion unsigned int sigVersion

The documentation for this struct was generated from the following file:

- [extern/libembroidery/src/embroidery_internal.h](#)

17.88 UiObject_Struct Reference

This covers the inbuilt designs: Dolphin, Snowflake and Heart. Covers Rotate, Scale and Point UI events.

```
#include <embroidermodder.h>
```

Public Attributes

- char `fname` [200]
- char `command` [200]
- bool `firstRun`
- `EmbVector controlPoints` [10]
- char `controlPointLabels` [10][200]
- int `n_controlPoints`
- int `numPoints`
- int `minPoints`
- int `maxPoints`
- `EmbVector center`
- `EmbVector scale`
- `EmbReal rotation`
- `uint32_t mode`
- char `path_desc` [1000]
- char `text` [200]
- int `textJustify`
- char `textFont` [200]
- `EmbReal textHeight`
- `EmbReal textRotation`
- char `id` [200]
- int `pattern_index`
- char `type` [200]
- int `object_index`
- bool `selectable`
- `EmbColor color`

17.88.1 Detailed Description

This covers the inbuilt designs: Dolphin, Snowflake and Heart. Covers Rotate, Scale and Point UI events.
scale is how radii are stored if the object is a circle, or the semi-major and semi-minor axes if the object is an ellipse.
center perhaps should be the "anchor" instead which is the center for some objects and top left for rectangular objects.

Do angles need special storage? angleRef, angleNew

This chapter details how the source code achieves the design detailed in the previous chapter. For the low-level details, read the later chapters.

Dear ImGui (CITATION NEEDED)

17.88.1.1 Geometry Objects At all times the EmbPattern has all of the information about the pattern however, editing information like the rubber text labels needs to be stored during runtime. Also editing ghosts like when a rotate command is half executed.

To deal with this we have a generic object that can act as any other object that updates the associated pattern struct as changes are made.

Note that the editor state is separate from this since that is per view, not per object.

Selecting is done via this interface.

17.88.2 Member Data Documentation

17.88.2.1 center `EmbVector` center

17.88.2.2 color `EmbColor` color

17.88.2.3 command `char` command[200]

17.88.2.4 controlPointLabels `char` controlPointLabels[10][200]

17.88.2.5 controlPoints `EmbVector` controlPoints[10]

17.88.2.6 firstRun `bool` firstRun

17.88.2.7 fname `char` fname[200]

17.88.2.8 id `char` id[200]

17.88.2.9 maxPoints `int` maxPoints

17.88.2.10 minPoints `int` minPoints

17.88.2.11 mode `uint32_t` mode

17.88.2.12 n_controlPoints `int` n_controlPoints

17.88.2.13 numPoints `int` numPoints

17.88.2.14 object_index `int` object_index

17.88.2.15 path_desc `char` path_desc[1000]

17.88.2.16 pattern_index `int` pattern_index

17.88.2.17 rotation `EmbReal` rotation

17.88.2.18 scale `EmbVector` scale

17.88.2.19 selectable bool selectable

17.88.2.20 text char text[200]

17.88.2.21 textFont char textFont[200]

17.88.2.22 textHeight EmbReal textHeight

17.88.2.23 textJustify int textJustify

17.88.2.24 textRotation EmbReal textRotation

17.88.2.25 type char type[200]

The documentation for this struct was generated from the following file:

- [embroidermodder2/embroidermodder.h](#)

17.89 UndoableAddCommand Class Reference

```
#include <embroidermodder.h>
```

Public Member Functions

- [UndoableAddCommand](#) (const QString &text, [BaseObject](#) *obj, [View](#) *v, [QUndoCommand](#) *parent=0)
- void [undo](#) ()
- void [redo](#) ()

Public Attributes

- [BaseObject](#) * object
- [View](#) * gview

17.89.1 Constructor & Destructor Documentation

17.89.1.1 UndoableAddCommand() [UndoableAddCommand](#) (

```
    const QString & text,
    BaseObject * obj,
    View * v,
    QUndoCommand * parent = 0 )
```

17.89.2 Member Function Documentation

17.89.2.1 redo() void redo ()

17.89.2.2 undo() void undo ()

17.89.3 Member Data Documentation

17.89.3.1 gview `View* gview`

17.89.3.2 object `BaseObject* object`

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/undo-commands.cpp](#)

17.90 UndoableDeleteCommand Class Reference

```
#include <embroidermodder.h>
```

Public Member Functions

- [UndoableDeleteCommand \(const QString &text, BaseObject *obj, View *v, QUndoCommand *parent=0\)](#)
- [void undo \(\)](#)
- [void redo \(\)](#)

Public Attributes

- [BaseObject * object](#)
- [View * gview](#)

17.90.1 Constructor & Destructor Documentation

```
17.90.1.1 UndoableDeleteCommand() UndoableDeleteCommand (  
    const QString & text,  
    BaseObject * obj,  
    View * v,  
    QUndoCommand * parent = 0 )
```

17.90.2 Member Function Documentation

17.90.2.1 redo() `void redo ()`

17.90.2.2 undo() `void undo ()`

17.90.3 Member Data Documentation

17.90.3.1 gview `View* gview`

17.90.3.2 object `BaseObject* object`

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/undo-commands.cpp](#)

17.91 UndoableGripEditCommand Class Reference

```
#include <embroidermodder.h>
```

Public Member Functions

- `UndoableGripEditCommand (const QPointF beforePoint, const QPointF afterPoint, const QString &text, BaseObject *obj, View *v, QUndoCommand *parent=0)`
- `void undo ()`
- `void redo ()`

Public Attributes

- `BaseObject * object`
- `View * gview`
- `QPointF before`
- `QPointF after`

17.91.1 Constructor & Destructor Documentation

```
17.91.1.1 UndoableGripEditCommand() UndoableGripEditCommand (
```

```
    const QPointF beforePoint,
```

```
    const QPointF afterPoint,
```

```
    const QString & text,
```

```
    BaseObject * obj,
```

```
    View * v,
```

```
    QUndoCommand * parent = 0 )
```

17.91.2 Member Function Documentation

```
17.91.2.1 redo() void redo ( )
```

```
17.91.2.2 undo() void undo ( )
```

17.91.3 Member Data Documentation

```
17.91.3.1 after QPointF after
```

```
17.91.3.2 before QPointF before
```

```
17.91.3.3 gview View* gview
```

```
17.91.3.4 object BaseObject* object
```

The documentation for this class was generated from the following files:

- `embroidermodder2/embroidermodder.h`
- `embroidermodder2/undo-commands.cpp`

17.92 UndoableMirrorCommand Class Reference

```
#include <embroidermodder.h>
```

Public Member Functions

- `UndoableMirrorCommand (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, const QString &text, BaseObject *obj, View *v, QUndoCommand *parent=0)`
- `void undo ()`
- `void redo ()`
- `void mirror ()`

Public Attributes

- `BaseObject * object`
- `View * gview`
- `QLineF mirrorLine`

17.92.1 Constructor & Destructor Documentation

```
17.92.1.1 UndoableMirrorCommand() UndoableMirrorCommand (
    EmbReal x1,
    EmbReal y1,
    EmbReal x2,
    EmbReal y2,
    const QString & text,
    BaseObject * obj,
    View * v,
    QUndoCommand * parent = 0 )
```

17.92.2 Member Function Documentation

```
17.92.2.1 mirror() void mirror ( )
```

```
17.92.2.2 redo() void redo ( )
```

```
17.92.2.3 undo() void undo ( )
```

17.92.3 Member Data Documentation

```
17.92.3.1 gview View* gview
```

```
17.92.3.2 mirrorLine QLineF mirrorLine
```

```
17.92.3.3 object BaseObject* object
```

The documentation for this class was generated from the following files:

- `embroidermodder2/embroidermodder.h`
- `embroidermodder2/undo-commands.cpp`

17.93 UndoableMoveCommand Class Reference

```
#include <embroidermodder.h>
```

Public Member Functions

- `UndoableMoveCommand (EmbReal deltaX, EmbReal deltaY, const QString &text, BaseObject *obj, View *v, QUndoCommand *parent=0)`
- `void undo ()`
- `void redo ()`

Public Attributes

- `BaseObject * object`
- `View * gview`
- `EmbReal dx`
- `EmbReal dy`

17.93.1 Constructor & Destructor Documentation

```
17.93.1.1 UndoableMoveCommand() UndoableMoveCommand (
```

```
    EmbReal deltaX,
```

```
    EmbReal deltaY,
```

```
    const QString & text,
```

```
    BaseObject * obj,
```

```
    View * v,
```

```
    QUndoCommand * parent = 0 )
```

17.93.2 Member Function Documentation

```
17.93.2.1 redo() void redo ( )
```

```
17.93.2.2 undo() void undo ( )
```

17.93.3 Member Data Documentation

```
17.93.3.1 dx EmbReal dx
```

```
17.93.3.2 dy EmbReal dy
```

```
17.93.3.3 gview View* gview
```

```
17.93.3.4 object BaseObject* object
```

The documentation for this class was generated from the following files:

- `embroidermodder2/embroidermodder.h`
- `embroidermodder2/undo-commands.cpp`

17.94 UndoableNavCommand Class Reference

```
#include <embroidermodder.h>
```

Public Member Functions

- `UndoableNavCommand` (const QString &`type`, `View` *`v`, QUndoCommand *`parent`=0)
- int `id` () const
- bool `mergeWith` (const QUndoCommand *`command`)
- void `undo` ()
- void `redo` ()

Public Attributes

- QString `navType`
- QTransform `fromTransform`
- QTransform `toTransform`
- QPointF `fromCenter`
- QPointF `toCenter`
- bool `done`
- `View` *`gview`

17.94.1 Constructor & Destructor Documentation

```
17.94.1.1 UndoableNavCommand() UndoableNavCommand (  
    const QString & type,  
    View * v,  
    QUndoCommand * parent = 0 )
```

17.94.2 Member Function Documentation

```
17.94.2.1 id() int id () const [inline]
```

```
17.94.2.2 mergeWith() bool mergeWith (  
    const QUndoCommand * command )
```

```
17.94.2.3 redo() void redo ()
```

```
17.94.2.4 undo() void undo ()
```

17.94.3 Member Data Documentation

```
17.94.3.1 done bool done
```

```
17.94.3.2 fromCenter QPointF fromCenter
```

17.94.3.3 fromTransform `QTransform fromTransform`

17.94.3.4 gview `View* gview`

17.94.3.5 navType `QString navType`

17.94.3.6 toCenter `QPointF toCenter`

17.94.3.7 toTransform `QTransform toTransform`

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/undo-commands.cpp](#)

17.95 UndoableRotateCommand Class Reference

```
#include <embroidermodder.h>
```

Public Member Functions

- `UndoableRotateCommand (EmbReal pivotPointX, EmbReal pivotPointY, EmbReal rotAngle, const QString &text, BaseObject *obj, View *v, QUndoCommand *parent=0)`
- `void undo ()`
- `void redo ()`
- `void rotate (EmbReal x, EmbReal y, EmbReal rot)`

Public Attributes

- `BaseObject * object`
- `View * gview`
- `EmbReal pivotX`
- `EmbReal pivotY`
- `EmbReal angle`

17.95.1 Constructor & Destructor Documentation

17.95.1.1 UndoableRotateCommand() `UndoableRotateCommand (`

```
    EmbReal pivotPointX,
    EmbReal pivotPointY,
    EmbReal rotAngle,
    const QString & text,
    BaseObject * obj,
    View * v,
    QUndoCommand * parent = 0 )
```

17.95.2 Member Function Documentation

17.95.2.1 redo() `void redo ()`

```
17.95.2.2 rotate() void rotate (
    EmbReal x,
    EmbReal y,
    EmbReal rot )
```

```
17.95.2.3 undo() void undo ( )
```

17.95.3 Member Data Documentation

```
17.95.3.1 angle EmbReal angle
```

```
17.95.3.2 gview View* gview
```

```
17.95.3.3 object BaseObject* object
```

```
17.95.3.4 pivotX EmbReal pivotX
```

```
17.95.3.5 pivotY EmbReal pivotY
```

The documentation for this class was generated from the following files:

- embroidermodder2/[embroidermodder.h](#)
- embroidermodder2/[undo-commands.cpp](#)

17.96 UndoableScaleCommand Class Reference

```
#include <embroidermodder.h>
```

Public Member Functions

- [UndoableScaleCommand](#) (EmbReal x, EmbReal y, EmbReal scaleFactor, const QString &text, BaseObject *obj, View *v, QUndoCommand *parent=0)
- void [undo](#) ()
- void [redo](#) ()

Public Attributes

- BaseObject * [object](#)
- View * [gview](#)
- EmbReal [dx](#)
- EmbReal [dy](#)
- EmbReal [factor](#)

17.96.1 Constructor & Destructor Documentation

```
17.96.1.1 UndoableScaleCommand() UndoableScaleCommand (
    EmbReal x,
    EmbReal y,
    EmbReal scaleFactor,
    const QString & text,
    BaseObject * obj,
    View * v,
    QUndoCommand * parent = 0 )
```

17.96.2 Member Function Documentation

17.96.2.1 redo() `void redo ()`

17.96.2.2 undo() `void undo ()`

17.96.3 Member Data Documentation

17.96.3.1 dx `EmbReal dx`

17.96.3.2 dy `EmbReal dy`

17.96.3.3 factor `EmbReal factor`

17.96.3.4 gview `View* gview`

17.96.3.5 object `BaseObject* object`

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/undo-commands.cpp](#)

17.97 UndoEditor Class Reference

#include <embroidermodder.h>

Public Slots

- void [undo \(\)](#)
- void [redo \(\)](#)
- void [updateCleanIcon \(bool opened\)](#)

Public Member Functions

- [UndoEditor \(const QString &iconDirectory=QString\(\), QWidget *widgetToFocus=0, QWidget *parent=0\)](#)
- [~UndoEditor \(\)](#)
- void [addStack \(QUndoStack *stack\)](#)
- bool [canUndo \(\) const](#)
- bool [canRedo \(\) const](#)

- `QString undoText () const`
- `QString redoText () const`

Public Attributes

- `QWidget * focusWidget`
- `QString iconDir`
- `int iconSize`
- `QUndoGroup * undoGroup`
- `QUndoView * undoView`

17.97.1 Constructor & Destructor Documentation

17.97.1.1 UndoEditor() `UndoEditor (`
 `const QString & iconDirectory = QString(),`
 `QWidget * widgetToFocus = 0,`
 `QWidget * parent = 0)`

17.97.1.2 ~UndoEditor() `~UndoEditor ()`

17.97.2 Member Function Documentation

17.97.2.1 addStack() `void addStack (`
 `QUndoStack * stack)`

17.97.2.2 canRedo() `bool canRedo () const`

17.97.2.3 canUndo() `bool canUndo () const`

17.97.2.4 redo `void redo () [slot]`

17.97.2.5 redoText() `QString redoText () const`

17.97.2.6 undo `void undo () [slot]`

17.97.2.7 undoText() `QString undoText () const`

17.97.2.8 updateCleanIcon `void updateCleanIcon (`
 `bool opened) [slot]`

17.97.3 Member Data Documentation

17.97.3.1 focusWidget QWidget* focusWidget

17.97.3.2 iconDir QString iconDir

17.97.3.3 iconSize int iconSize

17.97.3.4 undoGroup QUndoGroup* undoGroup

17.97.3.5 undoView QUndoView* undoView

The documentation for this class was generated from the following files:

- embroidermodder2/embroidermodder.h
- embroidermodder2/undo-editor.cpp

17.98 UndoHistory_ Struct Reference

```
#include <embroidermodder.h>
```

Public Attributes

- std::vector< std::string > **data**
- int **position**

17.98.1 Detailed Description

Todo document this.

17.98.2 Member Data Documentation

17.98.2.1 data std::vector<std::string> **data**

17.98.2.2 position int **position**

The documentation for this struct was generated from the following file:

- embroidermodder2/embroidermodder.h

17.99 View Class Reference

```
#include <embroidermodder.h>
```

Public Slots

- void **zoomIn** ()
- void **zoomOut** ()
- void **zoomWindow** ()
- void **zoomSelected** ()
- void **zoomExtents** ()
- void **panRealTime** ()
- void **panPoint** ()
- void **panLeft** ()

- void `panRight ()`
- void `panUp ()`
- void `panDown ()`
- void `selectAll ()`
- void `selectionChanged ()`
- void `clearSelection ()`
- void `deleteSelected ()`
- void `moveSelected (EmbReal dx, EmbReal dy)`
- void `cut ()`
- void `copy ()`
- void `paste ()`
- void `repeatAction ()`
- void `moveAction ()`
- void `scaleAction ()`
- void `scaleSelected (EmbReal x, EmbReal y, EmbReal factor)`
- void `rotateAction ()`
- void `rotateSelected (EmbReal x, EmbReal y, EmbReal rot)`
- void `mirrorSelected (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2)`
- int `numSelected ()`
- void `deletePressed ()`
- void `escapePressed ()`
- void `cornerButtonClicked ()`
- void `showScrollBars (bool val)`
- void `setCornerButton ()`
- void `setCrossHairColor (QRgb color)`
- void `setCrossHairSize (quint8 percent)`
- void `setBackgroundColor (QRgb color)`
- void `setSelectBoxColors (QRgb colorL, QRgb fillL, QRgb colorR, QRgb fillR, int alpha)`
- void `toggleSnap (bool on)`
- void `toggleGrid (bool on)`
- void `toggleRuler (bool on)`
- void `toggleOrtho (bool on)`
- void `togglePolar (bool on)`
- void `toggleQSnap (bool on)`
- void `toggleQTrack (bool on)`
- void `toggleLwt (bool on)`
- void `toggleReal (bool on)`
- bool `isLwtEnabled ()`
- bool `isRealEnabled ()`
- void `setGridColor (QRgb color)`
- void `createGrid (const QString &gridType)`
- void `setRulerColor (QRgb color)`
- void `previewOn (int clone, int mode, EmbReal x, EmbReal y, EmbReal data)`
- void `previewOff ()`
- void `enableMoveRapidFire ()`
- void `disableMoveRapidFire ()`
- bool `allowRubber ()`
- void `addToRubberRoom (QGraphicsItem *item)`
- void `vulcanizeRubberRoom ()`
- void `clearRubberRoom ()`
- void `spareRubber (qint64 id)`
- void `setRubberMode (int mode)`
- void `setRubberPoint (const QString &key, const QPointF &point)`
- void `setRubberText (const QString &key, const QString &txt)`

Public Member Functions

- `View (MainWindow *mw, QGraphicsScene *theScene, QWidget *parent)`
- `~View ()`
- `bool allowZoomIn ()`
- `bool allowZoomOut ()`
- `void recalculateLimits ()`
- `void zoomToPoint (const QPoint &mousePoint, int zoomDir)`
- `void centerAt (const QPointF ¢erPoint)`
- `QPointF center ()`
- `QUndoStack * getUndoStack ()`
- `void addObject (BaseObject *obj)`
- `void deleteObject (BaseObject *obj)`
- `void vulcanizeObject (BaseObject *obj)`

Protected Member Functions

- `void mouseDoubleClickEvent (QMouseEvent *event)`
- `void mousePressEvent (QMouseEvent *event)`
- `void mouseMoveEvent (QMouseEvent *event)`
- `void mouseReleaseEvent (QMouseEvent *event)`
- `void wheelEvent (QWheelEvent *event)`
- `void contextMenuEvent (QContextMenuEvent *event)`
- `void drawBackground (QPainter *painter, const QRectF &rect)`
- `void drawForeground (QPainter *painter, const QRectF &rect)`
- `void enterEvent (QEvent *event)`

Private Member Functions

- `void createGridRect ()`
- `void createGridPolar ()`
- `void createGridIso ()`
- `void createOrigin ()`
- `void loadRulerSettings ()`
- `bool willUnderflowInt32 (qint64 a, qint64 b)`
- `bool willOverflowInt32 (qint64 a, qint64 b)`
- `int roundToMultiple (bool roundUp, int numToRound, int multiple)`
- `QPainterPath createRulerTextPath (float x, EmbReal y, QString str, EmbReal height)`
- `QList< QGraphicsItem * > createObjectList (QList< QGraphicsItem * > list)`
- `void copySelected ()`
- `void startGripping (BaseObject *obj)`
- `void stopGripping (bool accept=false)`
- `void updateMouseCoords (int x, int y)`
- `void panStart (const QPoint &point)`
- `void alignScenePointWithViewPoint (const QPointF &scenePoint, const QPoint &viewPoint)`

Private Attributes

- `QHash< qint64, QGraphicsItem * > hashDeletedObjects`
- `QList< qint64 > spareRubberList`
- `QColor gridColor`
- `QPainterPath gridPath`
- `QPainterPath originPath`
- `bool rulerMetric`
- `QColor rulerColor`
- `quint8 rulerPixelSize`

- QList< QGraphicsItem * > previewObjectList
- QGraphicsItemGroup * previewObjectItemGroup
- QPointF previewPoint
- EmbReal previewData
- int previewMode
- QPointF cutCopyMousePoint
- QGraphicsItemGroup * pasteObjectItemGroup
- QPointF pasteDelta
- QList< QGraphicsItem * > rubberRoomList
- bool grippingActive
- bool rapidMoveActive
- bool previewActive
- bool pastingActive
- bool movingActive
- bool selectingActive
- bool zoomWindowActive
- bool panningRealTimeActive
- bool panningPointActive
- bool panningActive
- bool qSnapActive
- bool qSnapToggle
- BaseObject * gripBaseObj
- BaseObject * tempBaseObj
- MainWindow * mainWin
- QGraphicsScene * gscene
- QUndoStack * undoStack
- SelectBox * selectBox
- QPointF scenePressPoint
- QPoint pressPoint
- QPointF sceneMovePoint
- QPoint movePoint
- QPointF sceneReleasePoint
- QPoint releasePoint
- QPointF sceneGripPoint
- QPoint viewMousePoint
- QPointF sceneMousePoint
- QRgb qsnapLocatorColor
- quint8 qsnapLocatorSize
- quint8 qsnapApertureSize
- QRgb gripColorCool
- QRgb gripColorHot
- quint8 gripSize
- quint8 pickBoxSize
- QRgb crosshairColor
- quint32 crosshairSize
- int panDistance
- int panStartX
- int panStartY

17.99.1 Constructor & Destructor Documentation

17.99.1.1 `View()` `View (`
 `MainWindow * mw,`
 `QGraphicsScene * theScene,`
 `QWidget * parent)`

17.99.1.2 `~View()` `~View ()`

17.99.2 Member Function Documentation

17.99.2.1 `addObject()` `void addObject (`
 `BaseObject * obj)`

17.99.2.2 `addToRubberRoom()` `void addToRubberRoom (`
 `QGraphicsItem * item) [slot]`

17.99.2.3 `alignScenePointWithViewPoint()` `void alignScenePointWithViewPoint (`
 `const QPointF & scenePoint,`
 `const QPoint & viewPoint) [private]`

17.99.2.4 `allowRubber()` `bool allowRubber () [slot]`

17.99.2.5 `allowZoomIn()` `bool allowZoomIn ()`

17.99.2.6 `allowZoomOut()` `bool allowZoomOut ()`

17.99.2.7 `center()` `QPointF center () [inline]`

17.99.2.8 `centerAt()` `void centerAt (`
 `const QPointF & centerPoint)`

17.99.2.9 `clearRubberRoom()` `void clearRubberRoom () [slot]`

17.99.2.10 `clearSelection()` `void clearSelection () [slot]`

17.99.2.11 `contextMenuEvent()` `void contextMenuEvent (`
 `QContextMenuEvent * event) [protected]`

17.99.2.12 `copy()` `void copy () [slot]`

17.99.2.13 copySelected() void copySelected () [private]

17.99.2.14 cornerButtonClicked void cornerButtonClicked () [slot]

17.99.2.15 createGrid void createGrid (const QString & gridType) [slot]

17.99.2.16 createGridIso() void createGridIso () [private]

17.99.2.17 createGridPolar() void createGridPolar () [private]

17.99.2.18 createGridRect() void createGridRect () [private]

17.99.2.19 createObjectList() QList< QGraphicsItem * > createObjectList (QList< QGraphicsItem * > list) [private]

17.99.2.20 createOrigin() void createOrigin () [private]

17.99.2.21 createRulerTextPath() QPainterPath createRulerTextPath (float x, EmbReal y, QString str, EmbReal height) [private]

17.99.2.22 cut void cut () [slot]

17.99.2.23 deleteObject() void deleteObject (BaseObject * obj)

17.99.2.24 deletePressed void deletePressed () [slot]

17.99.2.25 deleteSelected void deleteSelected () [slot]

17.99.2.26 disableMoveRapidFire void disableMoveRapidFire () [slot]

17.99.2.27 drawBackground() void drawBackground (QPainter * painter, const QRectF & rect) [protected]

17.99.2.28 drawForeground() void drawForeground (
 QPainter * painter,
 const QRectF & rect) [protected]

17.99.2.29 enableMoveRapidFire void enableMoveRapidFire () [slot]

17.99.2.30 enterEvent() void enterEvent (
 QEvent * event) [protected]

17.99.2.31 escapePressed void escapePressed () [slot]

17.99.2.32 getUndoStack() QUndoStack * getUndoStack () [inline]

17.99.2.33 isLwtEnabled bool isLwtEnabled () [slot]

17.99.2.34 isRealEnabled bool isRealEnabled () [slot]

17.99.2.35 loadRulerSettings() void loadRulerSettings () [private]

17.99.2.36 mirrorSelected void mirrorSelected (
 EmbrReal x1,
 EmbrReal y1,
 EmbrReal x2,
 EmbrReal y2) [slot]

17.99.2.37 mouseDoubleClickEvent() void mouseDoubleClickEvent (
 QMouseEvent * event) [protected]

17.99.2.38 mouseMoveEvent() void mouseMoveEvent (
 QMouseEvent * event) [protected]

17.99.2.39 mousePressEvent() void mousePressEvent (
 QMouseEvent * event) [protected]

17.99.2.40 mouseReleaseEvent() void mouseReleaseEvent (
 QMouseEvent * event) [protected]

17.99.2.41 moveAction void moveAction () [slot]

17.99.2.42 moveSelected void moveSelected (EmbReal dx, EmbReal dy) [slot]

17.99.2.43 numSelected int numSelected () [slot]

17.99.2.44 panDown void panDown () [slot]

17.99.2.45 panLeft void panLeft () [slot]

17.99.2.46 panPoint void panPoint () [slot]

17.99.2.47 panRealTime void panRealTime () [slot]

17.99.2.48 panRight void panRight () [slot]

17.99.2.49 panStart() void panStart (const QPoint & point) [private]

17.99.2.50 panUp void panUp () [slot]

17.99.2.51 paste void paste () [slot]

17.99.2.52 previewOff void previewOff () [slot]

17.99.2.53 previewOn void previewOn (int clone, int mode, EmbReal x, EmbReal y, EmbReal data) [slot]

17.99.2.54 recalculateLimits() void recalculateLimits ()

17.99.2.55 repeatAction void repeatAction () [slot]

17.99.2.56 rotateAction void rotateAction () [slot]

17.99.2.57 `rotateSelected` void rotateSelected (EmbReal *x*, EmbReal *y*, EmbReal *rot*) [slot]

17.99.2.58 `roundToMultiple()` int roundToMultiple (bool *roundUp*, int *numToRound*, int *multiple*) [private]

17.99.2.59 `scaleAction` void scaleAction () [slot]

17.99.2.60 `scaleSelected` void scaleSelected (EmbReal *x*, EmbReal *y*, EmbReal *factor*) [slot]

17.99.2.61 `selectAll` void selectAll () [slot]

17.99.2.62 `selectionChanged` void selectionChanged () [slot]

17.99.2.63 `setBackgroundColor` void setBackgroundColor (QRgb *color*) [slot]

17.99.2.64 `setCornerButton` void setCornerButton () [slot]

17.99.2.65 `setCrossHairColor` void setCrossHairColor (QRgb *color*) [slot]

17.99.2.66 `setCrossHairSize` void setCrossHairSize (quint8 *percent*) [slot]

17.99.2.67 `setGridColor` void setGridColor (QRgb *color*) [slot]

17.99.2.68 `setRubberMode` void setRubberMode (int *mode*) [slot]

17.99.2.69 `setRubberPoint` void setRubberPoint (const QString & *key*, const QPointF & *point*) [slot]

17.99.2.70 `setRubberText` void setRubberText (const QString & key, const QString & txt) [slot]

17.99.2.71 `setRulerColor` void setRulerColor (QRgb color) [slot]

17.99.2.72 `setSelectBoxColors` void setSelectBoxColors (QRgb colorL, QRgb fillL, QRgb colorR, QRgb fillR, int alpha) [slot]

17.99.2.73 `showScrollBars` void showScrollBars (bool val) [slot]

17.99.2.74 `spareRubber` void spareRubber (qint64 id) [slot]

17.99.2.75 `startGripping()` void startGripping (BaseObject * obj) [private]

17.99.2.76 `stopGripping()` void stopGripping (bool accept = false) [private]

17.99.2.77 `toggleGrid` void toggleGrid (bool on) [slot]

17.99.2.78 `toggleLwt` void toggleLwt (bool on) [slot]

17.99.2.79 `toggleOrtho` void toggleOrtho (bool on) [slot]

17.99.2.80 `togglePolar` void togglePolar (bool on) [slot]

17.99.2.81 `toggleQSnap` void toggleQSnap (bool on) [slot]

17.99.2.82 `toggleQTrack` void toggleQTrack (bool on) [slot]

17.99.2.83 `toggleReal` void toggleReal (bool on) [slot]

17.99.2.84 `toggleRuler` void toggleRuler (bool on) [slot]

17.99.2.85 `toggleSnap` void toggleSnap (bool on) [slot]

17.99.2.86 `updateMouseCoords()` void updateMouseCoords (int x, int y) [private]

17.99.2.87 `vulcanizeObject()` void vulcanizeObject (BaseObject * obj)

17.99.2.88 `vulcanizeRubberRoom` void vulcanizeRubberRoom () [slot]

17.99.2.89 `wheelEvent()` void wheelEvent (QWheelEvent * event) [protected]

17.99.2.90 `willOverflowInt32()` bool willOverflowInt32 (qint64 a, qint64 b) [private]

17.99.2.91 `willUnderflowInt32()` bool willUnderflowInt32 (qint64 a, qint64 b) [private]

17.99.2.92 `zoomExtents` void zoomExtents () [slot]

17.99.2.93 `zoomIn` void zoomIn () [slot]

17.99.2.94 `zoomOut` void zoomOut () [slot]

17.99.2.95 `zoomSelected` void zoomSelected () [slot]

17.99.2.96 `zoomToPoint()` void zoomToPoint (const QPoint & mousePoint, int zoomDir)

17.99.2.97 zoomWindow void zoomWindow () [slot]

17.99.3 Member Data Documentation

17.99.3.1 crosshairColor QRgb crosshairColor [private]

17.99.3.2 crosshairSize quint32 crosshairSize [private]

17.99.3.3 cutCopyMousePoint QPointF cutCopyMousePoint [private]

17.99.3.4 gridColor QColor gridColor [private]

17.99.3.5 gridPath QPainterPath gridPath [private]

17.99.3.6 gripBaseObj BaseObject* gripBaseObj [private]

17.99.3.7 gripColorCool QRgb gripColorCool [private]

17.99.3.8 gripColorHot QRgb gripColorHot [private]

17.99.3.9 grippingActive bool grippingActive [private]

17.99.3.10 gripSize quint8 gripSize [private]

17.99.3.11 gscene QGraphicsScene* gscene [private]

17.99.3.12 hashDeletedObjects QHash<qint64, QGraphicsItem*> hashDeletedObjects [private]

17.99.3.13 mainWin MainWindow* mainWin [private]

17.99.3.14 movePoint QPoint movePoint [private]

17.99.3.15 movingActive bool movingActive [private]

17.99.3.16 originPath QPainterPath originPath [private]

17.99.3.17 panDistance int panDistance [private]

17.99.3.18 panningActive bool panningActive [private]

17.99.3.19 panningPointActive bool panningPointActive [private]

17.99.3.20 panningRealTimeActive bool panningRealTimeActive [private]

17.99.3.21 panStartX int panStartX [private]

17.99.3.22 panStartY int panStartY [private]

17.99.3.23 pasteDelta QPointF pasteDelta [private]

17.99.3.24 pasteObjectItemGroup QGraphicsItemGroup* pasteObjectItemGroup [private]

17.99.3.25 pastingActive bool pastingActive [private]

17.99.3.26 pickBoxSize quint8 pickBoxSize [private]

17.99.3.27 pressPoint QPoint pressPoint [private]

17.99.3.28 previewActive bool previewActive [private]

17.99.3.29 previewData EmbReal previewData [private]

17.99.3.30 previewMode int previewMode [private]

17.99.3.31 previewObjectItemGroup QGraphicsItemGroup* previewObjectItemGroup [private]

17.99.3.32 previewObjectList QList<QGraphicsItem*> previewObjectList [private]

17.99.3.33 previewPoint QPointF previewPoint [private]

17.99.3.34 qSnapActive bool qSnapActive [private]

17.99.3.35 qsnapApertureSize quint8 qsnapApertureSize [private]

17.99.3.36 qsnapLocatorColor QRgb qsnapLocatorColor [private]

17.99.3.37 qsnapLocatorSize quint8 qsnapLocatorSize [private]

17.99.3.38 qSnapToggle bool qSnapToggle [private]

17.99.3.39 rapidMoveActive bool rapidMoveActive [private]

17.99.3.40 releasePoint QPoint releasePoint [private]

17.99.3.41 rubberRoomList QList<QGraphicsItem*> rubberRoomList [private]

17.99.3.42 rulerColor QColor rulerColor [private]

17.99.3.43 rulerMetric bool rulerMetric [private]

17.99.3.44 rulerPixelSize quint8 rulerPixelSize [private]

17.99.3.45 sceneGripPoint QPointF sceneGripPoint [private]

17.99.3.46 sceneMousePoint QPointF sceneMousePoint [private]

17.99.3.47 sceneMovePoint QPointF sceneMovePoint [private]

17.99.3.48 scenePressPoint QPointF scenePressPoint [private]

17.99.3.49 sceneReleasePoint QPointF sceneReleasePoint [private]

17.99.3.50 selectBox SelectBox* selectBox [private]

17.99.3.51 selectingActive bool selectingActive [private]

17.99.3.52 spareRubberList QList<qint64> spareRubberList [private]

17.99.3.53 tempBaseObj `BaseObject*` `tempBaseObj` [private]

17.99.3.54 undoStack `QUndoStack*` `undoStack` [private]

17.99.3.55 viewMousePoint `QPoint` `viewMousePoint` [private]

17.99.3.56 zoomWindowActive `bool` `zoomWindowActive` [private]

The documentation for this class was generated from the following files:

- [embroidermodder2/embroidermodder.h](#)
- [embroidermodder2/view.cpp](#)

17.100 VipHeader_ Struct Reference

#include <embroidery_internal.h>

Public Attributes

- int `magicCode`
- int `numberOfStitches`
- int `numberOfColors`
- short `positiveXHoopSize`
- short `positiveYHoopSize`
- short `negativeXHoopSize`
- short `negativeYHoopSize`
- int `attributeOffset`
- int `xOffset`
- int `yOffset`
- unsigned char `stringVal` [8]
- short `unknown`
- int `colorLength`

17.100.1 Member Data Documentation

17.100.1.1 attributeOffset int `attributeOffset`

17.100.1.2 colorLength int `colorLength`

17.100.1.3 magicCode int `magicCode`

17.100.1.4 negativeXHoopSize short `negativeXHoopSize`

17.100.1.5 negativeYHoopSize short `negativeYHoopSize`

17.100.1.6 numberOfColors int `numberOfColors`

17.100.1.7 `numberOfStitches` `int numberOfStitches`

17.100.1.8 `positiveXHoopSize` `short positiveXHoopSize`

17.100.1.9 `positiveYHoopSize` `short positiveYHoopSize`

17.100.1.10 `stringVal` `unsigned char stringVal[8]`

17.100.1.11 `unknown` `short unknown`

17.100.1.12 `xOffset` `int xOffset`

17.100.1.13 `yOffset` `int yOffset`

The documentation for this struct was generated from the following file:

- `extern/libembroidery/src/embroidery_internal.h`

18 File Documentation

18.1 CODE_OF_CONDUCT.md File Reference

18.2 embroidermodder2/cmdprompt.cpp File Reference

```
#include "embroidermodder.h"
```

18.3 embroidermodder2/embdetails-dialog.cpp File Reference

```
#include "embroidermodder.h"
```

18.4 embroidermodder2/embroidermodder.cpp File Reference

```
#include "embroidermodder.h"
```

Functions

- static void `usage` (void)
usage
- static void `version` ()
version
- int `main` (int argc, char *argv[])
qMain

Variables

- static const char * `_appName_` = "Embroidermodder"
- static const char * `_appVer_` = "v2.0 alpha"
- static bool `exitApp` = false

18.4.1 Function Documentation

18.4.1.1 main() int main (int argc, char * argv[])
qMain

Parameters

| | |
|------|--|
| argc | |
| argv | |

Returns

18.4.1.2 usage() static void usage (void) [static]
usage

18.4.1.3 version() static void version () [static]
version

18.4.2 Variable Documentation

18.4.2.1 _appName_ const char* _appName_ = "Embroidermodder" [static]

18.4.2.2 _appVer_ const char* _appVer_ = "v2.0 alpha" [static]

18.4.2.3 exitApp bool exitApp = false [static]

18.5 embroidermodder2/embroidermodder.h File Reference

```
#include <cstdio>
#include <cstdlib>
#include <cstring>
#include <cstdint>
#include <cmath>
#include <ctime>
#include <cinttypes>
#include <vector>
#include <unordered_map>
#include <string>
#include <filesystem>
#include "embroidery.h"
#include "toml.h"
#include <QAction>
#include < QApplication>
```

```
#include <QComboBox>
#include <QContextMenuEvent>
#include <QClipboard>
#include <QDateTime>
#include <QDebug>
#include <QDialogButtonBox>
#include <QFile>
#include <QFrame>
#include <QGraphicsScene>
#include <QGraphicsPathItem>
#include <QGridLayout>
#include <QGroupBox>
#include <QLabel>
#include <QLineEdit>
#include <QList>
#include <QMainWindow>
#include <QMdiArea>
#include <QMenu>
#include <QMetaObject>
#include <QMessageBox>
#include <QObject>
#include <QPainter>
#include <QSplitter>
#include <QTextLayout>
#include <QTextStream>
#include <QTimer>
#include <QToolBar>
#include <QScrollArea>
#include <QUndoStack>
#include <QVBoxLayout>
#include <QtPrintSupport>
```

Classes

- struct [UndoHistory_](#)
- struct [UiObject_](#)

This covers the inbuilt designs: Dolphin, Snowflake and Heart. Covers Rotate, Scale and Point UI events.
- struct [EmbView_](#)
- struct [Settings_](#)

Settings System.
- class [BaseObject](#)
- class [ArcObject](#)
- class [CircleObject](#)
- class [DimLeaderObject](#)
- class [EllipseObject](#)
- class [ImageObject](#)
- class [LineObject](#)
- class [PathObject](#)
- class [PointObject](#)
- class [PolygonObject](#)
- class [PolylineObject](#)
- class [RectObject](#)
- class [SaveObject](#)
- class [TextSingleObject](#)
- class [Application](#)
- class [CmdPromptInput](#)

- class [CmdPromptHistory](#)
The Command Prompt History class.
- class [CmdPromptSplitter](#)
- class [CmdPromptHandle](#)
- class [CmdPrompt](#)
- class [EmbDetailsDialog](#)
- class [ImageWidget](#)
- class [LayerManager](#)
- class [MainWindow](#)
The MainWindow class.
- class [MdiWindow](#)
- class [MdiArea](#)
- class [PreviewDialog](#)
- class [PropertyEditor](#)
- class [SelectBox](#)
- class [Settings_Dialog](#)
- class [StatusBar](#)
- class [StatusBarButton](#)
- class [UndoEditor](#)
- class [UndoableAddCommand](#)
- class [UndoableDeleteCommand](#)
- class [UndoableMoveCommand](#)
- class [UndoableRotateCommand](#)
- class [UndoableScaleCommand](#)
- class [UndoableNavCommand](#)
- class [UndoableGripEditCommand](#)
- class [UndoableMirrorCommand](#)
- class [View](#)
- struct [Action__](#)

Typedefs

- typedef std::unordered_map< std::string, std::string > [Dictionary](#)
- typedef struct [UndoHistory_ UndoHistory](#)
- typedef std::unordered_map< std::string, [Dictionary](#) > [Index](#)
- typedef struct [UiObject_ UiObject](#)
This covers the inbuilt designs: Dolphin, Snowflake and Heart. Covers Rotate, Scale and Point UI events.
- typedef struct [EmbView_ EmbView](#)
- typedef struct [Settings_ Settings](#)
Settings System.
- typedef struct [Action__ Action](#)

Enumerations

- enum [UiMode](#) {
 [DEFAULT_MODE](#) , [CIRCLE_MODE_1P_RAD](#) , [CIRCLE_MODE_1P_DIA](#) , [CIRCLE_MODE_2P](#) ,
 [CIRCLE_MODE_3P](#) , [CIRCLE_MODE_TTR](#) , [ELLIPSE_MODE_MAJORDIAMETER_MINORRADIUS](#) ,
 [ELLIPSE_MODE_MAJORRADIUS_MINORRADIUS](#) ,
 [ELLIPSE_MODE_ELLIPSE_ROTATION](#) , [DOLPHIN_MODE_NUM_POINTS](#) , [DOLPHIN_MODE_XSCALE](#) ,
 [DOLPHIN_MODE_YSCALE](#) ,
 [HEART_MODE_NUM_POINTS](#) , [HEART_MODE_STYLE](#) , [HEART_MODE_XSCALE](#) , [HEART_MODE_YSCALE](#) ,
 [ROTATE_MODE_NORMAL](#) , [ROTATE_MODE_REFERENCE](#) , [SCALE_MODE_NORMAL](#) , [SCALE_MODE_REFERENCE](#) ,
 [SINGLE_LINE_TEXT_MODE_JUSTIFY](#) , [SINGLE_LINE_TEXT_MODE_SETFONT](#) , [SINGLE_LINE_TEXT_MODE_SETGEOM](#)

```

, SINGLE_LINE_TEXT_MODE_RAPID ,
STAR_MODE_NUM_POINTS , STAR_MODE_CENTER_PT , STAR_MODE_RAD_OUTER , STAR_MODE_RAD_INNER
,
SNOWFLAKE_MODE_NUM_POINTS , SNOWFLAKE_MODE_XSCALE , SNOWFLAKE_MODE_YSCALE }

• enum OBJ_KEYS {
OBJ_TYPE = 0 , OBJ_NAME = 1 , OBJ_LAYER = 2 , OBJ_COLOR = 3 ,
OBJ_LTYPE = 4 , OBJ_LWT = 5 , OBJ_RUBBER = 6 }

• enum OBJ_TYPE_VALUES {
OBJ_TYPE_NULL = 0 , OBJ_TYPE_BASE = 100000 , OBJ_TYPE_ARC = 100001 , OBJ_TYPE_BLOCK =
100002 ,
OBJ_TYPE_CIRCLE = 100003 , OBJ_TYPE_DIMALIGNED = 100004 , OBJ_TYPE_DIMANGULAR =
100005 , OBJ_TYPE_DIMARCLENGTH = 100006 ,
OBJ_TYPE_DIMDIAMETER = 100007 , OBJ_TYPE_DIMLEADER = 100008 , OBJ_TYPE_DIMLINEAR =
100009 , OBJ_TYPE_DIMORDINATE = 100010 ,
OBJ_TYPE_DIMRADIUS = 100011 , OBJ_TYPE_ELLIPSE = 100012 , OBJ_TYPE_ELLIPSEARC = 100013
, OBJ_TYPE_RUBBER = 100014 ,
OBJ_TYPE_GRID = 100015 , OBJ_TYPE_HATCH = 100016 , OBJ_TYPE_IMAGE = 100017 ,
OBJ_TYPE_INFINITELINE = 100018 ,
OBJ_TYPE_LINE = 100019 , OBJ_TYPE_PATH = 100020 , OBJ_TYPE_POINT = 100021 , OBJ_TYPE_POLYGON =
100022 ,
OBJ_TYPE_POLYLINE = 100023 , OBJ_TYPE_RAY = 100024 , OBJ_TYPE_RECTANGLE = 100025 ,
OBJ_TYPE_SLOT = 100026 ,
OBJ_TYPE_SPLINE = 100027 , OBJ_TYPE_TEXTMULTI = 100028 , OBJ_TYPE_TEXTSINGLE = 100029 }

• enum OBJ_LTYPE_VALUES {
OBJ_LTYPE_CONT = 0 , OBJ_LTYPE_CENTER = 1 , OBJ_LTYPE_DOT = 2 , OBJ_LTYPE_HIDDEN = 3 ,
OBJ_LTYPE_PHANTOM = 4 , OBJ_LTYPE_ZIGZAG = 5 , OBJ_LTYPE_RUNNING = 6 , OBJ_LTYPE_SATIN =
7 ,
OBJ_LTYPE_FISHBONE = 8 }

• enum OBJ_LWT_VALUES {
OBJ_LWT_BYLAYER = -2 , OBJ_LWT_BYBLOCK = -1 , OBJ_LWT_DEFAULT = 0 , OBJ_LWT_01 = 1 ,
OBJ_LWT_02 = 2 , OBJ_LWT_03 = 3 , OBJ_LWT_04 = 4 , OBJ_LWT_05 = 5 ,
OBJ_LWT_06 = 6 , OBJ_LWT_07 = 7 , OBJ_LWT_08 = 8 , OBJ_LWT_09 = 9 ,
OBJ_LWT_10 = 10 , OBJ_LWT_11 = 11 , OBJ_LWT_12 = 12 , OBJ_LWT_13 = 13 ,
OBJ_LWT_14 = 14 , OBJ_LWT_15 = 15 , OBJ_LWT_16 = 16 , OBJ_LWT_17 = 17 ,
OBJ_LWT_18 = 18 , OBJ_LWT_19 = 19 , OBJ_LWT_20 = 20 , OBJ_LWT_21 = 21 ,
OBJ_LWT_22 = 22 , OBJ_LWT_23 = 23 , OBJ_LWT_24 = 24 }

• enum OBJ_SNAP_VALUES {
OBJ_SNAP_NULL = 0 , OBJ_SNAP_ENDPOINT = 1 , OBJ_SNAP_MIDPOINT = 2 , OBJ_SNAP_CENTER =
3 ,
OBJ_SNAP_NODE = 4 , OBJ_SNAP_QUADRANT = 5 , OBJ_SNAP_INTERSECTION = 6 , OBJ_SNAP_EXTENSION =
7 ,
OBJ_SNAP_INSERTION = 8 , OBJ_SNAP_PERPENDICULAR = 9 , OBJ_SNAP_TANGENT = 10 ,
OBJ_SNAP_NEAREST = 11 ,
OBJ_SNAP_APPINTERSECTION = 12 , OBJ_SNAP_PARALLEL = 13 }

• enum OBJ_RUBBER_VALUES {
OBJ_RUBBER_OFF = 0 , OBJ_RUBBER_ON = 1 , OBJ_RUBBER_CIRCLE_1P_RAD , OBJ_RUBBER_CIRCLE_1P_DIA
,
OBJ_RUBBER_CIRCLE_2P , OBJ_RUBBER_CIRCLE_3P , OBJ_RUBBER_CIRCLE_TTR , OBJ_RUBBER_CIRCLE_TTT
,
OBJ_RUBBER_DIMLEADER_LINE , OBJ_RUBBER_ELLIPSE_LINE , OBJ_RUBBER_ELLIPSE_MAJOR_DIAMETER_MINOR
, OBJ_RUBBER_ELLIPSE_MAJOR_RADIUS_MINOR_RADIUS ,
OBJ_RUBBER_ELLIPSE_ROTATION , OBJ_RUBBER_GRIP , OBJ_RUBBER_LINE , OBJ_RUBBER_POLYGON
,
OBJ_RUBBER_POLYGON_INSCRIBE , OBJ_RUBBER_POLYGON_CIRCUMSCRIBE , OBJ_RUBBER_POLYLINE
, OBJ_RUBBER_IMAGE ,
OBJ_RUBBER_RECTANGLE , OBJ_RUBBER_TEXTSINGLE }

• enum SPARE_RUBBER_VALUES { SPARE_RUBBER_OFF = 0 , SPARE_RUBBER_PATH , SPARE_RUBBER_POLYGON
, SPARE_RUBBER_POLYLINE }

```

- enum `PREVIEW_CLONE_VALUES` { `PREVIEW_CLONE_NULL` = 0 , `PREVIEW_CLONE_SELECTED` , `PREVIEW_CLONE_RUBBER` }
- enum `PREVIEW_MODE_VALUES` { `PREVIEW_MODE_NULL` = 0 , `PREVIEW_MODE_MOVE` , `PREVIEW_MODE_ROTATE` , `PREVIEW_MODE_SCALE` }

Functions

- int `read_settings` (const char *`settings_file`)

Read the settings from file which aren't editable by the user. These files need to be placed in the install folder.
- void `write_settings` (const char *`fname`)
- `QPointF to_QPointF` (`EmbVector` a)
- `EmbVector to_EmbVector` (`QPointF` a)
- `EmbVector operator+` (`EmbVector` a, `EmbVector` b)
- `EmbVector operator-` (`EmbVector` a, `EmbVector` b)
- `EmbReal radians` (`EmbReal degrees`)
- `EmbReal degrees` (`EmbReal radian`)
- `MainWindow * mainWin` ()

mainWin
- int `get_action_index` (std::string cmd)

Variables

- static const `EmbReal emb_constant_pi` = 3.14159265358979323846
- `Settings settings`
- `Settings dialog`
- std::vector< `Action` > `action_table`
- std::vector< std::string > `file_toolbar`
- std::vector< std::string > `edit_toolbar`
- std::vector< std::string > `view_toolbar`
- std::vector< std::string > `zoom_toolbar`

18.5.1 Detailed Description

The only header for the GUI part: a good overview of this source code.

18.5.2 Typedef Documentation

18.5.2.1 Action `typedef struct Action__ Action`

18.5.2.2 Dictionary `typedef std::unordered_map<std::string, std::string> Dictionary`

18.5.2.3 EmbView `typedef struct EmbView_ EmbView`

18.5.3 EmbViews

The EmbView describes how the render is displayed.

18.5.3.1 Index `typedef std::unordered_map<std::string, Dictionary> Index`

Todo document this.

18.5.3.2 Settings `typedef struct Settings_ Settings`
Settings System.

Rather than pollute the global namespace, we collect together all the global settings into a structure that stores them. This also allows us to create a complete copy of the settings for the purpose of restoring them if the user cancels out of the Settings Dialog.

Like all of our structs, it's C99 compliant.

18.5.3.3 UiObject `typedef struct UiObject_ UiObject`

This covers the inbuilt designs: Dolphin, Snowflake and Heart. Covers Rotate, Scale and Point UI events.
scale is how radii are stored if the object is a circle, or the semi-major and semi-minor axes if the object is an ellipse.
center perhaps should be the "anchor" instead which is the center for some objects and top left for rectangular objects.

Do angles need special storage? angleRef, angleNew

This chapter details how the source code achieves the design detailed in the previous chapter. For the low-level details, read the later chapters.

Dear ImGui (CITATION NEEDED)

18.5.3.4 Geometry Objects At all times the EmbPattern has all of the information about the pattern however, editing information like the rubber text labels needs to be stored during runtime. Also editing ghosts like when a rotate command is half executed.

To deal with this we have a generic object that can act as any other object that updates the associated pattern struct as changes are made.

Note that the editor state is separate from this since that is per view, not per object.

Selecting is done via this interface.

18.5.3.5 UndoHistory `typedef struct UndoHistory_ UndoHistory`

Todo document this.

18.5.4 Enumeration Type Documentation**18.5.4.1 OBJ_KEYS** `enum OBJ_KEYS`

Enumerator

| | |
|------------|--|
| OBJ_TYPE | |
| OBJ_NAME | |
| OBJ_LAYER | |
| OBJ_COLOR | |
| OBJ_LTYPE | |
| OBJ_LWT | |
| OBJ_RUBBER | |

18.5.4.2 OBJ_LTYPE_VALUES `enum OBJ_LTYPE_VALUES`

Enumerator

| | |
|-------------------|--|
| OBJ_LTYPE_CONT | |
| OBJ_LTYPE_CENTER | |
| OBJ_LTYPE_DOT | |
| OBJ_LTYPE_HIDDEN | |
| OBJ_LTYPE_PHANTOM | |

Enumerator

| | |
|--------------------|--|
| OBJ_LTYPE_ZIGZAG | |
| OBJ_LTYPE_RUNNING | |
| OBJ_LTYPE_SATIN | |
| OBJ_LTYPE_FISHBONE | |

18.5.4.3 OBJ_LWT_VALUES enum [OBJ_LWT_VALUES](#)

Enumerator

| | |
|-----------------|--|
| OBJ_LWT_BYLAYER | |
| OBJ_LWT_BYBLOCK | |
| OBJ_LWT_DEFAULT | |
| OBJ_LWT_01 | |
| OBJ_LWT_02 | |
| OBJ_LWT_03 | |
| OBJ_LWT_04 | |
| OBJ_LWT_05 | |
| OBJ_LWT_06 | |
| OBJ_LWT_07 | |
| OBJ_LWT_08 | |
| OBJ_LWT_09 | |
| OBJ_LWT_10 | |
| OBJ_LWT_11 | |
| OBJ_LWT_12 | |
| OBJ_LWT_13 | |
| OBJ_LWT_14 | |
| OBJ_LWT_15 | |
| OBJ_LWT_16 | |
| OBJ_LWT_17 | |
| OBJ_LWT_18 | |
| OBJ_LWT_19 | |
| OBJ_LWT_20 | |
| OBJ_LWT_21 | |
| OBJ_LWT_22 | |
| OBJ_LWT_23 | |
| OBJ_LWT_24 | |

18.5.4.4 OBJ_RUBBER_VALUES enum [OBJ_RUBBER_VALUES](#)

Enumerator

| | |
|--------------------------|---|
| OBJ_RUBBER_OFF | |
| OBJ_RUBBER_ON | |
| OBJ_RUBBER_CIRCLE_1P_RAD | For the circle object currently focussed, show two rubber points: one for the centre (the anchor) and the other at some point on the radius to adjust the radius. |

Enumerator

| | |
|--|---|
| OBJ_RUBBER_CIRCLE_1P_DIA | For the circle object currently focussed, show two rubber points: one for the left of the diameter and one for the right. These rubber points can be moved around the circle, but they always oppose one another. |
| OBJ_RUBBER_CIRCLE_2P | |
| OBJ_RUBBER_CIRCLE_3P | |
| OBJ_RUBBER_CIRCLE_TTR | |
| OBJ_RUBBER_CIRCLE_TTT | |
| OBJ_RUBBER_DIMLEADER_LINE | |
| OBJ_RUBBER_ELLIPSE_LINE | |
| OBJ_RUBBER_ELLIPSE_MAJORDIAMETER ↔ MINORRADIUS | |
| OBJ_RUBBER_ELLIPSE_MAJORRADIUS ↔ MINORRADIUS | |
| OBJ_RUBBER_ELLIPSE_ROTATION | |
| OBJ_RUBBER_GRIP | |
| OBJ_RUBBER_LINE | |
| OBJ_RUBBER_POLYGON | |
| OBJ_RUBBER_POLYGON_INSCRIBE | |
| OBJ_RUBBER_POLYGON_CIRCUMSCRIBE | |
| OBJ_RUBBER_POLYLINE | |
| OBJ_RUBBER_IMAGE | |
| OBJ_RUBBER_RECTANGLE | |
| OBJ_RUBBER_TEXTSINGLE | |

18.5.4.5 OBJ_SNAP_VALUES enum [OBJ_SNAP_VALUES](#)

Enumerator

| | |
|--------------------------|--|
| OBJ_SNAP_NULL | |
| OBJ_SNAP_ENDPOINT | |
| OBJ_SNAP_MIDPOINT | |
| OBJ_SNAP_CENTER | |
| OBJ_SNAP_NODE | |
| OBJ_SNAP_QUADRANT | |
| OBJ_SNAP_INTERSECTION | |
| OBJ_SNAP_EXTENSION | |
| OBJ_SNAP_INSERTION | |
| OBJ_SNAP_PERPENDICULAR | |
| OBJ_SNAP_TANGENT | |
| OBJ_SNAP_NEAREST | |
| OBJ_SNAP_APPINTERSECTION | |
| OBJ_SNAP_PARALLEL | |

18.5.4.6 OBJ_TYPE_VALUES enum [OBJ_TYPE_VALUES](#)

Enumerator

| |
|-----------------------|
| OBJ_TYPE_NULL |
| OBJ_TYPE_BASE |
| OBJ_TYPE_ARC |
| OBJ_TYPE_BLOCK |
| OBJ_TYPE_CIRCLE |
| OBJ_TYPE_DIMALIGNED |
| OBJ_TYPE_DIMANGULAR |
| OBJ_TYPE_DIMARCLENGTH |
| OBJ_TYPE_DIMDIAMETER |
| OBJ_TYPE_DIMLEADER |
| OBJ_TYPE_DIMLINEAR |
| OBJ_TYPE_DIMORDINATE |
| OBJ_TYPE_DIMRADIUS |
| OBJ_TYPE_ELLIPSE |
| OBJ_TYPE_ELLIPSEARC |
| OBJ_TYPE_RUBBER |
| OBJ_TYPE_GRID |
| OBJ_TYPE_HATCH |
| OBJ_TYPE_IMAGE |
| OBJ_TYPE_INFINITELINE |
| OBJ_TYPE_LINE |
| OBJ_TYPE_PATH |
| OBJ_TYPE_POINT |
| OBJ_TYPE_POLYGON |
| OBJ_TYPE_POLYLINE |
| OBJ_TYPE_RAY |
| OBJ_TYPE_RECTANGLE |
| OBJ_TYPE_SLOT |
| OBJ_TYPE_SPLINE |
| OBJ_TYPE_TEXTMULTI |
| OBJ_TYPE_TEXTSINGLE |

18.5.4.7 PREVIEW_CLONE_VALUES enum [PREVIEW_CLONE_VALUES](#)

Enumerator

| |
|------------------------|
| PREVIEW_CLONE_NULL |
| PREVIEW_CLONE_SELECTED |
| PREVIEW_CLONE_RUBBER |

18.5.4.8 PREVIEW_MODE_VALUES enum [PREVIEW_MODE_VALUES](#)

Enumerator

| |
|---------------------|
| PREVIEW_MODE_NULL |
| PREVIEW_MODE_MOVE |
| PREVIEW_MODE_ROTATE |

Enumerator

| | |
|--------------------|--------------------------|
| PREVIEW_MODE_SCALE | <input type="checkbox"/> |
|--------------------|--------------------------|

18.5.4.9 SPARE_RUBBER_VALUES enum [SPARE_RUBBER_VALUES](#)

Enumerator

| | |
|-----------------------|--------------------------|
| SPARE_RUBBER_OFF | <input type="checkbox"/> |
| SPARE_RUBBER_PATH | <input type="checkbox"/> |
| SPARE_RUBBER_POLYGON | <input type="checkbox"/> |
| SPARE_RUBBER_POLYLINE | <input type="checkbox"/> |

18.5.4.10 UiMode enum [UiMode](#)

Enumerator

| | |
|--|--------------------------|
| DEFAULT_MODE | <input type="checkbox"/> |
| CIRCLE_MODE_1P_RAD | <input type="checkbox"/> |
| CIRCLE_MODE_1P_DIA | <input type="checkbox"/> |
| CIRCLE_MODE_2P | <input type="checkbox"/> |
| CIRCLE_MODE_3P | <input type="checkbox"/> |
| CIRCLE_MODE_TTR | <input type="checkbox"/> |
| ELLIPSE_MODE_MAJORDIAMETER_MINORRADIUS | <input type="checkbox"/> |
| ELLIPSE_MODE_MAJORRADIUS_MINORRADIUS | <input type="checkbox"/> |
| ELLIPSE_MODE_ELLIPSE_ROTATION | <input type="checkbox"/> |
| DOLPHIN_MODE_NUM_POINTS | <input type="checkbox"/> |
| DOLPHIN_MODE_XSCALE | <input type="checkbox"/> |
| DOLPHIN_MODE_YSCALE | <input type="checkbox"/> |
| HEART_MODE_NUM_POINTS | <input type="checkbox"/> |
| HEART_MODE_STYLE | <input type="checkbox"/> |
| HEART_MODE_XSCALE | <input type="checkbox"/> |
| HEART_MODE_YSCALE | <input type="checkbox"/> |
| ROTATE_MODE_NORMAL | <input type="checkbox"/> |
| ROTATE_MODE_REFERENCE | <input type="checkbox"/> |
| SCALE_MODE_NORMAL | <input type="checkbox"/> |
| SCALE_MODE_REFERENCE | <input type="checkbox"/> |
| SINGLE_LINE_TEXT_MODE_JUSTIFY | <input type="checkbox"/> |
| SINGLE_LINE_TEXT_MODE_SETFONT | <input type="checkbox"/> |
| SINGLE_LINE_TEXT_MODE_SETGEOM | <input type="checkbox"/> |
| SINGLE_LINE_TEXT_MODE_RAPID | <input type="checkbox"/> |
| STAR_MODE_NUM_POINTS | <input type="checkbox"/> |
| STAR_MODE_CENTER_PT | <input type="checkbox"/> |
| STAR_MODE_RAD_OUTER | <input type="checkbox"/> |
| STAR_MODE_RAD_INNER | <input type="checkbox"/> |
| SNOWFLAKE_MODE_NUM_POINTS | <input type="checkbox"/> |
| SNOWFLAKE_MODE_XSCALE | <input type="checkbox"/> |
| SNOWFLAKE_MODE_YSCALE | <input type="checkbox"/> |

18.5.5 Function Documentation

18.5.5.1 degrees() `EmbReal degrees (`
`EmbReal radian) [inline]`

18.5.5.2 get_action_index() `int get_action_index (`
`std::string cmd)`

18.5.5.3 mainWin() `MainWindow * mainWin ()`
mainWin

Returns

18.5.5.4 operator+() `EmbVector operator+ (`
`EmbVector a,`
`EmbVector b) [inline]`

18.5.5.5 operator-() `EmbVector operator- (`
`EmbVector a,`
`EmbVector b) [inline]`

18.5.5.6 radians() `EmbReal radians (`
`EmbReal degrees) [inline]`

18.5.5.7 read_settings() `int read_settings (`
`const char * settings_file)`

Read the settings from file which aren't editable by the user. These files need to be placed in the install folder.

18.5.5.8 to_EmbVector() `EmbVector to_EmbVector (`
`QPointF a) [inline]`

18.5.5.9 to_QPointF() `QPointF to_QPointF (`
`EmbVector a) [inline]`

18.5.5.10 write_settings() `void write_settings (`
`const char * fname)`

18.5.6 Variable Documentation

18.5.6.1 action_table `std::vector<Action> action_table [extern]`

18.5.6.2 dialog `Settings` `dialog` [extern]

These copies of the settings struct are for restoring the state if the user doesn't want to accept their changes in the settings dialog.

18.5.6.3 edit_toolbar `std::vector<std::string>` `edit_toolbar` [extern]**18.5.6.4 emb_constant_pi** `const EmbReal` `emb_constant_pi = 3.14159265358979323846` [static]**18.5.6.5 file_toolbar** `std::vector<std::string>` `file_toolbar` [extern]**18.5.6.6 settings** `Settings` `settings` [extern]

The actuator changes the program state via these global variables.

18.5.6.7 view_toolbar `std::vector<std::string>` `view_toolbar` [extern]**18.5.6.8 zoom_toolbar** `std::vector<std::string>` `zoom_toolbar` [extern]

18.6 embroidermodder.h

[Go to the documentation of this file.](#)

```
00001 /*
00002 *   Embroidermodder 2.
00003 *
00004 * -----
00005 *
00006 * Copyright 2013-2023 The Embroidermodder Team
00007 * Embroidermodder 2 is Open Source Software.
00008 * See LICENSE for licensing terms.
00009 *
00010 * -----
00011 *
00012 * Use Python's PEP7 style guide.
00013 *     https://peps.python.org/pep-0007/
00014 */
00015
00021 #ifndef __EMBROIDERMODDER.Utility_H__
00022 #define __EMBROIDERMODDER.Utility_H__
00023
00024 #include <cstdio>
00025 #include <cstdlib>
00026 #include <cstring>
00027 #include <cstdint>
00028 #include <cmath>
00029 #include <ctime>
00030 #include <cinttypes>
00031
00032 #include <vector>
00033 #include <unordered_map>
00034 #include <string>
00035 #include <filesystem>
00036
00037 #include "embroidery.h"
00038 #include "toml.h"
00039
00040 #include <QAction>
00041 #include <QApplication>
00042 #include <QComboBox>
00043 #include <QContextMenuEvent>
00044 #include <QClipboard>
00045 #include <QDateTime>
00046 #include <QDebug>
00047 #include <QDialogButtonBox>
00048 #include <QFile>
00049 #include <QFrame>
00050 #include <QGraphicsScene>
00051 #include <QGraphicsPathItem>
00052 #include <QGridLayout>
00053 #include <QGroupBox>
00054 #include <QLabel>
```

```
00055 #include <QLineEdit>
00056 #include <QList>
00057 #include <QMainWindow>
00058 #include <QMdiArea>
00059 #include <QMenu>
00060 #include <QMetaObject>
00061 #include <QMessageBox>
00062 #include <QObject>
00063 #include <QPainter>
00064 #include <QSplitter>
00065 #include <QTextLayout>
00066 #include <QTextStream>
00067 #include <QTimer>
00068 #include <QToolBar>
00069 #include <QScrollArea>
00070 #include <QUndoStack>
00071 #include <QVBoxLayout>
00072
00073 #include <QtPrintSupport>
00074
00075 class ImageWidget;
00076 class MdiArea;
00077 class MdiWindow;
00078 class View;
00079 class StatusBar;
00080 class StatusBarButton;
00081 class CmdPrompt;
00082 class PropertyEditor;
00083 class UndoEditor;
00084 class MainWindow;
00085
00086 class BaseObject;
00087 class SelectBox;
00088 class ArcObject;
00089 class BlockObject;
00090 class CircleObject;
00091 class DimAlignedObject;
00092 class DimAngularObject;
00093 class DimArcLengthObject;
00094 class DimDiameterObject;
00095 class DimLeaderObject;
00096 class DimLinearObject;
00097 class DimOrdinateObject;
00098 class DimRadiusObject;
00099 class EllipseObject;
00100 class EllipseArcObject;
00101 class HatchObject;
00102 class ImageObject;
00103 class InfiniteLineObject;
00104 class LineObject;
00105 class PathObject;
00106 class PointObject;
00107 class PolygonObject;
00108 class PolylineObject;
00109 class RayObject;
00110 class RectObject;
00111 class SplineObject;
00112 class TextMultiObject;
00113 class TextSingleObject;
00114
00119 typedef std::unordered_map<std::string, std::string> Dictionary;
00120
00124 typedef struct UndoHistory_ {
00125     std::vector<std::string> data; /*< \todo document this */
00126     int position; /*< \todo document this */
00127 } UndoHistory;
00128
00132 typedef std::unordered_map<std::string, Dictionary> Index;
00133
00166 typedef struct UiObject_ {
00167     char fname[200]; /*< \todo document this */
00168     char command[200]; /*< \todo document this */
00169     bool firstRun; /*< If this UiObject has been put through the
00170                     user interaction processor. */
00171     EmbVector controlPoints[10]; /*< Storage for however many Rubber Points the
00172                               design needs. */
00173     char controlPointLabels[10][200]; /*< Storage for the labels for the Rubber Points
00174                                         using the same indexing. */
00175     int n_controlPoints; /*< The number of entries in the controlPoints
00176                           and controlPointLabels. */
00177     int numPoints; /*< The number of points if we consider the object as a Polygon. */
00178     int minPoints; /*< The minimum number of points needed to make the
00179                   polygon look somewhat like the design. */
00180     int maxPoints; /*< The maximum number of points before adding more will
00181                   do nothing but slow down the program. */
00182     EmbVector center; /*< Where the polygon is centered. */
00183     EmbVector scale; /*< The scale of the object: note that the default
```

```

00184             may not be (1.0, 1.0). */
00185     EmbReal rotation; /*< \todo document this */
00186     uint32_t mode; /*< The mode argument records what kind of design we are
00187                 using and how to interact with it. */
00188     char path_desc[1000]; /*< The SVG style path spec. */
00189     char text[200]; /*< The text to be rendered to the scene. */
00190     int textJustify; /*< One of the JUSTIFY_* constants representing what kind
00191                 of alignment to use. */
00192     char textFont[200]; /*< The file name of the font to use. */
00193     EmbReal textHeight; /*< The text height. */
00194     EmbReal textRotation; /*< The rotation of the text in the scene. */
00195     GLuint texture_id; /*< Pointer to a texture that may be rendered to the object. */
00196     char id[200]; /*< \todo document this */
00197     int pattern_index; /*< \todo document this */
00198     char type[200]; /*< \todo document this */
00199     int object_index; /*< \todo document this */
00200     bool selectable; /*< \todo document this */
00201     EmbColor color; /*< \todo document this */
00202 } UiObject;
00203
00211 typedef struct EmbView_ {
00212     EmbPattern *pattern; /*< \todo document this */
00213     EmbVector origin; /*< \todo document this */
00214     EmbReal scale; /*< \todo document this */
00215     char grid_type[200]; /*< \todo document this */
00216     int ui_mode; /*< \todo document this */
00217     bool snap_mode; /*< \todo document this */
00218     bool grid_mode; /*< \todo document this */
00219     bool ruler_mode; /*< \todo document this */
00220     bool ortho_mode; /*< \todo document this */
00221     bool polar_mode; /*< \todo document this */
00222     bool qsnap_mode; /*< \todo document this */
00223     bool qtrack_mode; /*< \todo document this */
00224     bool lwt_mode; /*< \todo document this */
00225     bool real_render; /*< \todo document this */
00226     bool metric; /*< \todo document this */
00227     bool simulate; /*< \todo document this */
00228     clock_t simulation_start; /*< \todo document this */
00229     char text_font[200]; /*< \todo document this */
00230     EmbReal text_size; /*< \todo document this */
00231     EmbReal text_angle; /*< \todo document this */
00232     bool text_style_bold; /*< \todo document this */
00233     bool text_style_italic; /*< \todo document this */
00234     bool text_style_underline; /*< \todo document this */
00235     bool text_style_overline; /*< \todo document this */
00236     bool text_style_strikethrough; /*< \todo document this */
00237     char filename[200]; /*< \todo document this */
00238     UndoHistory undo_history; /*< \todo document this */
00239     int selected[100]; /*< \todo document this */
00240     int n_selected; /*< \todo document this */
00241     int rubber_mode; /*< . */
00242 } EmbView;
00243
00254 typedef struct Settings_ {
00255     char version[200]; /*< \todo document this */
00256     bool running; /*< \todo document this */
00257     bool testing; /*< \todo document this */
00258     int debug_mode; /*< \todo document this */
00259     bool show_about_dialog; /*< \todo document this */
00260     bool show_settings_editor; /*< \todo document this */
00261     bool show_editor; /*< \todo document this */
00262     bool show_details_dialog; /*< \todo document this */
00263     bool show_open_file_dialog; /*< \todo document this */
00264     int icon_size; /*< \todo document this */
00265     char icon_theme[200]; /*< \todo document this */
00266     int pattern_index; /*< \todo document this */
00267     char assets_dir[200]; /*< \todo document this */
00268     bool use_translation; /*< \todo document this */
00269     char language[200]; /*< \todo document this */
00270     bool mdi_bg_use_logo; /*< \todo document this */
00271     bool mdi_bg_use_texture; /*< \todo document this */
00272     bool mdi_bg_use_color; /*< \todo document this */
00273     char general_mdi_bg_logo[200]; /*< \todo document this */
00274     char general_mdi_bg_texture[200]; /*< \todo document this */
00275     uint32_t general_mdi_bg_color; /*< \todo document this */
00276     bool tip_of_the_day; /*< \todo document this */
00277     uint32_t general_current_tip; /*< \todo document this */
00278     bool general_system_help_browser; /*< \todo document this */
00279     bool general_check_for_updates; /*< \todo document this */
00280     bool display_use_opengl; /*< \todo document this */
00281     bool display_renderhint_aa; /*< \todo document this */
00282     bool display_renderhint_text_aa; /*< \todo document this */
00283     bool display_renderhint_smooth_pix; /*< \todo document this */
00284     bool display_renderhint_high_aa; /*< \todo document this */
00285     bool display_renderhint_noncosmetic; /*< \todo document this */
00286     bool display_show_scrollbars; /*< \todo document this */
00287     int display_scrollbar_widget_num; /*< \todo document this */

```

```

00288     uint32_t display_crosshair_color; /*< \todo document this */
00289     uint32_t display_bg_color; /*< \todo document this */
00290     uint32_t display_selectbox_left_color; /*< \todo document this */
00291     uint32_t display_selectbox_left_fill; /*< \todo document this */
00292     uint32_t display_selectbox_right_color; /*< \todo document this */
00293     uint32_t display_selectbox_right_fill; /*< \todo document this */
00294     uint8_t display_selectbox_alpha; /*< \todo document this */
00295     EmbReal display_zoomscale_in; /*< \todo document this */
00296     EmbReal display_zoomscale_out; /*< \todo document this */
00297     uint8_t display_crosshair_percent; /*< \todo document this */
00298     std::string display_units; /*< \todo document this */
00299     std::string opensave_custom_filter; /*< \todo document this */
00300     std::string opensave_open_format; /*< \todo document this */
00301     bool opensave_open_thumbnail; /*< \todo document this */
00302     std::string opensave_save_format; /*< \todo document this */
00303     bool opensave_save_thumbnail; /*< \todo document this */
00304     uint8_t opensave_recent_max_files; /*< \todo document this */
00305     std::vector<std::string> opensave_recent_list_of_files; /*< \todo document this */
00306     std::string opensave_recent_directory; /*< \todo document this */
00307     uint8_t opensave_trim_dst_num_jumps; /*< \todo document this */
00308     std::string printing_default_device; /*< \todo document this */
00309     bool printing_use_last_device; /*< \todo document this */
00310     bool printing_disable_bg; /*< \todo document this */
00311     bool grid_show_on_load; /*< \todo document this */
00312     bool grid_show_origin; /*< \todo document this */
00313     bool grid_color_match_crosshair; /*< \todo document this */
00314     uint32_t grid_color; /*< \todo document this */
00315     bool grid_load_from_file; /*< \todo document this */
00316     std::string grid_type; /*< \todo document this */
00317     bool grid_center_on_origin; /*< \todo document this */
00318     EmbVector grid_center; /*< \todo document this */
00319     EmbVector grid_size; /*< \todo document this */
00320     EmbVector grid_spacing; /*< \todo document this */
00321     EmbReal grid_size_radius; /*< \todo document this */
00322     EmbReal grid_spacing_radius; /*< \todo document this */
00323     EmbReal grid_spacing_angle; /*< \todo document this */
00324     bool ruler_show_on_load; /*< \todo document this */
00325     bool ruler_metric; /*< \todo document this */
00326     uint32_t ruler_color; /*< \todo document this */
00327     uint8_t ruler_pixel_size; /*< \todo document this */
00328     bool qsnap_enabled; /*< \todo document this */
00329     uint32_t qsnap_locator_color; /*< \todo document this */
00330     uint8_t qsnap_locator_size; /*< \todo document this */
00331     uint8_t qsnap_aperture_size; /*< \todo document this */
00332     bool qsnap_endpoint; /*< \todo document this */
00333     bool qsnap_midpoint; /*< \todo document this */
00334     bool qsnap_center; /*< \todo document this */
00335     bool qsnap_node; /*< \todo document this */
00336     bool qsnap_quadrant; /*< \todo document this */
00337     bool qsnap_intersection; /*< \todo document this */
00338     bool qsnap_extension; /*< \todo document this */
00339     bool qsnap_insertion; /*< \todo document this */
00340     bool qsnap_perpendicular; /*< \todo document this */
00341     bool qsnap_tangent; /*< \todo document this */
00342     bool qsnap_nearest; /*< \todo document this */
00343     bool qsnap_apparent; /*< \todo document this */
00344     bool qsnap_parallel; /*< \todo document this */
00345     bool lwt_show_lwt; /*< \todo document this */
00346     bool lwt_real_render; /*< \todo document this */
00347     bool shift_hold; /*< \todo document this */
00348     EmbReal lwt_default_lwt; /*< \todo document this */
00349     bool selection_mode_pickfirst; /*< \todo document this */
00350     bool selection_mode_pickadd; /*< \todo document this */
00351     bool selection_mode_pickdrag; /*< \todo document this */
00352     uint32_t selection_coolgrip_color; /*< \todo document this */
00353     uint32_t selection_hotgrip_color; /*< \todo document this */
00354     uint8_t selection_grip_size; /*< \todo document this */
00355     uint8_t selection_pickbox_size; /*< \todo document this */
00356     char text_font[200]; /*< \todo document this */
00357     EmbReal text_size; /*< \todo document this */
00358     EmbReal text_angle; /*< \todo document this */
00359     bool text_style_bold; /*< \todo document this */
00360     bool text_style_italic; /*< \todo document this */
00361     bool text_style_underline; /*< \todo document this */
00362     bool text_style_overline; /*< \todo document this */
00363     bool text_style_strikethrough; /*< \todo document this */
00364     Dictionary *texture_list; /*< \todo document this */
00365     uint32_t ticks_color; /*< \todo document this */
00366     uint32_t shine_color; /*< \todo document this */
00367     char to_open[200]; /*< \todo document this */
00368     char menu_action[200]; /*< \todo document this */
00369     char current_directory[200]; /*< \todo document this */
00370     EmbReal zoomInLimit; /*< */
00371     EmbReal zoomOutLimit; /*< */
00372     EmbReal ruler_width; /*< */
00373     EmbReal tick_depth; /*< */
00374     EmbReal major_tick_separation; /*< */

```

```

00375     EmbReal needle_speed; /*< */
00376     EmbReal stitch_time; /*< */
00377 } Settings;
00378
00379 enum UiMode {
00380     DEFAULT_MODE,
00381
00382     CIRCLE_MODE_1P_RAD,
00383     CIRCLE_MODE_1P_DIA,
00384     CIRCLE_MODE_2P,
00385     CIRCLE_MODE_3P,
00386     CIRCLE_MODE_TTR,
00387
00388     ELLIPSE_MODE_MAJORDIAMETER_MINORRADIUS,
00389     ELLIPSE_MODE_MAJORRADIUS_MINORRADIUS,
00390     ELLIPSE_MODE_ELLIPSE_ROTATION,
00391
00392     DOLPHIN_MODE_NUM_POINTS,
00393     DOLPHIN_MODE_XSCALE,
00394     DOLPHIN_MODE_YSCALE,
00395
00396     HEART_MODE_NUM_POINTS,
00397     HEART_MODE_STYLE,
00398     HEART_MODE_XSCALE,
00399     HEART_MODE_YSCALE,
00400
00401     ROTATE_MODE_NORMAL,
00402     ROTATE_MODE_REFERENCE,
00403
00404     SCALE_MODE_NORMAL,
00405     SCALE_MODE_REFERENCE,
00406
00407     SINGLE_LINE_TEXT_MODE_JUSTIFY,
00408     SINGLE_LINE_TEXT_MODE_SETFONT,
00409     SINGLE_LINE_TEXT_MODE_SETGEOM,
00410     SINGLE_LINE_TEXT_MODE_RAPID,
00411
00412     STAR_MODE_NUM_POINTS,
00413     STAR_MODE_CENTER_PT,
00414     STAR_MODE_RAD_OUTER,
00415     STAR_MODE_RAD_INNER,
00416
00417     SNOWFLAKE_MODE_NUM_POINTS,
00418     SNOWFLAKE_MODE_XSCALE,
00419     SNOWFLAKE_MODE_YSCALE
00420 };
00421
00422 //Custom Data used in QGraphicsItems
00423
00424 // ( int, const QVariant)
00425 //I.E. object.setData(OBJ_TYPE, OBJ_TYPE_LINE);
00426 //I.E. object.setData(OBJ_LAYER, "OUTLINE");
00427 //I.E. object.setData(OBJ_COLOR, 123);
00428 //I.E. object.setData(OBJ_LTYPE, OBJ_LTYPE_CONT);
00429
00430 //Keys
00431 enum OBJ_KEYS {
00432     OBJ_TYPE = 0, //value type - int: See OBJ_TYPE_VALUES
00433     OBJ_NAME = 1, //value type - str: See OBJ_NAME_VALUES
00434     OBJ_LAYER = 2, //value type - str: "USER", "DEFINED", "STRINGS", etc...
00435     OBJ_COLOR = 3, //value type - int: 0-255 //TODO: Use color chart in formats/format-dxf.h for this
00436     OBJ_LTYPE = 4, //value type - int: See OBJ_LTYPE_VALUES
00437     OBJ_LWT = 5, //value type - int: 0-27
00438     OBJ_RUBBER = 6 //value type - int: See OBJ_RUBBER_VALUES
00439 };
00440
00441 //Values
00442 enum OBJ_TYPE_VALUES {
00443     OBJ_TYPE_NULL = 0,
00444     /*< NOTE: Allow this enum to evaluate false */
00445     OBJ_TYPE_BASE = 100000,
00446     /*< NOTE: Values >= 65536 ensure compatibility with qgraphicsitem_cast() */
00447     OBJ_TYPE_ARC = 100001,
00448     OBJ_TYPE_BLOCK = 100002,
00449     /*< For the block type, that has to exist for SVG. */
00450     OBJ_TYPE_CIRCLE = 100003,
00451     OBJ_TYPE_DIMALIGNED = 100004,
00452     /*< For the Aligned Dimension, that has to exist for DXF drawings. */
00453     OBJ_TYPE_DIMANGULAR = 100005,
00454     /*< For the Angular Dimension, that has to exist for DXF drawings. */
00455     OBJ_TYPE_DIMARCLENGTH = 100006,
00456     /*< For the Arc Length Dimension, that has to exist for DXF drawings. */
00457     OBJ_TYPE_DIMDIAMETER = 100007,
00458     OBJ_TYPE_DIMLEADER = 100008,
00459     OBJ_TYPE_DIMLINEAR = 100009,
00460     /*< For the Linear Dimension, that has to exist for DXF drawings. */
00461     OBJ_TYPE_DIMORDINATE = 100010,

```

```

00462     /*< For the Ordinate Dimension, that has to exist for DXF drawings. */
00463     OBJ_TYPE_DIMRADIUS = 100011,
00464     /*< For the Radial Dimension, that has to exist for DXF drawings. */
00465     OBJ_TYPE_ELLIPSE = 100012,
00466     OBJ_TYPE_ELLIPSEARC = 100013,
00467     OBJ_TYPE_RUBBER = 100014,
00468     OBJ_TYPE_GRID = 100015,
00469     OBJ_TYPE_HATCH = 100016,
00470     OBJ_TYPE_IMAGE = 100017,
00471     OBJ_TYPE_INFINITELINE = 100018,
00472     /*< For the Infinite Line object. Which should be removed from output as it exists
00473      for drafting reasons. */
00474     OBJ_TYPE_LINE = 100019,
00475     OBJ_TYPE_PATH = 100020,
00476     OBJ_TYPE_POINT = 100021,
00477     OBJ_TYPE_POLYGON = 100022,
00478     OBJ_TYPE_POLYLINE = 100023,
00479     OBJ_TYPE_RAY = 100024,
00480     /*< For the Ray object. */
00481     OBJ_TYPE_RECTANGLE = 100025,
00482     OBJ_TYPE_SLOT = 100026,
00483     OBJ_TYPE_SPLINE = 100027,
00484     OBJ_TYPE_TEXTMULTI = 100028,
00485     OBJ_TYPE_TEXTSINGLE = 100029
00486 };
00487
00488 enum OBJ_LTYPE_VALUES {
00489 //CAD Linetypes
00490     OBJ_LTYPE_CONT = 0,
00491     OBJ_LTYPE_CENTER = 1,
00492     OBJ_LTYPE_DOT = 2,
00493     OBJ_LTYPE_HIDDEN = 3,
00494     OBJ_LTYPE_PHANTOM = 4,
00495     OBJ_LTYPE_ZIGZAG = 5,
00496 //Embroidery Stitchtypes
00497     OBJ_LTYPE_RUNNING = 6, // _____
00498     OBJ_LTYPE_SATIN = 7, // vvvvvvvvvv
00499     OBJ_LTYPE_FISHBONE = 8, // >>>>
00500 };
00501
00502 enum OBJ_LWT_VALUES {
00503     OBJ_LWT_BYLAYER = -2,
00504     OBJ_LWT_BYBLOCK = -1,
00505     OBJ_LWT_DEFAULT = 0,
00506     OBJ_LWT_01 = 1,
00507     OBJ_LWT_02 = 2,
00508     OBJ_LWT_03 = 3,
00509     OBJ_LWT_04 = 4,
00510     OBJ_LWT_05 = 5,
00511     OBJ_LWT_06 = 6,
00512     OBJ_LWT_07 = 7,
00513     OBJ_LWT_08 = 8,
00514     OBJ_LWT_09 = 9,
00515     OBJ_LWT_10 = 10,
00516     OBJ_LWT_11 = 11,
00517     OBJ_LWT_12 = 12,
00518     OBJ_LWT_13 = 13,
00519     OBJ_LWT_14 = 14,
00520     OBJ_LWT_15 = 15,
00521     OBJ_LWT_16 = 16,
00522     OBJ_LWT_17 = 17,
00523     OBJ_LWT_18 = 18,
00524     OBJ_LWT_19 = 19,
00525     OBJ_LWT_20 = 20,
00526     OBJ_LWT_21 = 21,
00527     OBJ_LWT_22 = 22,
00528     OBJ_LWT_23 = 23,
00529     OBJ_LWT_24 = 24
00530 };
00531
00532 enum OBJ_SNAP_VALUES {
00533     OBJ_SNAP_NULL = 0, //NOTE: Allow this enum to evaluate false
00534     OBJ_SNAP_ENDPOINT = 1,
00535     OBJ_SNAP_MIDPOINT = 2,
00536     OBJ_SNAP_CENTER = 3,
00537     OBJ_SNAP_NODE = 4,
00538     OBJ_SNAP_QUADRANT = 5,
00539     OBJ_SNAP_INTERSECTION = 6,
00540     OBJ_SNAP_EXTENSION = 7,
00541     OBJ_SNAP_INSERTION = 8,
00542     OBJ_SNAP_PERPENDICULAR = 9,
00543     OBJ_SNAP_TANGENT = 10,
00544     OBJ_SNAP_NEAREST = 11,
00545     OBJ_SNAP_APPINTERSECTION = 12,
00546     OBJ_SNAP_PARALLEL = 13
00547 };
00548

```

```

00549 enum OBJ_RUBBER_VALUES {
00550     OBJ_RUBBER_OFF = 0, //NOTE: Allow this enum to evaluate false
00551     OBJ_RUBBER_ON = 1, //NOTE: Allow this enum to evaluate true
00552
00553     OBJ_RUBBER_CIRCLE_1P_RAD,
00554     OBJ_RUBBER_CIRCLE_1P_DIA,
00555     OBJ_RUBBER_CIRCLE_2P,
00556     OBJ_RUBBER_CIRCLE_3P,
00557     OBJ_RUBBER_CIRCLE_TTR,
00558     OBJ_RUBBER_CIRCLE_TTT,
00559
00560     OBJ_RUBBER_DIMLEADER_LINE,
00561
00562     OBJ_RUBBER_ELLIPSE_LINE,
00563     OBJ_RUBBER_ELLIPSE_MAJOR_DIAMETER_MINOR_RADIUS,
00564     OBJ_RUBBER_ELLIPSE_MAJOR_RADIUS_MINOR_RADIUS,
00565     OBJ_RUBBER_ELLIPSE_ROTATION,
00566
00567     OBJ_RUBBER_GRIP,
00568
00569     OBJ_RUBBER_LINE,
00570
00571     OBJ_RUBBER_POLYGON,
00572     OBJ_RUBBER_POLYGON_INSCRIBE,
00573     OBJ_RUBBER_POLYGON_CIRCUMSCRIBE,
00574
00575     OBJ_RUBBER_POLYLINE,
00576
00577     OBJ_RUBBER_IMAGE,
00578
00579     OBJ_RUBBER_RECTANGLE,
00580
00581     OBJ_RUBBER_TEXTSINGLE
00582 };
00583
00584 enum SPARE_RUBBER_VALUES {
00585     SPARE_RUBBER_PATH,
00586     SPARE_RUBBER_POLYGON,
00587     SPARE_RUBBER_POLYLINE
00588 };
00589
00590
00591 enum PREVIEW_CLONE_VALUES {
00592     PREVIEW_CLONE_NULL = 0, //NOTE: Allow this enum to evaluate false
00593     PREVIEW_CLONE_SELECTED,
00594     PREVIEW_CLONE_RUBBER
00595 };
00596
00597
00598 enum PREVIEW_MODE_VALUES {
00599     PREVIEW_MODE_MOVE,
00600     PREVIEW_MODE_ROTATE,
00601     PREVIEW_MODE_SCALE
00602 };
00603
00604 static const EmbReal emb_constant_pi = 3.14159265358979323846;
00605
00606 /*
00607     * \brief Convert \a a to a QPointF.
00608 */
00609 inline QPointF
00610 to_QPointF(EmbVector a)
00611 {
00612     QPointF result(a.x(), a.y());
00613     return result;
00614 }
00615
00616 /*
00617     * \brief Convert \a a to an EmbVector.
00618 */
00619 inline EmbVector
00620 to_EmbVector(QPointF a)
00621 {
00622     EmbVector v;
00623     v.x() = a.x();
00624     v.y() = a.y();
00625     return v;
00626 }
00627
00628 /*
00629     * \brief Wrapper for embVector_add to use the syntax \a a + \a b.
00630 */
00631 inline EmbVector
00632 operator+(EmbVector a, EmbVector b)
00633 {
00634     EmbVector v;
00635     v.x() = a.x() + b.x();
00636     v.y() = a.y() + b.y();
00637     return v;
00638 }
00639
00640 /*
00641     * \brief Wrapper for embVector_sub to use the syntax \a a - \a b.
00642 */
00643 inline EmbVector
00644 operator-(EmbVector a, EmbVector b)
00645 {
00646     EmbVector v;
00647     v.x() = a.x() - b.x();
00648     v.y() = a.y() - b.y();
00649     return v;
00650 }
00651
00652 /*
00653     * \brief Wrapper for embVector_div to use the syntax \a a / \a b.
00654 */
00655 inline EmbVector
00656 operator/(EmbVector a, EmbVector b)
00657 {
00658     EmbVector v;
00659     v.x() = a.x() / b.x();
00660     v.y() = a.y() / b.y();
00661     return v;
00662 }
00663
00664 /*
00665     * \brief Wrapper for embVector_dot to use the syntax \a a . \a b.
00666 */
00667 inline EmbVector
00668 operator*(EmbVector a, EmbVector b)
00669 {
00670     EmbVector v;
00671     v.x() = a.x() * b.x();
00672     v.y() = a.y() * b.y();
00673     return v;
00674 }
00675
00676 /*
00677     * \brief Wrapper for embVector_cross to use the syntax \a a ^ \a b.
00678 */
00679 inline EmbVector
00680 operator^(EmbVector a, EmbVector b)
00681 {
00682     EmbVector v;
00683     v.x() = a.x() * b.y() - a.y() * b.x();
00684     v.y() = a.y() * b.x() - a.x() * b.y();
00685     return v;
00686 }
00687
00688 /*
00689     * \brief Wrapper for embVector_norm to use the syntax \a a ^> \a b.
00690 */
00691 inline EmbVector
00692 operator>(EmbVector a, EmbVector b)
00693 {
00694     EmbVector v;
00695     v.x() = a.x() > b.x();
00696     v.y() = a.y() > b.y();
00697     return v;
00698 }
00699
00700 /*
00701     * \brief Wrapper for embVector_lt to use the syntax \a a < \a b.
00702 */
00703 inline EmbVector
00704 operator<(EmbVector a, EmbVector b)
00705 {
00706     EmbVector v;
00707     v.x() = a.x() < b.x();
00708     v.y() = a.y() < b.y();
00709     return v;
00710 }
00711
00712 /*
00713     * \brief Wrapper for embVector_leq to use the syntax \a a <= \a b.
00714 */
00715 inline EmbVector
00716 operator<=(EmbVector a, EmbVector b)
00717 {
00718     EmbVector v;
00719     v.x() = a.x() <= b.x();
00720     v.y() = a.y() <= b.y();
00721     return v;
00722 }
00723
00724 /*
00725     * \brief Wrapper for embVector_geq to use the syntax \a a >= \a b.
00726 */
00727 inline EmbVector
00728 operator>=(EmbVector a, EmbVector b)
00729 {
00730     EmbVector v;
00731     v.x() = a.x() >= b.x();
00732     v.y() = a.y() >= b.y();
00733     return v;
00734 }
00735
00736 /*
00737     * \brief Wrapper for embVector_eq to use the syntax \a a == \a b.
00738 */
00739 inline EmbVector
00740 operator==(EmbVector a, EmbVector b)
00741 {
00742     EmbVector v;
00743     v.x() = a.x() == b.x();
00744     v.y() = a.y() == b.y();
00745     return v;
00746 }
00747
00748 /*
00749     * \brief Wrapper for embVector_neq to use the syntax \a a != \a b.
00750 */
00751 inline EmbVector
00752 operator!=(EmbVector a, EmbVector b)
00753 {
00754     EmbVector v;
00755     v.x() = a.x() != b.x();
00756     v.y() = a.y() != b.y();
00757     return v;
00758 }
00759
00760 /*
00761     * \brief Wrapper for embVector_min to use the syntax \a a < \a b.
00762 */
00763 inline EmbVector
00764 operator<(EmbVector a, EmbVector b)
00765 {
00766     EmbVector v;
00767     v.x() = a.x() < b.x();
00768     v.y() = a.y() < b.y();
00769     return v;
00770 }
00771
00772 /*
00773     * \brief Wrapper for embVector_max to use the syntax \a a > \a b.
00774 */
00775 inline EmbVector
00776 operator>(EmbVector a, EmbVector b)
00777 {
00778     EmbVector v;
00779     v.x() = a.x() > b.x();
00780     v.y() = a.y() > b.y();
00781     return v;
00782 }
00783
00784 /*
00785     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
00786 */
00787 inline EmbVector
00788 operator<(EmbVector a, EmbVector b)
00789 {
00790     EmbVector v;
00791     v.x() = a.x() < b.x();
00792     v.y() = a.y() < b.y();
00793     return v;
00794 }
00795
00796 /*
00797     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
00798 */
00799 inline EmbVector
00800 operator>(EmbVector a, EmbVector b)
00801 {
00802     EmbVector v;
00803     v.x() = a.x() > b.x();
00804     v.y() = a.y() > b.y();
00805     return v;
00806 }
00807
00808 /*
00809     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
00810 */
00811 inline EmbVector
00812 operator<(EmbVector a, EmbVector b)
00813 {
00814     EmbVector v;
00815     v.x() = a.x() < b.x();
00816     v.y() = a.y() < b.y();
00817     return v;
00818 }
00819
00820 /*
00821     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
00822 */
00823 inline EmbVector
00824 operator>(EmbVector a, EmbVector b)
00825 {
00826     EmbVector v;
00827     v.x() = a.x() > b.x();
00828     v.y() = a.y() > b.y();
00829     return v;
00830 }
00831
00832 /*
00833     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
00834 */
00835 inline EmbVector
00836 operator<(EmbVector a, EmbVector b)
00837 {
00838     EmbVector v;
00839     v.x() = a.x() < b.x();
00840     v.y() = a.y() < b.y();
00841     return v;
00842 }
00843
00844 /*
00845     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
00846 */
00847 inline EmbVector
00848 operator>(EmbVector a, EmbVector b)
00849 {
00850     EmbVector v;
00851     v.x() = a.x() > b.x();
00852     v.y() = a.y() > b.y();
00853     return v;
00854 }
00855
00856 /*
00857     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
00858 */
00859 inline EmbVector
00860 operator<(EmbVector a, EmbVector b)
00861 {
00862     EmbVector v;
00863     v.x() = a.x() < b.x();
00864     v.y() = a.y() < b.y();
00865     return v;
00866 }
00867
00868 /*
00869     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
00870 */
00871 inline EmbVector
00872 operator>(EmbVector a, EmbVector b)
00873 {
00874     EmbVector v;
00875     v.x() = a.x() > b.x();
00876     v.y() = a.y() > b.y();
00877     return v;
00878 }
00879
00880 /*
00881     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
00882 */
00883 inline EmbVector
00884 operator<(EmbVector a, EmbVector b)
00885 {
00886     EmbVector v;
00887     v.x() = a.x() < b.x();
00888     v.y() = a.y() < b.y();
00889     return v;
00890 }
00891
00892 /*
00893     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
00894 */
00895 inline EmbVector
00896 operator>(EmbVector a, EmbVector b)
00897 {
00898     EmbVector v;
00899     v.x() = a.x() > b.x();
00900     v.y() = a.y() > b.y();
00901     return v;
00902 }
00903
00904 /*
00905     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
00906 */
00907 inline EmbVector
00908 operator<(EmbVector a, EmbVector b)
00909 {
00910     EmbVector v;
00911     v.x() = a.x() < b.x();
00912     v.y() = a.y() < b.y();
00913     return v;
00914 }
00915
00916 /*
00917     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
00918 */
00919 inline EmbVector
00920 operator>(EmbVector a, EmbVector b)
00921 {
00922     EmbVector v;
00923     v.x() = a.x() > b.x();
00924     v.y() = a.y() > b.y();
00925     return v;
00926 }
00927
00928 /*
00929     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
00930 */
00931 inline EmbVector
00932 operator<(EmbVector a, EmbVector b)
00933 {
00934     EmbVector v;
00935     v.x() = a.x() < b.x();
00936     v.y() = a.y() < b.y();
00937     return v;
00938 }
00939
00940 /*
00941     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
00942 */
00943 inline EmbVector
00944 operator>(EmbVector a, EmbVector b)
00945 {
00946     EmbVector v;
00947     v.x() = a.x() > b.x();
00948     v.y() = a.y() > b.y();
00949     return v;
00950 }
00951
00952 /*
00953     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
00954 */
00955 inline EmbVector
00956 operator<(EmbVector a, EmbVector b)
00957 {
00958     EmbVector v;
00959     v.x() = a.x() < b.x();
00960     v.y() = a.y() < b.y();
00961     return v;
00962 }
00963
00964 /*
00965     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
00966 */
00967 inline EmbVector
00968 operator>(EmbVector a, EmbVector b)
00969 {
00970     EmbVector v;
00971     v.x() = a.x() > b.x();
00972     v.y() = a.y() > b.y();
00973     return v;
00974 }
00975
00976 /*
00977     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
00978 */
00979 inline EmbVector
00980 operator<(EmbVector a, EmbVector b)
00981 {
00982     EmbVector v;
00983     v.x() = a.x() < b.x();
00984     v.y() = a.y() < b.y();
00985     return v;
00986 }
00987
00988 /*
00989     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
00990 */
00991 inline EmbVector
00992 operator>(EmbVector a, EmbVector b)
00993 {
00994     EmbVector v;
00995     v.x() = a.x() > b.x();
00996     v.y() = a.y() > b.y();
00997     return v;
00998 }
00999
01000 /*
01001     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01002 */
01003 inline EmbVector
01004 operator<(EmbVector a, EmbVector b)
01005 {
01006     EmbVector v;
01007     v.x() = a.x() < b.x();
01008     v.y() = a.y() < b.y();
01009     return v;
01010 }
01011
01012 /*
01013     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01014 */
01015 inline EmbVector
01016 operator>(EmbVector a, EmbVector b)
01017 {
01018     EmbVector v;
01019     v.x() = a.x() > b.x();
01020     v.y() = a.y() > b.y();
01021     return v;
01022 }
01023
01024 /*
01025     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01026 */
01027 inline EmbVector
01028 operator<(EmbVector a, EmbVector b)
01029 {
01030     EmbVector v;
01031     v.x() = a.x() < b.x();
01032     v.y() = a.y() < b.y();
01033     return v;
01034 }
01035
01036 /*
01037     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01038 */
01039 inline EmbVector
01040 operator>(EmbVector a, EmbVector b)
01041 {
01042     EmbVector v;
01043     v.x() = a.x() > b.x();
01044     v.y() = a.y() > b.y();
01045     return v;
01046 }
01047
01048 /*
01049     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01050 */
01051 inline EmbVector
01052 operator<(EmbVector a, EmbVector b)
01053 {
01054     EmbVector v;
01055     v.x() = a.x() < b.x();
01056     v.y() = a.y() < b.y();
01057     return v;
01058 }
01059
01060 /*
01061     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01062 */
01063 inline EmbVector
01064 operator>(EmbVector a, EmbVector b)
01065 {
01066     EmbVector v;
01067     v.x() = a.x() > b.x();
01068     v.y() = a.y() > b.y();
01069     return v;
01070 }
01071
01072 /*
01073     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01074 */
01075 inline EmbVector
01076 operator<(EmbVector a, EmbVector b)
01077 {
01078     EmbVector v;
01079     v.x() = a.x() < b.x();
01080     v.y() = a.y() < b.y();
01081     return v;
01082 }
01083
01084 /*
01085     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01086 */
01087 inline EmbVector
01088 operator>(EmbVector a, EmbVector b)
01089 {
01090     EmbVector v;
01091     v.x() = a.x() > b.x();
01092     v.y() = a.y() > b.y();
01093     return v;
01094 }
01095
01096 /*
01097     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01098 */
01099 inline EmbVector
01100 operator<(EmbVector a, EmbVector b)
01101 {
01102     EmbVector v;
01103     v.x() = a.x() < b.x();
01104     v.y() = a.y() < b.y();
01105     return v;
01106 }
01107
01108 /*
01109     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01110 */
01111 inline EmbVector
01112 operator>(EmbVector a, EmbVector b)
01113 {
01114     EmbVector v;
01115     v.x() = a.x() > b.x();
01116     v.y() = a.y() > b.y();
01117     return v;
01118 }
01119
01120 /*
01121     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01122 */
01123 inline EmbVector
01124 operator<(EmbVector a, EmbVector b)
01125 {
01126     EmbVector v;
01127     v.x() = a.x() < b.x();
01128     v.y() = a.y() < b.y();
01129     return v;
01130 }
01131
01132 /*
01133     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01134 */
01135 inline EmbVector
01136 operator>(EmbVector a, EmbVector b)
01137 {
01138     EmbVector v;
01139     v.x() = a.x() > b.x();
01140     v.y() = a.y() > b.y();
01141     return v;
01142 }
01143
01144 /*
01145     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01146 */
01147 inline EmbVector
01148 operator<(EmbVector a, EmbVector b)
01149 {
01150     EmbVector v;
01151     v.x() = a.x() < b.x();
01152     v.y() = a.y() < b.y();
01153     return v;
01154 }
01155
01156 /*
01157     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01158 */
01159 inline EmbVector
01160 operator>(EmbVector a, EmbVector b)
01161 {
01162     EmbVector v;
01163     v.x() = a.x() > b.x();
01164     v.y() = a.y() > b.y();
01165     return v;
01166 }
01167
01168 /*
01169     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01170 */
01171 inline EmbVector
01172 operator<(EmbVector a, EmbVector b)
01173 {
01174     EmbVector v;
01175     v.x() = a.x() < b.x();
01176     v.y() = a.y() < b.y();
01177     return v;
01178 }
01179
01180 /*
01181     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01182 */
01183 inline EmbVector
01184 operator>(EmbVector a, EmbVector b)
01185 {
01186     EmbVector v;
01187     v.x() = a.x() > b.x();
01188     v.y() = a.y() > b.y();
01189     return v;
01190 }
01191
01192 /*
01193     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01194 */
01195 inline EmbVector
01196 operator<(EmbVector a, EmbVector b)
01197 {
01198     EmbVector v;
01199     v.x() = a.x() < b.x();
01200     v.y() = a.y() < b.y();
01201     return v;
01202 }
01203
01204 /*
01205     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01206 */
01207 inline EmbVector
01208 operator>(EmbVector a, EmbVector b)
01209 {
01210     EmbVector v;
01211     v.x() = a.x() > b.x();
01212     v.y() = a.y() > b.y();
01213     return v;
01214 }
01215
01216 /*
01217     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01218 */
01219 inline EmbVector
01220 operator<(EmbVector a, EmbVector b)
01221 {
01222     EmbVector v;
01223     v.x() = a.x() < b.x();
01224     v.y() = a.y() < b.y();
01225     return v;
01226 }
01227
01228 /*
01229     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01230 */
01231 inline EmbVector
01232 operator>(EmbVector a, EmbVector b)
01233 {
01234     EmbVector v;
01235     v.x() = a.x() > b.x();
01236     v.y() = a.y() > b.y();
01237     return v;
01238 }
01239
01240 /*
01241     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01242 */
01243 inline EmbVector
01244 operator<(EmbVector a, EmbVector b)
01245 {
01246     EmbVector v;
01247     v.x() = a.x() < b.x();
01248     v.y() = a.y() < b.y();
01249     return v;
01250 }
01251
01252 /*
01253     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01254 */
01255 inline EmbVector
01256 operator>(EmbVector a, EmbVector b)
01257 {
01258     EmbVector v;
01259     v.x() = a.x() > b.x();
01260     v.y() = a.y() > b.y();
01261     return v;
01262 }
01263
01264 /*
01265     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01266 */
01267 inline EmbVector
01268 operator<(EmbVector a, EmbVector b)
01269 {
01270     EmbVector v;
01271     v.x() = a.x() < b.x();
01272     v.y() = a.y() < b.y();
01273     return v;
01274 }
01275
01276 /*
01277     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01278 */
01279 inline EmbVector
01280 operator>(EmbVector a, EmbVector b)
01281 {
01282     EmbVector v;
01283     v.x() = a.x() > b.x();
01284     v.y() = a.y() > b.y();
01285     return v;
01286 }
01287
01288 /*
01289     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01290 */
01291 inline EmbVector
01292 operator<(EmbVector a, EmbVector b)
01293 {
01294     EmbVector v;
01295     v.x() = a.x() < b.x();
01296     v.y() = a.y() < b.y();
01297     return v;
01298 }
01299
01300 /*
01301     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01302 */
01303 inline EmbVector
01304 operator>(EmbVector a, EmbVector b)
01305 {
01306     EmbVector v;
01307     v.x() = a.x() > b.x();
01308     v.y() = a.y() > b.y();
01309     return v;
01310 }
01311
01312 /*
01313     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01314 */
01315 inline EmbVector
01316 operator<(EmbVector a, EmbVector b)
01317 {
01318     EmbVector v;
01319     v.x() = a.x() < b.x();
01320     v.y() = a.y() < b.y();
01321     return v;
01322 }
01323
01324 /*
01325     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01326 */
01327 inline EmbVector
01328 operator>(EmbVector a, EmbVector b)
01329 {
01330     EmbVector v;
01331     v.x() = a.x() > b.x();
01332     v.y() = a.y() > b.y();
01333     return v;
01334 }
01335
01336 /*
01337     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01338 */
01339 inline EmbVector
01340 operator<(EmbVector a, EmbVector b)
01341 {
01342     EmbVector v;
01343     v.x() = a.x() < b.x();
01344     v.y() = a.y() < b.y();
01345     return v;
01346 }
01347
01348 /*
01349     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01350 */
01351 inline EmbVector
01352 operator>(EmbVector a, EmbVector b)
01353 {
01354     EmbVector v;
01355     v.x() = a.x() > b.x();
01356     v.y() = a.y() > b.y();
01357     return v;
01358 }
01359
01360 /*
01361     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01362 */
01363 inline EmbVector
01364 operator<(EmbVector a, EmbVector b)
01365 {
01366     EmbVector v;
01367     v.x() = a.x() < b.x();
01368     v.y() = a.y() < b.y();
01369     return v;
01370 }
01371
01372 /*
01373     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01374 */
01375 inline EmbVector
01376 operator>(EmbVector a, EmbVector b)
01377 {
01378     EmbVector v;
01379     v.x() = a.x() > b.x();
01380     v.y() = a.y() > b.y();
01381     return v;
01382 }
01383
01384 /*
01385     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01386 */
01387 inline EmbVector
01388 operator<(EmbVector a, EmbVector b)
01389 {
01390     EmbVector v;
01391     v.x() = a.x() < b.x();
01392     v.y() = a.y() < b.y();
01393     return v;
01394 }
01395
01396 /*
01397     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01398 */
01399 inline EmbVector
01400 operator>(EmbVector a, EmbVector b)
01401 {
01402     EmbVector v;
01403     v.x() = a.x() > b.x();
01404     v.y() = a.y() > b.y();
01405     return v;
01406 }
01407
01408 /*
01409     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01410 */
01411 inline EmbVector
01412 operator<(EmbVector a, EmbVector b)
01413 {
01414     EmbVector v;
01415     v.x() = a.x() < b.x();
01416     v.y() = a.y() < b.y();
01417     return v;
01418 }
01419
01420 /*
01421     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01422 */
01423 inline EmbVector
01424 operator>(EmbVector a, EmbVector b)
01425 {
01426     EmbVector v;
01427     v.x() = a.x() > b.x();
01428     v.y() = a.y() > b.y();
01429     return v;
01430 }
01431
01432 /*
01433     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01434 */
01435 inline EmbVector
01436 operator<(EmbVector a, EmbVector b)
01437 {
01438     EmbVector v;
01439     v.x() = a.x() < b.x();
01440     v.y() = a.y() < b.y();
01441     return v;
01442 }
01443
01444 /*
01445     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01446 */
01447 inline EmbVector
01448 operator>(EmbVector a, EmbVector b)
01449 {
01450     EmbVector v;
01451     v.x() = a.x() > b.x();
01452     v.y() = a.y() > b.y();
01453     return v;
01454 }
01455
01456 /*
01457     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01458 */
01459 inline EmbVector
01460 operator<(EmbVector a, EmbVector b)
01461 {
01462     EmbVector v;
01463     v.x() = a.x() < b.x();
01464     v.y() = a.y() < b.y();
01465     return v;
01466 }
01467
01468 /*
01469     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01470 */
01471 inline EmbVector
01472 operator>(EmbVector a, EmbVector b)
01473 {
01474     EmbVector v;
01475     v.x() = a.x() > b.x();
01476     v.y() = a.y() > b.y();
01477     return v;
01478 }
01479
01480 /*
01481     * \brief Wrapper for embVector_minmax to use the syntax \a a < \a b.
01482 */
01483 inline EmbVector
01484 operator<(EmbVector a, EmbVector b)
01485 {
01486     EmbVector v;
01487     v.x() = a.x() < b.x();
01488     v.y() = a.y() < b.y();
01489     return v;
01490 }
01491
01492 /*
01493     * \brief Wrapper for embVector_maxmin to use the syntax \a a > \a b.
01494 */
01495 inline EmbVector
01496 operator>(EmbVector a, EmbVector b)
01497 {
01498     EmbVector v;
01499     v.x() = a.x() >
```

```

00643 {
00644     return embVector_add(a, b);
00645 }
00646
00647 /*
00648  * \brief Wrapper for embVector_subtract to use the syntax \a a - \a b.
00649  */
00650 inline EmbVector
00651 operator-(EmbVector a, EmbVector b)
00652 {
00653     return embVector_subtract(a, b);
00654 }
00655
00656 inline EmbReal
00657 radians(EmbReal degrees)
00658 {
00659     return (degrees*emb_constant_pi/180.0);
00660 }
00661
00662 inline EmbReal
00663 degrees(EmbReal radian)
00664 {
00665     return (radian*180.0/emb_constant_pi);
00666 }
00667
00668 class BaseObject : public QGraphicsPathItem
00669 {
00670     public:
00671         BaseObject(QGraphicsItem* parent = 0);
00672         virtual ~BaseObject();
00673
00674     enum { Type = OBJ_TYPE_BASE };
00675     virtual int type() const { return Type; }
00676
00677     QPen objPen;
00678     QPen lwtPen;
00679     QLineF objLine;
00680     int objRubberMode;
00681     QHash<QString, QPointF> objRubberPoints;
00682     QHash<QString, QString> objRubberTexts;
00683     qint64 objID;
00684
00685     qint64 objectID() const { return objID; }
00686     QPen objectPen() const { return objPen; }
00687     QColor objectColor() const { return objPen.color(); }
00688     QRgb objectColorRGB() const { return objPen.color().rgb(); }
00689     Qt::PenStyle objectLineType() const { return objPen.style(); }
00690     EmbReal objectLineWidth() const { return lwtPen.widthF(); }
00691     QPainterPath objectPath() const { return path(); }
00692     int objectRubberMode() const { return objRubberMode; }
00693     QPointF objectRubberPoint(const QString& key) const;
00694     QString objectRubberText(const QString& key) const;
00695
00696     QPointF objectCenter() const { return scenePos(); }
00697     EmbReal objectCenterX() const { return scenePos().x(); }
00698     EmbReal objectCenterY() const { return scenePos().y(); }
00699
00700     void setObjectCenter(EmbVector center)
00701     {
00702         setPos(center.x, center.y);
00703     }
00704     void setObjectCenterX(EmbReal centerX) { setX(centerX); }
00705     void setObjectCenterY(EmbReal centerY) { setY(centerY); }
00706
00707     QRectF rect() const { return path().boundingRect(); }
00708     void setRect(const QRectF& r) { QPainterPath p; p.addRect(r); setPath(p); }
00709     void setRect(EmbReal x, EmbReal y, EmbReal w, EmbReal h) { QPainterPath p; p.addRect(x,y,w,h);
00710         setPath(p); }
00711     QLineF line() const { return objLine; }
00712     void setLine(const QLineF& li) { QPainterPath p; p.moveTo(li.p1()); p.lineTo(li.p2()); setPath(p);
00713     objLine = li; }
00714     void setLine(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2) { QPainterPath p; p.moveTo(x1,y1);
00715     p.lineTo(x2,y2); setPath(p); objLine.setLine(x1,y1,x2,y2); }
00716
00717     void setObjectColor(const QColor& color);
00718     void setObjectColorRGB(QRgb rgb);
00719     void setObjectLineType(Qt::PenStyle lineType);
00720     void setObjectLineWidth(EmbReal lineWidth);
00721     void setObjectPath(const QPainterPath& p) { setPath(p); }
00722     void setObjectRubberMode(int mode) { objRubberMode = mode; }
00723     void setObjectRubberPoint(const QString& key, const QPointF& point) { objRubberPoints.insert(key,
00724     point); }
00725     void setObjectRubberText(const QString& key, const QString& txt) { objRubberTexts.insert(key,
00726     txt); }
00727
00728     virtual QRectF boundingRect() const;
00729     virtual QPainterPath shape() const { return path(); }

```

```

00734     void drawRubberLine(const QLineF& rubLine, QPainter* painter = 0, const char* colorFromScene = 0);
00735
00736     virtual void vulcanize() = 0;
00737     virtual QPointF mouseSnapPoint(const QPointF& mousePoint) = 0;
00738     virtual QList<QPointF> allGripPoints() = 0;
00739     virtual void gripEdit(const QPointF& before, const QPointF& after) = 0;
00740
00741 protected:
00742     QPen lineWeightPen() const { return lwtPen; }
00743     void realRender(QPainter* painter, const QPainterPath& renderPath);
00744 };
00745
00746
00747
00748 class ArcObject : public BaseObject
00749 {
00750 public:
00751     QPointF arcStartPoint;
00752     QPointF arcMidPoint;
00753     QPointF arcEndPoint;
00754
00755     ArcObject(EmbArc arc, QRgb rgb, QGraphicsItem* parent = 0);
00756     ArcObject(EmbReal startX, EmbReal startY, EmbReal midX, EmbReal midY, EmbReal endX, EmbReal endY,
00757     QRgb rgb, QGraphicsItem* parent = 0);
00758     ArcObject(ArcObject* obj, QGraphicsItem* parent = 0);
00759     ~ArcObject();
00760
00761     enum { Type = OBJ_TYPE_ARC };
00762     virtual int type() const { return Type; }
00763
00764     void init(EmbReal startX, EmbReal startY, EmbReal midX, EmbReal midY, EmbReal endX, EmbReal endY,
00765     QRgb rgb, Qt::PenStyle lineType);
00766     void updatePath();
00767
00768     void calculateArcData(EmbReal startX, EmbReal startY, EmbReal midX, EmbReal midY, EmbReal endX,
00769     EmbReal endY);
00770     void updateArcRect(EmbReal radius);
00771
00772     EmbReal objectRadius() const { return rect().width()/2.0*scale(); }
00773     EmbReal objectStartAngle() const;
00774     EmbReal objectEndAngle() const;
00775     QPointF objectStartPoint() const;
00776     EmbReal objectStartX() const;
00777     EmbReal objectStartY() const;
00778     QPointF objectMidPoint() const;
00779     EmbReal objectMidX() const;
00780     EmbReal objectMidY() const;
00781     QPointF objectEndPoint() const;
00782     EmbReal objectEndX() const;
00783     EmbReal objectEndY() const;
00784     EmbReal objectArea() const;
00785     EmbReal objectArcLength() const;
00786     EmbReal objectChord() const;
00787     EmbReal objectIncludedAngle() const;
00788     bool objectClockwise() const;
00789
00790     void setObjectRadius(EmbReal radius);
00791     void setObjectStartAngle(EmbReal angle);
00792     void setObjectEndAngle(EmbReal angle);
00793     void setObjectStartPoint(const QPointF& point);
00794     void setObjectEndPoint(EmbReal pointX, EmbReal pointY);
00795     void setObjectMidPoint(EmbReal pointX, EmbReal pointY);
00796     void setObjectEndPoint(const QPointF& point);
00797     void setObjectEndPoint(EmbReal pointX, EmbReal pointY);
00798
00799     void updateRubber(QPainter* painter = 0);
00800     virtual void vulcanize();
00801     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
00802     virtual QList<QPointF> allGripPoints();
00803     virtual void gripEdit(const QPointF& before, const QPointF& after);
00804 protected:
00805     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
00806 };
00807
00808
00809 class CircleObject : public BaseObject
00810 {
00811 public:
00812     CircleObject(EmbReal centerX, EmbReal centerY, EmbReal radius, QRgb rgb, QGraphicsItem* parent =
00813     0);
00813     CircleObject(CircleObject* obj, QGraphicsItem* parent = 0);
00814     ~CircleObject();
00815
00816     void init(EmbReal centerX, EmbReal centerY, EmbReal radius, QRgb rgb, Qt::PenStyle lineType);
00817     void updatePath();
00818
00819     enum { Type = OBJ_TYPE_CIRCLE };

```

```

00820     virtual int type() const { return Type; }
00821
00822     QPainterPath objectSavePath() const;
00823
00824     EmbReal objectRadius() const { return rect().width()/2.0*scale(); }
00825     EmbReal objectDiameter() const { return rect().width()*scale(); }
00826     EmbReal objectArea() const { return emb_constant_pi*objectRadius()*objectRadius(); }
00827     EmbReal objectCircumference() const { return emb_constant_pi*objectDiameter(); }
00828     QPointF objectQuadrant0() const { return objectCenter() + QPointF(objectRadius(), 0); }
00829     QPointF objectQuadrant90() const { return objectCenter() + QPointF(0,-objectRadius()); }
00830     QPointF objectQuadrant180() const { return objectCenter() + QPointF(-objectRadius(),0); }
00831     QPointF objectQuadrant270() const { return objectCenter() + QPointF(0, objectRadius()); }
00832
00833     void setObjectRadius(EmbReal radius);
00834     void setObjectDiameter(EmbReal diameter);
00835     void setObjectArea(EmbReal area);
00836     void setObjectCircumference(EmbReal circumference);
00837
00838     void updateRubber(QPainter* painter = 0);
00839     virtual void vulcanize();
00840     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
00841     virtual QList<QPointF> allGripPoints();
00842     virtual void gripEdit(const QPointF& before, const QPointF& after);
00843 protected:
00844     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
00845 };
00846
00850 class DimLeaderObject : public BaseObject
00851 {
00852 public:
00853     DimLeaderObject(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, QRgb rgb, QGraphicsItem* parent = 0);
00854     DimLeaderObject(DimLeaderObject* obj, QGraphicsItem* parent = 0);
00855     ~DimLeaderObject();
00856
00857     void init(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, QRgb rgb, Qt::PenStyle lineType);
00858
00859     bool curved;
00860     bool filled;
00861     void updateLeader();
00862     QPainterPath lineStylePath;
00863     QPainterPath arrowStylePath;
00864     EmbReal arrowStyleAngle;
00865     EmbReal arrowStyleLength;
00866     EmbReal lineStyleAngle;
00867     EmbReal lineStyleLength;
00868
00869     enum ArrowStyle
00870     {
00871         NoArrow, //NOTE: Allow this enum to evaluate false
00872         Open,
00873         Closed,
00874         Dot,
00875         Box,
00876         Tick
00877     };
00878
00879     enum lineStyle
00880     {
00881         NoLine, //NOTE: Allow this enum to evaluate false
00882         Flared,
00883         Fletching
00884     };
00885
00886     enum { Type = OBJ_TYPE_DIMLEADER };
00887     virtual int type() const { return Type; }
00888
00889     QPointF objectEndPoint1() const;
00890     QPointF objectEndPoint2() const;
00891     QPointF objectMidPoint() const;
00892     EmbReal objectX1() const { return objectEndPoint1().x(); }
00893     EmbReal objectY1() const { return objectEndPoint1().y(); }
00894     EmbReal objectX2() const { return objectEndPoint2().x(); }
00895     EmbReal objectY2() const { return objectEndPoint2().y(); }
00896     EmbReal objectDeltaX() const { return (objectX2() - objectX1()); }
00897     EmbReal objectDeltaY() const { return (objectY2() - objectY1()); }
00898     EmbReal objectAngle() const;
00899     EmbReal objectLength() const { return line().length(); }
00900
00901     void setObjectEndPoint1(const QPointF& endPt1);
00902     void setObjectEndPoint1(EmbReal x1, EmbReal y1);
00903     void setObjectEndPoint2(const QPointF& endPt2);
00904     void setObjectEndPoint2(EmbReal x2, EmbReal y2);
00905     void setObjectX1(EmbReal x) { setObjectEndPoint1(x, objectY1()); }
00906     void setObjectY1(EmbReal y) { setObjectEndPoint1(objectX1(), y); }
00907     void setObjectX2(EmbReal x) { setObjectEndPoint2(x, objectY2()); }
00908     void setObjectY2(EmbReal y) { setObjectEndPoint2(objectX2(), y); }

```

```

00910     void updateRubber(QPainter* painter = 0);
00911     virtual void vulcanize();
00912     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
00913     virtual QList<QPointF> allGripPoints();
00914     virtual void gripEdit(const QPointF& before, const QPointF& after);
00915 protected:
00916     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
00917 };
00918
00919
00920 class EllipseObject : public BaseObject
00921 {
00922 public:
00923     EllipseObject(EmbReal centerX, EmbReal centerY, EmbReal width, EmbReal height, QRgb rgb,
00924     QGraphicsItem* parent = 0);
00925     EllipseObject(EllipseObject* obj, QGraphicsItem* parent = 0);
00926     ~EllipseObject();
00927
00928     void init(EmbReal centerX, EmbReal centerY, EmbReal width, EmbReal height, QRgb rgb, Qt::PenStyle
00929     lineType);
00930     void updatePath();
00931
00932     enum { Type = OBJ_TYPE_ELLIPSE };
00933     virtual int type() const { return Type; }
00934
00935     QPainterPath objectSavePath() const;
00936
00937     EmbReal objectRadiusMajor() const { return qMax(rect().width(), rect().height())/2.0*scale(); }
00938     EmbReal objectRadiusMinor() const { return qMin(rect().width(), rect().height())/2.0*scale(); }
00939     EmbReal objectDiameterMajor() const { return qMax(rect().width(), rect().height())*scale(); }
00940     EmbReal objectDiameterMinor() const { return qMin(rect().width(), rect().height())*scale(); }
00941     EmbReal objectWidth() const { return rect().width()*scale(); }
00942     EmbReal objectHeight() const { return rect().height()*scale(); }
00943     QPointF objectQuadrant0() const;
00944     QPointF objectQuadrant90() const;
00945     QPointF objectQuadrant180() const;
00946     QPointF objectQuadrant270() const;
00947
00948     void setObjectSize(EmbReal width, EmbReal height);
00949     void setObjectRadiusMajor(EmbReal radius);
00950     void setObjectRadiusMinor(EmbReal radius);
00951     void setObjectDiameterMajor(EmbReal diameter);
00952     void setObjectDiameterMinor(EmbReal diameter);
00953
00954     void updateRubber(QPainter* painter = 0);
00955     virtual void vulcanize();
00956     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
00957     virtual QList<QPointF> allGripPoints();
00958     virtual void gripEdit(const QPointF& before, const QPointF& after);
00959
00960 protected:
00961     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
00962 };
00963
00964
00965 class ImageObject : public BaseObject
00966 {
00967 public:
00968     ImageObject(EmbReal x, EmbReal y, EmbReal w, EmbReal h, QRgb rgb, QGraphicsItem* parent = 0);
00969     ImageObject(ImageObject* obj, QGraphicsItem* parent = 0);
00970     ~ImageObject();
00971
00972     void init(EmbReal x, EmbReal y, EmbReal w, EmbReal h, QRgb rgb, Qt::PenStyle lineType);
00973     void updatePath();
00974
00975     enum { Type = OBJ_TYPE_IMAGE };
00976     virtual int type() const { return Type; }
00977
00978     QPointF objectTopLeft() const;
00979     QPointF objectTopRight() const;
00980     QPointF objectBottomLeft() const;
00981     QPointF objectBottomRight() const;
00982     EmbReal objectWidth() const { return rect().width()*scale(); }
00983     EmbReal objectHeight() const { return rect().height()*scale(); }
00984     EmbReal objectArea() const { return qAbs(objectWidth()*objectHeight()); }
00985
00986     void setObjectRect(EmbReal x, EmbReal y, EmbReal w, EmbReal h);
00987
00988     void updateRubber(QPainter* painter = 0);
00989     virtual void vulcanize();
00990     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
00991     virtual QList<QPointF> allGripPoints();
00992     virtual void gripEdit(const QPointF& before, const QPointF& after);
00993
00994 protected:
00995     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
00996 };
00997
00998 };
00999 
```

```

01000
01004 class LineObject : public BaseObject
01005 {
01006     public:
01007         LineObject(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, QRgb rgb, QGraphicsItem* parent = 0);
01008         LineObject(LineObject* obj, QGraphicsItem* parent = 0);
01009         ~LineObject();
01010
01011     void init(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, QRgb rgb, Qt::PenStyle lineType);
01012
01013     enum { Type = OBJ_TYPE_LINE };
01014     virtual int type() const { return Type; }
01015
01016     QPainterPath objectSavePath() const;
01017
01018     QPointF objectEndPoint1() const { return scenePos(); }
01019     QPointF objectEndPoint2() const;
01020     QPointF objectMidPoint() const;
01021     EmbReal objectX1() const { return objectEndPoint1().x(); }
01022     EmbReal objectY1() const { return objectEndPoint1().y(); }
01023     EmbReal objectX2() const { return objectEndPoint2().x(); }
01024     EmbReal objectY2() const { return objectEndPoint2().y(); }
01025     EmbReal objectDeltaX() const { return (objectX2() - objectX1()); }
01026     EmbReal objectDeltaY() const { return (objectY2() - objectY1()); }
01027     EmbReal objectAngle() const;
01028     EmbReal objectLength() const { return line().length()*scale(); }
01029
01030     void setObjectEndPoint1(const QPointF& endPt1);
01031     void setObjectEndPoint1(EmbReal x1, EmbReal y1);
01032     void setObjectEndPoint2(const QPointF& endPt2);
01033     void setObjectEndPoint2(EmbReal x2, EmbReal y2);
01034     void setObjectX1(EmbReal x) { setObjectEndPoint1(x, objectY1()); }
01035     void setObjectY1(EmbReal y) { setObjectEndPoint1(objectX1(), y); }
01036     void setObjectX2(EmbReal x) { setObjectEndPoint2(x, objectY2()); }
01037     void setObjectY2(EmbReal y) { setObjectEndPoint2(objectX2(), y); }
01038
01039     void updateRubber(QPainter* painter = 0);
01040     virtual void vulcanize();
01041     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
01042     virtual QList<QPointF> allGripPoints();
01043     virtual void gripEdit(const QPointF& before, const QPointF& after);
01044 protected:
01045     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
01046 };
01047
01048
01052 class PathObject : public BaseObject
01053 {
01054     public:
01055         PathObject(EmbReal x, EmbReal y, const QPainterPath p, QRgb rgb, QGraphicsItem* parent = 0);
01056         PathObject(PathObject* obj, QGraphicsItem* parent = 0);
01057         ~PathObject();
01058
01059     enum { Type = OBJ_TYPE_PATH };
01060     virtual int type() const { return Type; }
01061
01062     void init(EmbReal x, EmbReal y, const QPainterPath& p, QRgb rgb, Qt::PenStyle lineType);
01063     void updatePath(const QPainterPath& p);
01064     QPainterPath normalPath;
01065     //TODO: make paths similar to polylines. Review and implement any missing functions/members.
01066
01067     QPainterPath objectCopyPath() const;
01068     QPainterPath objectSavePath() const;
01069
01070     QPointF objectPos() const { return scenePos(); }
01071     EmbReal objectX() const { return scenePos().x(); }
01072     EmbReal objectY() const { return scenePos().y(); }
01073
01074     void setObjectPos(const QPointF& point) { setPos(point.x(), point.y()); }
01075     void setObjectPos(EmbReal x, EmbReal y) { setPos(x, y); }
01076     void setObjectX(EmbReal x) { setObjectPos(x, objectY()); }
01077     void setObjectY(EmbReal y) { setObjectPos(objectX(), y); }
01078
01079     void updateRubber(QPainter* painter = 0);
01080     virtual void vulcanize();
01081     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
01082     virtual QList<QPointF> allGripPoints();
01083     virtual void gripEdit(const QPointF& before, const QPointF& after);
01084 protected:
01085     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
01086 };
01087
01088
01092 class PointObject : public BaseObject
01093 {
01094     public:
01095         PointObject(EmbReal x, EmbReal y, QRgb rgb, QGraphicsItem* parent = 0);

```

```

01096 PointObject(PointObject* obj, QGraphicsItem* parent = 0);
01097 ~PointObject();
01098
01099 void init(EmbReal x, EmbReal y, QRgb rgb, Qt::PenStyle lineType);
01100
01101 enum { Type = OBJ_TYPE_POINT };
01102 virtual int type() const { return Type; }
01103
01104 QPainterPath objectSavePath() const;
01105
01106 QPointF objectPos() const { return scenePos(); }
01107 EmbReal objectX() const { return scenePos().x(); }
01108 EmbReal objectY() const { return scenePos().y(); }
01109
01110 void setObjectPos(const QPointF& point) { setPos(point.x(), point.y()); }
01111 void setObjectPos(EmbReal x, EmbReal y) { setPos(x, y); }
01112 void setObjectX(EmbReal x) { setObjectPos(x, objectY()); }
01113 void setObjectY(EmbReal y) { setObjectPos(objectX(), y); }
01114
01115 void updateRubber(QPainter* painter = 0);
01116 virtual void vulcanize();
01117 virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
01118 virtual QList<QPointF> allGripPoints();
01119 virtual void gripEdit(const QPointF& before, const QPointF& after);
01120 protected:
01121     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
01122 };
01123
01124
01125 class PolygonObject : public BaseObject
01126 {
01127 public:
01128     PolygonObject(EmbReal x, EmbReal y, const QPainterPath& p, QRgb rgb, QGraphicsItem* parent = 0);
01129     PolygonObject(PolygonObject* obj, QGraphicsItem* parent = 0);
01130     ~PolygonObject();
01131
01132     enum { Type = OBJ_TYPE_POLYGON };
01133     virtual int type() const { return Type; }
01134
01135     void init(EmbReal x, EmbReal y, const QPainterPath& p, QRgb rgb, Qt::PenStyle lineType);
01136     void updatePath(const QPainterPath& p);
01137     QPainterPath normalPath;
01138     int findIndex(const QPointF& point);
01139     int gripIndex;
01140
01141     QPainterPath objectCopyPath() const;
01142     QPainterPath objectSavePath() const;
01143
01144     QPointF objectPos() const { return scenePos(); }
01145     EmbReal objectX() const { return scenePos().x(); }
01146     EmbReal objectY() const { return scenePos().y(); }
01147
01148     void setObjectPos(const QPointF& point) { setPos(point.x(), point.y()); }
01149     void setObjectPos(EmbReal x, EmbReal y) { setPos(x, y); }
01150     void setObjectX(EmbReal x) { setObjectPos(x, objectY()); }
01151     void setObjectY(EmbReal y) { setObjectPos(objectX(), y); }
01152
01153     void updateRubber(QPainter* painter = 0);
01154     virtual void vulcanize();
01155     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
01156     virtual QList<QPointF> allGripPoints();
01157     virtual void gripEdit(const QPointF& before, const QPointF& after);
01158 protected:
01159     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
01160 };
01161
01162
01163 class PolylineObject : public BaseObject
01164 {
01165 public:
01166     PolylineObject(EmbReal x, EmbReal y, const QPainterPath& p, QRgb rgb, QGraphicsItem* parent = 0);
01167     PolylineObject(PolylineObject* obj, QGraphicsItem* parent = 0);
01168     ~PolylineObject();
01169
01170     enum { Type = OBJ_TYPE_POLYLINE };
01171     virtual int type() const { return Type; }
01172
01173     void init(EmbReal x, EmbReal y, const QPainterPath& p, QRgb rgb, Qt::PenStyle lineType);
01174     void updatePath(const QPainterPath& p);
01175     QPainterPath normalPath;
01176     int findIndex(const QPointF& point);
01177     int gripIndex;
01178
01179     QPainterPath objectCopyPath() const;
01180     QPainterPath objectSavePath() const;
01181
01182     QPointF objectPos() const { return scenePos(); }
01183     EmbReal objectX() const { return scenePos().x(); }
01184
01185     void setObjectPos(const QPointF& point) { setPos(point.x(), point.y()); }
01186     void setObjectPos(EmbReal x, EmbReal y) { setPos(x, y); }
01187     void setObjectX(EmbReal x) { setObjectPos(x, objectY()); }
01188     void setObjectY(EmbReal y) { setObjectPos(objectX(), y); }
01189
01190     void updateRubber(QPainter* painter = 0);
01191     virtual void vulcanize();
01192     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
01193     virtual QList<QPointF> allGripPoints();
01194     virtual void gripEdit(const QPointF& before, const QPointF& after);
01195 protected:
01196     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
01197 };

```

```

01189     EmbReal objectY() const { return scenePos().y(); }
01190
01191     void setObjectPos(const QPointF& point) { setPos(point.x(), point.y()); }
01192     void setObjectPos(EmbReal x, EmbReal y) { setPos(x, y); }
01193     void setObjectX(EmbReal x) { setObjectPos(x, objectY()); }
01194     void setObjectY(EmbReal y) { setObjectPos(objectX(), y); }
01195
01196     void updateRubber(QPainter* painter = 0);
01197     virtual void vulcanize();
01198     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
01199     virtual QList<QPointF> allGripPoints();
01200     virtual void gripEdit(const QPointF& before, const QPointF& after);
01201 protected:
01202     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
01203 };
01204
01205 class RectObject : public BaseObject
01206 {
01207 public:
01208     RectObject(EmbReal x, EmbReal y, EmbReal w, EmbReal h, QRgb rgb, QGraphicsItem* parent = 0);
01209     RectObject(RectObject* obj, QGraphicsItem* parent = 0);
01210     ~RectObject();
01211
01212     enum { Type = OBJ_TYPE_RECTANGLE };
01213     virtual int type() const { return Type; }
01214
01215     QPainterPath objectSavePath() const;
01216
01217     void init(EmbReal x, EmbReal y, EmbReal w, EmbReal h, QRgb rgb, Qt::PenStyle lineType);
01218     void updatePath();
01219
01220     QPointF objectPos() const { return scenePos(); }
01221
01222     QPointF objectTopLeft() const;
01223     QPointF objectTopRight() const;
01224     QPointF objectBottomLeft() const;
01225     QPointF objectBottomRight() const;
01226     EmbReal objectWidth() const { return rect().width()*scale(); }
01227     EmbReal objectHeight() const { return rect().height()*scale(); }
01228     EmbReal objectArea() const { return qAbs(objectWidth()*objectHeight()); }
01229
01230     void setObjectRect(EmbReal x, EmbReal y, EmbReal w, EmbReal h);
01231
01232     void updateRubber(QPainter* painter = 0);
01233     virtual void vulcanize();
01234     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
01235     virtual QList<QPointF> allGripPoints();
01236     virtual void gripEdit(const QPointF& before, const QPointF& after);
01237 protected:
01238     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
01239 };
01240
01241 class SaveObject : public QObject
01242 {
01243     Q_OBJECT
01244
01245 public:
01246     SaveObject(QGraphicsScene* theScene, QObject* parent = 0);
01247     ~SaveObject();
01248
01249     bool save(const QString &fileName);
01250
01251     void addArc          (EmbPattern* pattern, QGraphicsItem* item);
01252     void addBlock         (EmbPattern* pattern, QGraphicsItem* item);
01253     void addCircle        (EmbPattern* pattern, QGraphicsItem* item);
01254     void addDimAligned    (EmbPattern* pattern, QGraphicsItem* item);
01255     void addDimAngular   (EmbPattern* pattern, QGraphicsItem* item);
01256     void addDimArcLength (EmbPattern* pattern, QGraphicsItem* item);
01257     void addDimDiameter  (EmbPattern* pattern, QGraphicsItem* item);
01258     void addDimLeader    (EmbPattern* pattern, QGraphicsItem* item);
01259     void addDimLinear    (EmbPattern* pattern, QGraphicsItem* item);
01260     void addDimOrdinate  (EmbPattern* pattern, QGraphicsItem* item);
01261     void addDimRadius    (EmbPattern* pattern, QGraphicsItem* item);
01262     void addEllipse       (EmbPattern* pattern, QGraphicsItem* item);
01263     void addEllipseArc   (EmbPattern* pattern, QGraphicsItem* item);
01264     void addGrid          (EmbPattern* pattern, QGraphicsItem* item);
01265     void addHatch         (EmbPattern* pattern, QGraphicsItem* item);
01266     void addImage         (EmbPattern* pattern, QGraphicsItem* item);
01267     void addInfiniteLine (EmbPattern* pattern, QGraphicsItem* item);
01268     void addLine          (EmbPattern* pattern, QGraphicsItem* item);
01269     void addPath          (EmbPattern* pattern, QGraphicsItem* item);
01270     void addPoint         (EmbPattern* pattern, QGraphicsItem* item);
01271     void addPolygon       (EmbPattern* pattern, QGraphicsItem* item);
01272     void addPolyline      (EmbPattern* pattern, QGraphicsItem* item);
01273     void addRay           (EmbPattern* pattern, QGraphicsItem* item);
01274     void addRectangle     (EmbPattern* pattern, QGraphicsItem* item);
01275     void addSlot          (EmbPattern* pattern, QGraphicsItem* item);

```

```

01282     void addSpline      (EmbPattern* pattern, QGraphicsItem* item);
01283     void addTextMulti   (EmbPattern* pattern, QGraphicsItem* item);
01284     void addTextSingle  (EmbPattern* pattern, QGraphicsItem* item);
01285
01286     QGraphicsScene* gscene;
01287     int formatType;
01288
01289     void toPolyline(EmbPattern* pattern, const QPointF& objPos, const QPainterPath& objPath, const
01290                      QString& layer, const QColor& color, const QString& lineType, const QString& lineWeight);
01290 };
01291
01292 class TextSingleObject : public BaseObject
01293 {
01294 public:
01295     TextSingleObject(const QString& str, EmbReal x, EmbReal y, QRgb rgb, QGraphicsItem* parent = 0);
01296     TextSingleObject(TextSingleObject* obj, QGraphicsItem* parent = 0);
01297     ~TextSingleObject();
01298
01299     enum { Type = OBJ_TYPE_TEXTSINGLE };
01300     virtual int type() const { return Type; }
01301
01302     void init(const QString& str, EmbReal x, EmbReal y, QRgb rgb, Qt::PenStyle lineType);
01303
01304     QString objText;
01305     QString objTextFont;
01306     QString objTextJustify;
01307     EmbReal objTextSize;
01308     bool objTextBold;
01309     bool objTextItalic;
01310     bool objTextUnderline;
01311     bool objTextStrikeOut;
01312     bool objTextOverline;
01313     bool objTextBackward;
01314     bool objTextUpsideDown;
01315     QPainterPath objTextPath;
01316
01317     QList<QPainterPath> objectSavePathList() const { return subPathList(); }
01318     QList<QPainterPath> subPathList() const;
01319
01320     QPointF objectPos() const { return scenePos(); }
01321     EmbReal objectX() const { return scenePos().x(); }
01322     EmbReal objectY() const { return scenePos().y(); }
01323
01324     QStringList objectTextJustifyList() const;
01325
01326     void setObjectText(const QString& str);
01327     void setObjectTextFont(const QString& font);
01328     void setObjectTextJustify(const QString& justify);
01329     void setObjectTextSize(EmbReal size);
01330     void setObjectTextStyle(bool bold, bool italic, bool under, bool strike, bool over);
01331     void setObjectTextBold(bool val);
01332     void setObjectTextItalic(bool val);
01333     void setObjectTextUnderline(bool val);
01334     void setObjectTextStrikeOut(bool val);
01335     void setObjectTextOverline(bool val);
01336     void setObjectTextBackward(bool val);
01337     void setObjectTextUpsideDown(bool val);
01338
01339     void setObjectPos(const QPointF& point) { setPos(point.x(), point.y()); }
01340     void setObjectPos(EmbReal x, EmbReal y) { setPos(x, y); }
01341     void setObjectX(EmbReal x) { setObjectPos(x, objectY()); }
01342     void setObjectY(EmbReal y) { setObjectPos(objectX(), y); }
01343
01344     void updateRubber(QPainter* painter = 0);
01345     virtual void vulcanize();
01346     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
01347     virtual QList<QPointF> allGripPoints();
01348     virtual void gripEdit(const QPointF& before, const QPointF& after);
01349
01350 protected:
01351     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
01352 };
01353
01354 class Application : public QApplication
01355 {
01356     Q_OBJECT
01357 public:
01358     Application(int argc, char **argv);
01359     void setMainWin(MainWindow* mainWin) { _mainWin = mainWin; }
01360     MainWindow* _mainWin;
01361
01362 protected:
01363     virtual bool event(QEvent *e);
01364 };
01365
01366 class CmdPromptInput : public QLineEdit
01367 {
01368     Q_OBJECT
01369
01370
01371
01372
01373
01374
01375
01376
01377
01378

```

```
01379 public:
01380     CmdPromptInput(QWidget* parent = 0);
01381     ~CmdPromptInput();
01382
01383     QString curText;
01384     QString defaultPrefix;
01385     QString prefix;
01386
01387     QString lastCmd;
01388     QString curCmd;
01389     bool cmdActive;
01390
01391     bool rapidFireEnabled;
01392     bool isBlinking;
01393
01394     QHash<QString, QString>* aliasHash;
01395
01396     void changeFormatting(const QList<QTextLayout::FormatRange>& formats);
01397     void clearFormatting();
01398     void applyFormatting();
01399
01400 protected:
01401     void contextMenuEvent(QContextMenuEvent *event);
01402     bool eventFilter(QObject *obj, QEvent *event);
01403
01404 signals:
01405     void appendHistory(const QString& txt, int prefixLength);
01406
01407     //These connect to the CmdPrompt signals
01408     void startCommand(const QString& cmd);
01409     void runCommand(const QString& cmd, const QString& cmdtxt);
01410     void deletePressed();
01411     void tabPressed();
01412     void escapePressed();
01413     void upPressed();
01414     void downPressed();
01415     void F1Pressed();
01416     void F2Pressed();
01417     void F3Pressed();
01418     void F4Pressed();
01419     void F5Pressed();
01420     void F6Pressed();
01421     void F7Pressed();
01422     void F8Pressed();
01423     void F9Pressed();
01424     void F10Pressed();
01425     void F11Pressed();
01426     void F12Pressed();
01427     void cutPressed();
01428     void copyPressed();
01429     void pastePressed();
01430     void selectAllPressed();
01431     void undoPressed();
01432     void redoPressed();
01433
01434     void shiftPressed();
01435     void shiftReleased();
01436
01437     void showSettings();
01438
01439     void stopBlinking();
01440
01441 public slots:
01442     void addCommand(const QString& alias, const QString& cmd);
01443     void endCommand();
01444     void processInput(void);
01445     void checkSelection();
01446     void updateCurrentText(const QString& txt);
01447     void checkEditedText(const QString& txt);
01448     void checkChangedText(const QString& txt);
01449     void checkCursorPosition(int oldpos, int newpos);
01450 private slots:
01451     void copyClip();
01452     void pasteClip();
01453 };
01454
01455 class CmdPromptHistory : public QTextBrowser
01456 {
01457     Q_OBJECT
01458
01459     public:
01460         CmdPromptHistory(QWidget* parent = 0);
01461         ~CmdPromptHistory();
01462
01463         int tmpHeight;
01464         QString applyFormatting(const QString& txt, int prefixLength);
01465
01466
01467
01468
```

```

01469 protected:
01470     void contextMenuEvent(QContextMenuEvent* event);
01471
01472 public slots:
01473     void appendHistory(const QString& txt, int prefixLength);
01474     void startResizeHistory(int y);
01475     void stopResizeHistory(int y);
01476     void resizeHistory(int y);
01477
01478 signals:
01479     void historyAppended(const QString& txt);
01480 };
01481
01482 class CmdPromptSplitter : public QSplitter
01483 {
01484     Q_OBJECT
01485
01486 public:
01487     CmdPromptSplitter(QWidget* parent = 0);
01488     ~CmdPromptSplitter();
01489
01490 protected:
01491     QSplitterHandle* createHandle();
01492
01493 signals:
01494     void pressResizeHistory(int y);
01495     void releaseResizeHistory(int y);
01496     void moveResizeHistory(int y);
01497 };
01498
01499 class CmdPromptHandle : public QSplitterHandle
01500 {
01501     Q_OBJECT
01502
01503 public:
01504     CmdPromptHandle(Qt::Orientation orientation, QSplitter* parent);
01505     ~CmdPromptHandle();
01506
01507     int pressY;
01508     int releaseY;
01509     int moveY;
01510
01511 protected:
01512     void mousePressEvent(QMouseEvent* e);
01513     void mouseReleaseEvent(QMouseEvent* e);
01514     void mouseMoveEvent(QMouseEvent* e);
01515
01516 signals:
01517     void handlePressed(int y);
01518     void handleReleased(int y);
01519     void handleMoved(int y);
01520
01521 };
01522
01523 class CmdPrompt : public QWidget
01524 {
01525     Q_OBJECT
01526
01527 public:
01528     CmdPrompt(QWidget* parent = 0);
01529     ~CmdPrompt();
01530
01531     CmdPromptInput* promptInput;
01532     CmdPromptHistory* promptHistory;
01533     QVBoxLayout* promptVBoxLayout;
01534     QFrame* promptDivider;
01535
01536     CmdPromptSplitter* promptSplitter;
01537
01538     QHash<QString, QString>* styleHash;
01539     void updateStyle();
01540     QTimer* blinkTimer;
01541     bool blinkState;
01542
01543 protected:
01544
01545 public slots:
01546     QString getHistory() { return promptHistory->toHtml(); }
01547     QString getPrefix() { return promptInput->prefix; }
01548     QString getCurrentText() { return promptInput->curText; }
01549     void setCurrentText(const QString& txt) { promptInput->curText = promptInput->prefix + txt;
01550         promptInput->setText(promptInput->curText); }
01551     void setHistory(const QString& txt) { promptHistory->setHtml(txt);
01552         promptHistory->moveCursor(QTextCursor::End, QTextCursor::MoveAnchor); }
01553     void setPrefix(const QString& txt);
01554     void appendHistory(const QString& txt);
01555     void startResizingTheHistory(int y) { promptHistory->startResizeHistory(y); }
01556     void stopResizingTheHistory(int y) { promptHistory->stopResizeHistory(y); }

```

```
01563     void resizeTheHistory(int y) { promptHistory->resizeHistory(y); }
01564     void addCommand(const QString& alias, const QString& cmd) { promptInput->addCommand(alias, cmd); }
01565     void endCommand() { promptInput->endCommand(); }
01566     bool isCommandActive() { return promptInput->cmdActive; }
01567     QString activeCommand() { return promptInput->curCmd; }
01568     QString lastCommand() { return promptInput->lastCmd; }
01569     void processInput() { promptInput->processInput(); }
01570     void enableRapidFire() { promptInput->rapidFireEnabled = true; }
01571     void disableRapidFire() { promptInput->rapidFireEnabled = false; }
01572     bool isRapidFireEnabled() { return promptInput->rapidFireEnabled; }
01573
01574     void alert(const QString& txt);
01575
01576     void startBlinking();
01577     void stopBlinking();
01578     void blink();
01579
01580     void setPromptTextColor(const QColor&);
01581     void setPromptBackgroundColor(const QColor&);
01582     void setPromptFontFamily(const QString&);
01583     void setPromptFontStyle(const QString&);
01584     void setPromptFontSize(int);
01585
01586     void floatingChanged(bool);
01587
01588     void saveHistory(const QString& fileName, bool html);
01589
01590 private slots:
01591
01592 signals:
01593     void appendTheHistory(const QString& txt, int prefixLength);
01594
01595 //For connecting outside of command prompt
01596     void startCommand(const QString& cmd);
01597     void runCommand(const QString& cmd, const QString& cmdtxt);
01598     void deletePressed();
01599     void tabPressed();
01600     void escapePressed();
01601     void upPressed();
01602     void downPressed();
01603     void F1Pressed();
01604     void F2Pressed();
01605     void F3Pressed();
01606     void F4Pressed();
01607     void F5Pressed();
01608     void F6Pressed();
01609     void F7Pressed();
01610     void F8Pressed();
01611     void F9Pressed();
01612     void F10Pressed();
01613     void F11Pressed();
01614     void F12Pressed();
01615     void cutPressed();
01616     void copyPressed();
01617     void pastePressed();
01618     void selectAllPressed();
01619     void undoPressed();
01620     void redoPressed();
01621
01622     void shiftPressed();
01623     void shiftReleased();
01624
01625     void showSettings();
01626
01627     void historyAppended(const QString& txt);
01628 };
01629
01630 class EmbDetailsDialog : public QDialog
01631 {
01632     Q_OBJECT
01633
01634 public:
01635     EmbDetailsDialog(QGraphicsScene* theScene, QWidget *parent = 0);
01636     ~EmbDetailsDialog();
01637
01638     QWidget* mainWidget;
01639
01640     void getInfo();
01641     QWidget* createMainWidget();
01642     QWidget* createHistogram();
01643
01644     QDialButtonBox* buttonBox;
01645
01646     quint32 stitchesTotal;
01647     quint32 stitchesReal;
01648     quint32 stitchesJump;
01649     quint32 stitchesTrim;
```

```

01653     quint32 colorTotal;
01654     quint32 colorChanges;
01655
01656     QRectF boundingRect;
01657 };
01658
01659 class ImageWidget : public QWidget
01660 {
01661     Q_OBJECT
01662
01663     public:
01664         QImage img;
01665         ImageWidget(const QString &filename, QWidget* parent = 0);
01666         ~ImageWidget();
01667
01668     bool load(const QString &fileName);
01669     bool save(const QString &fileName);
01670
01671     protected:
01672         void paintEvent(QPaintEvent* event);
01673 };
01674
01675 class LayerManager : public QDialog
01676 {
01677     Q_OBJECT
01678
01679     public:
01680         QStandardItemModel* layerModel;
01681         QSortFilterProxyModel* layerModelSorted;
01682         QTreeView* treeView;
01683
01684         LayerManager(MainWindow* mw, QWidget *parent = 0);
01685         ~LayerManager();
01686
01687     void addLayer(const QString& name,
01688                 const bool visible,
01689                 const bool frozen,
01690                 const EmbReal zValue,
01691                 const QRgb color,
01692                 const QString& lineType,
01693                 const QString& lineWeight,
01694                 const bool print);
01695 };
01696
01697 class MainWindow: public QMainWindow
01698 {
01699     Q_OBJECT
01700
01701     public:
01702         MainWindow();
01703         ~MainWindow();
01704
01705         QString settings_general_language;
01706         QString settings_general_icon_theme;
01707         int settings_general_icon_size;
01708         bool settings_general_mdi_bg_use_logo;
01709         bool settings_general_mdi_bg_use_texture;
01710         bool settings_general_mdi_bg_use_color;
01711         QString settings_general_mdi_bg_logo;
01712         QString settings_general_mdi_bg_texture;
01713         QRgb settings_general_mdi_bg_color;
01714         bool settings_general_tip_of_the_day;
01715         quint16 settings_general_current_tip;
01716         bool settings_general_system_help_browser;
01717         bool settings_general_check_for_updates;
01718         bool settings_display_use_opengl;
01719         bool settings_display_renderhint_aa;
01720         bool settings_display_renderhint_text_aa;
01721         bool settings_display_renderhint_smooth_pix;
01722         bool settings_display_renderhint_high_aa;
01723         bool settings_display_renderhint_noncosmetic;
01724         bool settings_display_show_scrollbars;
01725         int settings_display_scrollbar_widget_num;
01726         QRgb settings_display_crosshair_color;
01727         QRgb settings_display_bg_color;
01728         QRgb settings_display_selectbox_left_color;
01729         QRgb settings_display_selectbox_left_fill;
01730         QRgb settings_display_selectbox_right_color;
01731         QRgb settings_display_selectbox_right_fill;
01732         uint8_t settings_display_selectbox_alpha;
01733         EmbReal settings_display_zoomscale_in;
01734         EmbReal settings_display_zoomscale_out;
01735         uint8_t settings_display_crosshair_percent;
01736         QString settings_display_units;
01737         QRgb settings_prompt_text_color;
01738         QRgb settings_prompt_bg_color;
01739         QString settings_prompt_font_family;

```

```
01749     QString settings_prompt_font_style;
01750     uint8_t settings_prompt_font_size;
01751     bool settings_prompt_save_history;
01752     bool settings_prompt_save_history_as_html;
01753     QString settings_prompt_save_history_filename;
01754     QString settings_opensave_custom_filter;
01755     QString settings_opensave_open_format;
01756     bool settings_opensave_open_thumbnail;
01757     QString settings_opensave_save_format;
01758     bool settings_opensave_save_thumbnail;
01759     uint8_t settings_opensave_recent_max_files;
01760     QStringList settings_opensave_recent_list_of_files;
01761     QString settings_opensave_recent_directory;
01762     uint8_t settings_opensave_trim_dst_num_jumps;
01763     QString settings_printing_default_device;
01764     bool settings_printing_use_last_device;
01765     bool settings_printing_disable_bg;
01766     bool settings_grid_show_on_load;
01767     bool settings_grid_show_origin;
01768     bool settings_grid_color_match_crosshair;
01769     QRgb settings_grid_color;
01770     bool settings_grid_load_from_file;
01771     QString settings_grid_type;
01772     bool settings_grid_center_on_origin;
01773     EmbReal settings_grid_center_x;
01774     EmbReal settings_grid_center_y;
01775     EmbReal settings_grid_size_x;
01776     EmbReal settings_grid_size_y;
01777     EmbReal settings_grid_spacing_x;
01778     EmbReal settings_grid_spacing_y;
01779     EmbReal settings_grid_size_radius;
01780     EmbReal settings_grid_spacing_radius;
01781     EmbReal settings_grid_spacing_angle;
01782     bool settings_ruler_show_on_load;
01783     bool settings_ruler_metric;
01784     QRgb settings_ruler_color;
01785     uint8_t settings_ruler_pixel_size;
01786     bool settings_qsnap_enabled;
01787     QRgb settings_qsnap_locator_color;
01788     uint8_t settings_qsnap_locator_size;
01789     uint8_t settings_qsnap_aperture_size;
01790     bool settings_qsnap_endpoint;
01791     bool settings_qsnap_midpoint;
01792     bool settings_qsnap_center;
01793     bool settings_qsnap_node;
01794     bool settings_qsnap_quadrant;
01795     bool settings_qsnap_intersection;
01796     bool settings_qsnap_extension;
01797     bool settings_qsnap_insertion;
01798     bool settings_qsnap_perpendicular;
01799     bool settings_qsnap_tangent;
01800     bool settings_qsnap_nearest;
01801     bool settings_qsnap_apparent;
01802     bool settings_qsnap_parallel;
01803     bool settings_lwt_show_lwt;
01804     bool settings_lwt_real_render;
01805     EmbReal settings_lwt_default_lwt;
01806     bool settings_selection_mode_pickfirst;
01807     bool settings_selection_mode_pickadd;
01808     bool settings_selection_mode_pickdrag;
01809     QRgb settings_selection_coolgrip_color;
01810     QRgb settings_selection_hotgrip_color;
01811     uint8_t settings_selection_grip_size;
01812     uint8_t settings_selection_pickbox_size;
01813     QString settings_text_font;
01814     EmbReal settings_text_size;
01815     EmbReal settings_text_angle;
01816     bool settings_text_style_bold;
01817     bool settings_text_style_italic;
01818     bool settings_text_style_underline;
01819     bool settings_text_style_overline;
01820     bool settings_text_style_strikeout;
01821
01822     MdiArea* getMdiArea();
01823     MainWindow* getApplication();
01824     MdiWindow* activeMdiWindow();
01825     View* activeView();
01826     QGraphicsScene* activeScene();
01827     QUndoStack* activeUndoStack();
01828
01829     void setUndoCleanIcon(bool opened);
01830
01831     virtual void updateMenuToolbarStatusbar();
01832
01833     MainWindow* mainWin;
01834     MdiArea* mdiArea;
01835     CmdPrompt* prompt;
```

```

01836     PropertyEditor* dockPropEdit;
01837     UndoEditor* dockUndoEdit;
01838     StatusBar* statusbar;
01839
01840     QList<QGraphicsItem*> cutCopyObjectList;
01841
01842     std::string actuator(std::string command);
01843     std::string run_script_file(std::string fname);
01844     std::string run_script(std::vector<std::string> script);
01845     void LoadCommand(QString cmdName);
01846
01847     QAction* actionHash[200];
01848     QHash<QString, QToolBar*> toolbarHash;
01849     QHash<QString, QMenu*> menuHash;
01850
01851     QString formatFilterOpen;
01852     QString formatFilterSave;
01853
01854     bool isCommandActive() { return prompt->isCommandActive(); }
01855     QString activeCommand() { return prompt->activeCommand(); }
01856     QIcon create_icon(QString stub);
01857     void create_toolbar(QToolBar* toolbar, std::string label, std::vector<std::string> entries);
01858
01859     QString platformString();
01860
01861 public slots:
01862
01863     void enablePromptRapidFire();
01864     void disablePromptRapidFire();
01865
01866     void enableMoveRapidFire();
01867     void disableMoveRapidFire();
01868
01869     void onCloseWindow();
01870     virtual void onCloseMdiWin(MdiWindow*);
01871
01872     void recentMenuAboutToShow();
01873
01874     void onWindowActivated(QMdiSubWindow* w);
01875     void windowMenuAboutToShow();
01876     void windowMenuActivated( bool checked/*int id*/ );
01877     QAction*           getAction(int actionEnum);
01878
01879     void updateAllViewScrollBars(bool val);
01880     void updateAllViewCrossHairColors(QRgb color);
01881     void updateAllViewBackgroundColors(QRgb color);
01882     void updateAllViewSelectBoxColors(QRgb colorL, QRgb fillL, QRgb colorR, QRgb fillR, int alpha);
01883     void updateAllViewGridColors(QRgb color);
01884     void updateAllViewRulerColors(QRgb color);
01885
01886     void updatePickAddMode(bool val);
01887     void pickAddModeToggled();
01888
01889     void settingsPrompt();
01890
01891     void settingsDialog(const QString& showTab = QString());
01892     void readSettings();
01893     void writeSettings();
01894
01895     static bool validFileFormat(const QString &fileName);
01896
01897 protected:
01898     virtual void resizeEvent(QResizeEvent* );
01899     void closeEvent(QCloseEvent *event);
01900     QAction* getFileSeparator();
01901     void loadFormats();
01902
01903     bool shiftKeyPressedState;
01904
01905     QByteArray layoutState;
01906
01907     int numOfDocs;
01908     int docIndex;
01909
01910     QList<MdiWindow*> listMdiWin;
01911     QMdiSubWindow* findMdiWindow(const QString &fileName);
01912     QString openFilePath;
01913
01914     QAction* myFileSeparator;
01915
01916     QWizard* wizardTipOfDay;
01917     QLabel* labelTipOfDay;
01918     QCheckBox* checkBoxTipOfDay;
01919     QStringList listTipOfDay;
01920
01921     void createAllActions();
01922

```

```
01923 //Toolbars
01924 //=====
01925 void createAllToolbars();
01926 void createPanToolbar();
01927 void createIconToolbar();
01928 void createHelpToolbar();
01929 void createLayerToolbar();
01930 void createPropertiesToolbar();
01931 void createTextToolbar();
01932 void createPromptToolbar();
01933
01934 QToolBar* toolbarFile;
01935 QToolBar* toolbarEdit;
01936 QToolBar* toolbarView;
01937 QToolBar* toolbarZoom;
01938 QToolBar* toolbarPan;
01939 QToolBar* toolbarIcon;
01940 QToolBar* toolbarHelp;
01941 QToolBar* toolbarLayer;
01942 QToolBar* toolbarText;
01943 QToolBar* toolbarProperties;
01944 QToolBar* toolbarPrompt;
01945
01946 //Selectors
01947 //=====
01948 QComboBox* layerSelector;
01949 QComboBox* colorSelector;
01950 QComboBox* linetypeSelector;
01951 QComboBox* linewidthSelector;
01952 QFontComboBox* textFontSelector;
01953 QComboBox* textSizeSelector;
01954
01955 //Menus
01956 //=====
01957 void createAllMenus();
01958 void createFileMenu();
01959 void createEditMenu();
01960 void createViewMenu();
01961 void createSettingsMenu();
01962 void createWindowMenu();
01963 void createHelpMenu();
01964
01965 QMenu* fileMenu;
01966 QMenu* editMenu;
01967 QMenu* viewMenu;
01968 QMenu* settingsMenu;
01969 QMenu* windowMenu;
01970 QMenu* helpMenu;
01971
01972 //SubMenus
01973 //=====
01974 QMenu* recentMenu;
01975 QMenu* zoomMenu;
01976 QMenu* panMenu;
01977
01978 private slots:
01979     void hideUnimplemented();
01980
01981 public slots:
01982     void stub_implement(QString txt);
01983     void stub_testing();
01985
01986     void promptHistoryAppended(const QString& txt);
01987     void logPromptInput(const QString& txt);
01988     void promptInputPrevious();
01989     void promptInputNext();
01990
01991     void runCommand();
01992     void runCommandMain(const QString& cmd);
01993     void runCommandClick(const QString& cmd, EmbReal x, EmbReal y);
01994     void runCommandMove(const QString& cmd, EmbReal x, EmbReal y);
01995     void runCommandContext(const QString& cmd, const QString& str);
01996     void runCommandPrompt(const QString& cmd, const QString& str);
01997
01998     void newFile();
01999     void openFile(bool recent = false, const QString& recentFile = "");
02000     void openFileSelected(const QStringList& );
02001     void openrecentfile();
02002     void savefile();
02003     void saveasfile();
02004     void print();
02005     void designDetails();
02006     void exit();
02007     void quit();
02008     void checkForUpdates();
02009     // Help Menu
```

```
02010 void tipOfTheDay();
02011 void buttonTipOfTheDayClicked(int);
02012 void checkBoxTipOfTheDayStateChanged(int);
02013 void help();
02014 void changelog();
02015 void about();
02016 void whatsThisContextHelp();
02017
02018 void cut();
02019 void copy();
02020 void paste();
02021 void selectAll();
02022
02023 void closeToolBar(QAction* );
02024 void floatingChangedToolBar(bool);
02025
02026 void toggleGrid();
02027 void toggleRuler();
02028 void toggleLwt();
02029
02030 // Icons
02031 void iconResize(int iconSize);
02032
02033 //Selectors
02034 void layerSelectorIndexChanged(int index);
02035 void colorSelectorIndexChanged(int index);
02036 void linetypeSelectorIndexChanged(int index);
02037 void linewidthSelectorIndexChanged(int index);
02038 void textFontSelectorCurrentFontChanged(const QFont& font);
02039 void textSizeSelectorIndexChanged(int index);
02040
02041 QString textFont();
02042 EmbReal textSize();
02043 EmbReal textAngle();
02044 bool textBold();
02045 bool textItalic();
02046 bool textUnderline();
02047 bool textStrikeOut();
02048 bool textOverline();
02049
02050 void setTextFont(const QString& str);
02051 void setTextSize(EmbReal num);
02052 void setTextAngle(EmbReal num);
02053 void setTextBold(bool val);
02054 void setTextItalic(bool val);
02055 void setTextUnderline(bool val);
02056 void setTextStrikeOut(bool val);
02057 void setTextOverline(bool val);
02058
02059 QString getCurrentLayer();
02060 QRgb getCurrentColor();
02061 QString getCurrentLineType();
02062 QString getCurrentLineWeight();
02063
02064 // Standard Slots
02065 void undo();
02066 void redo();
02067
02068 bool isShiftPressed();
02069 void setShiftPressed();
02070 void setShiftReleased();
02071
02072 void deletePressed();
02073 void escapePressed();
02074
02075 // Layer Toolbar
02076 void makeLayerActive();
02077 void layerManager();
02078 void layerPrevious();
02079 // Zoom Toolbar
02080 void zoomRealtime();
02081 void zoomPrevious();
02082 void zoomWindow();
02083 void zoomDynamic();
02084 void zoomScale();
02085 void zoomCenter();
02086 void zoomIn();
02087 void zoomOut();
02088 void zoomSelected();
02089 void zoomAll();
02090 void zoomExtents();
02091 // Pan SubMenu
02092 void panrealtime();
02093 void panpoint();
02094 void panLeft();
02095 void panRight();
02096 void panUp();
```

```
02097     void panDown();
02098
02099     void dayVision();
02100     void nightVision();
02101
02102     void doNothing();
02103
02104 public:
02105     //Natives
02106     void nativeAlert(const QString& txt);
02107     void nativeBlinkPrompt();
02108     void nativeSetPromptPrefix(const QString& txt);
02109     void nativeAppendPromptHistory(const QString& txt);
02110     void nativeEnablePromptRapidFire();
02111     void nativeDisablePromptRapidFire();
02112     void nativeInitCommand();
02113     void nativeEndCommand();
02114
02115     void nativeEnableMoveRapidFire();
02116     void nativeDisableMoveRapidFire();
02117
02118     void nativeNewFile();
02119     void nativeOpenFile();
02120
02121     void nativeExit();
02122     void nativeTipOfTheDay();
02123     void nativeWindowCascade();
02124     void nativeWindowTile();
02125     void nativeWindowClose();
02126     void nativeWindowCloseAll();
02127     void nativeWindowNext();
02128     void nativeWindowPrevious();
02129
02130     void nativeMessageBox(const QString& type, const QString& title, const QString& text);
02131
02132     void nativePrintArea(EmbReal x, EmbReal y, EmbReal w, EmbReal h);
02133
02134     void nativeSetBackgroundColor(uint8_t r, uint8_t g, uint8_t b);
02135     void nativeSetCrossHairColor(uint8_t r, uint8_t g, uint8_t b);
02136     void nativeSetGridColor(uint8_t r, uint8_t g, uint8_t b);
02137
02138     QString nativeTextFont();
02139     EmbReal nativeTextSize();
02140     EmbReal nativeTextAngle();
02141     bool nativeTextBold();
02142     bool nativeTextItalic();
02143     bool nativeTextUnderline();
02144     bool nativeTextStrikeOut();
02145     bool nativeTextOverline();
02146
02147     void nativePreviewOn(int clone, int mode, EmbReal x, EmbReal y, EmbReal data);
02148     void nativePreviewOff();
02149
02150     void nativeVulcanize();
02151     void nativeClearRubber();
02152     bool nativeAllowRubber();
02153     void nativeSpareRubber(qint64 id);
02154     //TODO: void nativeSetRubberFilter(qint64 id); //TODO: This is so more than 1 rubber object can
02155     //exist at one time without updating all rubber objects at once
02156     void nativeSetRubberMode(int mode);
02157     void nativeSetRubberPoint(const QString& key, EmbReal x, EmbReal y);
02158
02159     void nativeAddTextMulti(const QString& str, EmbReal x, EmbReal y, EmbReal rot, bool fill, int
02160     rubberMode);
02161     void nativeAddTextSingle(const QString& str, EmbReal x, EmbReal y, EmbReal rot, bool fill, int
02162     rubberMode);
02163     void nativeAddInfiniteLine(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal rot);
02164     void nativeAddRay(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal rot);
02165     void nativeAddLine(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal rot, int rubberMode);
02166     void nativeAddTriangle(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal x3, EmbReal y3,
02167     EmbReal rot, bool fill);
02168     void nativeAddRectangle(EmbReal x, EmbReal y, EmbReal w, EmbReal h, EmbReal rot, bool fill, int
02169     rubberMode);
02170     void nativeAddRoundedRectangle(EmbReal x, EmbReal y, EmbReal w, EmbReal h, EmbReal rad, EmbReal
02171     rot, bool fill);
02172     void nativeAddArc(EmbReal startX, EmbReal startY, EmbReal midX, EmbReal midY, EmbReal endX,
02173     EmbReal endY, int rubberMode);
02174     void nativeAddCircle(EmbReal centerX, EmbReal centerY, EmbReal radius, bool fill, int rubberMode);
02175     void nativeAddSlot(EmbReal centerX, EmbReal centerY, EmbReal diameter, EmbReal length, EmbReal
02176     rot, bool fill, int rubberMode);
02177     void nativeAddEllipse(EmbReal centerX, EmbReal centerY, EmbReal width, EmbReal height, EmbReal
02178     rot, bool fill, int rubberMode);
02179     void nativeAddPoint(EmbReal x, EmbReal y);
02180     void nativeAddRegularPolygon(EmbReal centerX, EmbReal centerY, quint16 sides, uint8_t mode,
02181     EmbReal rad, EmbReal rot, bool fill);
```

```

02174     void nativeAddPolygon(EmbReal startX, EmbReal startY, const QPainterPath& p, int rubberMode);
02175     void nativeAddPolyline(EmbReal startX, EmbReal startY, const QPainterPath& p, int rubberMode);
02176     void nativeAddPath(EmbReal startX, EmbReal startY, const QPainterPath& p, int rubberMode);
02177     void nativeAddHorizontalDimension(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal
02178         legHeight);
02179     void nativeAddVerticalDimension(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal
02180         legHeight);
02181     void nativeAddImage(const QString& img, EmbReal x, EmbReal y, EmbReal w, EmbReal h, EmbReal rot);
02182     void nativeAddDimLeader(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal rot, int
02183         rubberMode);
02184     void nativeSetCursorShape(const QString& str);
02185     EmbReal nativeCalculateAngle(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2);
02186     EmbReal nativeCalculateDistance(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2);
02187     EmbReal nativePerpendicularDistance(EmbReal px, EmbReal py, EmbReal x1, EmbReal y1, EmbReal x2,
02188         EmbReal y2);
02189     int nativeNumSelected();
02190     void nativeSelectAll();
02191     void nativeAddToSelection(const QPainterPath path, Qt::ItemSelectionMode mode);
02192     void nativeClearSelection();
02193     void nativeDeleteSelected();
02194     void nativeCutSelected(EmbReal x, EmbReal y);
02195     void nativeCopySelected(EmbReal x, EmbReal y);
02196     void nativePasteSelected(EmbReal x, EmbReal y);
02197     void nativeMoveSelected(EmbReal dx, EmbReal dy);
02198     void nativeScaleSelected(EmbReal x, EmbReal y, EmbReal factor);
02199     void nativeRotateSelected(EmbReal x, EmbReal y, EmbReal rot);
02200     void nativeMirrorSelected(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2);
02201     EmbReal nativeQSnapX();
02202     EmbReal nativeQSnapY();
02203     EmbReal nativeMouseX();
02204     EmbReal nativeMouseY();
02205 };
02206
02207 MainWindow* mainWin();
02208
02209
02210 class MdiWindow: public QMdiSubWindow
02211 {
02212     Q_OBJECT
02213
02214 public:
02215     MdiWindow(const int theIndex, MainWindow* mw, QMdiArea* parent, Qt::WindowFlags wflags);
02216     ~MdiWindow();
02217
02218     virtual QSize           sizeHint() const;
02219     QString    getCurrentFile() { return curFile; }
02220     QString    getShortCurrentFile();
02221     View*      getView() { return gview; }
02222     QGraphicsScene*   getScene() { return gscene; }
02223     QString    getCurrentLayer() { return curLayer; }
02224     QRgb      getCurrentColor() { return curColor; }
02225     QString    getCurrentLineType() { return curLineType; }
02226     QString    getCurrentLineWidth() { return curLineWidth; }
02227     void      setCurrentLayer(const QString& layer) { curLayer = layer; }
02228     void      setCurrentColor(const QRgb& color) { curColor = color; }
02229     void      setCurrentLineType(const QString& lineType) { curLineType = lineType; }
02230     void      setCurrentLineWidth(const QString& lineWidth) { curLineWidth = lineWidth; }
02231     void      designDetails();
02232     bool     loadFile(const QString &fileName);
02233     bool     saveFile(const QString &fileName);
02234 signals:
02235     void sendCloseMdiWin(MdiWindow* );
02236
02237 public slots:
02238     void closeEvent(QCloseEvent* e);
02239     void onWindowActivated();
02240
02241     void currentLayerChanged(const QString& layer);
02242     void currentColorChanged(const QRgb& color);
02243     void currentLinetypeChanged(const QString& type);
02244     void currentLinewidthChanged(const QString& weight);
02245
02246     void updateColorLinetypeLinewidth();
02247     void deletePressed();
02248     void escapePressed();
02249
02250     void showViewScrollBars(bool val);
02251     void setViewCrossHairColor(QRgb color);
02252     void setViewBackgroundColor(QRgb color);
02253     void setViewSelectBoxColors(QRgb colorL, QRgb fillL, QRgb colorR, QRgb fillR, int alpha);
02254     void setViewGridColor(QRgb color);
02255     void setViewRulerColor(QRgb color);
02256

```

```
02257     void print();
02258     void saveBMC();
02259
02260     void promptHistoryAppended(const QString& txt);
02261     void logPromptInput(const QString& txt);
02262     void promptInputPrevious();
02263     void promptInputNext();
02264
02265 protected:
02266
02267 private:
02268     MainWindow*           mainWin;
02269     QMdiArea*             mdiArea;
02270     QGraphicsScene*       gscene;
02271     View*                 gview;
02272
02273     bool fileWasLoaded;
02274
02275     QString promptHistory;
02276     QList<QString> promptInputList;
02277     int promptInputNum;
02278
02279     QPrinter               printer;
02280
02281     QString curFile;
02282     void setCurrentFile(const QString& fileName);
02283     QString fileExtension(const QString& fileName);
02284
02285     int myIndex;
02286
02287     QString curLayer;
02288     QRgb curColor;
02289     QString curLineType;
02290     QString curLineWeight;
02291
02292     void promptInputPrevNext(bool prev);
02293 };
02294
02295 class MdiArea : public QMdiArea
02296 {
02297     Q_OBJECT
02298
02299 public:
02300     MainWindow* mainWin;
02301
02302     bool useLogo;
02303     bool useTexture;
02304     bool useColor;
02305
02306     QPixmap bgLogo;
02307     QPixmap bgTexture;
02308     QColor bgColor;
02309
02310     void zoomExtentsAllSubWindows();
02311     void forceRepaint();
02312
02313     MdiArea(MainWindow* mw, QWidget* parent = 0);
02314     ~MdiArea();
02315
02316     void useBackgroundLogo(bool use);
02317     void useBackgroundTexture(bool use);
02318     void useBackgroundColor(bool use);
02319
02320     void setBackgroundLogo(const QString& fileName);
02321     void setBackgroundTexture(const QString& fileName);
02322     void setBackgroundColor(const QColor& color);
02323
02324 public slots:
02325     void cascade();
02326     void tile();
02327 protected:
02328     virtual void mouseDoubleClickEvent(QMouseEvent* e);
02329     virtual void paintEvent(QPaintEvent* e);
02330 };
02331
02332 class PreviewDialog : public QFileDialog
02333 {
02334     Q_OBJECT
02335
02336 public:
02337     PreviewDialog(QWidget* parent = 0,
02338                   const QString& caption = QString(),
02339                   const QString& directory = QString(),
02340                   const QString& filter = QString());
02341     ~PreviewDialog();
02342
02343     ImageWidget* imgWidget;
```

```

02350 };
02351
02352
02353 class PropertyEditor : public QDockWidget
02354 {
02355     Q_OBJECT
02356
02357 public:
02358     PropertyEditor(const QString& iconDirectory = QString(), bool pickAddMode = true, QWidget*
02359     widgetToFocus = 0, QWidget* parent = 0); //, Qt::WindowFlags flags = 0);
02360     ~PropertyEditor();
02361 protected:
02362     bool eventFilter(QObject *obj, QEvent *event);
02363
02364 signals:
02365     void pickAddModeToggled();
02366
02367 public slots:
02368     void setSelectedItems(QList<QGraphicsItem*> itemList);
02369     void updatePickAddModeButton(bool pickAddMode);
02370
02371 private slots:
02372     void fieldEdited(QObject* fieldObj);
02373     void showGroups(int objType);
02374     void showOneType(int index);
02375     void hideAllGroups();
02376     void clearAllFields();
02377     void togglePickAddMode();
02378
02379 private:
02380     QWidget* focusWidget;
02381
02382     QString iconDir;
02383     int iconSize;
02384     Qt::ToolButtonStyle propertyEditorButtonStyle;
02385
02386     bool pickAdd;
02387
02388     QList<QGraphicsItem*> selectedItemList;
02389
02390     ArcObject* tempArcObj;
02391     BlockObject* tempBlockObj;
02392     CircleObject* tempCircleObj;
02393     DimAlignedObject* tempDimAlignedObj;
02394     DimAngularObject* tempDimAngularObj;
02395     DimArcLengthObject* tempDimArcLenObj;
02396     DimDiameterObject* tempDimDiamObj;
02397     DimLeaderObject* tempDimLeaderObj;
02398     DimLinearObject* tempDimLinearObj;
02399     DimOrdinateObject* tempDimOrdObj;
02400     DimRadiusObject* tempDimRadiusObj;
02401     EllipseObject* tempEllipseObj;
02402     EllipseArcObject* tempEllipseArcObj;
02403     HatchObject* tempHatchObj;
02404     ImageObject* tempImageObj;
02405     InfiniteLineObject* tempInflLineObj;
02406     LineObject* tempLineObj;
02407     PathObject* tempPathObj;
02408     PointObject* tempPointObj;
02409     PolygonObject* tempPolygonObj;
02410     PolylineObject* tempPolylineObj;
02411     RayObject* tempRayObj;
02412     RectObject* tempRectObj;
02413     SplineObject* tempSplineObj;
02414     TextMultiObject* tempTextMultiObj;
02415     TextSingleObject* tempTextSingleObj;
02416
02417     //Helper functions
02418     QToolButton* createToolButton(const QString& iconName, const QString& txt);
02419     QLineEdit* createLineEdit(const QString& validatorType = QString(), bool readOnly = false);
02420     QComboBox* createComboBox(bool disable = false);
02421     QFontComboBox* createFontComboBox(bool disable = false);
02422
02423     int precisionAngle;
02424     int precisionLength;
02425
02426     //Used when checking if fields vary
02427     QString fieldOldText;
02428     QString fieldNewText;
02429     QString fieldVariesText;
02430     QString fieldYesText;
02431     QString fieldNoText;
02432     QString fieldOnText;
02433     QString fieldOffText;
02434
02435     void updateLineEditStrIfVaries(LineEdit* lineEdit, const QString& str);

```

```

02436     void updateLineEditNumIfVaries(QLineEdit* lineEdit, EmbReal num, bool useAnglePrecision);
02437     void updateFontComboBoxStrIfVaries(QFontComboBox* fontComboBox, const QString& str);
02438     void updateComboBoxStrIfVaries(QComboBox* comboBox, const QString& str, const QStringList&
02439         strList);
02440     void updateComboBoxBoolIfVaries(QComboBox* comboBox, bool val, bool yesOrNoText);
02441     QSignalMapper* signalMapper;
02442     void mapSignal(QObject* fieldObj, const QString& name, QVariant value);
02443
02444 //=====
02445 //Selection
02446 //=====
02447     QComboBox* createComboBoxSelected();
02448     QToolButton* createToolButtonQSelect();
02449     QToolButton* createToolButtonPickAdd();
02450
02451     QComboBox* comboBoxSelected;
02452     QToolButton* toolButtonQSelect;
02453     QToolButton* toolButtonPickAdd;
02454
02455 //TODO: Alphabetic/Categorized TabWidget
02456
02457     QGroupBox* createGroupBoxGeneral();
02458
02459     QGroupBox* createGroupBoxGeometryArc();
02460     QGroupBox* createGroupBoxMiscArc();
02461     QGroupBox* createGroupBoxGeometryBlock();
02462     QGroupBox* createGroupBoxGeometryCircle();
02463     QGroupBox* createGroupBoxGeometryDimAligned();
02464     QGroupBox* createGroupBoxGeometryDimAngular();
02465     QGroupBox* createGroupBoxGeometryDimArcLength();
02466     QGroupBox* createGroupBoxGeometryDimDiameter();
02467     QGroupBox* createGroupBoxGeometryDimLeader();
02468     QGroupBox* createGroupBoxGeometryDimLinear();
02469     QGroupBox* createGroupBoxGeometryDimOrdinate();
02470     QGroupBox* createGroupBoxGeometryDimRadius();
02471     QGroupBox* createGroupBoxGeometryEllipse();
02472     QGroupBox* createGroupBoxGeometryImage();
02473     QGroupBox* createGroupBoxMiscImage();
02474     QGroupBox* createGroupBoxGeometryInfiniteLine();
02475     QGroupBox* createGroupBoxGeometryLine();
02476     QGroupBox* createGroupBoxGeometryPath();
02477     QGroupBox* createGroupBoxMiscPath();
02478     QGroupBox* createGroupBoxGeometryPoint();
02479     QGroupBox* createGroupBoxGeometryPolygon();
02480     QGroupBox* createGroupBoxGeometryPolyline();
02481     QGroupBox* createGroupBoxMiscPolyline();
02482     QGroupBox* createGroupBoxGeometryRay();
02483     QGroupBox* createGroupBoxGeometryRectangle();
02484     QGroupBox* createGroupBoxGeometryTextMulti();
02485     QGroupBox* createGroupBoxTextTextSingle();
02486     QGroupBox* createGroupBoxGeometryTextSingle();
02487     QGroupBox* createGroupBoxMiscTextSingle();
02488 };
02489
02490
02491 class SelectBox : public QRubberBand
02492 {
02493     Q_OBJECT
02494
02495 public:
02496     SelectBox(Shape s, QWidget* parent = 0);
02497
02498     QColor leftBrushColor;
02499     QColor rightBrushColor;
02500     QColor leftPenColor;
02501     QColor rightPenColor;
02502     quint8 alpha;
02503
02504     QBrush dirBrush;
02505     QBrush leftBrush;
02506     QBrush rightBrush;
02507
02508     QPen dirPen;
02509     QPen leftPen;
02510     QPen rightPen;
02511
02512     bool boxDir;
02513
02514     void forceRepaint();
02515
02516 public slots:
02517     void setDirection(int dir);
02518     void setColors(const QColor& colorL, const QColor& fillL, const QColor& colorR, const QColor&
02519         fillR, int newAlpha);
02520 protected:

```

```

02521     void paintEvent (QPaintEvent* );
02522 };
02523
02527 class Settings_Dialog : public QDialog
02528 {
02529     Q_OBJECT
02530
02531 public:
02532     Settings_Dialog(MainWindow* mw, const QString& showTab = QString(), QWidget *parent = 0);
02533     ~Settings_Dialog();
02534
02535     MainWindow* mainWin;
02536
02537     QTabWidget* tabWidget;
02538
02539     QWidget* createTabGeneral();
02540     QWidget* createTabFilePaths();
02541     QWidget* createTabDisplay();
02542     QWidget* createTabPrompt();
02543     QWidget* createTabOpenSave();
02544     QWidget* createTabPrinting();
02545     QWidget* createTabSnap();
02546     QWidget* createTabGridRuler();
02547     QWidget* createTabOrthoPolar();
02548     QWidget* createTabQuickSnap();
02549     QWidget* createTabQuickTrack();
02550     QWidget* createTabLineWeight();
02551     QWidget* createTabSelection();
02552
02553     QDialogButtonBox* buttonBox;
02554
02555     void addColorsToComboBox(QComboBox* comboBox);
02556
02557     //Temporary for instant preview
02558     bool preview_general_mdi_bg_use_logo;
02559     bool preview_general_mdi_bg_use_texture;
02560     bool preview_general_mdi_bg_use_color;
02561
02562     QString accept_general_mdi_bg_logo;
02563     QString accept_general_mdi_bg_texture;
02564     QRgb preview_general_mdi_bg_color;
02565     QRgb accept_general_mdi_bg_color;
02566
02567     bool preview_display_show_scrollbars;
02568
02569     QRgb preview_display_crosshair_color;
02570     QRgb accept_display_crosshair_color;
02571     QRgb preview_display_bg_color;
02572     QRgb accept_display_bg_color;
02573
02574     QRgb preview_display_selectbox_left_color;
02575     QRgb accept_display_selectbox_left_color;
02576     QRgb preview_display_selectbox_left_fill;
02577     QRgb accept_display_selectbox_left_fill;
02578     QRgb preview_display_selectbox_right_color;
02579     QRgb accept_display_selectbox_right_color;
02580     QRgb preview_display_selectbox_right_fill;
02581     QRgb accept_display_selectbox_right_fill;
02582     quint8 preview_display_selectbox_alpha;
02583
02584     QRgb preview_prompt_text_color;
02585     QRgb accept_prompt_text_color;
02586
02587     QRgb preview_prompt_bg_color;
02588     QRgb accept_prompt_bg_color;
02589
02590     QString preview_prompt_font_family;
02591     QString preview_prompt_font_style;
02592     quint8 preview_prompt_font_size;
02593
02594     QRgb preview_grid_color;
02595     QRgb accept_grid_color;
02596
02597     QRgb preview_ruler_color;
02598     QRgb accept_ruler_color;
02599
02600     bool preview_lwt_show_lwt;
02601     bool preview_lwt_real_render;
02602
02603     //Temporary until changes are accepted
02604     QString dialog_general_language;
02605     QString dialog_general_icon_theme;
02606     int dialog_general_icon_size;
02607     bool dialog_general_mdi_bg_use_logo;
02608     bool dialog_general_mdi_bg_use_texture;
02609     bool dialog_general_mdi_bg_use_color;
02610     QString dialog_general_mdi_bg_logo;

```

```
02611     QString dialog_general_mdi_bg_texture;
02612     QRgb dialog_general_mdi_bg_color;
02613     bool dialog_general_tip_of_the_day;
02614     bool dialog_general_system_help_browser;
02615     bool dialog_display_use_opengl;
02616     bool dialog_display_renderhint_aa;
02617     bool dialog_display_renderhint_text_aa;
02618     bool dialog_display_renderhint_smooth_pix;
02619     bool dialog_display_renderhint_high_aa;
02620     bool dialog_display_renderhint_noncosmetic;
02621     bool dialog_display_show_scrollbars;
02622     int dialog_display_scrollbar_widget_num;
02623     QRgb dialog_display_crosshair_color;
02624     QRgb dialog_display_bg_color;
02625     QRgb dialog_display_selectbox_left_color;
02626     QRgb dialog_display_selectbox_left_fill;
02627     QRgb dialog_display_selectbox_right_color;
02628     QRgb dialog_display_selectbox_right_fill;
02629     quint8 dialog_display_selectbox_alpha;
02630     EmbReal dialog_display_zoomscale_in;
02631     EmbReal dialog_display_zoomscale_out;
02632     quint8 dialog_display_crosshair_percent;
02633     QString dialog_display_units;
02634     QRgb dialog_prompt_text_color;
02635     QRgb dialog_prompt_bg_color;
02636     QString dialog_prompt_font_family;
02637     QString dialog_prompt_font_style;
02638     quint8 dialog_prompt_font_size;
02639     bool dialog_prompt_save_history;
02640     bool dialog_prompt_save_history_as_html;
02641     QString dialog_prompt_save_history_filename;
02642     QString dialog_opensave_custom_filter;
02643     QString dialog_opensave_open_format;
02644     bool dialog_opensave_open_thumbnail;
02645     QString dialog_opensave_save_format;
02646     bool dialog_opensave_save_thumbnail;
02647     quint8 dialog_opensave_recent_max_files;
02648     quint8 dialog_opensave_trim_dst_num_jumps;
02649     QString dialog_printing_default_device;
02650     bool dialog_printing_use_last_device;
02651     bool dialog_printing_disable_bg;
02652     bool dialog_grid_show_on_load;
02653     bool dialog_grid_show_origin;
02654     bool dialog_grid_color_match_crosshair;
02655     QRgb dialog_grid_color;
02656     bool dialog_grid_load_from_file;
02657     QString dialog_grid_type;
02658     bool dialog_grid_center_on_origin;
02659     EmbReal dialog_grid_center_x;
02660     EmbReal dialog_grid_center_y;
02661     EmbReal dialog_grid_size_x;
02662     EmbReal dialog_grid_size_y;
02663     EmbReal dialog_grid_spacing_x;
02664     EmbReal dialog_grid_spacing_y;
02665     EmbReal dialog_grid_size_radius;
02666     EmbReal dialog_grid_spacing_radius;
02667     EmbReal dialog_grid_spacing_angle;
02668     bool dialog_ruler_show_on_load;
02669     bool dialog_ruler_metric;
02670     QRgb dialog_ruler_color;
02671     quint8 dialog_ruler_pixel_size;
02672     bool dialog_qsnap_enabled;
02673     QRgb dialog_qsnap_locator_color;
02674     quint8 dialog_qsnap_locator_size;
02675     quint8 dialog_qsnap_aperture_size;
02676     bool dialog_qsnap_endpoint;
02677     bool dialog_qsnap_midpoint;
02678     bool dialog_qsnap_center;
02679     bool dialog_qsnap_node;
02680     bool dialog_qsnap_quadrant;
02681     bool dialog_qsnap_intersection;
02682     bool dialog_qsnap_extension;
02683     bool dialog_qsnap_insertion;
02684     bool dialog_qsnap_perpendicular;
02685     bool dialog_qsnap_tangent;
02686     bool dialog_qsnap_nearest;
02687     bool dialog_qsnap_apparent;
02688     bool dialog_qsnap_parallel;
02689     bool dialog_lwt_show_lwt;
02690     bool dialog_lwt_real_render;
02691     EmbReal dialog_lwt_default_lwt;
02692     bool dialog_selection_mode_pickfirst;
02693     bool dialog_selection_mode_pickadd;
02694     bool dialog_selection_mode_pickdrag;
02695     QRgb dialog_selection_coolgrip_color;
02696     QRgb dialog_selection_hotgrip_color;
02697     quint8 dialog_selection_grip_size;
```

```

02698     quint8 dialog_selection_pickbox_size;
02699
02700 private slots:
02701     void comboBoxLanguageCurrentIndexChanged(const QString&);
02702     void comboBoxIconThemeCurrentIndexChanged(const QString&);
02703     void comboBoxIconSizeCurrentIndexChanged(int);
02704     void checkBoxGeneralMdiBGUseLogoStateChanged(int);
02705     void chooseGeneralMdiBackgroundLogo();
02706     void checkBoxGeneralMdiBGUseTextureStateChanged(int);
02707     void chooseGeneralMdiBackgroundTexture();
02708     void checkBoxGeneralMdiBGUseColorStateChanged(int);
02709     void chooseGeneralMdiBackgroundColor();
02710     void currentGeneralMdiBackgroundColorChanged(const QColor&);
02711     void checkBoxTipOfTheDayStateChanged(int);
02712     void checkBoxUseOpenGLStateChanged(int);
02713     void checkBoxRenderHintAAStateChanged(int);
02714     void checkBoxRenderHintTextAAStateChanged(int);
02715     void checkBoxRenderHintSmoothPixStateChanged(int);
02716     void checkBoxRenderHintHighAAStateChanged(int);
02717     void checkBoxRenderHintNonCosmeticStateChanged(int);
02718     void checkBoxShowScrollBarsStateChanged(int);
02719     void comboBoxScrollBarWidgetCurrentIndexChanged(int);
02720     void spinBoxZoomScaleInValueChanged(double);
02721     void spinBoxZoomScaleOutValueChanged(double);
02722     void checkBoxDisableBGStateChanged(int);
02723     void chooseDisplayCrossHairColor();
02724     void currentDisplayCrossHairColorChanged(const QColor&);
02725     void chooseDisplayBackgroundColor();
02726     void currentDisplayBackgroundColorChanged(const QColor&);
02727     void chooseDisplaySelectBoxLeftColor();
02728     void currentDisplaySelectBoxLeftColorChanged(const QColor&);
02729     void chooseDisplaySelectBoxLeftFill();
02730     void currentDisplaySelectBoxLeftFillChanged(const QColor&);
02731     void chooseDisplaySelectBoxRightColor();
02732     void currentDisplaySelectBoxRightColorChanged(const QColor&);
02733     void chooseDisplaySelectBoxRightFill();
02734     void currentDisplaySelectBoxRightFillChanged(const QColor&);
02735     void spinBoxDisplaySelectBoxAlphaValueChanged(int);
02736     void choosePromptTextColor();
02737     void currentPromptTextColorChanged(const QColor&);
02738     void choosePromptBackgroundColor();
02739     void currentPromptBackgroundColorChanged(const QColor&);
02740     void comboBoxPromptFontFamilyCurrentIndexChanged(const QString&);
02741     void comboBoxPromptFontStyleCurrentIndexChanged(const QString&);
02742     void spinBoxPromptFontSizeValueChanged(int);
02743     void checkBoxPromptSaveHistoryStateChanged(int);
02744     void checkBoxPromptSaveHistoryAsHtmlStateChanged(int);
02745     void checkBoxCustomFilterStateChanged(int);
02746     void buttonCustomFilterSelectAllClicked();
02747     void buttonCustomFilterClearAllClicked();
02748     void spinBoxRecentMaxFilesValueChanged(int);
02749     void spinBoxTrimDstNumJumpsValueChanged(int);
02750     void checkBoxGridShowOnLoadStateChanged(int);
02751     void checkBoxGridShowOriginStateChanged(int);
02752     void checkBoxGridColorMatchCrossHairStateChanged(int);
02753     void chooseGridColor();
02754     void currentGridColorChanged(const QColor&);
02755     void checkBoxGridLoadFromFileStateChanged(int);
02756     void comboBoxGridTypeCurrentIndexChanged(const QString&);
02757     void checkBoxGridCenterOnOriginStateChanged(int);
02758     void spinBoxGridCenterXValueChanged(double);
02759     void spinBoxGridCenterYValueChanged(double);
02760     void spinBoxGridSizeXValueChanged(double);
02761     void spinBoxGridSizeYValueChanged(double);
02762     void spinBoxGridSpacingXValueChanged(double);
02763     void spinBoxGridSpacingYValueChanged(double);
02764     void spinBoxGridSizeRadiusValueChanged(double);
02765     void spinBoxGridSpacingRadiusValueChanged(double);
02766     void spinBoxGridSpacingAngleValueChanged(double);
02767     void checkBoxRulerShowOnLoadStateChanged(int);
02768     void comboBoxRulerMetricCurrentIndexChanged(int);
02769     void chooseRulerColor();
02770     void currentRulerColorChanged(const QColor&);
02771     void spinBoxRulerPixelSizeValueChanged(double);
02772     void checkBoxQSnapEndPointStateChanged(int);
02773     void checkBoxQSnapMidPointStateChanged(int);
02774     void checkBoxQSnapCenterStateChanged(int);
02775     void checkBoxQSnapNodeStateChanged(int);
02776     void checkBoxQSnapQuadrantStateChanged(int);
02777     void checkBoxQSnapIntersectionStateChanged(int);
02778     void checkBoxQSnapExtensionStateChanged(int);
02779     void checkBoxQSnapInsertionStateChanged(int);
02780     void checkBoxQSnapPerpendicularStateChanged(int);
02781     void checkBoxQSnapTangentStateChanged(int);
02782     void checkBoxQSnapNearestStateChanged(int);
02783     void checkBoxQSnapApparentStateChanged(int);
02784     void checkBoxQSnapParallelStateChanged(int);

```

```
02785     void buttonQSnapSelectAllClicked();
02786     void buttonQSnapClearAllClicked();
02787     void comboBoxQSnapLocatorColorCurrentIndexChanged(int);
02788     void sliderQSnapLocatorsSizeValueChanged(int);
02789     void sliderQSnapApertureSizeValueChanged(int);
02790     void checkBoxLwtShowLwtStateChanged(int);
02791     void checkBoxLwtRealRenderStateChanged(int);
02792     void checkBoxSelectionModePickFirstStateChanged(int);
02793     void checkBoxSelectionModePickAddStateChanged(int);
02794     void checkBoxSelectionModePickDragStateChanged(int);
02795     void comboBoxSelectionCoolGripColorCurrentIndexChanged(int);
02796     void comboBoxSelectionHotGripColorCurrentIndexChanged(int);
02797     void sliderSelectionGripSizeValueChanged(int);
02798     void sliderSelectionPickBoxSizeValueChanged(int);
02799
02800     void acceptChanges();
02801     void rejectChanges();
02802
02803 signals:
02804     void buttonCustomFilterSelectAll(bool);
02805     void buttonCustomFilterClearAll(bool);
02806     void buttonQSnapSelectAll(bool);
02807     void buttonQSnapClearAll(bool);
02808 };
02809
02810
02811 class StatusBar : public QStatusBar
02812 {
02813     Q_OBJECT
02814
02815 public:
02816     StatusBar(MainWindow* mw, QWidget* parent = 0);
02817
02818     StatusBarButton* statusBarSnapButton;
02819     StatusBarButton* statusBarGridButton;
02820     StatusBarButton* statusBarRulerButton;
02821     StatusBarButton* statusBarOrthoButton;
02822     StatusBarButton* statusBarPolarButton;
02823     StatusBarButton* statusBarQSnapButton;
02824     StatusBarButton* statusBarQTrackButton;
02825     StatusBarButton* statusBarLwtButton;
02826     QLabel* statusBarMouseCoord;
02827
02828     void setMouseCoord(EmbReal x, EmbReal y);
02829 };
02830
02834 class StatusBarButton : public QToolButton
02835 {
02836     Q_OBJECT
02837
02838 public:
02839     StatusBarButton(QString buttonText, MainWindow* mw, StatusBar* statbar, QWidget *parent = 0);
02840
02841     MainWindow* mainWin;
02842     StatusBar* statusbar;
02843
02844 protected:
02845     void contextMenuEvent(QContextMenuEvent *event = 0);
02846
02847 private slots:
02848     void settingsSnap();
02849     void settingsGrid();
02850     void settingsRuler();
02851     void settingsOrtho();
02852     void settingsPolar();
02853     void settingsQSnap();
02854     void settingsQTrack();
02855     void settingsLwt();
02856     void toggleSnap(bool on);
02857     void toggleGrid(bool on);
02858     void toggleRuler(bool on);
02859     void toggleOrtho(bool on);
02860     void togglePolar(bool on);
02861     void toggleQSnap(bool on);
02862     void toggleQTrack(bool on);
02863     void toggleLwt(bool on);
02864 public slots:
02865     void enableLwt();
02866     void disableLwt();
02867     void enableReal();
02868     void disableReal();
02869 };
02870
02874 class UndoEditor : public QDockWidget
02875 {
02876     Q_OBJECT
02877
```

```

02878 public:
02879     UndoEditor(const QString& iconDirectory = QString(), QWidget* widgetToFocus = 0, QWidget* parent =
02880         0); //, Qt::WindowFlags flags = 0);
02881     ~UndoEditor();
02882     void addStack(QUndoStack* stack);
02883
02884     bool canUndo() const;
02885     bool canRedo() const;
02886
02887     QWidget* focusWidget;
02888
02889     QString iconDir;
02890     int iconSize;
02891
02892     QUndoGroup* undoGroup;
02893     QUndoView* undoView;
02894
02895     QString undoText() const;
02896     QString redoText() const;
02897 protected:
02898
02899 public slots:
02900     void undo();
02901     void redo();
02902
02903     void updateCleanIcon(bool opened);
02904 };
02905
02906 class UndoableAddCommand : public QUndoCommand
02907 {
02908 public:
02909     UndoableAddCommand(const QString& text, BaseObject* obj, View* v, QUndoCommand* parent = 0);
02910
02911     void undo();
02912     void redo();
02913
02914     BaseObject* object;
02915     View* gview;
02916
02917 };
02918
02919 class UndoableDeleteCommand : public QUndoCommand
02920 {
02921 public:
02922     UndoableDeleteCommand(const QString& text, BaseObject* obj, View* v, QUndoCommand* parent = 0);
02923
02924     void undo();
02925     void redo();
02926
02927     BaseObject* object;
02928     View* gview;
02929
02930 };
02931
02932 class UndoableMoveCommand : public QUndoCommand
02933 {
02934 public:
02935     UndoableMoveCommand(EmbReal deltaX, EmbReal deltaY, const QString& text, BaseObject* obj, View* v,
02936     QUndoCommand* parent = 0);
02937
02938     void undo();
02939     void redo();
02940
02941     BaseObject* object;
02942     View* gview;
02943     EmbReal dx;
02944     EmbReal dy;
02945
02946 };
02947
02948 class UndoableRotateCommand : public QUndoCommand
02949 {
02950 public:
02951     UndoableRotateCommand(EmbReal pivotPointX, EmbReal pivotPointY, EmbReal rotAngle, const QString&
02952     text, BaseObject* obj, View* v, QUndoCommand* parent = 0);
02953
02954     void undo();
02955     void redo();
02956
02957     void rotate(EmbReal x, EmbReal y, EmbReal rot);
02958
02959     BaseObject* object;
02960     View* gview;
02961     EmbReal pivotX;
02962     EmbReal pivotY;
02963     EmbReal angle;
02964
02965 };
02966
02967 class UndoableScaleCommand : public QUndoCommand

```

```
02977 {
02978     public:
02979         UndoableScaleCommand(EmbReal x, EmbReal y, EmbReal scaleFactor, const QString& text, BaseObject* obj, View* v, QUndoCommand* parent = 0);
02980
02981     void undo();
02982     void redo();
02983
02984     BaseObject* object;
02985     View* gview;
02986     EmbReal dx;
02987     EmbReal dy;
02988     EmbReal factor;
02989 };
02990
02994 class UndoableNavCommand : public QUndoCommand
02995 {
02996     public:
02997         UndoableNavCommand(const QString& type, View* v, QUndoCommand* parent = 0);
02998
02999     int id() const { return 1234; }
03000     bool mergeWith(const QUndoCommand* command);
03001     void undo();
03002     void redo();
03003
03004     QString navType;
03005     QTransform fromTransform;
03006     QTransform toTransform;
03007     QPointF fromCenter;
03008     QPointF toCenter;
03009     bool done;
03010     View* gview;
03011 };
03012
03016 class UndoableGripEditCommand : public QUndoCommand
03017 {
03018     public:
03019         UndoableGripEditCommand(const QPointF beforePoint, const QPointF afterPoint, const QString& text,
03020         BaseObject* obj, View* v, QUndoCommand* parent = 0);
03021
03022     void undo();
03023     void redo();
03024
03025     BaseObject* object;
03026     View* gview;
03027     QPointF before;
03028     QPointF after;
03029
03033 class UndoableMirrorCommand : public QUndoCommand
03034 {
03035     public:
03036         UndoableMirrorCommand(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, const QString& text,
03037         BaseObject* obj, View* v, QUndoCommand* parent = 0);
03038
03039     void undo();
03040     void redo();
03041     void mirror();
03042
03043     BaseObject* object;
03044     View* gview;
03045     QLineF mirrorLine;
03046 };
03047
03051 class View : public QGraphicsView
03052 {
03053     Q_OBJECT
03054
03055     public:
03056         View(MainWindow* mw, QGraphicsScene* theScene, QWidget* parent);
03057         ~View();
03058
03059         bool allowZoomIn();
03060         bool allowZoomOut();
03061
03062         void recalculateLimits();
03063         void zoomToPoint(const QPoint& mousePoint, int zoomDir);
03064         void centerAt(const QPointF& centerPoint);
03065         QPointF center() { return mapToScene(rect().center()); }
03066
03067         QUndoStack* getUndoStack() { return undoStack; }
03068         void addObject(BaseObject* obj);
03069         void deleteObject(BaseObject* obj);
03070         void vulcanizeObject(BaseObject* obj);
03071
03072     public slots:
```

```

03073     void zoomIn();
03074     void zoomOut();
03075     void zoomWindow();
03076     void zoomSelected();
03077     void zoomExtents();
03078     void panRealTime();
03079     void panPoint();
03080     void panLeft();
03081     void panRight();
03082     void panUp();
03083     void panDown();
03084     void selectAll();
03085     void selectionChanged();
03086     void clearSelection();
03087     void deleteSelected();
03088     void moveSelected(EmbReal dx, EmbReal dy);
03089     void cut();
03090     void copy();
03091     void paste();
03092     void repeatAction();
03093     void moveAction();
03094     void scaleAction();
03095     void scaleSelected(EmbReal x, EmbReal y, EmbReal factor);
03096     void rotateAction();
03097     void rotateSelected(EmbReal x, EmbReal y, EmbReal rot);
03098     void mirrorSelected(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2);
03099     int numSelected();
03100
03101     void deletePressed();
03102     void escapePressed();
03103
03104     void cornerButtonClicked();
03105
03106     void showScrollBars(bool val);
03107     void setCornerButton();
03108     void setCrossHairColor(QRgb color);
03109     void setCrossHairSize(qint8 percent);
03110     void setBackgroundColor(QRgb color);
03111     void setSelectBoxColors(QRgb colorL, QRgb fillL, QRgb colorR, QRgb fillR, int alpha);
03112     void toggleSnap(bool on);
03113     void toggleGrid(bool on);
03114     void toggleRuler(bool on);
03115     void toggleOrtho(bool on);
03116     void togglePolar(bool on);
03117     void toggleQSnap(bool on);
03118     void toggleQTrack(bool on);
03119     void toggleLwt(bool on);
03120     void toggleReal(bool on);
03121     bool isLwtEnabled();
03122     bool isRealEnabled();
03123
03124     void setGridColor(QRgb color);
03125     void createGrid(const QString& gridType);
03126     void setRulerColor(QRgb color);
03127
03128     void previewOn(int clone, int mode, EmbReal x, EmbReal y, EmbReal data);
03129     void previewOff();
03130
03131     void enableMoveRapidFire();
03132     void disableMoveRapidFire();
03133
03134     bool allowRubber();
03135     void addToRubberRoom(QGraphicsItem* item);
03136     void vulcanizeRubberRoom();
03137     void clearRubberRoom();
03138     void spareRubber(qint64 id);
03139     void setRubberMode(int mode);
03140     void setRubberPoint(const QString& key, const QPointF& point);
03141     void setRubberText(const QString& key, const QString& txt);
03142
03143 protected:
03144     void mouseDoubleClickEvent(QMouseEvent* event);
03145     void mousePressEvent(QMouseEvent* event);
03146     void mouseMoveEvent(QMouseEvent* event);
03147     void mouseReleaseEvent(QMouseEvent* event);
03148     void wheelEvent(QWheelEvent* event);
03149     void contextMenuEvent(QContextMenuEvent* event);
03150     void drawBackground(QPainter* painter, const QRectF& rect);
03151     void drawForeground(QPainter* painter, const QRectF& rect);
03152     void enterEvent(QEvent* event);
03153
03154 private:
03155     QHash<qint64, QGraphicsItem*> hashDeletedObjects;
03156
03157     QList<qint64> spareRubberList;
03158     QColor gridColumn;

```

```
03160     QPainterPath gridPath;
03161     void createGridRect();
03162     void createGridPolar();
03163     void createGridIso();
03164     QPainterPath originPath;
03165     void createOrigin();
03166
03167     bool rulerMetric;
03168     QColor rulerColor;
03169     quint8 rulerPixelSize;
03170     void loadRulerSettings();
03171
03172     bool willUnderflowInt32(qint64 a, qint64 b);
03173     bool willOverflowInt32(qint64 a, qint64 b);
03174     int roundToMultiple(bool roundUp, int numToRound, int multiple);
03175     QPainterPath createRulerTextPath(float x, EmbReal y, QString str, EmbReal height);
03176
03177     QList<QGraphicsItem*> previewObjectList;
03178     QGraphicsItemGroup* previewObjectItemGroup;
03179     QPointF previewPoint;
03180     EmbReal previewData;
03181     int previewMode;
03182
03183     QList<QGraphicsItem*> createObjectList(QList<QGraphicsItem*> list);
03184     QPointF cutCopyMousePoint;
03185     QGraphicsItemGroup* pasteObjectItemGroup;
03186     QPointF pasteDelta;
03187
03188     QList<QGraphicsItem*> rubberRoomList;
03189
03190     void copySelected();
03191
03192     bool grippingActive;
03193     bool rapidMoveActive;
03194     bool previewActive;
03195     bool pastingActive;
03196     bool movingActive;
03197     bool selectingActive;
03198     bool zoomWindowActive;
03199     bool panningRealTimeActive;
03200     bool panningPointActive;
03201     bool panningActive;
03202     bool qSnapActive;
03203     bool qSnapToggle;
03204
03205     void startGripping(BaseObject* obj);
03206     void stopGripping(bool accept = false);
03207
03208     BaseObject* gripBaseObj;
03209     BaseObject* tempBaseObj;
03210
03211     MainWindow* mainWin;
03212     QGraphicsScene* gscene;
03213     QUndoStack* undoStack;
03214
03215     SelectBox* selectBox;
03216     QPointF scenePressPoint;
03217     QPoint pressPoint;
03218     QPointF sceneMovePoint;
03219     QPoint movePoint;
03220     QPointF sceneReleasePoint;
03221     QPoint releasePoint;
03222     QPointF sceneGripPoint;
03223
03224     void updateMouseCoords(int x, int y);
03225     QPoint viewMousePoint;
03226     QPointF sceneMousePoint;
03227     QRgb qsnapLocatorColor;
03228     quint8 qsnapLocatorSize;
03229     quint8 qsnapApertureSize;
03230     QRgb gripColorCool;
03231     QRgb gripColorHot;
03232     quint8 gripSize;
03233     quint8 pickBoxSize;
03234     QRgb crosshairColor;
03235     quint32 crosshairSize;
03236
03237     void panStart(const QPoint& point);
03238     int panDistance;
03239     int panStartX;
03240     int panStartY;
03241
03242     void alignScenePointWithViewPoint(const QPointF& scenePoint, const QPoint& viewPoint);
03243 };
03244
03245     typedef struct Action__ {
03246         int hash;
```

```
03250     /*< Index in the actionHash array. */
03251     std::string icon;
03252     /*< The stub used for the icon and the basic command. */
03253     std::string tooltip;
03254     /*< The label in the menus and the message that appears when
03255      you hover over an icon. */
03256     std::string statustip;
03257     /*< The message that appears at the bottom of the . */
03258     std::string shortcut;
03259     /*< The keyboard shortcut for this action. */
03260     std::vector<std::string> aliases;
03261     /*< A list of all alternative commands, if empty only
03262      the icon string will be . */
03263     std::vector<std::string> script;
03264     /*< If this is a compound action this will be a
03265      list of commands or it can allow for command line
03266      style command aliases. For example: icon16 would become
03267      the string list {"iconResize 16"}.
03268     std::string menu_name;
03269     /*< */
03270     int menu_position;
03271     /*< */
03272     std::string toolbar_name;
03273     /*< */
03274     int toolbar_position;
03275     /*< */
03276 } Action;
03277
03278 int get_action_index(std::string cmd);
03279
03280 /* */
03281 extern Settings settings;
03282 extern Settings dialog;
03283 extern std::vector<Action> action_table;
03284 extern std::vector<std::string> file_toolbar;
03285 extern std::vector<std::string> edit_toolbar;
03286 extern std::vector<std::string> view_toolbar;
03287 extern std::vector<std::string> zoom_toolbar;
03288
03289 #endif
```

18.7 embroidermodder2/imagedwidget.cpp File Reference

```
#include "embroidermodder.h"
```

18.8 embroidermodder2/layer-manager.cpp File Reference

```
#include "embroidermodder.h"
```

18.8.1 Detailed Description

Embroidermodder 2
Copyright 2013-2022 The Embroidermodder Team Embroidermodder 2 is Open Source Software. See LICENSE for licensing terms.
Use Python's PEP7 style guide. <https://peps.python.org/pep-0007/>

18.9 embroidermodder2/mainwindow-commands.cpp File Reference

```
#include "embroidermodder.h"
```

18.10 embroidermodder2/mainwindow-menus.cpp File Reference

```
#include "embroidermodder.h"
```

18.11 embroidermodder2/mainwindow-settings.cpp File Reference

```
#include "embroidermodder.h"
#include <string>
```

```
#include <iostream>
#include <fstream>
```

Functions

- `QString SettingsDir ()`
- `QString SettingsPath ()`
- `std::vector< std::string > to_string_vector (QStringList list)`
- `void read_configuration ()`
- `std::string write_setting (std::string label, int a)`
- `std::string write_setting (std::string label, QRgb a)`
- `std::string write_setting (std::string label, QString a)`
- `std::string write_setting (std::string label, float a)`
- `std::string write_setting (std::string label, bool a)`

18.11.1 Function Documentation

18.11.1.1 `read_configuration()` `void read_configuration ()`

18.11.1.2 `SettingsDir()` `QString SettingsDir ()`

Note: on Unix we include the trailing separator. For Windows compatibility we omit it.

18.11.1.3 `SettingsPath()` `QString SettingsPath ()`

18.11.1.4 `to_string_vector()` `std::vector< std::string > to_string_vector (QStringList list)`

18.11.1.5 `write_setting()` [1/5] `std::string write_setting (std::string label, bool a)`

18.11.1.6 `write_setting()` [2/5] `std::string write_setting (std::string label, float a)`

18.11.1.7 `write_setting()` [3/5] `std::string write_setting (std::string label, int a)`

18.11.1.8 `write_setting()` [4/5] `std::string write_setting (std::string label, QRgb a)`

18.11.1.9 `write_setting()` [5/5] `std::string write_setting (std::string label, QString a)`

18.12 embroidermodder2/mainwindow-toolbars.cpp File Reference

```
#include "embroidermodder.h"
```

Functions

- int [get_action_index](#) (std::string cmd)

Variables

- std::vector< std::string > [file_toolbar](#)
- std::vector< std::string > [edit_toolbar](#)
- std::vector< std::string > [view_toolbar](#)
- std::vector< std::string > [zoom_toolbar](#)
- std::vector< std::string > [pan_toolbar](#)
- std::vector< std::string > [icon_toolbar](#)
- std::vector< std::string > [help_toolbar](#)

18.12.1 Function Documentation

18.12.1.1 [get_action_index\(\)](#) int get_action_index (std::string cmd)

18.12.2 Variable Documentation

18.12.2.1 [edit_toolbar](#) std::vector<std::string> edit_toolbar

Initial value:

```
= {  
    "cut",  
    "copy",  
    "paste"  
}
```

18.12.2.2 [file_toolbar](#) std::vector<std::string> file_toolbar

Initial value:

```
= {  
    "new",  
    "open",  
    "save",  
    "saveas",  
    "print",  
    "designdetails",  
    "---",  
    "undo",  
    "redo",  
    "---",  
    "help"  
}
```

18.12.2.3 [help_toolbar](#) std::vector<std::string> help_toolbar

Initial value:

```
= {  
    "help",  
    "---",  
    "changelog",  
    "---",  
    "about",  
    "---",  
    "whatsthis"  
}
```

18.12.2.4 icon_toolbar std::vector<std::string> icon_toolbar**Initial value:**

```
= {
    "icon16",
    "icon24",
    "icon32",
    "icon48",
    "icon64",
    "icon128"
}
```

18.12.2.5 pan_toolbar std::vector<std::string> pan_toolbar**Initial value:**

```
= {
    "panrealtime",
    "panpoint",
    "---",
    "panleft",
    "panright",
    "panup",
    "pandown"
}
```

18.12.2.6 view_toolbar std::vector<std::string> view_toolbar**Initial value:**

```
= {
    "day",
    "night"
}
```

18.12.2.7 zoom_toolbar std::vector<std::string> zoom_toolbar**Initial value:**

```
= {
    "zoomwindow",
    "zoomdynamic",
    "zoomscale",
    "---",
    "zoomcenter",
    "zoomin",
    "zoomout",
    "---",
    "zoomselected",
    "zoomall",
    "zoomextents"
}
```

18.13 embroidermodder2/mainwindow.cpp File Reference

```
#include "embroidermodder.h"
#include <cerrno>
#include <iostream>
#include <fstream>
```

Classes

- struct [Parameter_](#)

TypeDefs

- typedef struct [Parameter_](#) [Parameter](#)

Functions

- std::string [read_string_setting](#) (toml_table_t *table, const char *key)
- int [read_settings](#) (const char *settings_file)

Read the settings from file which aren't editable by the user. These files need to be placed in the install folder.

- bool `validRGB` (int r, int g, int b)
- `MainWindow * mainWin ()`
mainWin
- std::string `convert_args_to_type` (std::string label, std::vector< std::string > args, const char *args_template, Parameter result[10])
Inspired by PyArg_ParseTupleAndKeywords allowing a uniform argument parsing framework.
 - std::string `Error` (Parameter args[10])
 - std::string `Todo` (Parameter result[10])

Variables

- static const int `CIRCLE_MODE_1P_RAD_` = 0
- static const int `CIRCLE_MODE_1P_DIA_` = 1
- static const int `CIRCLE_MODE_2P_` = 2
- static const int `CIRCLE_MODE_3P_` = 3
- static const int `CIRCLE_MODE_TTR_` = 4
- static const int `DOLPHIN_MODE_NUM_POINTS_` = 0
- static const int `DOLPHIN_MODE_XSCALE_` = 1
- static const int `DOLPHIN_MODE_YSCALE_` = 2
- static const int `SINGLE_LINE_TEXT_MODE_JUSTIFY_` = 0
- static const int `SINGLE_LINE_TEXT_MODE_SETFONT_` = 1
- static const int `SINGLE_LINE_TEXT_MODE_SETGEOM_` = 2
- static const int `SINGLE_LINE_TEXT_MODE_RAPID_` = 3
- static const int `STAR_MODE_NUM_POINTS_` = 0
- static const int `STAR_MODE_CENTER_PT_` = 1
- static const int `STAR_MODE_RAD_OUTER_` = 2
- static const int `STAR_MODE_RAD_INNER_` = 3
- `MainWindow * _mainWin` = 0
- std::vector< Action > `action_table`
- QStringList `action_labels`
- `Settings settings`
- `Settings dialog`
- `Settings preview`

18.13.1 Typedef Documentation

18.13.1.1 Parameter `typedef struct Parameter_ Parameter`

18.13.2 Function Documentation

18.13.2.1 `convert_args_to_type()` `std::string convert_args_to_type (`

```
    std::string label,
    std::vector< std::string > args,
    const char * args_template,
    Parameter result[10] )
```

Inspired by PyArg_ParseTupleAndKeywords allowing a uniform argument parsing framework.

Parameters

| | |
|----------------------------|--|
| <code>label</code> | The caller's name. |
| <code>args</code> | The list of strings passed from the user. |
| <code>args_template</code> | The string of characters describing the types of the output. |
| <code>result</code> | The fixed length array of results. |

Returns

An error message if an error occurred or an empty string if it passes.

```
18.13.2.2 Error() std::string Error (
    Parameter args[10] )
"debug": qDebug("%s", qPrintable(result[0].s_value));
```

```
18.13.2.3 mainWin() MainWindow * mainWin ( )
mainWin
```

Returns

```
18.13.2.4 read_settings() int read_settings (
    const char * settings_file )
```

Read the settings from file which aren't editable by the user. These files need to be placed in the install folder.

```
18.13.2.5 read_string_setting() std::string read_string_setting (
    toml_table_t * table,
    const char * key )
```

```
18.13.2.6 Todo() std::string Todo (
    Parameter result[10] )
```

```
18.13.2.7 validRGB() bool validRGB (
    int r,
    int g,
    int b )
```

18.13.3 Variable Documentation

```
18.13.3.1 _mainWin MainWindow* _mainWin = 0
```

```
18.13.3.2 action_labels QStringList action_labels
```

```
18.13.3.3 action_table std::vector<Action> action_table
```

```
18.13.3.4 CIRCLE_MODE_1P_DIA_ const int CIRCLE_MODE_1P_DIA_ = 1 [static]
```

```
18.13.3.5 CIRCLE_MODE_1P_RAD_ const int CIRCLE_MODE_1P_RAD_ = 0 [static]
```

18.13.3.6 CIRCLE_MODE_2P_ const int CIRCLE_MODE_2P_ = 2 [static]

18.13.3.7 CIRCLE_MODE_3P_ const int CIRCLE_MODE_3P_ = 3 [static]

18.13.3.8 CIRCLE_MODE_TTR_ const int CIRCLE_MODE_TTR_ = 4 [static]

18.13.3.9 dialog *Settings* dialog

These copies of the settings struct are for restoring the state if the user doesn't want to accept their changes in the settings dialog.

18.13.3.10 DOLPHIN_MODE_NUM_POINTS_ const int DOLPHIN_MODE_NUM_POINTS_ = 0 [static]

18.13.3.11 DOLPHIN_MODE_XSCALE_ const int DOLPHIN_MODE_XSCALE_ = 1 [static]

18.13.3.12 DOLPHIN_MODE_YSCALE_ const int DOLPHIN_MODE_YSCALE_ = 2 [static]

18.13.3.13 preview *Settings* preview

18.13.3.14 settings *Settings* settings

The actuator changes the program state via these global variables.

18.13.3.15 SINGLE_LINE_TEXT_MODE_JUSTIFY_ const int SINGLE_LINE_TEXT_MODE_JUSTIFY_ = 0 [static]

18.13.3.16 SINGLE_LINE_TEXT_MODE_RAPID_ const int SINGLE_LINE_TEXT_MODE_RAPID_ = 3 [static]

18.13.3.17 SINGLE_LINE_TEXT_MODE_SETFONT_ const int SINGLE_LINE_TEXT_MODE_SETFONT_ = 1 [static]

18.13.3.18 SINGLE_LINE_TEXT_MODE_SETGEOM_ const int SINGLE_LINE_TEXT_MODE_SETGEOM_ = 2 [static]

18.13.3.19 STAR_MODE_CENTER_PT_ const int STAR_MODE_CENTER_PT_ = 1 [static]

18.13.3.20 STAR_MODE_NUM_POINTS_ const int STAR_MODE_NUM_POINTS_ = 0 [static]

18.13.3.21 STAR_MODE_RAD_INNER_ const int STAR_MODE_RAD_INNER_ = 3 [static]

18.13.3.22 STAR_MODE_RAD_OUTER_ const int STAR_MODE_RAD_OUTER_ = 2 [static]

18.14 embroidermodder2/mdiarea.cpp File Reference

```
#include "embroidermodder.h"
```

18.15 embroidermodder2/mdiwindow.cpp File Reference

```
#include "embroidermodder.h"
```

18.16 embroidermodder2/object-arc.cpp File Reference

```
#include "embroidermodder.h"
```

Functions

- [EmbVector rotate_vector \(EmbVector v, EmbReal alpha\)](#)

18.16.1 Function Documentation

18.16.1.1 rotate_vector() [EmbVector](#) rotate_vector (

```
    EmbVector v,  
    EmbReal alpha )
```

Returns

18.17 embroidermodder2/object-base.cpp File Reference

```
#include "embroidermodder.h"
```

18.18 embroidermodder2/object-circle.cpp File Reference

```
#include "embroidermodder.h"
```

18.19 embroidermodder2/object-dimleader.cpp File Reference

```
#include "embroidermodder.h"
```

18.20 embroidermodder2/object-ellipse.cpp File Reference

```
#include "embroidermodder.h"
```

18.21 embroidermodder2/object-image.cpp File Reference

```
#include "embroidermodder.h"
```

18.22 embroidermodder2/object-line.cpp File Reference

```
#include "embroidermodder.h"
```

18.23 **embroidermodder2/object-path.cpp** File Reference

```
#include "embroidermodder.h"
```

18.24 **embroidermodder2/object-point.cpp** File Reference

```
#include "embroidermodder.h"
```

18.25 **embroidermodder2/object-polygon.cpp** File Reference

```
#include "embroidermodder.h"
```

18.26 **embroidermodder2/object-polyline.cpp** File Reference

```
#include "embroidermodder.h"
```

18.27 **embroidermodder2/object-rect.cpp** File Reference

```
#include "embroidermodder.h"
```

18.27.1 Detailed Description

Embroidermodder 2
Copyright 2013-2022 The Embroidermodder Team Embroidermodder 2 is Open Source Software. See LICENSE for licensing terms.
Use Python's PEP7 style guide. <https://peps.python.org/pep-0007/>

18.28 **embroidermodder2/object-save.cpp** File Reference

```
#include "embroidermodder.h"
```

18.29 **embroidermodder2/object-textsingle.cpp** File Reference

```
#include "embroidermodder.h"
```

18.30 **embroidermodder2/preview-dialog.cpp** File Reference

```
#include "embroidermodder.h"
```

18.31 **embroidermodder2/property-editor.cpp** File Reference

```
#include "embroidermodder.h"
```

Variables

- std::unordered_map< std::string, QGroupBox * > **groupBoxes**
- std::unordered_map< std::string, QComboBox * > **comboBoxes**
- std::unordered_map< std::string, QLineEdit * > **lineEdits**
- std::unordered_map< std::string, QToolButton * > **toolButtons**
- QGroupBox * **groupBoxGeometryCircle**

- QGroupBox * `groupBoxGeometryImage`
- QGroupBox * `groupBoxMiscImage`
- QGroupBox * `groupBoxGeometryInfiniteLine`
- QGroupBox * `groupBoxGeometryLine`
- QGroupBox * `groupBoxGeometryPolygon`
- QGroupBox * `groupBoxGeometryPolyline`
- QGroupBox * `groupBoxGeometryPath`
- QGroupBox * `groupBoxMiscPath`
- QGroupBox * `groupBoxGeometryPoint`
- QGroupBox * `groupBoxGeometryRay`
- QGroupBox * `groupBoxGeometryRectangle`
- QGroupBox * `groupBoxGeometryTextMulti`
- QGroupBox * `groupBoxTextTextSingle`
- QGroupBox * `groupBoxGeometryTextSingle`
- QGroupBox * `groupBoxMiscTextSingle`
- QGroupBox * `groupBoxMiscPolyline`
- QComboBox * `comboBoxGeneralLineType`
- QComboBox * `comboBoxGeneralLineWeight`
- QLineEdit * `lineEditArcCenterY`
- QLineEdit * `lineEditArcRadius`
- QLineEdit * `lineEditArcStartAngle`
- QLineEdit * `lineEditArcEndAngle`
- QLineEdit * `lineEditArcStartX`
- QLineEdit * `lineEditArcStartY`
- QLineEdit * `lineEditArcEndX`
- QLineEdit * `lineEditArcEndY`
- QLineEdit * `lineEditArcArea`
- QLineEdit * `lineEditArcLength`
- QLineEdit * `lineEditArcChord`
- QLineEdit * `lineEditArcIncAngle`
- QToolButton * `toolButtonArcClockwise`
- QComboBox * `comboBoxArcClockwise`
- QToolButton * `toolButtonBlockX`
- QToolButton * `toolButtonBlockY`
- QLineEdit * `lineEditBlockX`
- QLineEdit * `lineEditBlockY`
- QToolButton * `toolButtonCircleCenterX`
- QToolButton * `toolButtonCircleCenterY`
- QToolButton * `toolButtonCircleRadius`
- QToolButton * `toolButtonCircleDiameter`
- QToolButton * `toolButtonCircleArea`
- QToolButton * `toolButtonCircleCircumference`
- QLineEdit * `lineEditCircleCenterX`
- QLineEdit * `lineEditCircleCenterY`
- QLineEdit * `lineEditCircleRadius`
- QLineEdit * `lineEditCircleDiameter`
- QLineEdit * `lineEditCircleArea`
- QLineEdit * `lineEditCircleCircumference`
- QGroupBox * `groupBoxGeometryDimAligned`
- QGroupBox * `groupBoxGeometryDimAngular`
- QGroupBox * `groupBoxGeometryDimArcLength`
- QGroupBox * `groupBoxGeometryDimDiameter`
- QGroupBox * `groupBoxGeometryDimLeader`
- QGroupBox * `groupBoxGeometryDimLinear`
- QGroupBox * `groupBoxGeometryDimOrdinate`

- QGroupBox * `groupBoxGeometryDimRadius`
- QGroupBox * `groupBoxGeometryEllipse`
- QToolButton * `toolButtonEllipseCenterX`
- QToolButton * `toolButtonEllipseCenterY`
- QToolButton * `toolButtonEllipseRadiusMajor`
- QToolButton * `toolButtonEllipseRadiusMinor`
- QToolButton * `toolButtonEllipseDiameterMajor`
- QToolButton * `toolButtonEllipseDiameterMinor`
- QLineEdit * `lineEditEllipseCenterX`
- QLineEdit * `lineEditEllipseCenterY`
- QLineEdit * `lineEditEllipseRadiusMajor`
- QLineEdit * `lineEditEllipseRadiusMinor`
- QLineEdit * `lineEditEllipseDiameterMajor`
- QLineEdit * `lineEditEllipseDiameterMinor`
- QToolButton * `toolButtonImageX`
- QToolButton * `toolButtonImageY`
- QToolButton * `toolButtonImageWidth`
- QToolButton * `toolButtonImageHeight`
- QLineEdit * `lineEditImageX`
- QLineEdit * `lineEditImageY`
- QLineEdit * `lineEditImageWidth`
- QLineEdit * `lineEditImageHeight`
- QToolButton * `toolButtonImageName`
- QToolButton * `toolButtonImagePath`
- QLineEdit * `lineEditImageName`
- QLineEdit * `lineEditImagePath`
- QToolButton * `toolButtonInfiniteLineY1`
- QToolButton * `toolButtonInfiniteLineX2`
- QToolButton * `toolButtonInfiniteLineY2`
- QToolButton * `toolButtonInfiniteLineVectorX`
- QToolButton * `toolButtonInfiniteLineVectorY`
- QLineEdit * `lineEditInfiniteLineY1`
- QLineEdit * `lineEditInfiniteLineX2`
- QLineEdit * `lineEditInfiniteLineY2`
- QLineEdit * `lineEditInfiniteLineVectorX`
- QLineEdit * `lineEditInfiniteLineVectorY`
- QToolButton * `toolButtonLineStartX`
- QToolButton * `toolButtonLineStartY`
- QToolButton * `toolButtonLineEndX`
- QToolButton * `toolButtonLineEndY`
- QToolButton * `toolButtonLineDeltaX`
- QToolButton * `toolButtonLineDeltaY`
- QToolButton * `toolButtonLineAngle`
- QToolButton * `toolButtonLineLength`
- QLineEdit * `lineEditLineStartX`
- QLineEdit * `lineEditLineStartY`
- QLineEdit * `lineEditLineEndX`
- QLineEdit * `lineEditLineEndY`
- QLineEdit * `lineEditLineDeltaX`
- QLineEdit * `lineEditLineDeltaY`
- QLineEdit * `lineEditLineAngle`
- QLineEdit * `lineEditLineLength`
- QToolButton * `toolButtonPolygonCenterX`
- QToolButton * `toolButtonPolygonCenterY`
- QToolButton * `toolButtonPolygonRadiusVertex`

- QToolButton * `toolButtonPolygonRadiusSide`
- QToolButton * `toolButtonPolygonDiameterVertex`
- QToolButton * `toolButtonPolygonDiameterSide`
- QToolButton * `toolButtonPolygonInteriorAngle`
- QLineEdit * `lineEditPolygonCenterX`
- QLineEdit * `lineEditPolygonCenterY`
- QLineEdit * `lineEditPolygonRadiusVertex`
- QLineEdit * `lineEditPolygonRadiusSide`
- QLineEdit * `lineEditPolygonDiameterVertex`
- QLineEdit * `lineEditPolygonDiameterSide`
- QLineEdit * `lineEditPolygonInteriorAngle`
- QToolButton * `toolButtonPolylineVertexNum`
- QToolButton * `toolButtonPolylineVertexX`
- QToolButton * `toolButtonPolylineVertexY`
- QToolButton * `toolButtonPolylineArea`
- QToolButton * `toolButtonPolylineLength`
- QComboBox * `comboBoxPolylineVertexNum`
- QLineEdit * `lineEditPolylineVertexX`
- QLineEdit * `lineEditPolylineVertexY`
- QLineEdit * `lineEditPolylineArea`
- QLineEdit * `lineEditPolylineLength`
- QToolButton * `toolButtonPathVertexNum`
- QToolButton * `toolButtonPathVertexX`
- QToolButton * `toolButtonPathVertexY`
- QToolButton * `toolButtonPathArea`
- QToolButton * `toolButtonPathLength`
- QComboBox * `comboBoxPathVertexNum`
- QLineEdit * `lineEditPathVertexX`
- QLineEdit * `lineEditPathVertexY`
- QLineEdit * `lineEditPathArea`
- QLineEdit * `lineEditPathLength`
- QToolButton * `toolButtonPathClosed`
- QComboBox * `comboBoxPathClosed`
- QToolButton * `toolButtonPointX`
- QToolButton * `toolButtonPointY`
- QLineEdit * `lineEditPointX`
- QLineEdit * `lineEditPointY`
- QToolButton * `toolButtonPolylineClosed`
- QComboBox * `comboBoxPolylineClosed`
- QToolButton * `toolButtonRayX1`
- QToolButton * `toolButtonRayY1`
- QToolButton * `toolButtonRayX2`
- QToolButton * `toolButtonRayY2`
- QToolButton * `toolButtonRayVectorX`
- QToolButton * `toolButtonRayVectorY`
- QLineEdit * `lineEditRayX1`
- QLineEdit * `lineEditRayY1`
- QLineEdit * `lineEditRayX2`
- QLineEdit * `lineEditRayY2`
- QLineEdit * `lineEditRayVectorX`
- QLineEdit * `lineEditRayVectorY`
- QToolButton * `toolButtonRectangleCorner1X`
- QToolButton * `toolButtonRectangleCorner1Y`
- QToolButton * `toolButtonRectangleCorner2X`
- QToolButton * `toolButtonRectangleCorner2Y`

- QToolButton * `toolButtonRectangleCorner3X`
- QToolButton * `toolButtonRectangleCorner3Y`
- QToolButton * `toolButtonRectangleCorner4X`
- QToolButton * `toolButtonRectangleCorner4Y`
- QToolButton * `toolButtonRectangleWidth`
- QToolButton * `toolButtonRectangleHeight`
- QToolButton * `toolButtonRectangleArea`
- QLineEdit * `lineEditRectangleCorner1X`
- QLineEdit * `lineEditRectangleCorner1Y`
- QLineEdit * `lineEditRectangleCorner2X`
- QLineEdit * `lineEditRectangleCorner2Y`
- QLineEdit * `lineEditRectangleCorner3X`
- QLineEdit * `lineEditRectangleCorner3Y`
- QLineEdit * `lineEditRectangleCorner4X`
- QLineEdit * `lineEditRectangleCorner4Y`
- QLineEdit * `lineEditRectangleWidth`
- QLineEdit * `lineEditRectangleHeight`
- QLineEdit * `lineEditRectangleArea`
- QToolButton * `toolButtonTextMultiX`
- QToolButton * `toolButtonTextMultiY`
- QLineEdit * `lineEditTextMultiX`
- QLineEdit * `lineEditTextMultiY`
- QToolButton * `toolButtonTextSingleContents`
- QToolButton * `toolButtonTextSingleFont`
- QToolButton * `toolButtonTextSingleJustify`
- QToolButton * `toolButtonTextSingleHeight`
- QToolButton * `toolButtonTextSingleRotation`
- QLineEdit * `lineEditTextSingleContents`
- QFontComboBox * `comboBoxTextSingleFont`
- QComboBox * `comboBoxTextSingleJustify`
- QLineEdit * `lineEditTextSingleHeight`
- QLineEdit * `lineEditTextSingleRotation`
- QToolButton * `toolButtonTextSingleX`
- QToolButton * `toolButtonTextSingleY`
- QLineEdit * `lineEditTextSingleX`
- QLineEdit * `lineEditTextSingleY`
- QToolButton * `toolButtonTextSingleBackward`
- QToolButton * `toolButtonTextSingleUpsideDown`
- QComboBox * `comboBoxTextSingleBackward`
- QComboBox * `comboBoxTextSingleUpsideDown`

18.31.1 Variable Documentation

18.31.1.1 comboBoxArcClockwise `QComboBox* comboBoxArcClockwise`

18.31.1.2 comboBoxes `std::unordered_map<std::string, QComboBox *> comboBoxes`

18.31.1.3 comboBoxGeneralLineType `QComboBox* comboBoxGeneralLineType`

18.31.1.4 comboBoxGeneralLineWeight QComboBox* comboBoxGeneralLineWeight

18.31.1.5 comboBoxPathClosed QComboBox* comboBoxPathClosed

18.31.1.6 comboBoxPathVertexNum QComboBox* comboBoxPathVertexNum

18.31.1.7 comboBoxPolylineClosed QComboBox* comboBoxPolylineClosed

18.31.1.8 comboBoxPolylineVertexNum QComboBox* comboBoxPolylineVertexNum

18.31.1.9 comboBoxTextSingleBackward QComboBox* comboBoxTextSingleBackward

18.31.1.10 comboBoxTextSingleFont QFontComboBox* comboBoxTextSingleFont

18.31.1.11 comboBoxTextSingleJustify QComboBox* comboBoxTextSingleJustify

18.31.1.12 comboBoxTextSingleUpsideDown QComboBox* comboBoxTextSingleUpsideDown

18.31.1.13 groupBoxes std::unordered_map<std::string, QGroupBox *> groupBoxes

18.31.1.14 groupBoxGeometryCircle QGroupBox* groupBoxGeometryCircle

18.31.1.15 groupBoxGeometryDimAligned QGroupBox* groupBoxGeometryDimAligned

18.31.1.16 groupBoxGeometryDimAngular QGroupBox* groupBoxGeometryDimAngular

18.31.1.17 groupBoxGeometryDimArcLength QGroupBox* groupBoxGeometryDimArcLength

18.31.1.18 groupBoxGeometryDimDiameter QGroupBox* groupBoxGeometryDimDiameter

18.31.1.19 groupBoxGeometryDimLeader QGroupBox* groupBoxGeometryDimLeader

18.31.1.20 groupBoxGeometryDimLinear QGroupBox* groupBoxGeometryDimLinear

18.31.1.21 groupBoxGeometryDimOrdinate QGroupBox* groupBoxGeometryDimOrdinate

18.31.1.22 groupBoxGeometryDimRadius QGroupBox* groupBoxGeometryDimRadius

18.31.1.23 groupBoxGeometryEllipse QGroupBox* groupBoxGeometryEllipse

18.31.1.24 groupBoxGeometryImage QGroupBox* groupBoxGeometryImage

18.31.1.25 groupBoxGeometryInfiniteLine QGroupBox* groupBoxGeometryInfiniteLine

18.31.1.26 groupBoxGeometryLine QGroupBox* groupBoxGeometryLine

18.31.1.27 groupBoxGeometryPath QGroupBox* groupBoxGeometryPath

18.31.1.28 groupBoxGeometryPoint QGroupBox* groupBoxGeometryPoint

18.31.1.29 groupBoxGeometryPolygon QGroupBox* groupBoxGeometryPolygon

18.31.1.30 groupBoxGeometryPolyline QGroupBox* groupBoxGeometryPolyline

18.31.1.31 groupBoxGeometryRay QGroupBox* groupBoxGeometryRay

18.31.1.32 groupBoxGeometryRectangle QGroupBox* groupBoxGeometryRectangle

18.31.1.33 groupBoxGeometryTextMulti QGroupBox* groupBoxGeometryTextMulti

18.31.1.34 groupBoxGeometryTextSingle QGroupBox* groupBoxGeometryTextSingle

18.31.1.35 groupBoxMiscImage QGroupBox* groupBoxMiscImage

18.31.1.36 groupBoxMiscPath QGroupBox* groupBoxMiscPath

18.31.1.37 groupBoxMiscPolyline QGroupBox* groupBoxMiscPolyline

18.31.1.38 groupBoxMiscTextSingle QGroupBox* groupBoxMiscTextSingle

18.31.1.39 groupBoxTextTextSingle QGroupBox* groupBoxTextTextSingle

18.31.1.40 lineEditArcArea QLineEdit* lineEditArcArea

18.31.1.41 lineEditArcCenterY QLineEdit* lineEditArcCenterY

18.31.1.42 lineEditArcChord QLineEdit* lineEditArcChord

18.31.1.43 lineEditArcEndAngle QLineEdit* lineEditArcEndAngle

18.31.1.44 lineEditArcEndX QLineEdit* lineEditArcEndX

18.31.1.45 lineEditArcEndY QLineEdit* lineEditArcEndY

18.31.1.46 lineEditArcIncAngle QLineEdit* lineEditArcIncAngle

18.31.1.47 lineEditArcLength QLineEdit* lineEditArcLength

18.31.1.48 lineEditArcRadius QLineEdit* lineEditArcRadius

18.31.1.49 lineEditArcStartAngle QLineEdit* lineEditArcStartAngle

18.31.1.50 lineEditArcStartX QLineEdit* lineEditArcStartX

18.31.1.51 lineEditArcStartY QLineEdit* lineEditArcStartY

18.31.1.52 lineEditBlockX QLineEdit* lineEditBlockX

18.31.1.53 lineEditBlockY QLineEdit* lineEditBlockY

18.31.1.54 lineEditCircleArea QLineEdit* lineEditCircleArea

18.31.1.55 lineEditCircleCenterX QLineEdit* lineEditCircleCenterX

18.31.1.56 lineEditCircleCenterY QLineEdit* lineEditCircleCenterY

18.31.1.57 lineEditCircleCircumference QLineEdit* lineEditCircleCircumference

18.31.1.58 `lineEditCircleDiameter` `QLineEdit* lineEditCircleDiameter`

18.31.1.59 `lineEditCircleRadius` `QLineEdit* lineEditCircleRadius`

18.31.1.60 `lineEditEllipseCenterX` `QLineEdit* lineEditEllipseCenterX`

18.31.1.61 `lineEditEllipseCenterY` `QLineEdit* lineEditEllipseCenterY`

18.31.1.62 `lineEditEllipseDiameterMajor` `QLineEdit* lineEditEllipseDiameterMajor`

18.31.1.63 `lineEditEllipseDiameterMinor` `QLineEdit* lineEditEllipseDiameterMinor`

18.31.1.64 `lineEditEllipseRadiusMajor` `QLineEdit* lineEditEllipseRadiusMajor`

18.31.1.65 `lineEditEllipseRadiusMinor` `QLineEdit* lineEditEllipseRadiusMinor`

18.31.1.66 `lineEditImageHeight` `QLineEdit* lineEditImageHeight`

18.31.1.67 `lineEditImageName` `QLineEdit* lineEditImageName`

18.31.1.68 `lineEditImagePath` `QLineEdit* lineEditImagePath`

18.31.1.69 `lineEditImageWidth` `QLineEdit* lineEditImageWidth`

18.31.1.70 `lineEditImageX` `QLineEdit* lineEditImageX`

18.31.1.71 `lineEditImageY` `QLineEdit* lineEditImageY`

18.31.1.72 `lineEditInfiniteLineVectorX` `QLineEdit* lineEditInfiniteLineVectorX`

18.31.1.73 `lineEditInfiniteLineVectorY` `QLineEdit* lineEditInfiniteLineVectorY`

18.31.1.74 `lineEditInfiniteLineX2` `QLineEdit* lineEditInfiniteLineX2`

18.31.1.75 `lineEditInfiniteLineY1` `QLineEdit* lineEditInfiniteLineY1`

18.31.1.76 `lineEditInfiniteLineY2` `QLineEdit* lineEditInfiniteLineY2`

18.31.1.77 `lineEditLineAngle` `QLineEdit* lineEditLineAngle`

18.31.1.78 `lineEditLineDeltaX` `QLineEdit* lineEditLineDeltaX`

18.31.1.79 `lineEditLineDeltaY` `QLineEdit* lineEditLineDeltaY`

18.31.1.80 `lineEditLineEndX` `QLineEdit* lineEditLineEndX`

18.31.1.81 `lineEditLineEndY` `QLineEdit* lineEditLineEndY`

18.31.1.82 `lineEditLineLength` `QLineEdit* lineEditLineLength`

18.31.1.83 `lineEditLineStartX` `QLineEdit* lineEditLineStartX`

18.31.1.84 `lineEditLineStartY` `QLineEdit* lineEditLineStartY`

18.31.1.85 `lineEditPathArea` `QLineEdit* lineEditPathArea`

18.31.1.86 `lineEditPathLength` `QLineEdit* lineEditPathLength`

18.31.1.87 `lineEditPathVertexX` `QLineEdit* lineEditPathVertexX`

18.31.1.88 `lineEditPathVertexY` `QLineEdit* lineEditPathVertexY`

18.31.1.89 `lineEditPointX` `QLineEdit* lineEditPointX`

18.31.1.90 `lineEditPointY` `QLineEdit* lineEditPointY`

18.31.1.91 `lineEditPolygonCenterX` `QLineEdit* lineEditPolygonCenterX`

18.31.1.92 `lineEditPolygonCenterY` `QLineEdit* lineEditPolygonCenterY`

18.31.1.93 `lineEditPolygonDiameterSide` `QLineEdit* lineEditPolygonDiameterSide`

18.31.1.94 `lineEditPolygonDiameterVertex` `QLineEdit* lineEditPolygonDiameterVertex`

18.31.1.95 `lineEditPolygonInteriorAngle` `QLineEdit* lineEditPolygonInteriorAngle`

18.31.1.96 `lineEditPolygonRadiusSide` `QLineEdit* lineEditPolygonRadiusSide`

18.31.1.97 `lineEditPolygonRadiusVertex` `QLineEdit* lineEditPolygonRadiusVertex`

18.31.1.98 `lineEditPolylineArea` `QLineEdit* lineEditPolylineArea`

18.31.1.99 `lineEditPolylineLength` `QLineEdit* lineEditPolylineLength`

18.31.1.100 `lineEditPolylineVertexX` `QLineEdit* lineEditPolylineVertexX`

18.31.1.101 `lineEditPolylineVertexY` `QLineEdit* lineEditPolylineVertexY`

18.31.1.102 `lineEditRayVectorX` `QLineEdit* lineEditRayVectorX`

18.31.1.103 `lineEditRayVectorY` `QLineEdit* lineEditRayVectorY`

18.31.1.104 `lineEditRayX1` `QLineEdit* lineEditRayX1`

18.31.1.105 `lineEditRayX2` `QLineEdit* lineEditRayX2`

18.31.1.106 `lineEditRayY1` `QLineEdit* lineEditRayY1`

18.31.1.107 `lineEditRayY2` `QLineEdit* lineEditRayY2`

18.31.1.108 `lineEditRectangleArea` `QLineEdit* lineEditRectangleArea`

18.31.1.109 `lineEditRectangleCorner1X` `QLineEdit* lineEditRectangleCorner1X`

18.31.1.110 `lineEditRectangleCorner1Y` `QLineEdit* lineEditRectangleCorner1Y`

18.31.1.111 `lineEditRectangleCorner2X` `QLineEdit* lineEditRectangleCorner2X`

18.31.1.112 `lineEditRectangleCorner2Y` `QLineEdit* lineEditRectangleCorner2Y`

18.31.1.113 `lineEditRectangleCorner3X` `QLineEdit* lineEditRectangleCorner3X`

18.31.1.114 `lineEditRectangleCorner3Y` `QLineEdit* lineEditRectangleCorner3Y`

18.31.1.115 `lineEditRectangleCorner4X` `QLineEdit* lineEditRectangleCorner4X`

18.31.1.116 `lineEditRectangleCorner4Y` `QLineEdit* lineEditRectangleCorner4Y`

18.31.1.117 `lineEditRectangleHeight` `QLineEdit* lineEditRectangleHeight`

18.31.1.118 `lineEditRectangleWidth` `QLineEdit* lineEditRectangleWidth`

18.31.1.119 `lineEdits` `std::unordered_map<std::string, QLineEdit *> lineEdits`

18.31.1.120 `lineEditTextMultiX` `QLineEdit* lineEditTextMultiX`

18.31.1.121 `lineEditTextMultiY` `QLineEdit* lineEditTextMultiY`

18.31.1.122 `lineEditTextSingleContents` `QLineEdit* lineEditTextSingleContents`

18.31.1.123 `lineEditTextSingleHeight` `QLineEdit* lineEditTextSingleHeight`

18.31.1.124 `lineEditTextSingleRotation` `QLineEdit* lineEditTextSingleRotation`

18.31.1.125 `lineEditTextSingleX` `QLineEdit* lineEditTextSingleX`

18.31.1.126 `lineEditTextSingleY` `QLineEdit* lineEditTextSingleY`

18.31.1.127 `toolButtonArcClockwise` `QToolButton* toolButtonArcClockwise`

18.31.1.128 `toolButtonBlockX` `QToolButton* toolButtonBlockX`

18.31.1.129 `toolButtonBlockY` `QToolButton* toolButtonBlockY`

18.31.1.130 toolButtonCircleArea QToolButton* toolButtonCircleArea

18.31.1.131 toolButtonCircleCenterX QToolButton* toolButtonCircleCenterX

18.31.1.132 toolButtonCircleCenterY QToolButton* toolButtonCircleCenterY

18.31.1.133 toolButtonCircleCircumference QToolButton* toolButtonCircleCircumference

18.31.1.134 toolButtonCircleDiameter QToolButton* toolButtonCircleDiameter

18.31.1.135 toolButtonCircleRadius QToolButton* toolButtonCircleRadius

18.31.1.136 toolButtonEllipseCenterX QToolButton* toolButtonEllipseCenterX

18.31.1.137 toolButtonEllipseCenterY QToolButton* toolButtonEllipseCenterY

18.31.1.138 toolButtonEllipseDiameterMajor QToolButton* toolButtonEllipseDiameterMajor

18.31.1.139 toolButtonEllipseDiameterMinor QToolButton* toolButtonEllipseDiameterMinor

18.31.1.140 toolButtonEllipseRadiusMajor QToolButton* toolButtonEllipseRadiusMajor

18.31.1.141 toolButtonEllipseRadiusMinor QToolButton* toolButtonEllipseRadiusMinor

18.31.1.142 toolButtonImageHeight QToolButton* toolButtonImageHeight

18.31.1.143 toolButtonImageName QToolButton* toolButtonImageName

18.31.1.144 toolButtonImagePath QToolButton* toolButtonImagePath

18.31.1.145 toolButtonImageWidth QToolButton* toolButtonImageWidth

18.31.1.146 toolButtonImageX QToolButton* toolButtonImageX

18.31.1.147 toolButtonImageY QToolButton* toolButtonImageY

18.31.1.148 toolButtonInfiniteLineVectorX QToolButton* toolButtonInfiniteLineVectorX

18.31.1.149 toolButtonInfiniteLineVectorY QToolButton* toolButtonInfiniteLineVectorY

18.31.1.150 toolButtonInfiniteLineX2 QToolButton* toolButtonInfiniteLineX2

18.31.1.151 toolButtonInfiniteLineY1 QToolButton* toolButtonInfiniteLineY1

18.31.1.152 toolButtonInfiniteLineY2 QToolButton* toolButtonInfiniteLineY2

18.31.1.153 toolButtonLineAngle QToolButton* toolButtonLineAngle

18.31.1.154 toolButtonLineDeltaX QToolButton* toolButtonLineDeltaX

18.31.1.155 toolButtonLineDeltaY QToolButton* toolButtonLineDeltaY

18.31.1.156 toolButtonLineEndX QToolButton* toolButtonLineEndX

18.31.1.157 toolButtonLineEndY QToolButton* toolButtonLineEndY

18.31.1.158 toolButtonLineLength QToolButton* toolButtonLineLength

18.31.1.159 toolButtonLineStartX QToolButton* toolButtonLineStartX

18.31.1.160 toolButtonLineStartY QToolButton* toolButtonLineStartY

18.31.1.161 toolButtonPathArea QToolButton* toolButtonPathArea

18.31.1.162 toolButtonPathClosed QToolButton* toolButtonPathClosed

18.31.1.163 toolButtonPathLength QToolButton* toolButtonPathLength

18.31.1.164 toolButtonPathVertexNum QToolButton* toolButtonPathVertexNum

18.31.1.165 toolButtonPathVertexX QToolButton* toolButtonPathVertexX

18.31.1.166 toolButtonPathVertexY QToolButton* toolButtonPathVertexY

18.31.1.167 toolButtonPointX QToolButton* toolButtonPointX

18.31.1.168 toolButtonPointY QToolButton* toolButtonPointY

18.31.1.169 toolButtonPolygonCenterX QToolButton* toolButtonPolygonCenterX

18.31.1.170 toolButtonPolygonCenterY QToolButton* toolButtonPolygonCenterY

18.31.1.171 toolButtonPolygonDiameterSide QToolButton* toolButtonPolygonDiameterSide

18.31.1.172 toolButtonPolygonDiameterVertex QToolButton* toolButtonPolygonDiameterVertex

18.31.1.173 toolButtonPolygonInteriorAngle QToolButton* toolButtonPolygonInteriorAngle

18.31.1.174 toolButtonPolygonRadiusSide QToolButton* toolButtonPolygonRadiusSide

18.31.1.175 toolButtonPolygonRadiusVertex QToolButton* toolButtonPolygonRadiusVertex

18.31.1.176 toolButtonPolylineArea QToolButton* toolButtonPolylineArea

18.31.1.177 toolButtonPolylineClosed QToolButton* toolButtonPolylineClosed

18.31.1.178 toolButtonPolylineLength QToolButton* toolButtonPolylineLength

18.31.1.179 toolButtonPolylineVertexNum QToolButton* toolButtonPolylineVertexNum

18.31.1.180 toolButtonPolylineVertexX QToolButton* toolButtonPolylineVertexX

18.31.1.181 toolButtonPolylineVertexY QToolButton* toolButtonPolylineVertexY

18.31.1.182 toolButtonRayVectorX QToolButton* toolButtonRayVectorX

18.31.1.183 toolButtonRayVectorY QToolButton* toolButtonRayVectorY

18.31.1.184 toolButtonRayX1 QToolButton* toolButtonRayX1

18.31.1.185 toolButtonRayX2 QToolButton* toolButtonRayX2

18.31.1.186 toolButtonRayY1 QToolButton* toolButtonRayY1

18.31.1.187 toolButtonRayY2 QToolButton* toolButtonRayY2

18.31.1.188 toolButtonRectangleArea QToolButton* toolButtonRectangleArea

18.31.1.189 toolButtonRectangleCorner1X QToolButton* toolButtonRectangleCorner1X

18.31.1.190 toolButtonRectangleCorner1Y QToolButton* toolButtonRectangleCorner1Y

18.31.1.191 toolButtonRectangleCorner2X QToolButton* toolButtonRectangleCorner2X

18.31.1.192 toolButtonRectangleCorner2Y QToolButton* toolButtonRectangleCorner2Y

18.31.1.193 toolButtonRectangleCorner3X QToolButton* toolButtonRectangleCorner3X

18.31.1.194 toolButtonRectangleCorner3Y QToolButton* toolButtonRectangleCorner3Y

18.31.1.195 toolButtonRectangleCorner4X QToolButton* toolButtonRectangleCorner4X

18.31.1.196 toolButtonRectangleCorner4Y QToolButton* toolButtonRectangleCorner4Y

18.31.1.197 toolButtonRectangleHeight QToolButton* toolButtonRectangleHeight

18.31.1.198 toolButtonRectangleWidth QToolButton* toolButtonRectangleWidth

18.31.1.199 toolButtons std::unordered_map<std::string, QToolButton * > toolButtons

18.31.1.200 toolButtonTextMultiX QToolButton* toolButtonTextMultiX

18.31.1.201 toolButtonTextMultiY QToolButton* toolButtonTextMultiY

18.31.1.202 toolButtonTextSingleBackward QToolButton* toolButtonTextSingleBackward

18.31.1.203 toolButtonTextSingleContents QToolButton* toolButtonTextSingleContents

18.31.1.204 toolButtonTextSingleFont QToolButton* toolButtonTextSingleFont

18.31.1.205 toolButtonTextSingleHeight QToolButton* toolButtonTextSingleHeight

18.31.1.206 toolButtonTextSingleJustify QToolButton* toolButtonTextSingleJustify

18.31.1.207 toolButtonTextSingleRotation QToolButton* toolButtonTextSingleRotation

18.31.1.208 toolButtonTextSingleUpsideDown QToolButton* toolButtonTextSingleUpsideDown

18.31.1.209 toolButtonTextSingleX QToolButton* toolButtonTextSingleX

18.31.1.210 toolButtonTextSingleY QToolButton* toolButtonTextSingleY

18.32 embroidermodder2/README.md File Reference

18.33 embroidermodder2/selectbox.cpp File Reference

```
#include "embroidermodder.h"
```

18.34 embroidermodder2/settings-dialog.cpp File Reference

```
#include "embroidermodder.h"
```

18.35 embroidermodder2/statusbar-button.cpp File Reference

```
#include "embroidermodder.h"
```

18.36 embroidermodder2/statusbar.cpp File Reference

```
#include "embroidermodder.h"
```

18.37 embroidermodder2/undo-commands.cpp File Reference

```
#include "embroidermodder.h"
```

18.38 embroidermodder2/undo-editor.cpp File Reference

```
#include "embroidermodder.h"
```

18.38.1 Detailed Description

Embroidermodder 2.

Copyright 2013-2022 The Embroidermodder Team Embroidermodder 2 is Open Source Software. See LICENSE for licensing terms.

Use Python's PEP7 style guide. <https://peps.python.org/pep-0007/>

18.39 embroidermodder2/view.cpp File Reference

```
#include "embroidermodder.h"
```

18.39.1 Detailed Description

Embroidermodder 2.

Copyright 2013-2022 The Embroidermodder Team Embroidermodder 2 is Open Source Software. See LICENSE for licensing terms.

Use Python's PEP7 style guide. <https://peps.python.org/pep-0007/>

18.40 extern/libembroidery/src/array.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "embroidery_internal.h"
```

Functions

- `EmbArray * embArray_create (int type)`
- `int embArray_resize (EmbArray *a)`
- `void embArray_copy (EmbArray *dst, EmbArray *src)`
- `int embArray_addArc (EmbArray *a, EmbArc b)`
- `int embArray_addCircle (EmbArray *a, EmbCircle b)`
- `int embArray_addEllipse (EmbArray *a, EmbEllipse b)`
- `int embArray_addFlag (EmbArray *a, EmbFlag b)`
- `int embArray_addLine (EmbArray *a, EmbLine b)`
- `int embArray_addPath (EmbArray *a, EmbPath b)`
- `int embArray_addPoint (EmbArray *a, EmbPoint b)`
- `int embArray_addPolyline (EmbArray *a, EmbPolyline b)`
- `int embArray_addPolygon (EmbArray *a, EmbPolygon b)`
- `int embArray_addRect (EmbArray *a, EmbRect b)`
- `int embArray_addStitch (EmbArray *a, EmbStitch b)`
- `int embArray_addVector (EmbArray *a, EmbVector b)`
- `void embArray_free (EmbArray *a)`

18.40.1 Function Documentation

18.40.1.1 `embArray_addArc()` `int embArray_addArc (`

```
    EmbArray * a,
    EmbArc b )
```

Add an arc *b* to the EmbArray *a* and it returns if the element was successfully added.

18.40.1.2 `embArray_addCircle()` `int embArray_addCircle (`

```
    EmbArray * a,
    EmbCircle b )
```

Add a circle *b* to the EmbArray *a* and it returns if the element was successfully added.

```
18.40.1.3 embArray_addEllipse() int embArray_addEllipse (
    EmbArray * a,
    EmbEllipse b )
```

Add an ellipse *b* to the EmbArray *a* and it returns if the element was successfully added.

```
18.40.1.4 embArray_addFlag() int embArray_addFlag (
    EmbArray * a,
    EmbFlag b )
```

Add a flag *b* to the EmbArray *a* and it returns if the element was successfully added.

```
18.40.1.5 embArray_addLine() int embArray_addLine (
    EmbArray * a,
    EmbLine b )
```

Add a line *b* to the EmbArray *a* and it returns if the element was successfully added.

```
18.40.1.6 embArray_addPath() int embArray_addPath (
    EmbArray * a,
    EmbPath b )
```

Add a path *b* to the EmbArray *a* and it returns if the element was successfully added.

```
18.40.1.7 embArray_addPoint() int embArray_addPoint (
    EmbArray * a,
    EmbPoint b )
```

Add a point *b* to the EmbArray *a* and it returns if the element was successfully added.

```
18.40.1.8 embArray_addPolygon() int embArray_addPolygon (
    EmbArray * a,
    EmbPolygon b )
```

Add a polygon *b* to the EmbArray *a* and it returns if the element was successfully added.

```
18.40.1.9 embArray_addPolyline() int embArray_addPolyline (
    EmbArray * a,
    EmbPolyline b )
```

Add a polyline *b* to the EmbArray *a* and it returns if the element was successfully added.

```
18.40.1.10 embArray_addRect() int embArray_addRect (
    EmbArray * a,
    EmbRect b )
```

Add a rectangle *b* to the EmbArray *a* and it returns if the element was successfully added.

```
18.40.1.11 embArray_addStitch() int embArray_addStitch (
    EmbArray * a,
    EmbStitch b )
```

Add a stitch *b* to the EmbArray *a* and it returns if the element was successfully added.

```
18.40.1.12 embArray_addVector() int embArray_addVector (
    EmbArray * a,
    EmbVector b )
```

Add a vector *b* to the EmbArray *a* and it returns if the element was successfully added.

```
18.40.1.13 embArray_copy() void embArray_copy (
    EmbArray * dst,
    EmbArray * src )
```

Copies all entries in the EmbArray struct from *src* to *dst*.

18.40.1.14 embArray_create() `EmbArray * embArray_create (int type)`

Allocates memory for an EmbArray of the type determined by the argument *type*.

18.40.1.15 embArray_free() `void embArray_free (EmbArray * a)`

Free the memory of EmbArray *a*, recursively if necessary.

18.40.1.16 embArray_resize() `int embArray_resize (EmbArray * a)`

Resizes the array *a* to be CHUNK_SIZE entries longer if and only if the amount of room left is less than 3 entries.

18.41 extern/libembroidery/src/compress.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "embroidery_internal.h"
```

Functions

- int `hus_compress (char *data, int length, char *output, int *output_length)`
- void `huffman_build_table (huffman *h)`
- int * `huffman_lookup (huffman h, int byte_lookup)`
- void `compress_init ()`
- int `compress_get_bits (compress *c, int length)`
- int `compress_pop (compress *c, int bit_count)`
- int `compress_peek (compress *c, int bit_count)`
- int `compress_read_variable_length (compress *c)`
- void `compress_load_character_length_huffman (compress *c)`
- void `compress_load_character_huffman (compress *c)`
- void `compress_load_distance_huffman (compress *c)`
- void `compress_load_block (compress *c)`
- int `compress_get_token (compress *c)`
- int `compress_get_position (compress *c)`
- int `hus_decompress (char *data, int length, char *output, int *output_length)`

Variables

- int `huffman_lookup_data [2]`

18.41.1 Detailed Description

This is a work in progress.

Thanks to Jason Weiler for describing the binary formats of the HUS and VIP formats at:

<http://www.jasonweiler.com/HUSandVIPFileInfo.html>

Further thanks to github user tatarize for solving the mystery of the compression in:

<https://github.com/EmbroidePy/pyembroidery>

with a description of that work here:

<https://stackoverflow.com/questions/7852670/greenleaf-archive-library>

This is based on their work.

18.41.2 Function Documentation

18.41.2.1 compress_get_bits() int compress_get_bits (

```
    compress * c,
    int length )
```

c.length Returns .

18.41.2.2 compress_get_position() int compress_get_position (

```
    compress * c )
```

c. Returns the position as an int.

18.41.2.3 compress_get_token() int compress_get_token (

```
    compress * c )
```

c. Returns the token as an int.

18.41.2.4 compress_init() void compress_init ()

18.41.2.5 compress_load_block() void compress_load_block (

```
    compress * c )
```

c. Returns nothing.

18.41.2.6 compress_load_character_huffman() void compress_load_character_huffman (

```
    compress * c )
```

Load character table to compress struct *c*. Returns nothing.

18.41.2.7 compress_load_character_length_huffman() void compress_load_character_length_huffman (

```
(    compress * c )
```

c. Returns.

18.41.2.8 compress_load_distance_huffman() void compress_load_distance_huffman (

```
    compress * c )
```

c. Returns nothing.

18.41.2.9 compress_peek() int compress_peek (

```
    compress * c,
    int bit_count )
```

c.bit_count. Returns.

18.41.2.10 compress_pop() int compress_pop (

```
    compress * c,
    int bit_count )
```

c.bit_count. Returns.

18.41.2.11 compress_read_variable_length() int compress_read_variable_length (

```
    compress * c )
```

c. Returns.

18.41.2.12 huffman_build_table() void huffman_build_table (

```
    huffman * h )
```

These next 2 functions represent the [Huffman](#) class in tartarize's code. *h*

18.41.2.13 huffman_lookup() int * huffman_lookup (

```
    huffman h,
    int byte_lookup )
```

Lookup *byte_lookup* in huffman table *h* return result as two bytes using the memory *huffman_lookup_data*.

```
18.41.2.14 hus_compress() int hus_compress (
    char * data,
    int length,
    char * output,
    int * output_length )
```

data length output output_length. Returns whether it was successful as an int.

This avoids the now unnecessary compression by placing a minimal header of 6 bytes and using only literals in the huffman compressed part (see the sources above).

```
18.41.2.15 hus_decompress() int hus_decompress (
    char * data,
    int length,
    char * output,
    int * output_length )
```

data length output output_length. Returns whether the decompression was successful.

18.41.3 Variable Documentation

18.41.3.1 huffman_lookup_data int huffman_lookup_data[2]

18.42 extern/libembroidery/src/embroidery.h File Reference

Classes

- struct EmbColor_
- struct EmbVector_
- struct EmblImage_
- struct EmbBlock_
- struct EmbAlignedDim_
- struct EmbAngularDim_
- struct EmbArcLengthDim_
- struct EmbDiameterDim_
- struct EmbLeaderDim_
- struct EmbLinearDim_
- struct EmbOrdinateDim_
- struct EmbRadiusDim_
- struct EmbInfiniteLine_
- struct EmbRay_
- struct EmbTextMulti_
- struct EmbTextSingle_
- struct EmbTime_
- struct EmbPoint_
- struct EmbLine_
- struct EmbPath_
- struct EmbStitch_
- struct EmbThread_
- struct thread_color_
- struct EmbArc_
absolute position (not relative)
- struct EmbRect_
- struct EmbCircle_
- struct EmbSatinOutline_
- struct EmbEllipse_
- struct EmbBezier_

- struct EmbSpline_
- struct LSYSTEM
- struct EmbGeometry_
- struct EmbArray_
- struct EmbLayer_
- struct EmbPattern_
- struct EmbFormatList_

Macros

- #define LIBEMBROIDERY_EMBEDDED_VERSION 0
- #define NORMAL 0 /*! stitch to (x, y) */
- #define JUMP 1 /*! move to (x, y) */
- #define TRIM 2 /*! trim + move to (x, y) */
- #define STOP 4 /*! pause machine for thread change */
- #define SEQUIN 8 /*! sequin */
- #define END 16 /*! end of program */
- #define EMB_FORMAT_100 0
- #define EMB_FORMAT_10O 1
- #define EMB_FORMAT_ART 2
- #define EMB_FORMAT_BMC 3
- #define EMB_FORMAT_BRO 4
- #define EMB_FORMAT_CND 5
- #define EMB_FORMAT_COL 6
- #define EMB_FORMAT_CSD 7
- #define EMB_FORMAT_CSV 8
- #define EMB_FORMAT_DAT 9
- #define EMB_FORMAT_DEM 10
- #define EMB_FORMAT_DSB 11
- #define EMB_FORMAT_DST 12
- #define EMB_FORMAT_DSZ 13
- #define EMB_FORMAT_DXF 14
- #define EMB_FORMAT_EDR 15
- #define EMB_FORMAT_EMD 16
- #define EMB_FORMAT_EXP 17
- #define EMB_FORMAT_EXY 18
- #define EMB_FORMAT_EYS 19
- #define EMB_FORMAT_FXY 20
- #define EMB_FORMAT_GC 21
- #define EMB_FORMAT_GNC 22
- #define EMB_FORMAT_GT 23
- #define EMB_FORMAT_HUS 24
- #define EMB_FORMAT_INB 25
- #define EMB_FORMAT_INF 26
- #define EMB_FORMAT_JEF 27
- #define EMB_FORMAT_KSM 28
- #define EMB_FORMAT_MAX 29
- #define EMB_FORMAT_MIT 30
- #define EMB_FORMAT_NEW 31
- #define EMB_FORMAT_OFM 32
- #define EMB_FORMAT_PCD 33
- #define EMB_FORMAT_PCM 34
- #define EMB_FORMAT_PCQ 35
- #define EMB_FORMAT_PCS 36
- #define EMB_FORMAT_PEC 37

- #define EMB_FORMAT_PEL 38
- #define EMB_FORMAT_PEM 39
- #define EMB_FORMAT_PES 40
- #define EMB_FORMAT_PHB 41
- #define EMB_FORMAT_PHC 42
- #define EMB_FORMAT_PLT 43
- #define EMB_FORMAT_RGB 44
- #define EMB_FORMAT_SEW 45
- #define EMB_FORMAT_SHV 46
- #define EMB_FORMAT_SST 47
- #define EMB_FORMAT_STX 48
- #define EMB_FORMAT_SVG 49
- #define EMB_FORMAT_T01 50
- #define EMB_FORMAT_T09 51
- #define EMB_FORMAT_TAP 52
- #define EMB_FORMAT_THR 53
- #define EMB_FORMAT_TXT 54
- #define EMB_FORMAT_U00 55
- #define EMB_FORMAT_U01 56
- #define EMB_FORMAT_VIP 57
- #define EMB_FORMAT_VP3 58
- #define EMB_FORMAT_XXX 59
- #define EMB_FORMAT_ZSK 60
- #define Arc_Polyester 0
- #define Arc_Rayon 1
- #define CoatsAndClark_Rayon 2
- #define Exquisite_Polyester 3
- #define Fufu_Polyester 4
- #define Fufu_Rayon 5
- #define Hemingworth_Polyester 6
- #define Isacord_Polyester 7
- #define Isafil_Rayon 8
- #define Marathon_Polyester 9
- #define Marathon_Rayon 10
- #define Madeira_Polyester 11
- #define Madeira_Rayon 12
- #define Metro_Polyester 13
- #define Pantone 14
- #define RobisonAnton_Polyester 15
- #define RobisonAnton_Rayon 16
- #define Sigma_Polyester 17
- #define Sulky_Rayon 18
- #define ThreadArt_Rayon 19
- #define ThreadArt_Polyester 20
- #define ThreaDelight_Polyester 21
- #define Z102_Isacord_Polyester 22
- #define SVG_Colors 23
- #define hus_thread 24
- #define jef_thread 25
- #define pcm_thread 26
- #define pec_thread 27
- #define shv_thread 28
- #define dxf_color 29
- #define EMB_ARRAY 0
- #define EMB_ARC 1

- #define EMB_CIRCLE 2
- #define EMB_DIM_DIAMETER 3
- #define EMB_DIM_LEADER 4
- #define EMB_ELLIPSE 5
- #define EMB_FLAG 6
- #define EMB_LINE 7
- #define EMB_IMAGE 8
- #define EMB_PATH 9
- #define EMB_POINT 10
- #define EMB_POLYGON 11
- #define EMB_POLYLINE 12
- #define EMB_RECT 13
- #define EMB_SPLINE 14
- #define EMB_STITCH 15
- #define EMB_TEXT_SINGLE 16
- #define EMB_TEXT_MULTI 17
- #define EMB_VECTOR 18
- #define EMB_THREAD 19
- #define EMBFORMAT_UNSUPPORTED 0
- #define EMBFORMAT_STITCHONLY 1
- #define EMBFORMAT_OBJECTONLY 2
- #define EMBFORMAT_STCHANDOBJ 3 /* binary operation: 1+2=3 */
- #define numberFormats 61
- #define CHUNK_SIZE 128
- #define EMB_MAX_LAYERS 10
- #define MAX_THREADS 256
- #define EMBFORMAT_MAXEXT 3
- #define EMBFORMAT_MAXDESC 50
- #define MAX_STITCHES 1000000
- #define EMB_PUBLIC

Typedefs

- typedef float EmbReal
- typedef struct EmbColor_ EmbColor
- typedef struct EmbVector_ EmbVector
- typedef struct EmbArray_ EmbArray
- typedef struct EmbImage_ EmbImage
- typedef struct EmbBlock_ EmbBlock
- typedef struct EmbAlignedDim_ EmbAlignedDim
- typedef struct EmbAngularDim_ EmbAngularDim
- typedef struct EmbArcLengthDim_ EmbArcLengthDim
- typedef struct EmbDiameterDim_ EmbDiameterDim
- typedef struct EmbLeaderDim_ EmbLeaderDim
- typedef struct EmbLinearDim_ EmbLinearDim
- typedef struct EmbOrdinateDim_ EmbOrdinateDim
- typedef struct EmbRadiusDim_ EmbRadiusDim
- typedef struct EmbInfiniteLine_ EmbInfiniteLine
- typedef struct EmbRay_ EmbRay
- typedef struct EmbTextMulti_ EmbTextMulti
- typedef struct EmbTextSingle_ EmbTextSingle
- typedef struct EmbTime_ EmbTime
- typedef struct EmbPoint_ EmbPoint
- typedef struct EmbLine_ EmbLine
- typedef struct EmbPath_ EmbPath

- `typedef struct EmbStitch_ EmbStitch`
- `typedef struct EmbThread_ EmbThread`
- `typedef struct thread_color_ thread_color`
- `typedef struct EmbArc_ EmbArc`
`absolute position (not relative)`
- `typedef struct EmbRect_ EmbRect`
- `typedef struct EmbCircle_ EmbCircle`
- `typedef EmbPath EmbPolygon`
- `typedef EmbPath EmbPolyline`
- `typedef int EmbFlag`
- `typedef struct EmbSatinOutline_ EmbSatinOutline`
- `typedef struct EmbEllipse_ EmbEllipse`
- `typedef struct EmbBezier_ EmbBezier`
- `typedef struct EmbSpline_ EmbSpline`
- `typedef struct LSYSTEM L_system`
- `typedef struct EmbGeometry_ EmbGeometry`
- `typedef struct EmbLayer_ EmbLayer`
- `typedef struct EmbPattern_ EmbPattern`
- `typedef struct EmbFormatList_ EmbFormatList`

Functions

- `EMB_PUBLIC int lindenmayer_system (L_system L, char *state, int iteration, int complete)`
- `EMB_PUBLIC int hilbert_curve (EmbPattern *pattern, int iterations)`
- `EMB_PUBLIC int emb_identify_format (const char *ending)`
`fileName`
- `EMB_PUBLIC void testMain (int level)`
- `EMB_PUBLIC int convert (const char *inf, const char *outf)`
- `EMB_PUBLIC EmbColor embColor_make (unsigned char r, unsigned char g, unsigned char b)`
- `EMB_PUBLIC EmbColor * embColor_create (unsigned char r, unsigned char g, unsigned char b)`
- `EMB_PUBLIC EmbColor embColor_fromHexStr (char *val)`
`Converts a 6 digit hex string (I.E. "00FF00") into an EmbColor and returns it.`
- `EMB_PUBLIC int embColor_distance (EmbColor a, EmbColor b)`
`a b`
- `EMB_PUBLIC EmbArray * embArray_create (int type)`
- `EMB_PUBLIC int embArray_resize (EmbArray *g)`
- `EMB_PUBLIC void embArray_copy (EmbArray *dst, EmbArray *src)`
- `EMB_PUBLIC int embArray_addArc (EmbArray *g, EmbArc arc)`
- `EMB_PUBLIC int embArray_addCircle (EmbArray *g, EmbCircle circle)`
- `EMB_PUBLIC int embArray_addEllipse (EmbArray *g, EmbEllipse ellipse)`
- `EMB_PUBLIC int embArray_addFlag (EmbArray *g, int flag)`
- `EMB_PUBLIC int embArray_addLine (EmbArray *g, EmbLine line)`
- `EMB_PUBLIC int embArray_addRect (EmbArray *g, EmbRect rect)`
- `EMB_PUBLIC int embArray_addPath (EmbArray *g, EmbPath p)`
- `EMB_PUBLIC int embArray_addPoint (EmbArray *g, EmbPoint p)`
- `EMB_PUBLIC int embArray_addPolygon (EmbArray *g, EmbPolygon p)`
- `EMB_PUBLIC int embArray_addPolyline (EmbArray *g, EmbPolyline p)`
- `EMB_PUBLIC int embArray_addStitch (EmbArray *g, EmbStitch st)`
- `EMB_PUBLIC int embArray_addThread (EmbArray *g, EmbThread p)`
- `EMB_PUBLIC int embArray_addVector (EmbArray *g, EmbVector)`
- `EMB_PUBLIC void embArray_free (EmbArray *p)`
- `EMB_PUBLIC EmbLine embLine_make (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2)`
- `EMB_PUBLIC void embLine_normalVector (EmbLine line, EmbVector *result, int clockwise)`
- `EMB_PUBLIC EmbVector embLine_intersectionPoint (EmbLine line1, EmbLine line2)`

- EMB_PUBLIC int embThread_findNearestColor (EmbColor color, EmbColor *colors, int n_colors)
 - EMB_PUBLIC int embThread_findNearestThread (EmbColor color, EmbThread *threads, int n_threads)

color thread_list n_threads
 - EMB_PUBLIC EmbThread embThread_getRandom (void)
 - EMB_PUBLIC void embVector_normalize (EmbVector vector, EmbVector *result)
 - EMB_PUBLIC void embVector_multiply (EmbVector vector, EmbReal magnitude, EmbVector *result)
 - EMB_PUBLIC EmbVector embVector_add (EmbVector v1, EmbVector v2)
 - EMB_PUBLIC EmbVector embVector_average (EmbVector v1, EmbVector v2)
 - EMB_PUBLIC EmbVector embVector_subtract (EmbVector v1, EmbVector v2)
 - EMB_PUBLIC EmbReal embVector_dot (EmbVector v1, EmbVector v2)
 - EMB_PUBLIC EmbReal embVector_cross (EmbVector v1, EmbVector v2)
- The "cross product" as vectors a and b returned as a real value.*
- EMB_PUBLIC void embVector_transpose_product (EmbVector v1, EmbVector v2, EmbVector *result)
 - EMB_PUBLIC EmbReal embVector_length (EmbVector vector)
 - EMB_PUBLIC EmbReal embVector_relativeX (EmbVector a1, EmbVector a2, EmbVector a3)
 - EMB_PUBLIC EmbReal embVector_relativeY (EmbVector a1, EmbVector a2, EmbVector a3)
 - EMB_PUBLIC EmbReal embVector_angle (EmbVector v)
 - EMB_PUBLIC EmbReal embVector_distance (EmbVector a, EmbVector b)
 - EMB_PUBLIC EmbVector embVector_unit (EmbReal angle)
 - EMB_PUBLIC EmbArc embArc_init (void)
 - EMB_PUBLIC char embArc_clockwise (EmbArc arc)
 - EMB_PUBLIC void getArcCenter (EmbArc arc, EmbVector *arcCenter)
 - EMB_PUBLIC char getArcDataFromBulge (EmbReal bulge, EmbArc *arc, EmbReal *arcCenterX, EmbReal *arcCenterY, EmbReal *radius, EmbReal *diameter, EmbReal *chord, EmbReal *chordMidX, EmbReal *chordMidY, EmbReal *sagitta, EmbReal *apothem, EmbReal *incAngleInDegrees, char *clockwise)
 - EMB_PUBLIC EmbCircle embCircle_init (void)
 - EMB_PUBLIC int getCircleCircleIntersections (EmbCircle c0, EmbCircle c1, EmbVector *v0, EmbVector *v1)
 - EMB_PUBLIC int getCircleTangentPoints (EmbCircle c, EmbVector p, EmbVector *v0, EmbVector *v1)
 - EMB_PUBLIC EmbEllipse embEllipse_init (void)
 - EMB_PUBLIC EmbEllipse embEllipse_make (EmbReal cx, EmbReal cy, EmbReal rx, EmbReal ry)
 - EMB_PUBLIC EmbReal embEllipse_diameterX (EmbEllipse ellipse)
 - EMB_PUBLIC EmbReal embEllipse_diameterY (EmbEllipse ellipse)
 - EMB_PUBLIC EmbReal embEllipse_width (EmbEllipse ellipse)
 - EMB_PUBLIC EmbReal embEllipse_height (EmbEllipse ellipse)
 - EMB_PUBLIC EmbReal embEllipse_area (EmbEllipse ellipse)
 - EMB_PUBLIC EmbReal embEllipse_perimeter (EmbEllipse ellipse)
 - EMB_PUBLIC EmblImage emblImage_create (int, int)
 - EMB_PUBLIC void emblImage_read (EmblImage *image, char *fname)
 - EMB_PUBLIC int emblImage_write (EmblImage *image, char *fname)
 - EMB_PUBLIC void emblImage_free (EmblImage *image)
 - EMB_PUBLIC EmbRect embRect_init (void)
 - EMB_PUBLIC EmbReal embRect_area (EmbRect)
 - EMB_PUBLIC int threadColor (const char *, int brand)
 - EMB_PUBLIC int threadColorNum (unsigned int color, int brand)
 - EMB_PUBLIC const char * threadColorName (unsigned int color, int brand)
 - EMB_PUBLIC void embTime_initNow (EmbTime *t)

t
 - EMB_PUBLIC EmbTime embTime_time (EmbTime *t)

t
 - EMB_PUBLIC void embSatinOutline_generateSatinOutline (EmbArray *lines, EmbReal thickness, EmbSatinOutline *result)

lines thickness result
 - EMB_PUBLIC EmbArray * embSatinOutline_renderStitches (EmbSatinOutline *result, EmbReal density)

result density

- **EMB_PUBLIC EmbGeometry * embGeometry_init (int type_in)**
Our generic object interface backends to each individual type.
- **EMB_PUBLIC void embGeometry_free (EmbGeometry *obj)**
Free the memory occupied by a non-stitch geometry object.
- **EMB_PUBLIC void embGeometry_move (EmbGeometry *obj, EmbVector delta)**
Translate obj by the vector delta.
- **EMB_PUBLIC EmbRect embGeometry_boundingRect (EmbGeometry *obj)**
Calculate the bounding box of geometry obj based on what kind of geometric object it is.
- **EMB_PUBLIC void embGeometry_vulcanize (EmbGeometry *obj)**
Toggle the rubber mode of the object.
- **EMB_PUBLIC EmbPattern * embPattern_create (void)**
- **EMB_PUBLIC void embPattern_hideStitchesOverLength (EmbPattern *p, int length)**
- **EMB_PUBLIC void embPattern_fixColorCount (EmbPattern *p)**
- **EMB_PUBLIC int embPattern_addThread (EmbPattern *p, EmbThread thread)**
- **EMB_PUBLIC void embPattern_addStitchAbs (EmbPattern *p, EmbReal x, EmbReal y, int flags, int isAuto←ColorIndex)**
- **EMB_PUBLIC void embPattern_addStitchRel (EmbPattern *p, EmbReal dx, EmbReal dy, int flags, int is←AutoColorIndex)**
- **EMB_PUBLIC void embPattern_changeColor (EmbPattern *p, int index)**
- **EMB_PUBLIC void embPattern_free (EmbPattern *p)**
- **EMB_PUBLIC void embPattern_scale (EmbPattern *p, EmbReal scale)**
- **EMB_PUBLIC EmbReal embPattern_totalStitchLength (EmbPattern *pattern)**
- **EMB_PUBLIC EmbReal embPattern_minimumStitchLength (EmbPattern *pattern)**
- **EMB_PUBLIC EmbReal embPattern_maximumStitchLength (EmbPattern *pattern)**
- **EMB_PUBLIC void embPattern_lengthHistogram (EmbPattern *pattern, int *bin, int NUMBINS)**
- **EMB_PUBLIC int embPattern_realStitches (EmbPattern *pattern)**
- **EMB_PUBLIC int embPattern_jumpStitches (EmbPattern *pattern)**
- **EMB_PUBLIC int embPattern_trimStitches (EmbPattern *pattern)**
- **EMB_PUBLIC EmbRect embPattern_calcBoundingBox (EmbPattern *p)**
- **EMB_PUBLIC void embPattern_flipHorizontal (EmbPattern *p)**
- **EMB_PUBLIC void embPattern_flipVertical (EmbPattern *p)**
- **EMB_PUBLIC void embPattern_flip (EmbPattern *p, int horz, int vert)**
- **EMB_PUBLIC void embPattern_combineJumpStitches (EmbPattern *p)**
- **EMB_PUBLIC void embPattern_correctForMaxStitchLength (EmbPattern *p, EmbReal maxStitchLength, EmbReal maxJumpLength)**
- **EMB_PUBLIC void embPattern_center (EmbPattern *p)**
- **EMB_PUBLIC void embPattern_loadExternalColorFile (EmbPattern *p, const char *fileName)**
- **EMB_PUBLIC void embPattern_convertGeometry (EmbPattern *p)**
- **EMB_PUBLIC void embPattern_designDetails (EmbPattern *p)**
- **EMB_PUBLIC EmbPattern * embPattern_combine (EmbPattern *p1, EmbPattern *p2)**
- **EMB_PUBLIC int embPattern_color_count (EmbPattern *pattern, EmbColor startColor)**
- **EMB_PUBLIC void embPattern_end (EmbPattern *p)**
- **EMB_PUBLIC void embPattern_crossstitch (EmbPattern *pattern, EmblImage *, int threshold)**
- **EMB_PUBLIC void embPattern_horizontal_fill (EmbPattern *pattern, EmblImage *, int threshold)**
- **EMB_PUBLIC int embPattern_render (EmbPattern *pattern, char *fname)**
- **EMB_PUBLIC int embPattern_simulate (EmbPattern *pattern, char *fname)**
- **EMB_PUBLIC void embPattern_addCircleAbs (EmbPattern *p, EmbCircle obj)**
- **EMB_PUBLIC void embPattern_addEllipseAbs (EmbPattern *p, EmbEllipse obj)**
- **EMB_PUBLIC void embPattern_addLineAbs (EmbPattern *p, EmbLine obj)**
- **EMB_PUBLIC void embPattern_addPathAbs (EmbPattern *p, EmbPath obj)**
- **EMB_PUBLIC void embPattern_addPointAbs (EmbPattern *p, EmbPoint obj)**
- **EMB_PUBLIC void embPattern_addPolygonAbs (EmbPattern *p, EmbPolygon obj)**
- **EMB_PUBLIC void embPattern_addPolylineAbs (EmbPattern *p, EmbPolyline obj)**
- **EMB_PUBLIC void embPattern_addRectAbs (EmbPattern *p, EmbRect obj)**

- EMB_PUBLIC void `embPattern_copyStitchListToPolylines` (EmbPattern *pattern)
- EMB_PUBLIC void `embPattern_copyPolylinesToStitchList` (EmbPattern *pattern)
- EMB_PUBLIC void `embPattern_moveStitchListToPolylines` (EmbPattern *pattern)
- EMB_PUBLIC void `embPattern_movePolylinesToStitchList` (EmbPattern *pattern)
- EMB_PUBLIC char `embPattern_read` (EmbPattern *pattern, const char *fileName, int format)
pattern fileName format
- EMB_PUBLIC char `embPattern_write` (EmbPattern *pattern, const char *fileName, int format)
pattern fileName format
- EMB_PUBLIC char `embPattern_readAuto` (EmbPattern *pattern, const char *fileName)
pattern fileName
- EMB_PUBLIC char `embPattern_writeAuto` (EmbPattern *pattern, const char *fileName)
pattern fileName
- EMB_PUBLIC void `report` (int result, char *label)
- EMB_PUBLIC int `full_test_matrix` (char *fname)
- EMB_PUBLIC int `emb_round` (EmbReal x)
- EMB_PUBLIC EmbReal `radians` (EmbReal degree)
- EMB_PUBLIC EmbReal `degrees` (EmbReal radian)

Variables

- EmbFormatList `formatTable` [numberOfFormats]
- const int `pecThreadCount`
- const int `shvThreadCount`
- const EmbReal `embConstantPi`
- const EmbThread `husThreads` []
- const EmbThread `jefThreads` []
- const EmbThread `shvThreads` []
- const EmbThread `pcmThreads` []
- const EmbThread `pecThreads` []
- const unsigned char `_dxfColorTable` [][][3]
- EmbThread `black_thread`
- const unsigned char `vipDecodingTable` []
- int `emb_error`
Error code storage for optional control flow blocking.
- int `emb_verbose`
Verbosity level.

18.42.1 Macro Definition Documentation

18.42.1.1 Arc_Polyester #define Arc_Polyester 0

18.42.1.2 Arc_Rayon #define Arc_Rayon 1

18.42.1.3 CHUNK_SIZE #define CHUNK_SIZE 128

18.42.1.4 CoatsAndClark_Rayon #define CoatsAndClark_Rayon 2

18.42.1.5 dxf_color #define dxf_color 29

18.42.1.6 EMB_ARC #define EMB_ARC 1

18.42.1.7 EMB_ARRAY #define EMB_ARRAY 0

18.42.1.8 EMB_CIRCLE #define EMB_CIRCLE 2

18.42.1.9 EMB_DIM_DIAMETER #define EMB_DIM_DIAMETER 3

18.42.1.10 EMB_DIM_LEADER #define EMB_DIM_LEADER 4

18.42.1.11 EMB_ELLIPSE #define EMB_ELLIPSE 5

18.42.1.12 EMB_FLAG #define EMB_FLAG 6

18.42.1.13 EMB_FORMAT_100 #define EMB_FORMAT_100 0

Format identifiers

18.42.1.14 EMB_FORMAT_10O #define EMB_FORMAT_10O 1

18.42.1.15 EMB_FORMAT_ART #define EMB_FORMAT_ART 2

18.42.1.16 EMB_FORMAT_BMC #define EMB_FORMAT_BMC 3

18.42.1.17 EMB_FORMAT_BRO #define EMB_FORMAT_BRO 4

18.42.1.18 EMB_FORMAT_CND #define EMB_FORMAT_CND 5

18.42.1.19 EMB_FORMAT_COL #define EMB_FORMAT_COL 6

18.42.1.20 EMB_FORMAT_CSD #define EMB_FORMAT_CSD 7

18.42.1.21 EMB_FORMAT_CSV #define EMB_FORMAT_CSV 8

18.42.1.22 EMB_FORMAT_DAT #define EMB_FORMAT_DAT 9

18.42.1.23 EMB_FORMATDEM #define EMB_FORMATDEM 10

18.42.1.24 EMB_FORMAT_DSB #define EMB_FORMAT_DSB 11

18.42.1.25 EMB_FORMAT_DST #define EMB_FORMAT_DST 12

18.42.1.26 EMB_FORMAT_DSZ #define EMB_FORMAT_DSZ 13

18.42.1.27 EMB_FORMAT_DXF #define EMB_FORMAT_DXF 14

18.42.1.28 EMB_FORMAT_EDR #define EMB_FORMAT_EDR 15

18.42.1.29 EMB_FORMAT_EMD #define EMB_FORMAT_EMD 16

18.42.1.30 EMB_FORMAT_EXP #define EMB_FORMAT_EXP 17

18.42.1.31 EMB_FORMAT_EXY #define EMB_FORMAT_EXY 18

18.42.1.32 EMB_FORMAT_EYS #define EMB_FORMAT_EYS 19

18.42.1.33 EMB_FORMAT_FXY #define EMB_FORMAT_FXY 20

18.42.1.34 EMB_FORMAT_GC #define EMB_FORMAT_GC 21

18.42.1.35 EMB_FORMAT_GNC #define EMB_FORMAT_GNC 22

18.42.1.36 EMB_FORMAT_GT #define EMB_FORMAT_GT 23

18.42.1.37 EMB_FORMAT_HUS #define EMB_FORMAT_HUS 24

18.42.1.38 EMB_FORMAT_INB #define EMB_FORMAT_INB 25

18.42.1.39 EMB_FORMAT_INF #define EMB_FORMAT_INF 26

18.42.1.40 EMB_FORMAT_JEF #define EMB_FORMAT_JEF 27

18.42.1.41 EMB_FORMAT_KSM #define EMB_FORMAT_KSM 28

18.42.1.42 EMB_FORMAT_MAX #define EMB_FORMAT_MAX 29

18.42.1.43 EMB_FORMAT_MIT #define EMB_FORMAT_MIT 30

18.42.1.44 EMB_FORMAT_NEW #define EMB_FORMAT_NEW 31

18.42.1.45 EMB_FORMAT_OFM #define EMB_FORMAT_OFM 32

18.42.1.46 EMB_FORMAT_PCD #define EMB_FORMAT_PCD 33

18.42.1.47 EMB_FORMAT_PCM #define EMB_FORMAT_PCM 34

18.42.1.48 EMB_FORMAT_PCQ #define EMB_FORMAT_PCQ 35

18.42.1.49 EMB_FORMAT_PCS #define EMB_FORMAT_PCS 36

18.42.1.50 EMB_FORMAT_PEC #define EMB_FORMAT_PEC 37

18.42.1.51 EMB_FORMAT_PEL #define EMB_FORMAT_PEL 38

18.42.1.52 EMB_FORMAT_PEM #define EMB_FORMAT_PEM 39

18.42.1.53 EMB_FORMAT_PES #define EMB_FORMAT_PES 40

18.42.1.54 EMB_FORMAT_PHB #define EMB_FORMAT_PHB 41

18.42.1.55 EMB_FORMAT_PHC #define EMB_FORMAT_PHC 42

18.42.1.56 EMB_FORMAT_PLT #define EMB_FORMAT_PLT 43

18.42.1.57 EMB_FORMAT_RGB #define EMB_FORMAT_RGB 44

18.42.1.58 EMB_FORMAT_SEW #define EMB_FORMAT_SEW 45

18.42.1.59 EMB_FORMAT_SHV #define EMB_FORMAT_SHV 46

18.42.1.60 EMB_FORMAT_SST #define EMB_FORMAT_SST 47

18.42.1.61 EMB_FORMAT_STX #define EMB_FORMAT_STX 48

18.42.1.62 EMB_FORMAT_SVG #define EMB_FORMAT_SVG 49

18.42.1.63 EMB_FORMAT_T01 #define EMB_FORMAT_T01 50

18.42.1.64 EMB_FORMAT_T09 #define EMB_FORMAT_T09 51

18.42.1.65 EMB_FORMAT_TAP #define EMB_FORMAT_TAP 52

18.42.1.66 EMB_FORMAT_THR #define EMB_FORMAT_THR 53

18.42.1.67 EMB_FORMAT_TXT #define EMB_FORMAT_TXT 54

18.42.1.68 EMB_FORMAT_U00 #define EMB_FORMAT_U00 55

18.42.1.69 EMB_FORMAT_U01 #define EMB_FORMAT_U01 56

18.42.1.70 EMB_FORMAT_VIP #define EMB_FORMAT_VIP 57

18.42.1.71 EMB_FORMAT_VP3 #define EMB_FORMAT_VP3 58

18.42.1.72 EMB_FORMAT_XXX #define EMB_FORMAT_XXX 59

18.42.1.73 EMB_FORMAT_ZSK #define EMB_FORMAT_ZSK 60

18.42.1.74 EMB_IMAGE #define EMB_IMAGE 8

18.42.1.75 EMB_LINE #define EMB_LINE 7

18.42.1.76 EMB_MAX_LAYERS #define EMB_MAX_LAYERS 10

18.42.1.77 EMB_PATH #define EMB_PATH 9

18.42.1.78 EMB_POINT #define EMB_POINT 10

18.42.1.79 EMB_POLYGON #define EMB_POLYGON 11

18.42.1.80 EMB_POLYLINE #define EMB_POLYLINE 12

18.42.1.81 EMB_PUBLIC #define EMB_PUBLIC

18.42.1.82 EMB_RECT #define EMB_RECT 13

18.42.1.83 EMB_SPLINE #define EMB_SPLINE 14

18.42.1.84 EMB_STITCH #define EMB_STITCH 15

18.42.1.85 EMB_TEXT_MULTI #define EMB_TEXT_MULTI 17

18.42.1.86 EMB_TEXT_SINGLE #define EMB_TEXT_SINGLE 16

18.42.1.87 EMB_THREAD #define EMB_THREAD 19

18.42.1.88 EMB_VECTOR #define EMB_VECTOR 18

18.42.1.89 EMBFORMAT_MAXDESC #define EMBFORMAT_MAXDESC 50

18.42.1.90 EMBFORMAT_MAXEXT #define EMBFORMAT_MAXEXT 3

18.42.1.91 EMBFORMAT_OBJECTONLY #define EMBFORMAT_OBJECTONLY 2

18.42.1.92 EMBFORMAT_STCHANDOBJ #define EMBFORMAT_STCHANDOBJ 3 /* binary operation←
: 1+2=3 */

18.42.1.93 EMBFORMAT_STITCHONLY #define EMBFORMAT_STITCHONLY 1

18.42.1.94 EMBFORMAT_UNSUPPORTED #define EMBFORMAT_UNSUPPORTED 0

18.42.1.95 END #define END 16 /*! end of program */

18.42.1.96 Exquisite_Polyester #define Exquisite_Polyester 3

18.42.1.97 Fufu_Polyester #define Fufu_Polyester 4

18.42.1.98 Fufu_Rayon #define Fufu_Rayon 5

18.42.1.99 Hemingworth_Polyester #define Hemingworth_Polyester 6

18.42.1.100 hus_thread #define hus_thread 24

18.42.1.101 Isacord_Polyester #define Isacord_Polyester 7

18.42.1.102 Isafil_Rayon #define Isafil_Rayon 8

18.42.1.103 jef_thread #define jef_thread 25

18.42.1.104 JUMP #define JUMP 1 /*! move to (x, y) */

18.42.1.105 LIBEMBROIDERY_EMBEDDED_VERSION #define LIBEMBROIDERY_EMBEDDED_VERSION 0

18.42.1.106 Madeira_Polyester #define Madeira_Polyester 11

18.42.1.107 Madeira_Rayon #define Madeira_Rayon 12

18.42.1.108 Marathon_Polyester #define Marathon_Polyester 9

18.42.1.109 Marathon_Rayon #define Marathon_Rayon 10

18.42.1.110 MAX_STITCHES #define MAX_STITCHES 1000000

18.42.1.111 MAX_THREADS #define MAX_THREADS 256

18.42.1.112 Metro_Polyester #define Metro_Polyester 13

18.42.1.113 NORMAL #define NORMAL 0 /*! stitch to (x, y) */

Machine codes for stitch flags

18.42.1.114 `numberOfFormats` #define numberOfFormats 61

18.42.1.115 `Pantone` #define Pantone 14

18.42.1.116 `pcm_thread` #define pcm_thread 26

18.42.1.117 `pec_thread` #define pec_thread 27

18.42.1.118 `RobisonAnton_Polyester` #define RobisonAnton_Polyester 15

18.42.1.119 `RobisonAnton_Rayon` #define RobisonAnton_Rayon 16

18.42.1.120 `SEQUIN` #define SEQUIN 8 /*! sequin */

18.42.1.121 `shv_thread` #define shv_thread 28

18.42.1.122 `Sigma_Polyester` #define Sigma_Polyester 17

18.42.1.123 `STOP` #define STOP 4 /*! pause machine for thread change */

18.42.1.124 `Sulky_Rayon` #define Sulky_Rayon 18

18.42.1.125 `SVG_Colors` #define SVG_Colors 23

18.42.1.126 `ThreadArt_Polyester` #define ThreadArt_Polyester 20

18.42.1.127 `ThreadArt_Rayon` #define ThreadArt_Rayon 19

18.42.1.128 `ThreaDelight_Polyester` #define ThreaDelight_Polyester 21

18.42.1.129 `TRIM` #define TRIM 2 /*! trim + move to (x, y) */

18.42.1.130 `Z102_Isacord_Polyester` #define Z102_Isacord_Polyester 22

18.42.2 Typedef Documentation

18.42.2.1 EmbAlignedDim `typedef struct EmbAlignedDim_ EmbAlignedDim`

18.42.2.2 EmbAngularDim `typedef struct EmbAngularDim_ EmbAngularDim`

18.42.2.3 EmbArc `typedef struct EmbArc_ EmbArc`
absolute position (not relative)

18.42.2.4 EmbArcLengthDim `typedef struct EmbArcLengthDim_ EmbArcLengthDim`

18.42.2.5 EmbArray `typedef struct EmbArray_ EmbArray`
The basic array type.

18.42.2.6 EmbBezier `typedef struct EmbBezier_ EmbBezier`

18.42.2.7 EmbBlock `typedef struct EmbBlock_ EmbBlock`

18.42.2.8 EmbCircle `typedef struct EmbCircle_ EmbCircle`

18.42.2.9 EmbColor `typedef struct EmbColor_ EmbColor`
EmbColor uses the light primaries: red, green, blue in that order.

18.42.2.10 EmbDiameterDim `typedef struct EmbDiameterDim_ EmbDiameterDim`

18.42.2.11 EmbEllipse `typedef struct EmbEllipse_ EmbEllipse`

18.42.2.12 EmbFlag `typedef int EmbFlag`

18.42.2.13 EmbFormatList `typedef struct EmbFormatList_ EmbFormatList`

18.42.2.14 EmbGeometry `typedef struct EmbGeometry_ EmbGeometry`

18.42.2.15 EmbImage `typedef struct EmbImage_ EmbImage`

18.42.2.16 EmbInfiniteLine `typedef struct EmbInfiniteLine_ EmbInfiniteLine`

18.42.2.17 EmbLayer `typedef struct EmbLayer_ EmbLayer`

18.42.2.18 EmbLeaderDim `typedef struct EmbLeaderDim_ EmbLeaderDim`

18.42.2.19 EmbLine `typedef struct EmbLine_ EmbLine`

18.42.2.20 EmbLinearDim `typedef struct EmbLinearDim_ EmbLinearDim`

18.42.2.21 EmbOrdinateDim `typedef struct EmbOrdinateDim_ EmbOrdinateDim`

18.42.2.22 EmbPath `typedef struct EmbPath_ EmbPath`

18.42.2.23 EmbPattern `typedef struct EmbPattern_ EmbPattern`

18.42.2.24 EmbPoint `typedef struct EmbPoint_ EmbPoint`

18.42.2.25 EmbPolygon `typedef EmbPath EmbPolygon`

18.42.2.26 EmbPolyline `typedef EmbPath EmbPolyline`

18.42.2.27 EmbRadiusDim `typedef struct EmbRadiusDim_ EmbRadiusDim`

18.42.2.28 EmbRay `typedef struct EmbRay_ EmbRay`

18.42.2.29 EmbReal `typedef float EmbReal`

18.42.2.30 EmbRect `typedef struct EmbRect_ EmbRect`

18.42.2.31 EmbSatinOutline `typedef struct EmbSatinOutline_ EmbSatinOutline`

18.42.2.32 EmbSpline `typedef struct EmbSpline_ EmbSpline`

18.42.2.33 EmbStitch `typedef struct EmbStitch_ EmbStitch`

18.42.2.34 EmbTextMulti `typedef struct EmbTextMulti_ EmbTextMulti`

18.42.2.35 EmbTextSingle `typedef struct EmbTextSingle_ EmbTextSingle`

18.42.2.36 EmbThread `typedef struct EmbThread_ EmbThread`

18.42.2.37 EmbTime `typedef struct EmbTime_ EmbTime`

18.42.2.38 EmbVector `typedef struct EmbVector_ EmbVector`

The basic type to represent points absolutely or represent directions.
Positive y is up, units are in mm.

18.42.2.39 L_system `typedef struct LSYSTEM L_system`

18.42.2.40 thread_color `typedef struct thread_color_ thread_color`

18.42.3 Function Documentation

18.42.3.1 convert() `EMB_PUBLIC int convert (`

```
    const char * inf,
    const char * outf )
```

18.42.3.2 degrees() `EMB_PUBLIC EmbReal degrees (`

```
    EmbReal radian ) [inline]
```

18.42.3.3 emb_identify_format() `EMB_PUBLIC int emb_identify_format (`

```
    const char * fileName )
```

fileName

Returns

`int`

18.42.3.4 emb_round() `EMB_PUBLIC int emb_round (`

```
    EmbReal x )
```

18.42.3.5 embArc_clockwise() `EMB_PUBLIC char embArc_clockwise (`

```
    EmbArc arc )
```

18.42.3.6 embArc_init() `EMB_PUBLIC EmbArc embArc_init (`

```
    void )
```

18.42.3.7 embArray_addArc() `EMB_PUBLIC int embArray_addArc (`

```
    EmbArray * a,
    EmbArc b )
```

Add an arc *b* to the EmbArray *a* and it returns if the element was successfully added.

18.42.3.8 embArray_addCircle() `EMB_PUBLIC int embArray_addCircle (`

```
    EmbArray * a,
    EmbCircle b )
```

Add a circle *b* to the EmbArray *a* and it returns if the element was successfully added.

18.42.3.9 embArray_addEllipse() `EMB_PUBLIC int embArray_addEllipse (EmbArray * a, EmbEllipse b)`

Add an ellipse *b* to the EmbArray *a* and it returns if the element was successfully added.

18.42.3.10 embArray_addFlag() `EMB_PUBLIC int embArray_addFlag (EmbArray * a, EmbFlag b)`

Add a flag *b* to the EmbArray *a* and it returns if the element was successfully added.

18.42.3.11 embArray_addLine() `EMB_PUBLIC int embArray_addLine (EmbArray * a, EmbLine b)`

Add a line *b* to the EmbArray *a* and it returns if the element was successfully added.

18.42.3.12 embArray_addPath() `EMB_PUBLIC int embArray_addPath (EmbArray * a, EmbPath b)`

Add a path *b* to the EmbArray *a* and it returns if the element was successfully added.

18.42.3.13 embArray_addPoint() `EMB_PUBLIC int embArray_addPoint (EmbArray * a, EmbPoint b)`

Add a point *b* to the EmbArray *a* and it returns if the element was successfully added.

18.42.3.14 embArray_addPolygon() `EMB_PUBLIC int embArray_addPolygon (EmbArray * a, EmbPolygon b)`

Add a polygon *b* to the EmbArray *a* and it returns if the element was successfully added.

18.42.3.15 embArray_addPolyline() `EMB_PUBLIC int embArray_addPolyline (EmbArray * a, EmbPolyline b)`

Add a polyline *b* to the EmbArray *a* and it returns if the element was successfully added.

18.42.3.16 embArray_addRect() `EMB_PUBLIC int embArray_addRect (EmbArray * a, EmbRect b)`

Add a rectangle *b* to the EmbArray *a* and it returns if the element was successfully added.

18.42.3.17 embArray_addStitch() `EMB_PUBLIC int embArray_addStitch (EmbArray * a, EmbStitch b)`

Add a stitch *b* to the EmbArray *a* and it returns if the element was successfully added.

18.42.3.18 embArray_addThread() `EMB_PUBLIC int embArray_addThread (EmbArray * g, EmbThread p)`

18.42.3.19 embArray_addVector() `EMB_PUBLIC int embArray_addVector (EmbArray * a, EmbVector b)`

Add a vector *b* to the EmbArray *a* and it returns if the element was successfully added.

18.42.3.20 embArray_copy() `EMB_PUBLIC void embArray_copy (`
 `EmbArray * dst,`
 `EmbArray * src)`

Copies all entries in the EmbArray struct from *src* to *dst*.

18.42.3.21 embArray_create() `EMB_PUBLIC EmbArray * embArray_create (`
 `int type)`

Allocates memory for an EmbArray of the type determined by the argument *type*.

18.42.3.22 embArray_free() `EMB_PUBLIC void embArray_free (`
 `EmbArray * a)`

Free the memory of EmbArray *a*, recursively if necessary.

18.42.3.23 embArray_resize() `EMB_PUBLIC int embArray_resize (`
 `EmbArray * a)`

Resizes the array *a* to be CHUNK_SIZE entries longer if and only if the amount of room left is less than 3 entries.

18.42.3.24 embCircle_init() `EMB_PUBLIC EmbCircle embCircle_init (`
 `void)`

18.42.3.25 embColor_create() `EMB_PUBLIC EmbColor * embColor_create (`
 `unsigned char r,`
 `unsigned char g,`
 `unsigned char b)`

18.42.3.26 embColor_distance() `EMB_PUBLIC int embColor_distance (`
 `EmbColor a,`
 `EmbColor b)`

a b

Returns

int

18.42.3.27 embColor_fromHexStr() `EMB_PUBLIC EmbColor embColor_fromHexStr (`
 `char * val)`

Converts a 6 digit hex string (I.E. "00FF00") into an EmbColor and returns it.

val/6 byte code describing the color as a hex string, doesn't require null termination.

Returns

EmbColor the same color as our internal type.

18.42.3.28 embColor_make() `EMB_PUBLIC EmbColor embColor_make (`
 `unsigned char r,`
 `unsigned char g,`
 `unsigned char b)`

18.42.3.29 embEllipse_area() `EMB_PUBLIC EmbReal embEllipse_area (`
 `EmbEllipse ellipse)`

18.42.3.30 embEllipse_diameterX() `EMB_PUBLIC EmbReal embEllipse_diameterX (`
`EmbEllipse ellipse)`

18.42.3.31 embEllipse_diameterY() `EMB_PUBLIC EmbReal embEllipse_diameterY (`
`EmbEllipse ellipse)`

18.42.3.32 embEllipse_height() `EMB_PUBLIC EmbReal embEllipse_height (`
`EmbEllipse ellipse)`

18.42.3.33 embEllipse_init() `EMB_PUBLIC EmbEllipse embEllipse_init (`
`void)`

18.42.3.34 embEllipse_make() `EMB_PUBLIC EmbEllipse embEllipse_make (`
`EmbReal cx,`
`EmbReal cy,`
`EmbReal rx,`
`EmbReal ry)`

18.42.3.35 embEllipse_perimeter() `EMB_PUBLIC EmbReal embEllipse_perimeter (`
`EmbEllipse ellipse)`

18.42.3.36 embEllipse_width() `EMB_PUBLIC EmbReal embEllipse_width (`
`EmbEllipse ellipse)`

18.42.3.37 embGeometry_boundingRect() `EMB_PUBLIC EmbRect embGeometry_boundingRect (`
`EmbGeometry * obj)`

Calculate the bounding box of geometry *obj* based on what kind of geometric object it is.

obj A pointer to the geometry memory.

Returns

`EmbRect` The bounding box in the same scale as the input geometry.

In the case of a failure the bounding box returned is always the unit square with top left corner at (0, 0).

18.42.3.38 embGeometry_free() `EMB_PUBLIC void embGeometry_free (`
`EmbGeometry * obj)`

Free the memory occupied by a non-stitch geometry object.

obj Pointer to geometry memory.

18.42.3.39 embGeometry_init() `EMB_PUBLIC EmbGeometry * embGeometry_init (`
`int type_in)`

Our generic object interface backends to each individual type.

type_in

Returns

`EmbGeometry*`

18.42.3.40 embGeometry_move() `EMB_PUBLIC void embGeometry_move (`
 `EmbGeometry * obj,`
 `EmbVector delta)`

Translate *obj* by the vector *delta*.

obj A pointer to the geometry memory. *delta* A vector in the 0.1mm scale to offset the geometry by.

18.42.3.41 embGeometry_vulcanize() `EMB_PUBLIC void embGeometry_vulcanize (`
 `EmbGeometry * obj)`

Toggle the rubber mode of the object.

obj

Todo Review. This could be controlled by a simple flag.

18.42.3.42 embImage_create() `EMB_PUBLIC EmbImage embImage_create (`
 `int ,`
 `int)`

18.42.3.43 embImage_free() `EMB_PUBLIC void embImage_free (`
 `EmbImage * image)`

18.42.3.44 embImage_read() `EMB_PUBLIC void embImage_read (`
 `EmbImage * image,`
 `char * fname)`

18.42.3.45 embImage_write() `EMB_PUBLIC int embImage_write (`
 `EmbImage * image,`
 `char * fname)`

18.42.3.46 embLine_intersectionPoint() `EMB_PUBLIC EmbVector embLine_intersectionPoint (`
 `EmbLine line1,`
 `EmbLine line2)`

18.42.3.47 embLine_make() `EMB_PUBLIC EmbLine embLine_make (`
 `EmbReal x1,`
 `EmbReal y1,`
 `EmbReal x2,`
 `EmbReal y2)`

18.42.3.48 embLine_normalVector() `EMB_PUBLIC void embLine_normalVector (`
 `EmbLine line,`
 `EmbVector * result,`
 `int clockwise)`

Finds the normalized vector perpendicular (clockwise) to the line given by v1->v2 (normal to the line)

18.42.3.49 embPattern_addCircleAbs() `EMB_PUBLIC void embPattern_addCircleAbs (`
 `EmbPattern * p,`
 `EmbCircle circle)`

Adds a circle object to pattern (*p*) with its center at the absolute position (*cx, cy*) with a radius of (*r*). Positive *y* is up.
Units are in millimeters.

```
18.42.3.50 embPattern_addEllipseAbs() EMB_PUBLIC void embPattern_addEllipseAbs (
    EmbPattern * p,
    EmbEllipse ellipse )
```

Adds an ellipse object to pattern (*p*) with its center at the absolute position (*cx,cy*) with radii of (*rx,ry*). Positive y is up. Units are in millimeters.

```
18.42.3.51 embPattern_addLineAbs() EMB_PUBLIC void embPattern_addLineAbs (
    EmbPattern * p,
    EmbLine line )
```

Adds a line object to pattern (*p*) starting at the absolute position (*x1,y1*) and ending at the absolute position (*x2,y2*). Positive y is up. Units are in millimeters.

```
18.42.3.52 embPattern_addPathAbs() EMB_PUBLIC void embPattern_addPathAbs (
    EmbPattern * p,
    EmbPath obj )
```

```
18.42.3.53 embPattern_addPointAbs() EMB_PUBLIC void embPattern_addPointAbs (
    EmbPattern * p,
    EmbPoint obj )
```

Adds a point object to pattern (*p*) at the absolute position (*x,y*). Positive y is up. Units are in millimeters.

```
18.42.3.54 embPattern_addPolygonAbs() EMB_PUBLIC void embPattern_addPolygonAbs (
    EmbPattern * p,
    EmbPolygon obj )
```

```
18.42.3.55 embPattern_addPolylineAbs() EMB_PUBLIC void embPattern_addPolylineAbs (
    EmbPattern * p,
    EmbPolyline obj )
```

```
18.42.3.56 embPattern_addRectAbs() EMB_PUBLIC void embPattern_addRectAbs (
    EmbPattern * p,
    EmbRect rect )
```

Adds a rectangle object to pattern (*p*) at the absolute position (*x,y*) with a width of (*w*) and a height of (*h*). Positive y is up. Units are in millimeters.

```
18.42.3.57 embPattern_addStitchAbs() EMB_PUBLIC void embPattern_addStitchAbs (
    EmbPattern * p,
    EmbReal x,
    EmbReal y,
    int flags,
    int isAutoColorIndex )
```

Adds a stitch to the pattern (*p*) at the absolute position (*x,y*). Positive y is up. Units are in millimeters.

```
18.42.3.58 embPattern_addStitchRel() EMB_PUBLIC void embPattern_addStitchRel (
    EmbPattern * p,
    EmbReal dx,
    EmbReal dy,
    int flags,
    int isAutoColorIndex )
```

Adds a stitch to the pattern (*p*) at the relative position (*dx,dy*) to the previous stitch. Positive y is up. Units are in millimeters.

18.42.3.59 embPattern_addThread() `EMB_PUBLIC int embPattern_addThread (`
 `EmbPattern * pattern,`
 `EmbThread thread)`

pattern *thread*

Returns

`int`

18.42.3.60 embPattern_calcBoundingBox() `EMB_PUBLIC EmbRect embPattern_calcBoundingBox (`
 `EmbPattern * p)`

Returns an `EmbRect` that encapsulates all stitches and objects in the pattern (*p*).

18.42.3.61 embPattern_center() `EMB_PUBLIC void embPattern_center (`
 `EmbPattern * p)`

Center the pattern *p*.

18.42.3.62 embPattern_changeColor() `EMB_PUBLIC void embPattern_changeColor (`
 `EmbPattern * p,`
 `int index)`

Change the `currentColorIndex` of pattern *p* to *index*.

18.42.3.63 embPattern_color_count() `EMB_PUBLIC int embPattern_color_count (`
 `EmbPattern * pattern,`
 `EmbColor startColor)`

18.42.3.64 embPattern_combine() `EMB_PUBLIC EmbPattern * embPattern_combine (`
 `EmbPattern * p1,`
 `EmbPattern * p2)`

p1 *p2*

Returns

`EmbPattern*`

18.42.3.65 embPattern_combineJumpStitches() `EMB_PUBLIC void embPattern_combineJumpStitches (`
 `EmbPattern * p)`

p

18.42.3.66 embPattern_convertGeometry() `EMB_PUBLIC void embPattern_convertGeometry (`
 `EmbPattern * p)`

p

18.42.3.67 embPattern_copyPolylinesToStitchList() `EMB_PUBLIC void embPattern_copyPolylinesTo←`
`StitchList (`
 `EmbPattern * pattern)`

18.42.3.68 embPattern_copyStitchListToPolylines() `EMB_PUBLIC void embPattern_copyStitchListTo←`
`Polylines (`
 `EmbPattern * pattern)`

```
18.42.3.69 embPattern_correctForMaxStitchLength() EMB_PUBLIC void embPattern_correctForMaxStitchLength (
    EmbPattern * p,
    EmbReal maxStitchLength,
    EmbReal maxJumpLength )
```

Todo The params determine the max XY movement rather than the length. They need renamed or clarified further.

```
18.42.3.70 embPattern_create() EMB_PUBLIC EmbPattern * embPattern_create (
    void )
```

Returns a pointer to an EmbPattern. It is created on the heap. The caller is responsible for freeing the allocated memory with [embPattern_free\(\)](#).

Returns

EmbPattern*

```
18.42.3.71 embPattern_crossstitch() EMB_PUBLIC void embPattern_crossstitch (
```

```
    EmbPattern * pattern,
    EmbImage * image,
    int threshold )
```

pattern image threshold

Uses a threshold method to determine where to put crosses in the fill.

To improve this, we can remove the vertical stitches when two crosses neighbour. Currently the simple way to do this is to chain crosses that are neighbours exactly one ahead.

```
18.42.3.72 embPattern_designDetails() EMB_PUBLIC void embPattern_designDetails (
    EmbPattern * p )
```

```
18.42.3.73 embPattern_end() EMB_PUBLIC void embPattern_end (
    EmbPattern * p )
```

```
18.42.3.74 embPattern_fixColorCount() EMB_PUBLIC void embPattern_fixColorCount (
    EmbPattern * p )
```

```
18.42.3.75 embPattern_flip() EMB_PUBLIC void embPattern_flip (
    EmbPattern * p,
    int horz,
    int vert )
```

Flips the entire pattern (*p*) horizontally about the x-axis if (*horz*) is true. Flips the entire pattern (*p*) vertically about the y-axis if (*vert*) is true.

```
18.42.3.76 embPattern_flipHorizontal() EMB_PUBLIC void embPattern_flipHorizontal (
    EmbPattern * p )
```

Flips the entire pattern (*p*) horizontally about the y-axis.

```
18.42.3.77 embPattern_flipVertical() EMB_PUBLIC void embPattern_flipVertical (
    EmbPattern * p )
```

Flips the entire pattern (*p*) vertically about the x-axis.

18.42.3.78 embPattern_free() `EMB_PUBLIC void embPattern_free (``EmbPattern * p)`

Frees all memory allocated in the pattern (*p*).

18.42.3.79 embPattern_hideStitchesOverLength() `EMB_PUBLIC void embPattern_hideStitchesOver←`
`Length (``EmbPattern * p,``int length)`

p length

18.42.3.80 embPattern_horizontal_fill() `EMB_PUBLIC void embPattern_horizontal_fill (``EmbPattern * pattern,``EmbImage * image,``int threshhold)`

pattern image threshhold

Uses a threshhold method to determine where to put lines in the fill.

Needs to pass a "donut test", i.e. an image with black pixels where: $10 < x*x + y*y < 20$ over the area (-30, 30) x (-30, 30).

Use render then image difference to see how well it passes.

18.42.3.81 embPattern_jumpStitches() `EMB_PUBLIC int embPattern_jumpStitches (``EmbPattern * pattern)`**18.42.3.82 embPattern_lengthHistogram()** `EMB_PUBLIC void embPattern_lengthHistogram (``EmbPattern * pattern,``int * bin,``int NUMBINS)`**18.42.3.83 embPattern_loadExternalColorFile()** `EMB_PUBLIC void embPattern_loadExternalColorFile (``EmbPattern * p,``const char * fileName)`

TODO: Description needed.

18.42.3.84 embPattern_maximumStitchLength() `EMB_PUBLIC EmbReal embPattern_maximumStitchLength`
(`EmbPattern * pattern)`**18.42.3.85 embPattern_minimumStitchLength()** `EMB_PUBLIC EmbReal embPattern_minimumStitchLength`
(`EmbPattern * pattern)`**18.42.3.86 embPattern_movePolylinesToStitchList()** `EMB_PUBLIC void embPattern_movePolylinesTo←`
`StitchList (``EmbPattern * pattern)`**18.42.3.87 embPattern_moveStitchListToPolylines()** `EMB_PUBLIC void embPattern_moveStitchListTo←`
`Polylines (``EmbPattern * pattern)`

18.42.3.88 embPattern_read() `EMB_PUBLIC char embPattern_read (`
 `EmbPattern * pattern,`
 `const char * fileName,`
 `int format)`

pattern fileName format

Returns

char

18.42.3.89 embPattern_readAuto() `EMB_PUBLIC char embPattern_readAuto (`
 `EmbPattern * pattern,`
 `const char * fileName)`

pattern fileName

Returns

char

18.42.3.90 embPattern_realStitches() `EMB_PUBLIC int embPattern_realStitches (`
 `EmbPattern * pattern)`

18.42.3.91 embPattern_render() `EMB_PUBLIC int embPattern_render (`
 `EmbPattern * pattern,`
 `char * fname)`

18.42.3.92 embPattern_scale() `EMB_PUBLIC void embPattern_scale (`
 `EmbPattern * p,`
 `EmbReal scale)`

Very simple scaling of the x and y axis for every point. Doesn't insert or delete stitches to preserve density.

18.42.3.93 embPattern_simulate() `EMB_PUBLIC int embPattern_simulate (`
 `EmbPattern * pattern,`
 `char * fname)`

18.42.3.94 embPattern_totalStitchLength() `EMB_PUBLIC EmbReal embPattern_totalStitchLength (`
 `EmbPattern * pattern)`

pattern

Returns

float

18.42.3.95 embPattern_trimStitches() `EMB_PUBLIC int embPattern_trimStitches (`
 `EmbPattern * pattern)`

18.42.3.96 embPattern_write() `EMB_PUBLIC char embPattern_write (`
 `EmbPattern * pattern,`
 `const char * fileName,`
 `int format)`
pattern fileName format

Returns

char

18.42.3.97 embPattern_writeAuto() `EMB_PUBLIC char embPattern_writeAuto (`
 `EmbPattern * pattern,`
 `const char * fileName)`
pattern fileName

Returns

char

18.42.3.98 embRect_area() `EMB_PUBLIC EmbReal embRect_area (`
 `EmbRect rect)`

18.42.3.99 embRect_init() `EMB_PUBLIC EmbRect embRect_init (`
 `void)`

18.42.3.100 embSatinOutline_generateSatinOutline() `EMB_PUBLIC void embSatinOutline_generate←`
SatinOutline (
 `EmbArray * lines,`
 `EmbReal thickness,`
 `EmbSatinOutline * result)`
lines thickness result

18.42.3.101 embSatinOutline_renderStitches() `EMB_PUBLIC EmbArray * embSatinOutline_render←`
Stitches (
 `EmbSatinOutline * result,`
 `EmbReal density)`
result density

Returns

`EmbArray*`

18.42.3.102 embThread_findNearestColor() `EMB_PUBLIC int embThread_findNearestColor (`
 `EmbColor color,`
 `EmbColor * color_list,`
 `int n_colors)`

Returns the closest color to the required color based on a list of available threads. The algorithm is a simple least squares search against the list. If the (square of) Euclidean 3-dimensional distance between the points in (red, green, blue) space is smaller then the index is saved and the remaining index is returned to the caller.

color The EmbColor color to match. *colors* The EmbThreadList pointer to start the search at. *mode* Is the argument an array of threads (0) or colors (1)?

Returns

`closestIndex` The entry in the ThreadList that matches.

18.42.3.103 `embThread_findNearestThread()` `EMB_PUBLIC int embThread_findNearestThread (`

```
    EmbColor color,
    EmbThread * thread_list,
    int n_threads )
```

`color thread_list n_threads`

Returns

`int`

18.42.3.104 `embThread_getRandom()` `EMB_PUBLIC EmbThread embThread_getRandom (`

```
    void )
```

Returns a random thread color, useful in filling in cases where the actual color of the thread doesn't matter but one needs to be declared to test or render a pattern.

Returns

`c` The resulting color.

18.42.3.105 `embTime_initNow()` `EMB_PUBLIC void embTime_initNow (`

```
    EmbTime * t )
```

`t`

18.42.3.106 `embTime_time()` `EMB_PUBLIC EmbTime embTime_time (`

```
    EmbTime * t )
```

`t`

Returns

`EmbTime`

18.42.3.107 `embVector_add()` `EMB_PUBLIC EmbVector embVector_add (`

```
    EmbVector a,
    EmbVector b )
```

The sum of vectors `a` and `b` returned as a vector.

Equivalent to:

$$\mathbf{c} = \mathbf{a} + \mathbf{b} = \begin{pmatrix} a_x + b_x \\ a_y + b_y \end{pmatrix}$$

18.42.3.108 `embVector_angle()` `EMB_PUBLIC EmbReal embVector_angle (`

```
    EmbVector v )
```

The angle, measured anti-clockwise from the x-axis, of a vector `v`.

```
18.42.3.109 embVector_average() EMB_PUBLIC EmbVector embVector_average (
    EmbVector a,
    EmbVector b )
```

The average of vectors $v1$ and $v2$ returned as a vector.

Equivalent to:

$$\mathbf{c} = \frac{\mathbf{a} + \mathbf{b}}{2} = \begin{pmatrix} \frac{a_x+b_x}{2} \\ \frac{a_y+b_y}{2} \end{pmatrix}$$

```
18.42.3.110 embVector_cross() EMB_PUBLIC EmbReal embVector_cross (
    EmbVector a,
    EmbVector b )
```

The "cross product" as vectors a and b returned as a real value.

Technically, this is the magnitude of the cross product when the embroidery is placed in the $z=0$ plane (since the cross product is defined for 3-dimensional vectors). That is:

$$|c| = \left| \begin{pmatrix} a_x \\ a_y \\ 0 \end{pmatrix} \times \begin{pmatrix} b_x \\ b_y \\ 0 \end{pmatrix} \right| = \left| \begin{pmatrix} 0 \\ 0 \\ a_x b_y - a_y b_x \end{pmatrix} \right| = a_x b_y - a_y b_x$$

```
18.42.3.111 embVector_distance() EMB_PUBLIC EmbReal embVector_distance (
    EmbVector a,
    EmbVector b )
```

The distance between a and b returned as a real value.

$$d = |\mathbf{a} - \mathbf{b}| = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$$

```
18.42.3.112 embVector_dot() EMB_PUBLIC EmbReal embVector_dot (
    EmbVector a,
    EmbVector b )
```

The dot product as vectors $v1$ and $v2$ returned as a EmbReal.

Equivalent to:

$$c = \mathbf{a} \cdot \mathbf{b} = a_x b_x + a_y b_y$$

```
18.42.3.113 embVector_length() EMB_PUBLIC EmbReal embVector_length (
    EmbVector vector )
```

The length or absolute value of the vector $vector$.

Equivalent to:

$$|v| = \sqrt{v_x^2 + v_y^2}$$

```
18.42.3.114 embVector_multiply() EMB_PUBLIC void embVector_multiply (
    EmbVector vector,
    EmbReal magnitude,
    EmbVector * result )
```

The scalar multiple $magnitude$ of a vector $vector$. Returned as $result$.

Todo make result return argument.

```
18.42.3.115 embVector_normalize() EMB_PUBLIC void embVector_normalize (
    EmbVector vector,
    EmbVector * result )
```

Finds the unit length vector *result* in the same direction as *vector*.

Equivalent to:

$$\mathbf{u} = \frac{\mathbf{v}}{|\mathbf{v}|}$$

Todo make result return argument.

```
18.42.3.116 embVector_relativeX() EMB_PUBLIC EmbReal embVector_relativeX (
    EmbVector a1,
    EmbVector a2,
    EmbVector a3 )
```

The x-component of the vector

```
18.42.3.117 embVector_relativeY() EMB_PUBLIC EmbReal embVector_relativeY (
    EmbVector a1,
    EmbVector a2,
    EmbVector a3 )
```

The y-component of the vector

```
18.42.3.118 embVector_subtract() EMB_PUBLIC EmbVector embVector_subtract (
    EmbVector v1,
    EmbVector v2 )
```

The difference between vectors *v1* and *v2* returned as *result*.

Equivalent to:

$$\mathbf{c} = \mathbf{a} - \mathbf{b} = \begin{pmatrix} a_x - b_x \\ a_y - b_y \end{pmatrix}$$

```
18.42.3.119 embVector_transpose_product() EMB_PUBLIC void embVector_transpose_product (
    EmbVector v1,
    EmbVector v2,
    EmbVector * result )
```

Since we aren't using full vector algebra here, all vectors are "vertical". so this is like the product $v1^T \cdot v2$ for our vectors *v1* and *v2* so a "component-wise product". The result is stored at the pointer *result*.

That is $(1 \ 0) \cdot (a) = (x \ y) \cdot (0 \ 1) \cdot (b) = (y \ b)$

```
18.42.3.120 embVector_unit() EMB_PUBLIC EmbVector embVector_unit (
    EmbReal alpha )
```

The unit vector in the direction *angle*.

$$\mathbf{a}_\alpha = \begin{pmatrix} \cos(\alpha) \\ \sin(\alpha) \end{pmatrix}$$

```
18.42.3.121 full_test_matrix() EMB_PUBLIC int full_test_matrix (
    char * fname )
```

```
18.42.3.122 getArcCenter() EMB_PUBLIC void getArcCenter (
    EmbArc arc,
    EmbVector * arcCenter )
```

```
18.42.3.123 getArcDataFromBulge() EMB_PUBLIC char getArcDataFromBulge (
    EmbReal bulge,
    EmbArc * arc,
    EmbReal * arcCenterX,
    EmbReal * arcCenterY,
    EmbReal * radius,
    EmbReal * diameter,
    EmbReal * chord,
    EmbReal * chordMidX,
    EmbReal * chordMidY,
    EmbReal * sagitta,
    EmbReal * apothem,
    EmbReal * incAngleInDegrees,
    char * clockwise )
```

```
18.42.3.124 getCircleCircleIntersections() EMB_PUBLIC int getCircleCircleIntersections (
    EmbCircle c0,
    EmbCircle c1,
    EmbVector * v0,
    EmbVector * v1 )
```

```
18.42.3.125 getCircleTangentPoints() EMB_PUBLIC int getCircleTangentPoints (
    EmbCircle c,
    EmbVector p,
    EmbVector * v0,
    EmbVector * v1 )
```

```
18.42.3.126 hilbert_curve() EMB_PUBLIC int hilbert_curve (
    EmbPattern * pattern,
    int iterations )
```

pattern iterations

https://en.wikipedia.org/wiki/Hilbert_curve

Using the Lindenmayer System, so we can save work across different functions.

```
18.42.3.127 lindenmayer_system() EMB_PUBLIC int lindenmayer_system (
    L_system L,
    char * state,
    int iterations,
    int complete )
```

L state iterations complete

Returns

int

This is a slow generation algorithm.

```
18.42.3.128 radians() EMB_PUBLIC EmbReal radians (
    EmbReal degrees ) [inline]
```

```
18.42.3.129 report() EMB_PUBLIC void report (
    int result,
    char * label )
```

```
18.42.3.130 testMain() EMB_PUBLIC void testMain (
    int level )

18.42.3.131 threadColor() EMB_PUBLIC int threadColor (
    const char * name,
    int brand )

18.42.3.132 threadColorName() EMB_PUBLIC const char * threadColorName (
    unsigned int color,
    int brand )

18.42.3.133 threadColorNum() EMB_PUBLIC int threadColorNum (
    unsigned int color,
    int brand )
```

18.42.4 Variable Documentation

18.42.4.1 _dxfColorTable const unsigned char _dxfColorTable[][3] [extern]

18.42.4.2 black_thread EmbThread black_thread [extern]

18.42.4.3 emb_error int emb_error [extern]
Error code storage for optional control flow blocking.

18.42.4.4 emb_verbose int emb_verbose [extern]
Verbosity level.

18.42.4.5 embConstantPi const EmbReal embConstantPi [extern]

18.42.4.6 formatTable EmbFormatList formatTable[numberOfFormats] [extern]

This file is part of libembroidery.

Copyright 2018-2022 The Embroidermodder Team Licensed under the terms of the zlib license.

This file contains all the read and write functions for the library.

Todo This list needs reviewed in case some stitch formats also can contain object data (EMBFORMAT_↔ STCHANDOBJ). *

18.42.4.7 husThreads const EmbThread husThreads[] [extern]

18.42.4.8 jefThreads const EmbThread jefThreads[] [extern]

18.42.4.9 pcmThreads const EmbThread pcmThreads[] [extern]

18.42.4.10 pecThreadCount const int pecThreadCount [extern]

18.42.4.11 pecThreads const EmbThread pecThreads[] [extern]

18.42.4.12 shvThreadCount const int shvThreadCount [extern]

18.42.4.13 shvThreads const EmbThread shvThreads[] [extern]

18.42.4.14 vipDecodingTable const unsigned char vipDecodingTable[] [extern]

18.42.4.15 Embroidery Format (.pcq) The Pfaff vip format is stitch-only.

18.43 embroidery.h

[Go to the documentation of this file.](#)

```
00001 #ifndef LIBEMBROIDERY_HEADER_
00002 #define LIBEMBROIDERY_HEADER_
00003
00004 #ifdef __cplusplus
00005 extern "C" {
00006 #endif
00007
00016 #ifndef LIBEMBROIDERY_EMBEDDED_VERSION
00017 #define LIBEMBROIDERY_EMBEDDED_VERSION 0
00018 #endif
00019
00020 /* MACROS
00021 *****/
00022
00026 #define NORMAL 0
00027 #define JUMP 1
00028 #define TRIM 2
00029 #define STOP 4
00030 #define SEQUIN 8
00031 #define END 16
00036 #define EMB_FORMAT_100 0
00037 #define EMB_FORMAT_100 1
00038 #define EMB_FORMAT_ART 2
00039 #define EMB_FORMAT_BMC 3
00040 #define EMB_FORMAT_BRO 4
00041 #define EMB_FORMAT_CND 5
00042 #define EMB_FORMAT_COL 6
00043 #define EMB_FORMAT_CSD 7
00044 #define EMB_FORMAT_CSV 8
00045 #define EMB_FORMAT_DAT 9
00046 #define EMB_FORMAT_DEM 10
00047 #define EMB_FORMAT_DSB 11
00048 #define EMB_FORMAT_DST 12
00049 #define EMB_FORMAT_DSZ 13
00050 #define EMB_FORMAT_DXF 14
00051 #define EMB_FORMAT_EDR 15
00052 #define EMB_FORMAT_EMD 16
00053 #define EMB_FORMAT_EXP 17
00054 #define EMB_FORMAT_EXY 18
00055 #define EMB_FORMAT_EYS 19
00056 #define EMB_FORMAT_FXY 20
00057 #define EMB_FORMAT_GC 21
00058 #define EMB_FORMAT_GNC 22
00059 #define EMB_FORMAT_GT 23
00060 #define EMB_FORMAT_HUS 24
00061 #define EMB_FORMAT_INB 25
00062 #define EMB_FORMAT_INF 26
00063 #define EMB_FORMAT_JEF 27
00064 #define EMB_FORMAT_KSM 28
00065 #define EMB_FORMAT_MAX 29
00066 #define EMB_FORMAT_MIT 30
00067 #define EMB_FORMAT_NEW 31
00068 #define EMB_FORMAT_OFM 32
00069 #define EMB_FORMAT_PCD 33
00070 #define EMB_FORMAT_PCM 34
00071 #define EMB_FORMAT_PCQ 35
00072 #define EMB_FORMAT_PCS 36
```

```
00073 #define EMB_FORMAT_PEC      37
00074 #define EMB_FORMAT_PEL      38
00075 #define EMB_FORMAT_PEM      39
00076 #define EMB_FORMAT_PES      40
00077 #define EMB_FORMAT_PHB      41
00078 #define EMB_FORMAT_PHC      42
00079 #define EMB_FORMAT_PLT      43
00080 #define EMB_FORMAT_RGB      44
00081 #define EMB_FORMAT_SEW      45
00082 #define EMB_FORMAT_SHV      46
00083 #define EMB_FORMAT_SST      47
00084 #define EMB_FORMAT_STX      48
00085 #define EMB_FORMAT_SVG      49
00086 #define EMB_FORMAT_T01      50
00087 #define EMB_FORMAT_T09      51
00088 #define EMB_FORMAT_TAP      52
00089 #define EMB_FORMAT_THR      53
00090 #define EMB_FORMAT_TXT      54
00091 #define EMB_FORMAT_U00      55
00092 #define EMB_FORMAT_U01      56
00093 #define EMB_FORMAT_VIP      57
00094 #define EMB_FORMAT_VP3      58
00095 #define EMB_FORMAT_XXX      59
00096 #define EMB_FORMAT_ZSK      60
00097
00098 /* Thread color */
00099 #define Arc_Polyester      0
00100 #define Arc_Rayon          1
00101 #define CoatsAndClark_Rayon 2
00102 #define Exquisite_Polyester 3
00103 #define Fufu_Polyester     4
00104 #define Fufu_Rayon          5
00105 #define Hemingworth_Polyester 6
00106 #define Isacord_Polyester   7
00107 #define Isafil_Rayon        8
00108 #define Marathon_Polyester 9
00109 #define Marathon_Rayon      10
00110 #define Madeira_Polyester   11
00111 #define Madeira_Rayon        12
00112 #define Metro_Polyester     13
00113 #define Pantone             14
00114 #define RobisonAnton_Polyester 15
00115 #define RobisonAnton_Rayon   16
00116 #define Sigma_Polyester     17
00117 #define Sulky_Rayon          18
00118 #define ThreadArt_Rayon     19
00119 #define ThreadArt_Polyester 20
00120 #define ThreaDelight_Polyester 21
00121 #define Z102_Isacord_Polyester 22
00122 #define SVG_Colors          23
00123 #define hus_thread          24
00124 #define jef_thread          25
00125 #define pcm_thread          26
00126 #define pec_thread          27
00127 #define shv_thread          28
00128 #define dxf_color           29
00129
00130 #define EMB_ARRAY            0
00131 #define EMB_ARC              1
00132 #define EMB_CIRCLE           2
00133 #define EMB_DIM_DIAMETER     3
00134 #define EMB_DIM_LEADER       4
00135 #define EMB_ELLIPSE          5
00136 #define EMB_FLAG              6
00137 #define EMB_LINE              7
00138 #define EMB_IMAGE             8
00139 #define EMB_PATH              9
00140 #define EMB_POINT             10
00141 #define EMB_POLYGON           11
00142 #define EMB_POLYLINE          12
00143 #define EMB_RECT              13
00144 #define EMB_SPLINE             14
00145 #define EMB_STITCH             15
00146 #define EMB_TEXT_SINGLE       16
00147 #define EMB_TEXT_MULTI         17
00148 #define EMB_VECTOR            18
00149 #define EMB_THREAD             19
00150
00151 #define EMBFORMAT_UNSUPPORTED 0
00152 #define EMBFORMAT_STITCHONLY  1
00153 #define EMBFORMAT_OBJECTONLY  2
00154 #define EMBFORMAT_STCHANDOBJ 3 /* binary operation: 1+2=3 */
00155
00156 #define numberFormats        61
00157
00158 #define CHUNK_SIZE            128
00159
```

```

00160 #define EMB_MAX_LAYERS          10
00161 #define MAX_THREADS           256
00162 #define EMBFORMAT_MAXEXT        3
00163 /* maximum length of extension without dot */
00164 #define EMBFORMAT_MAXDESC       50
00165 /* the longest possible description string length */
00166 #define MAX_STITCHES         1000000
00167
00168
00169
00170 #if defined(_WIN32) && !defined(WIN32)
00171 #define WIN32
00172 #endif
00173
00174 /* When building a shared library,
00175 * use the proper export keyword depending on the compiler */
00176 #define EMB_PUBLIC
00177 #if defined(LIBEMBROIDERY_SHARED)
00178     #undef EMB_PUBLIC
00179     #if defined(__WIN32__) || defined(WIN32)
00180         #define EMB_PUBLIC __declspec(dllexport)
00181     #else
00182         #define EMB_PUBLIC __attribute__ ((visibility("default")))
00183     #endif
00184 #endif
00185
00186 /* TYPEDEFS AND STRUCTS
00187 *****/
00188
00189 typedef float EmbReal;
00190
00194 typedef struct EmbColor_
00195 {
00196     unsigned char r;
00197     unsigned char g;
00198     unsigned char b;
00199 } EmbColor;
00200
00206 typedef struct EmbVector_
00207 {
00208     EmbReal x;
00209     EmbReal y;
00210 } EmbVector;
00211
00215 typedef struct EmbArray_ EmbArray;
00216
00217 typedef struct EmbImage_ {
00218     EmbVector position;
00219     EmbVector dimensions;
00220     unsigned char* data;
00221     int width;
00222     int height;
00223     char path[200];
00224     char name[200];
00225 } EmbImage;
00226
00231 typedef struct EmbBlock_ {
00232     EmbVector position;
00233 } EmbBlock;
00234
00239 typedef struct EmbAlignedDim_ {
00240     EmbVector position;
00241 } EmbAlignedDim;
00242
00247 typedef struct EmbAngularDim_ {
00248     EmbVector position;
00249 } EmbAngularDim;
00250
00255 typedef struct EmbArcLengthDim_ {
00256     EmbVector position;
00257 } EmbArcLengthDim;
00258
00263 typedef struct EmbDiameterDim_ {
00264     EmbVector position;
00265 } EmbDiameterDim;
00266
00271 typedef struct EmbLeaderDim_ {
00272     EmbVector position;
00273 } EmbLeaderDim;
00274
00279 typedef struct EmbLinearDim_ {
00280     EmbVector position;
00281 } EmbLinearDim;
00282
00287 typedef struct EmbOrdinateDim_ {
00288     EmbVector position;
00289 } EmbOrdinateDim;

```

```
00290
00295 typedef struct EmbRadiusDim_ {
00296     EmbVector position;
00297 } EmbRadiusDim;
00298
00299 typedef struct EmbInfiniteLine_ {
00300     EmbVector position;
00301 } EmbInfiniteLine;
00302
00303 typedef struct EmbRay_ {
00304     EmbVector position;
00305 } EmbRay;
00306
00307 typedef struct EmbTextMulti_ {
00308     EmbVector position;
00309     char text[200];
00310 } EmbTextMulti;
00311
00312 typedef struct EmbTextSingle_ {
00313     EmbVector position;
00314     char text[200];
00315 } EmbTextSingle;
00316
00317 typedef struct EmbTime_ {
00318 {
00319     unsigned int year;
00320     unsigned int month;
00321     unsigned int day;
00322     unsigned int hour;
00323     unsigned int minute;
00324     unsigned int second;
00325 } EmbTime;
00326
00327 typedef struct EmbPoint_ {
00328 {
00329     EmbVector position;
00330     int lineType;
00331     EmbColor color;
00332 } EmbPoint;
00333
00334 typedef struct EmbLine_ {
00335 {
00336     EmbVector start;
00337     EmbVector end;
00338     int lineType;
00339     EmbColor color;
00340 } EmbLine;
00341
00342 typedef struct EmbPath_ {
00343 {
00344     EmbArray* pointList;
00345     EmbArray* flagList;
00346     int lineType;
00347     EmbColor color;
00348 } EmbPath;
00349
00350 typedef struct EmbStitch_ {
00351 {
00352     int flags;
00353     EmbReal x;
00354     EmbReal y;
00355     int color;
00356 } EmbStitch;
00357
00358 typedef struct EmbThread_ {
00359 {
00360     EmbColor color;
00361     char description[50];
00362     char catalogNumber[30];
00363 } EmbThread;
00364
00365 typedef struct thread_color_ {
00366 {
00367     char name[22];
00368     unsigned int hex_code;
00369     int manufacturer_code;
00370 } thread_color;
00371
00372 typedef struct EmbArc_ {
00373 {
00374     EmbVector start;
00375     EmbVector mid;
00376     EmbVector end;
00377 } EmbArc;
00378
00379 typedef struct EmbRect_ {
00380 {
00381     EmbReal top;
```

```

00434     EmbReal left;
00435     EmbReal bottom;
00436     EmbReal right;
00437     EmbReal rotation;
00438     EmbReal radius;
00439 } EmbRect;
00440
00445 typedef struct EmbCircle_
00446 {
00447     EmbVector center;
00448     EmbReal radius;
00449 } EmbCircle;
00450
00455 typedef EmbPath EmbPolygon;
00456
00461 typedef EmbPath EmbPolyline;
00462
00467 typedef int EmbFlag;
00468
00473 typedef struct EmbSatinOutline_
00474 {
00475     int length;
00476     EmbArray* side1;
00477     EmbArray* side2;
00478 } EmbSatinOutline;
00479
00484 typedef struct EmbEllipse_
00485 {
00486     EmbVector center;
00487     EmbVector radius;
00488     EmbReal rotation;
00489 } EmbEllipse;
00490
00495 typedef struct EmbBezier_ {
00496     EmbVector start;
00497     EmbVector control1;
00498     EmbVector control2;
00499     EmbVector end;
00500 } EmbBezier;
00501
00506 typedef struct EmbSpline_ {
00507     EmbArray *beziers;
00508 } EmbSpline;
00509
00514 typedef struct LSYSTEM {
00515     char axiom;
00516     char *alphabet;
00517     char *constants;
00518     char **rules;
00519 } L_system;
00520
00525 typedef struct EmbGeometry_ {
00526     union {
00527         EmbArc arc;
00528         EmbCircle circle;
00529         EmbColor color;
00530         EmbEllipse ellipse;
00531         EmbLine line;
00532         EmbPath path;
00533         EmbPoint point;
00534         EmbPolygon polygon;
00535         EmbPolyline polyline;
00536         EmbRect rect;
00537         EmbSpline spline;
00538         EmbVector vector;
00539     } object;
00540     EmbStitch stitch;
00541     EmbThread thread;
00542     int flag;
00543     int type;
00544     int lineType;
00545     EmbColor color;
00546 } EmbGeometry;
00547
00552 struct EmbArray_ {
00553     EmbGeometry *geometry;
00554     EmbStitch *stitch;
00555     EmbThread *thread;
00556     int count;
00557     int length;
00558     int type;
00559 };
00560
00565 typedef struct EmbLayer_
00566 {
00567     char name[100];
00568     EmbArray *geometry;

```

```
00569 } EmbLayer;
00570
00575 typedef struct EmbPattern_
00576 {
00577     unsigned int dstJumpsPerTrim;
00578     EmbVector home;
00579     EmbReal hoop_width;
00580     EmbReal hoop_height;
00581     EmbArray *thread_list;
00582     EmbArray *stitch_list;
00583     EmbArray *geometry;
00584     EmbLayer layer[EMB_MAX_LAYERS];
00585     int currentColorIndex;
00586 } EmbPattern;
00587
00592 typedef struct EmbFormatList_
00593 {
00594     char extension[2 + EMBFORMAT_MAXEXT];
00595     char description[EMBFORMAT_MAXDESC];
00596     char reader_state;
00597     char writer_state;
00598     int type;
00599     int color_only;
00600     int check_for_color_file;
00601     int write_external_color_file;
00602 } EmbFormatList;
00603
00604 /* Function Declarations
00605 ****
00606 EMB_PUBLIC int lindenmayer_system(L_system L, char* state, int iteration, int complete);
00607 EMB_PUBLIC int hilbert_curve(EmbPattern *pattern, int iterations);
00608
00609 EMB_PUBLIC int emb_identify_format(const char *ending);
00610 EMB_PUBLIC void testMain(int level);
00611 EMB_PUBLIC int convert(const char *inf, const char *outf);
00612
00613 EMB_PUBLIC EmbColor embColor_make(unsigned char r, unsigned char g, unsigned char b);
00614 EMB_PUBLIC EmbColor* embColor_create(unsigned char r, unsigned char g, unsigned char b);
00615 EMB_PUBLIC EmbColor embColor_fromHexStr(char* val);
00616 EMB_PUBLIC int embColor_distance(EmbColor a, EmbColor b);
00617
00618 EMB_PUBLIC EmbArray* embArray_create(int type);
00619 EMB_PUBLIC int embArray_resize(EmbArray *g);
00620 EMB_PUBLIC void embArray_copy(EmbArray *dst, EmbArray *src);
00621 EMB_PUBLIC int embArray_addArc(EmbArray* g, EmbArc arc);
00622 EMB_PUBLIC int embArray_addCircle(EmbArray* g, EmbCircle circle);
00623 EMB_PUBLIC int embArray_addEllipse(EmbArray* g, EmbEllipse ellipse);
00624 EMB_PUBLIC int embArray_addFlag(EmbArray* g, int flag);
00625 EMB_PUBLIC int embArray_addLine(EmbArray* g, EmbLine line);
00626 EMB_PUBLIC int embArray_addRect(EmbArray* g, EmbRect rect);
00627 EMB_PUBLIC int embArray_addPath(EmbArray* g, EmbPath p);
00628 EMB_PUBLIC int embArray_addPoint(EmbArray* g, EmbPoint p);
00629 EMB_PUBLIC int embArray_addPolygon(EmbArray* g, EmbPolygon p);
00630 EMB_PUBLIC int embArray_addPolyline(EmbArray* g, EmbPolyline p);
00631 /* EMB_PUBLIC int embArray_addSpline(EmbArray* g, EmbSpline p); */
00632 EMB_PUBLIC int embArray_addStitch(EmbArray* g, EmbStitch st);
00633 EMB_PUBLIC int embArray_addThread(EmbArray* g, EmbThread p);
00634 EMB_PUBLIC int embArray_addVector(EmbArray* g, EmbVector);
00635 EMB_PUBLIC void embArray_free(EmbArray* p);
00636
00637 EMB_PUBLIC EmbLine embLine_make(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2);
00638
00639 EMB_PUBLIC void embLine_normalVector(EmbLine line, EmbVector* result, int clockwise);
00640 EMB_PUBLIC EmbVector embLine_intersectionPoint(EmbLine line1, EmbLine line2);
00641
00642 EMB_PUBLIC int embThread_findNearestColor(EmbColor color, EmbColor* colors, int n_colors);
00643 EMB_PUBLIC int embThread_findNearestThread(EmbColor color, EmbThread* threads, int n_threads);
00644 EMB_PUBLIC EmbThread embThread_getRandom(void);
00645
00646 EMB_PUBLIC void embVector_normalize(EmbVector vector, EmbVector* result);
00647 EMB_PUBLIC void embVector_multiply(EmbVector vector, EmbReal magnitude, EmbVector* result);
00648 EMB_PUBLIC EmbVector embVector_add(EmbVector v1, EmbVector v2);
00649 EMB_PUBLIC EmbVector embVector_average(EmbVector v1, EmbVector v2);
00650 EMB_PUBLIC EmbVector embVector_subtract(EmbVector v1, EmbVector v2);
00651 EMB_PUBLIC EmbReal embVector_dot(EmbVector v1, EmbVector v2);
00652 EMB_PUBLIC EmbReal embVector_cross(EmbVector v1, EmbVector v2);
00653 EMB_PUBLIC void embVector_transpose_product(EmbVector v1, EmbVector v2, EmbVector* result);
00654 EMB_PUBLIC EmbReal embVector_length(EmbVector vector);
00655 EMB_PUBLIC EmbReal embVector_relativeX(EmbVector a1, EmbVector a2, EmbVector a3);
00656 EMB_PUBLIC EmbReal embVector_relativeY(EmbVector a1, EmbVector a2, EmbVector a3);
00657 EMB_PUBLIC EmbReal embVector_angle(EmbVector v);
00658 EMB_PUBLIC EmbReal embVector_distance(EmbVector a, EmbVector b);
00659 EMB_PUBLIC EmbVector embVector_unit(EmbReal angle);
00660
00661 EMB_PUBLIC EmbArc embArc_init(void);
00662 EMB_PUBLIC char embArc_clockwise(EmbArc arc);
00663
```

```

00664 EMB_PUBLIC void getArcCenter(EmbArc arc, EmbVector *arcCenter);
00665 EMB_PUBLIC char getArcDataFromBulge(EmbReal bulge,
00666             EmbArc *arc,
00667             EmbReal* arcCenterX,           EmbReal* arcCenterY,
00668             EmbReal* radius,             EmbReal* diameter,
00669             EmbReal* chord,
00670             EmbReal* chordMidX,         EmbReal* chordMidY,
00671             EmbReal* sagitta,          EmbReal* apothem,
00672             EmbReal* incAngleInDegrees, char* clockwise);
00673
00674 EMB_PUBLIC EmbCircle embCircle_init(void);
00675 EMB_PUBLIC int getCircleCircleIntersections(
00676     EmbCircle c0, EmbCircle c1, EmbVector *v0, EmbVector *v1);
00677 EMB_PUBLIC int getCircleTangentPoints(
00678     EmbCircle c, EmbVector p, EmbVector *v0, EmbVector *v1);
00679
00680 EMB_PUBLIC EmbEllipse embEllipse_init(void);
00681 EMB_PUBLIC EmbEllipse embEllipse_make(EmbReal cx, EmbReal cy, EmbReal rx, EmbReal ry);
00682 EMB_PUBLIC EmbReal embEllipse_diameterX(EmbEllipse ellipse);
00683 EMB_PUBLIC EmbReal embEllipse_diameterY(EmbEllipse ellipse);
00684 EMB_PUBLIC EmbReal embEllipse_width(EmbEllipse ellipse);
00685 EMB_PUBLIC EmbReal embEllipse_height(EmbEllipse ellipse);
00686 EMB_PUBLIC EmbReal embEllipse_area(EmbEllipse ellipse);
00687 EMB_PUBLIC EmbReal embEllipse_perimeter(EmbEllipse ellipse);
00688
00689 EMB_PUBLIC EmbImage embImage_create(int, int);
00690 EMB_PUBLIC void embImage_read(EmbImage *image, char *fname);
00691 EMB_PUBLIC int embImage_write(EmbImage *image, char *fname);
00692 EMB_PUBLIC void embImage_free(EmbImage *image);
00693
00694 EMB_PUBLIC EmbRect embRect_init(void);
00695 EMB_PUBLIC EmbReal embRect_area(EmbRect);
00696
00697 EMB_PUBLIC int threadColor(const char*, int brand);
00698 EMB_PUBLIC int threadColorNum(unsigned int color, int brand);
00699 EMB_PUBLIC const char* threadColorName(unsigned int color, int brand);
00700
00701 EMB_PUBLIC void embTime_initNow(EmbTime* t);
00702 EMB_PUBLIC EmbTime embTime_time(EmbTime* t);
00703
00704 EMB_PUBLIC void embSatinOutline_generateSatinOutline(EmbArray* lines, EmbReal thickness,
00705             EmbSatinOutline* result);
00706 EMB_PUBLIC EmbArray* embSatinOutline_renderStitches(EmbSatinOutline* result, EmbReal density);
00707
00708 EMB_PUBLIC EmbGeometry *embGeometry_init(int type_in);
00709 EMB_PUBLIC void embGeometry_free(EmbGeometry *obj);
00710 EMB_PUBLIC void embGeometry_move(EmbGeometry *obj, EmbVector delta);
00711 EMB_PUBLIC void embGeometry_boundingRect(EmbGeometry *obj);
00712 EMB_PUBLIC void embGeometry_vulcanize(EmbGeometry *obj);
00713
00714 EMB_PUBLIC EmbPattern* embPattern_create(void);
00715 EMB_PUBLIC void embPattern_hideStitchesOverLength(EmbPattern* p, int length);
00716 EMB_PUBLIC void embPattern_fixColorCount(EmbPattern* p);
00717 EMB_PUBLIC int embPattern_addThread(EmbPattern* p, EmbThread thread);
00718 EMB_PUBLIC void embPattern_addStitchAbs(EmbPattern* p, EmbReal x, EmbReal y, int flags, int
00719             isAutoColorIndex);
00720 EMB_PUBLIC void embPattern_addStitchRel(EmbPattern* p, EmbReal dx, EmbReal dy, int flags, int
00721             isAutoColorIndex);
00722 EMB_PUBLIC void embPattern_changeColor(EmbPattern* p, int index);
00723 EMB_PUBLIC void embPattern_free(EmbPattern* p);
00724 EMB_PUBLIC void embPattern_scale(EmbPattern* p, EmbReal scale);
00725 EMB_PUBLIC EmbReal embPattern_totalStitchLength(EmbPattern *pattern);
00726 EMB_PUBLIC EmbReal embPattern_minimumStitchLength(EmbPattern *pattern);
00727 EMB_PUBLIC EmbReal embPattern_maximumStitchLength(EmbPattern *pattern);
00728 EMB_PUBLIC void embPattern_lengthHistogram(EmbPattern *pattern, int *bin, int NUMBINS);
00729 EMB_PUBLIC int embPattern_realStitches(EmbPattern *pattern);
00730 EMB_PUBLIC int embPattern_jumpStitches(EmbPattern *pattern);
00731 EMB_PUBLIC int embPattern_trimStitches(EmbPattern *pattern);
00732 EMB_PUBLIC void embPattern_calcBoundingBox(EmbPattern* p);
00733 EMB_PUBLIC void embPattern_flipHorizontal(EmbPattern* p);
00734 EMB_PUBLIC void embPattern_flipVertical(EmbPattern* p);
00735 EMB_PUBLIC void embPattern_center(EmbPattern* p);
00736 EMB_PUBLIC void embPattern_loadExternalColorFile(EmbPattern* p, const char* fileName);
00737 EMB_PUBLIC void embPattern_convertGeometry(EmbPattern* p);
00738 EMB_PUBLIC void embPattern_designDetails(EmbPattern *p);
00739 EMB_PUBLIC EmbPattern *embPattern_combine(EmbPattern *p1, EmbPattern *p2);
00740 EMB_PUBLIC int embPattern_color_count(EmbPattern *pattern, EmbColor startColor);
00741 EMB_PUBLIC void embPattern_end(EmbPattern* p);
00742 EMB_PUBLIC void embPattern_crossstitch(EmbPattern *pattern, EmbImage *, int threshold);
00743 EMB_PUBLIC void embPattern_horizontal_fill(EmbPattern *pattern, EmbImage *, int threshold);
00744 EMB_PUBLIC void embPattern_render(EmbPattern *pattern, char *fname);
00745 EMB_PUBLIC void embPattern_simulate(EmbPattern *pattern, char *fname);
00746

```

```

00747 EMB_PUBLIC void embPattern_addCircleAbs(EmbPattern* p, EmbCircle obj);
00748 EMB_PUBLIC void embPattern_addEllipseAbs(EmbPattern* p, EmbEllipse obj);
00749 EMB_PUBLIC void embPattern_addLineAbs(EmbPattern* p, EmbLine obj);
00750 EMB_PUBLIC void embPattern_addPathAbs(EmbPattern* p, EmbPath obj);
00751 EMB_PUBLIC void embPattern_addPointAbs(EmbPattern* p, EmbPoint obj);
00752 EMB_PUBLIC void embPattern_addPolygonAbs(EmbPattern* p, EmbPolygon obj);
00753 EMB_PUBLIC void embPattern_addPolylineAbs(EmbPattern* p, EmbPolyline obj);
00754 EMB_PUBLIC void embPattern_addRectAbs(EmbPattern* p, EmbRect obj);
00755
00756 EMB_PUBLIC void embPattern_copyStitchListToPolylines(EmbPattern* pattern);
00757 EMB_PUBLIC void embPattern_copyPolylinesToStitchList(EmbPattern* pattern);
00758 EMB_PUBLIC void embPattern_moveStitchListToPolylines(EmbPattern* pattern);
00759 EMB_PUBLIC void embPattern_movePolylinesToStitchList(EmbPattern* pattern);
00760
00761 EMB_PUBLIC char embPattern_read(EmbPattern *pattern, const char* fileName, int format);
00762 EMB_PUBLIC char embPattern_write(EmbPattern *pattern, const char* fileName, int format);
00763
00764 EMB_PUBLIC char embPattern_readAuto(EmbPattern *pattern, const char* fileName);
00765 EMB_PUBLIC char embPattern_writeAuto(EmbPattern *pattern, const char* fileName);
00766
00767 EMB_PUBLIC void report(int result, char *label);
00768 EMB_PUBLIC int full_test_matrix(char *fname);
00769
00770 EMB_PUBLIC int emb_round(EmbReal x);
00771 EMB_PUBLIC EmbReal radians(EmbReal degree);
00772 EMB_PUBLIC EmbReal degrees(EmbReal radian);
00773
00774 /* NON-MACRO CONSTANTS
00775 ****
00776
00777 extern EmbFormatList formatTable[numberOfFormats];
00778 extern const int pecThreadCount;
00779 extern const int shvThreadCount;
00780 extern const EmbReal embConstantPi;
00781 extern const EmbThread husThreads[];
00782 extern const EmbThread jefThreads[];
00783 extern const EmbThread shvThreads[];
00784 extern const EmbThread pcmThreads[];
00785 extern const EmbThread pecThreads[];
00786 extern const unsigned char _dxfColorTable[][][3];
00787 extern EmbThread black_thread;
00788 extern const unsigned char vipDecodingTable[];
00789
00790 /* VARIABLES
00791 ****
00792
00793 extern int emb_error;
00794
00795 extern int emb_verbose;
00796
00797 #ifdef __cplusplus
00798 }
00799 #endif /* __cplusplus */
00800
00801 #endif /* LIBEMBROIDERY_HEADER */
00802
00803
00804
00805
00806
00807
00808

```

18.44 extern/libembroidery/src/embroidery_internal.h File Reference

```
#include "embroidery.h"
#include <stdio.h>
```

Classes

- struct [_bcf_file_difat](#)
- struct [_bcf_file_fat](#)
- struct [_bcf_directory_entry](#)
- struct [_bcf_directory](#)
- struct [_bcf_file_header](#)
- struct [_bcf_file](#)
- struct [_vp3Hoop](#)
- struct [ThredHeader_](#)
- struct [ThredExtension_](#)
- struct [SubDescriptor_](#)
- struct [StxThread_](#)

- struct `VipHeader_`
- struct `SvgAttribute_`
- struct `Huffman`
- struct `Compress`

Macros

- `#define CompoundFileSector_MaxRegSector 0xFFFFFFFFFA`
- `#define CompoundFileSector_DIFAT_Sector 0xFFFFFFFFFC`
- `#define CompoundFileSector_FAT_Sector 0xFFFFFFFFFD`
- `#define CompoundFileSector_EndOfChain 0xFFFFFFFFFE`
- `#define CompoundFileSector_FreeSector 0xFFFFFFFFFF`
- `#define ObjectTypeUnknown 0x00`
- `#define ObjectTypeStorage 0x01`
- `#define ObjectTypeStream 0x02`
- `#define ObjectTypeRootEntry 0x05`
- `#define CompoundFileStreamId_MaxRegularStreamId 0xFFFFFFFFFA`
- `#define CompoundFileStreamId_NoStream 0xFFFFFFFFFF`
- `#define ELEMENT_XML 0`
- `#define ELEMENT_A 1`
- `#define ELEMENT_ANIMATE 2`
- `#define ELEMENT_ANIMATECOLOR 3`
- `#define ELEMENT_ANIMATEMOTION 4`
- `#define ELEMENT_ANIMATETRANSFORM 5`
- `#define ELEMENT_ANIMATION 6`
- `#define ELEMENT_AUDIO 7`
- `#define ELEMENT_CIRCLE 8`
- `#define ELEMENT_DEFS 9`
- `#define ELEMENT_DESC 10`
- `#define ELEMENT_DISCARD 11`
- `#define ELEMENT_ELLIPSE 12`
- `#define ELEMENT_FONT 13`
- `#define ELEMENT_FONT_FACE 14`
- `#define ELEMENT_FONT_FACE_SRC 15`
- `#define ELEMENT_FONT_FACE_URI 16`
- `#define ELEMENT_FOREIGN_OBJECT 17`
- `#define ELEMENT_G 18`
- `#define ELEMENT_GLYPH 19`
- `#define ELEMENT_HANDLER 20`
- `#define ELEMENT_HKERN 21`
- `#define ELEMENT_IMAGE 22`
- `#define ELEMENT_LINE 23`
- `#define ELEMENT_LINEAR_GRADIENT 24`
- `#define ELEMENT_LISTENER 25`
- `#define ELEMENT_METADATA 26`
- `#define ELEMENT_MISSING_GLYPH 27`
- `#define ELEMENT_MPATH 28`
- `#define ELEMENT_PATH 29`
- `#define ELEMENT_POLYGON 30`
- `#define ELEMENT_POLYLINE 31`
- `#define ELEMENT_PREFETCH 32`
- `#define ELEMENT_RADIAL_GRADIENT 33`
- `#define ELEMENT_RECT 34`
- `#define ELEMENT_SCRIPT 35`
- `#define ELEMENT_SET 36`

- #define ELEMENT_SOLID_COLOR 37
- #define ELEMENT_STOP 38
- #define ELEMENT_SVG 39
- #define ELEMENT_SWITCH 40
- #define ELEMENT_TBREAK 41
- #define ELEMENT_TEXT 42
- #define ELEMENT_TEXT_AREA 43
- #define ELEMENT_TITLE 44
- #define ELEMENT_TSPAN 45
- #define ELEMENT_USE 46
- #define ELEMENT_VIDEO 47
- #define RED_TERM_COLOR "\x1B[0;31m"
- #define GREEN_TERM_COLOR "\x1B[0;32m"
- #define YELLOW_TERM_COLOR "\x1B[1;33m"
- #define RESET_TERM_COLOR "\033[0m"
- #define HOOP_126X110 0
- #define HOOP_110X110 1
- #define HOOP_50X50 2
- #define HOOP_140X200 3
- #define HOOP_230X200 4
- #define EMB_MIN(A, B) (((A) < (B)) ? (A) : (B))
- #define EMB_MAX(A, B) (((A) > (B)) ? (A) : (B))
- #define EMB_BIG_ENDIAN 0
- #define EMB_LITTLE_ENDIAN 1
- #define ENDIAN_HOST EMB_LITTLE_ENDIAN
- #define EMB_INT16_BIG 2
- #define EMB_INT16_LITTLE 3
- #define EMB_INT32_BIG 4
- #define EMB_INT32_LITTLE 5
- #define PES0001 0
- #define PES0020 1
- #define PES0022 2
- #define PES0030 3
- #define PES0040 4
- #define PES0050 5
- #define PES0055 6
- #define PES0056 7
- #define PES0060 8
- #define PES0070 9
- #define PES0080 10
- #define PES0090 11
- #define PES0100 12
- #define N_PES VERSIONS 13
- #define DXF_VERSION_R10 "AC1006"
- #define DXF_VERSION_R11 "AC1009"
- #define DXF_VERSION_R12 "AC1009"
- #define DXF_VERSION_R13 "AC1012"
- #define DXF_VERSION_R14 "AC1014"
- #define DXF_VERSION_R15 "AC1015"
- #define DXF_VERSION_R18 "AC1018"
- #define DXF_VERSION_R21 "AC1021"
- #define DXF_VERSION_R24 "AC1024"
- #define DXF_VERSION_R27 "AC1027"
- #define DXF_VERSION_2000 "AC1015"
- #define DXF_VERSION_2002 "AC1015"

- #define DXF_VERSION_2004 "AC1018"
- #define DXF_VERSION_2006 "AC1018"
- #define DXF_VERSION_2007 "AC1021"
- #define DXF_VERSION_2009 "AC1021"
- #define DXF_VERSION_2010 "AC1024"
- #define DXF_VERSION_2013 "AC1027"
- #define SVG_CREATOR_NULL 0
- #define SVG_CREATOR_EMBROIDERMODDER 1
- #define SVG_CREATOR_ILLUSTRATOR 2
- #define SVG_CREATOR_INKSCAPE 3
- #define SVG_EXPECT_NULL 0
- #define SVG_EXPECT_ELEMENT 1
- #define SVG_EXPECT_ATTRIBUTE 2
- #define SVG_EXPECT_VALUE 3
- #define SVG_NULL 0
- #define SVG_ELEMENT 1
- #define SVG_PROPERTY 2
- #define SVG_MEDIA_PROPERTY 3
- #define SVG_ATTRIBUTE 4
- #define SVG_CATCH_ALL 5
- #define LINETO 0
- #define MOVETO 1
- #define BULGETOCONTROL 2
- #define BULGETOEND 4
- #define ELLIPSETORAD 8
- #define ELLIPSETOEND 16
- #define CUBICTOCONTROL1 32
- #define CUBICTOCONTROL2 64
- #define CUBICTOEND 128
- #define QUADTOCONTROL 256
- #define QUADTOEND 512

Typedefs

- typedef struct _bcf_file_difat bcf_file_difat
- typedef struct _bcf_file_fat bcf_file_fat
- typedef struct _bcf_directory_entry bcf_directory_entry
- typedef struct _bcf_directory bcf_directory
- typedef struct _bcf_file_header bcf_file_header
- typedef struct _bcf_file bcf_file
- typedef struct _vp3Hoop vp3Hoop
- typedef struct ThredHeader_ ThredHeader
- typedef struct ThredExtension_ ThredExtension
- typedef struct SubDescriptor_ SubDescriptor
- typedef struct StxThread_ StxThread
- typedef struct VipHeader_ VipHeader
- typedef struct SvgAttribute_ SvgAttribute
- typedef struct Huffman huffman
- typedef struct Compress compress

Enumerations

- enum CSV_EXPECT { CSV_EXPECT_NULL , CSV_EXPECT_QUOTE1 , CSV_EXPECT_QUOTE2 , CSV_EXPECT_COMMA }
- enum CSV_MODE { CSV_MODE_NULL , CSV_MODE_COMMENT , CSV_MODE_VARIABLE , CSV_MODE_THREAD , CSV_MODE_STITCH }

Functions

- void `huffman_build_table` (`huffman` **h*)
- int * `huffman_table_lookup` (`huffman` **h*, int `byte_lookup`, int *`lengths`)
- int `compress_get_bits` (`compress` **c*, int `length`)
- int `compress_pop` (`compress` **c*, int `bit_count`)
- int `compress_read_variable_length` (`compress` **c*)
- void `compress_load_character_length_huffman` (`compress` **c*)
- void `compress_load_character_huffman` (`compress` **c*)
- void `compress_load_distance_huffman` (`compress` **c*)
- void `compress_load_block` (`compress` **c*)
- int `compress_get_token` (`compress` **c*)
- int `compress_get_position` (`compress` **c*)
- void `readPecStitches` (`EmbPattern` **pattern*, FILE **file*)
- void `writePecStitches` (`EmbPattern` **pattern*, FILE **file*, const char **filename*)
- int `decodeNewStitch` (unsigned char *value*)
 - value*
- void `pfaffEncode` (FILE **file*, int *x*, int *y*, int *flags*)
- `EmbReal pfaffDecode` (unsigned char *a1*, unsigned char *a2*, unsigned char *a3*)
- unsigned char `mitEncodeStitch` (`EmbReal` *value*)
 - value*
- int `mitDecodeStitch` (unsigned char *value*)
 - value*
- int `encode_tajima_ternary` (unsigned char *b*[3], int *x*, int *y*)
- void `decode_tajima_ternary` (unsigned char *b*[3], int **x*, int **y*)
- void `encode_t01_record` (unsigned char *b*[3], int *x*, int *y*, int *flags*)
- int `decode_t01_record` (unsigned char *b*[3], int **x*, int **y*, int **flags*)
- void `readPESHeaderV5` (FILE **file*, `EmbPattern` **pattern*)
- void `readPESHeaderV6` (FILE **file*, `EmbPattern` **pattern*)
- void `readPESHeaderV7` (FILE **file*, `EmbPattern` **pattern*)
- void `readPESHeaderV8` (FILE **file*, `EmbPattern` **pattern*)
- void `readPESHeaderV9` (FILE **file*, `EmbPattern` **pattern*)
- void `readPESHeaderV10` (FILE **file*, `EmbPattern` **pattern*)
- void `readDescriptions` (FILE **file*, `EmbPattern` **pattern*)
- void `readHoopName` (FILE **file*, `EmbPattern` **pattern*)
- void `readImageString` (FILE **file*, `EmbPattern` **pattern*)
- void `readProgrammableFills` (FILE **file*, `EmbPattern` **pattern*)
- void `readMotifPatterns` (FILE **file*, `EmbPattern` **pattern*)
- void `readFeatherPatterns` (FILE **file*, `EmbPattern` **pattern*)
- void `readThreads` (FILE **file*, `EmbPattern` **pattern*)
- void `emblnt_read` (FILE **f*, char **label*, void **b*, int *mode*)
- void `emblnt_write` (FILE **f*, char **label*, void **b*, int *mode*)
- int `emb_readline` (FILE **file*, char **line*, int *maxLength*)
 - file line maxLength*
- int `bcfFile_read` (FILE **file*, `bcf_file` **bcfFile*)
 - file bcfFile*
- FILE * `GetFile` (`bcf_file` **bcfFile*, FILE **file*, char **fileToFind*)
 - Get the *File* object.
- void `bcf_file_free` (`bcf_file` **bcfFile*)
 - bcfFile*
- void `binaryReadString` (FILE **file*, char **buffer*, int *maxLength*)
 - file buffer maxLength*
- void `binaryReadUnicodeString` (FILE **file*, char **buffer*, const int *stringLength*)
 - file buffer stringLength*

- int `stringInArray` (const char *s, const char **array)
- void `fpad` (FILE *f, char c, int n)
 f
- char * `copy_trim` (char const *s)
 s
- char * `emb_optOut` (EmbReal num, char *str)
 Optimizes the number (num) for output to a text file and returns it as a string (str).
- void `write_24bit` (FILE *file, int)
 file x
- int `check_header_present` (FILE *file, int minimum_header_length)
 file minimum_header_length
- unsigned short `fread_uint16` (FILE *file)
 f
- short `fread_int16` (FILE *f)
 f
- int `fread_int32_be` (FILE *f)
 f
- void `safe_free` (void *data)
 data
- void `binaryWriteUIntBE` (FILE *f, unsigned int data)
 f data
- void `binaryWriteUInt` (FILE *f, unsigned int data)
 f data
- void `binaryWriteIntBE` (FILE *f, int data)
 f data
- void `binaryWriteInt` (FILE *f, int data)
 f data
- void `binaryWriteUShort` (FILE *f, unsigned short data)
 f data
- void `binaryWriteUShortBE` (FILE *f, unsigned short data)
 f data
- void `binaryWriteShort` (FILE *f, short data)
 f data
- `bcf_file_difat * bcf_difat_create` (FILE *file, unsigned int fatSectors, const unsigned int sectorSize)
 file fatSectors sectorSize
- unsigned int `readFullSector` (FILE *file, `bcf_file_difat` *bcfFile, unsigned int *numberOfDifatEntriesStillToRead)
 file bcfFile difatEntriesToRead
- unsigned int `numberOfEntriesInDifatSector` (`bcf_file_difat` *fat)
- void `bcf_file_difat_free` (`bcf_file_difat` *difat)
- unsigned int `entriesInDifatSector` (`bcf_file_difat` *fat)
 fat
- `bcf_file_fat * bcfFileFat_create` (const unsigned int sectorSize)
 sectorSize
- void `loadFatFromSector` (`bcf_file_fat` *fat, FILE *file)
 fat file
- void `bcf_file_fat_free` (`bcf_file_fat` **fat)
- `bcf_directory_entry * CompoundFileDialogEntry` (FILE *file)
 file
- `bcf_directory * CompoundFileDialog` (const unsigned int maxNumberOfDirectoryEntries)
 maxNumberOfDirectoryEntries

- void `readNextSector` (FILE *file, `bcf_directory` *dir)
file dir
- void `bcf_directory_free` (`bcf_directory` **dir)
dir
- `bcf_file_header bcfFileHeader_read` (FILE *file)
file
- int `bcfFileHeader_isValid` (`bcf_file_header` header)
- int `hus_compress` (char *input, int size, char *output, int *out_size)
- int `hus_decompress` (char *input, int size, char *output, int *out_size)
- void `testTangentPoints` (`EmbCircle` c, `EmbVector` p, `EmbVector` *t0, `EmbVector` *t1)
- void `printArcResults` (`EmbReal` bulge, `EmbArc` arc, `EmbReal` centerX, `EmbReal` centerY, `EmbReal` radius, `EmbReal` diameter, `EmbReal` chord, `EmbReal` chordMidX, `EmbReal` chordMidY, `EmbReal` sagitta, `EmbReal` apothem, `EmbReal` incAngle, char `clockwise`)
- int `create_test_file_1` (const char *outf)
- int `create_test_file_2` (const char *outf)
- int `create_test_file_3` (const char *outf)
- int `testEmbCircle` (void)
- int `testEmbCircle_2` (void)
- int `testGeomArc` (void)
- int `testThreadColor` (void)
- int `testEmbFormat` (void)
- void `embColor_read` (FILE *f, `EmbColor` *c, int toRead)
f c toRead
- void `embColor_write` (FILE *f, `EmbColor` c, int toWrite)
f c toWrite
- char `read100` (`EmbPattern` *pattern, FILE *file)
- char `write100` (`EmbPattern` *pattern, FILE *file)
- char `read10o` (`EmbPattern` *pattern, FILE *file)
- char `write10o` (`EmbPattern` *pattern, FILE *file)
- char `readArt` (`EmbPattern` *pattern, FILE *file)
- char `writeArt` (`EmbPattern` *pattern, FILE *file)
- char `readBmc` (`EmbPattern` *pattern, FILE *file)
- char `writeBmc` (`EmbPattern` *pattern, FILE *file)
- char `readBro` (`EmbPattern` *pattern, FILE *file)
- char `writeBro` (`EmbPattern` *pattern, FILE *file)
- char `readCnd` (`EmbPattern` *pattern, FILE *file)
- char `writeCnd` (`EmbPattern` *pattern, FILE *file)
- char `readCol` (`EmbPattern` *pattern, FILE *file)
- char `writeCol` (`EmbPattern` *pattern, FILE *file)
- char `readCsd` (`EmbPattern` *pattern, FILE *file)
- char `writeCsd` (`EmbPattern` *pattern, FILE *file)
- char `readCsv` (`EmbPattern` *pattern, FILE *file)
- char `writeCsv` (`EmbPattern` *pattern, FILE *file)
- char `readDat` (`EmbPattern` *pattern, FILE *file)
- char `writeDat` (`EmbPattern` *pattern, FILE *file)
- char `readDem` (`EmbPattern` *pattern, FILE *file)
- char `writeDem` (`EmbPattern` *pattern, FILE *file)
- char `readDsb` (`EmbPattern` *pattern, FILE *file)
- char `writeDsb` (`EmbPattern` *pattern, FILE *file)
- char `readDst` (`EmbPattern` *pattern, FILE *file)
- char `writeDst` (`EmbPattern` *pattern, FILE *file)
- char `readDsz` (`EmbPattern` *pattern, FILE *file)
- char `writeDsz` (`EmbPattern` *pattern, FILE *file)
- char `readDxf` (`EmbPattern` *pattern, FILE *file)

- char `writeDxf` (`EmbPattern` *pattern, `FILE` *file)
- char `readEdr` (`EmbPattern` *pattern, `FILE` *file)
- char `writeEdr` (`EmbPattern` *pattern, `FILE` *file)
- char `readEmd` (`EmbPattern` *pattern, `FILE` *file)
- char `writeEmd` (`EmbPattern` *pattern, `FILE` *file)
- char `readExp` (`EmbPattern` *pattern, `FILE` *file)
- char `writeExp` (`EmbPattern` *pattern, `FILE` *file)
- char `readExy` (`EmbPattern` *pattern, `FILE` *file)
- char `writeExy` (`EmbPattern` *pattern, `FILE` *file)
- char `readEys` (`EmbPattern` *pattern, `FILE` *file)
- char `writeEys` (`EmbPattern` *pattern, `FILE` *file)
- char `readFxy` (`EmbPattern` *pattern, `FILE` *file)
- char `writeFxy` (`EmbPattern` *pattern, `FILE` *file)
- char `readGc` (`EmbPattern` *pattern, `FILE` *file)
- char `writeGc` (`EmbPattern` *pattern, `FILE` *file)
- char `readGnc` (`EmbPattern` *pattern, `FILE` *file)
- char `writeGnc` (`EmbPattern` *pattern, `FILE` *file)
- char `readGt` (`EmbPattern` *pattern, `FILE` *file)
- char `writeGt` (`EmbPattern` *pattern, `FILE` *file)
- char `readHus` (`EmbPattern` *pattern, `FILE` *file)
- char `writeHus` (`EmbPattern` *pattern, `FILE` *file)
- char `readInb` (`EmbPattern` *pattern, `FILE` *file)
- char `writeInb` (`EmbPattern` *pattern, `FILE` *file)
- char `readInf` (`EmbPattern` *pattern, `FILE` *file)
- char `writeInf` (`EmbPattern` *pattern, `FILE` *file)
- char `readJef` (`EmbPattern` *pattern, `FILE` *file)
- char `writeJef` (`EmbPattern` *pattern, `FILE` *file)
- char `readKsm` (`EmbPattern` *pattern, `FILE` *file)
- char `writeKsm` (`EmbPattern` *pattern, `FILE` *file)
- char `readMax` (`EmbPattern` *pattern, `FILE` *file)
- char `writeMax` (`EmbPattern` *pattern, `FILE` *file)
- char `readMit` (`EmbPattern` *pattern, `FILE` *file)
- char `writeMit` (`EmbPattern` *pattern, `FILE` *file)
- char `readNew` (`EmbPattern` *pattern, `FILE` *file)
- char `writeNew` (`EmbPattern` *pattern, `FILE` *file)
- char `readOfm` (`EmbPattern` *pattern, `FILE` *file)
- char `writeOfm` (`EmbPattern` *pattern, `FILE` *file)
- char `readPcd` (`EmbPattern` *pattern, const `char` *fileName, `FILE` *file)
- char `writePcd` (`EmbPattern` *pattern, `FILE` *file)
- char `readPcm` (`EmbPattern` *pattern, `FILE` *file)
- char `writePcm` (`EmbPattern` *pattern, `FILE` *file)
- char `readPcq` (`EmbPattern` *pattern, const `char` *fileName, `FILE` *file)
- char `writePcq` (`EmbPattern` *pattern, `FILE` *file)
- char `readPcs` (`EmbPattern` *pattern, const `char` *fileName, `FILE` *file)
- char `writePcs` (`EmbPattern` *pattern, `FILE` *file)
- char `readPec` (`EmbPattern` *pattern, const `char` *fileName, `FILE` *file)
- char `writePec` (`EmbPattern` *pattern, const `char` *fileName, `FILE` *file)
- char `readPel` (`EmbPattern` *pattern, `FILE` *file)
- char `writePel` (`EmbPattern` *pattern, `FILE` *file)
- char `readPem` (`EmbPattern` *pattern, `FILE` *file)
- char `writePem` (`EmbPattern` *pattern, `FILE` *file)
- char `readPes` (`EmbPattern` *pattern, const `char` *fileName, `FILE` *file)
- char `writePes` (`EmbPattern` *pattern, const `char` *fileName, `FILE` *file)
- char `readPhb` (`EmbPattern` *pattern, `FILE` *file)
- char `writePhb` (`EmbPattern` *pattern, `FILE` *file)

- char `readPhc` (`EmbPattern` *pattern, `FILE` *file)
- char `writePhc` (`EmbPattern` *pattern, `FILE` *file)
- char `readPlt` (`EmbPattern` *pattern, `FILE` *file)
- char `writePlt` (`EmbPattern` *pattern, `FILE` *file)
- char `readRgb` (`EmbPattern` *pattern, `FILE` *file)
- char `writeRgb` (`EmbPattern` *pattern, `FILE` *file)
- char `readSew` (`EmbPattern` *pattern, `FILE` *file)
- char `writeSew` (`EmbPattern` *pattern, `FILE` *file)
- char `readShv` (`EmbPattern` *pattern, `FILE` *file)
- char `writeShv` (`EmbPattern` *pattern, `FILE` *file)
- char `readSst` (`EmbPattern` *pattern, `FILE` *file)
- char `writeSst` (`EmbPattern` *pattern, `FILE` *file)
- char `readStx` (`EmbPattern` *pattern, `FILE` *file)
- char `writeStx` (`EmbPattern` *pattern, `FILE` *file)
- char `readSvg` (`EmbPattern` *pattern, `FILE` *file)
- char `writeSvg` (`EmbPattern` *pattern, `FILE` *file)
- char `readT01` (`EmbPattern` *pattern, `FILE` *file)
- char `writeT01` (`EmbPattern` *pattern, `FILE` *file)
- char `readT09` (`EmbPattern` *pattern, `FILE` *file)
- char `writeT09` (`EmbPattern` *pattern, `FILE` *file)
- char `readTap` (`EmbPattern` *pattern, `FILE` *file)
- char `writeTap` (`EmbPattern` *pattern, `FILE` *file)
- char `readThr` (`EmbPattern` *pattern, `FILE` *file)
- char `writeThr` (`EmbPattern` *pattern, `FILE` *file)
- char `readTxt` (`EmbPattern` *pattern, `FILE` *file)
- char `writeTxt` (`EmbPattern` *pattern, `FILE` *file)
- char `readU00` (`EmbPattern` *pattern, `FILE` *file)
- char `writeU00` (`EmbPattern` *pattern, `FILE` *file)
- char `readU01` (`EmbPattern` *pattern, `FILE` *file)
- char `writeU01` (`EmbPattern` *pattern, `FILE` *file)
- char `readVip` (`EmbPattern` *pattern, `FILE` *file)
- char `writeVip` (`EmbPattern` *pattern, `FILE` *file)
- char `readVp3` (`EmbPattern` *pattern, `FILE` *file)
- char `writeVp3` (`EmbPattern` *pattern, `FILE` *file)
- char `readXxx` (`EmbPattern` *pattern, `FILE` *file)
- char `writeXxx` (`EmbPattern` *pattern, `FILE` *file)
- char `readZsk` (`EmbPattern` *pattern, `FILE` *file)
- char `writeZsk` (`EmbPattern` *pattern, `FILE` *file)

Variables

- const char `imageWithFrame` [38][48]

18.44.1 Macro Definition Documentation

18.44.1.1 BULGETOCONTROL `#define BULGETOCONTROL 2`

18.44.1.2 BULGETOEND `#define BULGETOEND 4`

18.44.1.3 CompoundFileSector_DIFAT_Sector `#define CompoundFileSector_DIFAT_Sector 0xFFFFFFFFC`

18.44.1.4 CompoundFileSector_EndOfChain #define CompoundFileSector_EndOfChain 0xFFFFFFFFFE

18.44.1.5 CompoundFileSector_FAT_Sector #define CompoundFileSector_FAT_Sector 0xFFFFFFFFFD

18.44.1.6 CompoundFileSector_FreeSector #define CompoundFileSector_FreeSector 0xFFFFFFFFFF

18.44.1.7 CompoundFileSector_MaxRegSector #define CompoundFileSector_MaxRegSector 0xFFFFFFFFFA
Type of sector

18.44.1.8 CompoundFileStreamId_MaxRegularStreamId #define CompoundFileStreamId_MaxRegular←
StreamId 0xFFFFFFFFFA
Special values for Stream Identifiers All real stream Ids are less than this

18.44.1.9 CompoundFileStreamId_NoStream #define CompoundFileStreamId_NoStream 0xFFFFFFFFFF
There is no valid stream Id

18.44.1.10 CUBICTOCONTROL1 #define CUBICTOCONTROL1 32

18.44.1.11 CUBICTOCONTROL2 #define CUBICTOCONTROL2 64

18.44.1.12 CUBICTOEND #define CUBICTOEND 128

18.44.1.13 DXF_VERSION_2000 #define DXF_VERSION_2000 "AC1015"

18.44.1.14 DXF_VERSION_2002 #define DXF_VERSION_2002 "AC1015"

18.44.1.15 DXF_VERSION_2004 #define DXF_VERSION_2004 "AC1018"

18.44.1.16 DXF_VERSION_2006 #define DXF_VERSION_2006 "AC1018"

18.44.1.17 DXF_VERSION_2007 #define DXF_VERSION_2007 "AC1021"

18.44.1.18 DXF_VERSION_2009 #define DXF_VERSION_2009 "AC1021"

18.44.1.19 DXF_VERSION_2010 #define DXF_VERSION_2010 "AC1024"

18.44.1.20 DXF_VERSION_2013 #define DXF_VERSION_2013 "AC1027"

18.44.1.21 DXF_VERSION_R10 #define DXF_VERSION_R10 "AC1006"

18.44.1.22 DXF_VERSION_R11 #define DXF_VERSION_R11 "AC1009"

18.44.1.23 DXF_VERSION_R12 #define DXF_VERSION_R12 "AC1009"

18.44.1.24 DXF_VERSION_R13 #define DXF_VERSION_R13 "AC1012"

18.44.1.25 DXF_VERSION_R14 #define DXF_VERSION_R14 "AC1014"

18.44.1.26 DXF_VERSION_R15 #define DXF_VERSION_R15 "AC1015"

18.44.1.27 DXF_VERSION_R18 #define DXF_VERSION_R18 "AC1018"

18.44.1.28 DXF_VERSION_R21 #define DXF_VERSION_R21 "AC1021"

18.44.1.29 DXF_VERSION_R24 #define DXF_VERSION_R24 "AC1024"

18.44.1.30 DXF_VERSION_R27 #define DXF_VERSION_R27 "AC1027"

18.44.1.31 ELEMENT_A #define ELEMENT_A 1

18.44.1.32 ELEMENT_ANIMATE #define ELEMENT_ANIMATE 2

18.44.1.33 ELEMENT_ANIMATECOLOR #define ELEMENT_ANIMATECOLOR 3

18.44.1.34 ELEMENT_ANIMATEMOTION #define ELEMENT_ANIMATEMOTION 4

18.44.1.35 ELEMENT_ANIMATETRANSFORM #define ELEMENT_ANIMATETRANSFORM 5

18.44.1.36 ELEMENT_ANIMATION #define ELEMENT_ANIMATION 6

18.44.1.37 ELEMENT_AUDIO #define ELEMENT_AUDIO 7

18.44.1.38 ELEMENT_CIRCLE #define ELEMENT_CIRCLE 8

18.44.1.39 ELEMENT_DEFS #define ELEMENT_DEFS 9

18.44.1.40 ELEMENT_DESC #define ELEMENT_DESC 10

18.44.1.41 ELEMENT_DISCARD #define ELEMENT_DISCARD 11

18.44.1.42 ELEMENT_ELLIPSE #define ELEMENT_ELLIPSE 12

18.44.1.43 ELEMENT_FONT #define ELEMENT_FONT 13

18.44.1.44 ELEMENT_FONT_FACE #define ELEMENT_FONT_FACE 14

18.44.1.45 ELEMENT_FONT_FACE_SRC #define ELEMENT_FONT_FACE_SRC 15

18.44.1.46 ELEMENT_FONT_FACE_URI #define ELEMENT_FONT_FACE_URI 16

18.44.1.47 ELEMENT_FOREIGN_OBJECT #define ELEMENT_FOREIGN_OBJECT 17

18.44.1.48 ELEMENT_G #define ELEMENT_G 18

18.44.1.49 ELEMENT_GLYPH #define ELEMENT_GLYPH 19

18.44.1.50 ELEMENT_HANDLER #define ELEMENT_HANDLER 20

18.44.1.51 ELEMENT_HKERN #define ELEMENT_HKERN 21

18.44.1.52 ELEMENT_IMAGE #define ELEMENT_IMAGE 22

18.44.1.53 ELEMENT_LINE #define ELEMENT_LINE 23

18.44.1.54 ELEMENT_LINEAR_GRADIENT #define ELEMENT_LINEAR_GRADIENT 24

18.44.1.55 ELEMENT_LISTENER #define ELEMENT_LISTENER 25

18.44.1.56 ELEMENT_METADATA #define ELEMENT_METADATA 26

18.44.1.57 ELEMENT_MISSING_GLYPH #define ELEMENT_MISSING_GLYPH 27

18.44.1.58 ELEMENT_MPATH #define ELEMENT_MPATH 28

18.44.1.59 ELEMENT_PATH #define ELEMENT_PATH 29

18.44.1.60 ELEMENT_POLYGON #define ELEMENT_POLYGON 30

18.44.1.61 ELEMENT_POLYLINE #define ELEMENT_POLYLINE 31

18.44.1.62 ELEMENT_PREFETCH #define ELEMENT_PREFETCH 32

18.44.1.63 ELEMENT_RADIAL_GRADIENT #define ELEMENT_RADIAL_GRADIENT 33

18.44.1.64 ELEMENT_RECT #define ELEMENT_RECT 34

18.44.1.65 ELEMENT_SCRIPT #define ELEMENT_SCRIPT 35

18.44.1.66 ELEMENT_SET #define ELEMENT_SET 36

18.44.1.67 ELEMENT_SOLID_COLOR #define ELEMENT_SOLID_COLOR 37

18.44.1.68 ELEMENT_STOP #define ELEMENT_STOP 38

18.44.1.69 ELEMENT_SVG #define ELEMENT_SVG 39

18.44.1.70 ELEMENT_SWITCH #define ELEMENT_SWITCH 40

18.44.1.71 ELEMENT_TBREAK #define ELEMENT_TBREAK 41

18.44.1.72 ELEMENT_TEXT #define ELEMENT_TEXT 42

18.44.1.73 ELEMENT_TEXT_AREA #define ELEMENT_TEXT_AREA 43

18.44.1.74 ELEMENT_TITLE #define ELEMENT_TITLE 44

18.44.1.75 ELEMENT_TSPAN #define ELEMENT_TSPAN 45

18.44.1.76 ELEMENT_USE #define ELEMENT_USE 46

18.44.1.77 ELEMENT_VIDEO #define ELEMENT_VIDEO 47

18.44.1.78 ELEMENT_XML #define ELEMENT_XML 0

18.44.1.79 ELLIPSETOEND #define ELLIPSETOEND 16

18.44.1.80 ELLIPSETORAD #define ELLIPSETORAD 8

18.44.1.81 EMB_BIG_ENDIAN #define EMB_BIG_ENDIAN 0

18.44.1.82 EMB_INT16_BIG #define EMB_INT16_BIG 2

18.44.1.83 EMB_INT16_LITTLE #define EMB_INT16_LITTLE 3

18.44.1.84 EMB_INT32_BIG #define EMB_INT32_BIG 4

18.44.1.85 EMB_INT32_LITTLE #define EMB_INT32_LITTLE 5

18.44.1.86 EMB_LITTLE_ENDIAN #define EMB_LITTLE_ENDIAN 1

18.44.1.87 EMB_MAX #define EMB_MAX(
 A,
 B) ((A) > (B)) ? (A) : (B))

18.44.1.88 EMB_MIN #define EMB_MIN(
 A,
 B) ((A) < (B)) ? (A) : (B))

18.44.1.89 ENDIAN_HOST #define ENDIAN_HOST [EMB_LITTLE_ENDIAN](#)

18.44.1.90 GREEN_TERM_COLOR #define GREEN_TERM_COLOR "\x1B[0;32m"

18.44.1.91 HOOP_110X110 #define HOOP_110X110 1

18.44.1.92 HOOP_126X110 #define HOOP_126X110 0

18.44.1.93 HOOP_140X200 #define HOOP_140X200 3

18.44.1.94 HOOP_230X200 #define HOOP_230X200 4

18.44.1.95 HOOP_50X50 #define HOOP_50X50 2

18.44.1.96 LINETO #define LINETO 0

18.44.1.97 MOVETO #define MOVETO 1

18.44.1.98 N_PES VERSIONS #define N_PES VERSIONS 13

18.44.1.99 ObjectTypeRootEntry #define ObjectTypeRootEntry 0x05
the root entry

18.44.1.100 ObjectTypeStorage #define ObjectTypeStorage 0x01
a directory type object

18.44.1.101 ObjectTypeStream #define ObjectTypeStream 0x02
a file type object

18.44.1.102 ObjectTypeUnknown #define ObjectTypeUnknown 0x00
Type of directory object Probably unallocated

18.44.1.103 PES0001 #define PES0001 0

18.44.1.104 PES0020 #define PES0020 1

18.44.1.105 PES0022 #define PES0022 2

18.44.1.106 PES0030 #define PES0030 3

18.44.1.107 PES0040 #define PES0040 4

18.44.1.108 PES0050 #define PES0050 5

18.44.1.109 PES0055 #define PES0055 6

18.44.1.110 PES0056 #define PES0056 7

18.44.1.111 PES0060 #define PES0060 8

18.44.1.112 PES0070 #define PES0070 9

18.44.1.113 PES0080 #define PES0080 10

18.44.1.114 PES0090 #define PES0090 11

18.44.1.115 PES0100 #define PES0100 12

18.44.1.116 QUADTOCONTROL #define QUADTOCONTROL 256

18.44.1.117 QUADTOEND #define QUADTOEND 512

18.44.1.118 RED_TERM_COLOR #define RED_TERM_COLOR "\x1B[0;31m"

18.44.1.119 RESET_TERM_COLOR #define RESET_TERM_COLOR "\033[0m"

18.44.1.120 SVG_ATTRIBUTE #define SVG_ATTRIBUTE 4

18.44.1.121 SVG_CATCH_ALL #define SVG_CATCH_ALL 5

18.44.1.122 SVG_CREATOR_EMBROIDERMODDER #define SVG_CREATOR_EMBROIDERMODDER 1

18.44.1.123 SVG_CREATOR_ILLUSTRATOR #define SVG_CREATOR_ILLUSTRATOR 2

18.44.1.124 SVG_CREATOR_INKSCAPE #define SVG_CREATOR_INKSCAPE 3

18.44.1.125 SVG_CREATOR_NULL #define SVG_CREATOR_NULL 0

18.44.1.126 SVG_ELEMENT #define SVG_ELEMENT 1

18.44.1.127 SVG_EXPECT_ATTRIBUTE #define SVG_EXPECT_ATTRIBUTE 2

18.44.1.128 SVG_EXPECT_ELEMENT #define SVG_EXPECT_ELEMENT 1

18.44.1.129 SVG_EXPECT_NULL #define SVG_EXPECT_NULL 0

18.44.1.130 SVG_EXPECT_VALUE #define SVG_EXPECT_VALUE 3

18.44.1.131 SVG_MEDIA_PROPERTY #define SVG_MEDIA_PROPERTY 3

18.44.1.132 SVG_NULL #define SVG_NULL 0

18.44.1.133 SVG_PROPERTY #define SVG_PROPERTY 2

18.44.1.134 YELLOW_TERM_COLOR #define YELLOW_TERM_COLOR "\x1B[1;33m"

18.44.2 Typedef Documentation

18.44.2.1 bcf_directory typedef struct [_bcf_directory](#) bcf_directory

Todo possibly add a directory tree in the future.

18.44.2.2 bcf_directory_entry typedef struct [_bcf_directory_entry](#) bcf_directory_entry

18.44.2.3 bcf_file typedef struct [_bcf_file](#) bcf_file

18.44.2.4 bcf_file_difat typedef struct [_bcf_file_difat](#) bcf_file_difat

18.44.2.5 bcf_file_fat typedef struct [_bcf_file_fat](#) bcf_file_fat

18.44.2.6 bcf_file_header typedef struct [_bcf_file_header](#) bcf_file_header

Todo CLSID should be a separate type.

18.44.2.7 compress typedef struct Compress compress

18.44.2.8 huffman typedef struct Huffman huffman

18.44.2.9 StxThread `typedef struct StxThread_ StxThread`

18.44.2.10 SubDescriptor `typedef struct SubDescriptor_ SubDescriptor`

18.44.2.11 SvgAttribute `typedef struct SvgAttribute_ SvgAttribute`

18.44.2.12 ThredExtension `typedef struct ThredExtension_ ThredExtension`

18.44.2.13 ThredHeader `typedef struct ThredHeader_ ThredHeader`

18.44.2.14 VipHeader `typedef struct VipHeader_ VipHeader`

18.44.2.15 vp3Hoop `typedef struct _vp3Hoop vp3Hoop`

18.44.3 Enumeration Type Documentation

18.44.3.1 CSV_EXPECT `enum CSV_EXPECT`

Enumerator

| | |
|-------------------|--|
| CSV_EXPECT_NULL | |
| CSV_EXPECT_QUOTE1 | |
| CSV_EXPECT_QUOTE2 | |
| CSV_EXPECT_COMMAS | |

18.44.3.2 CSV_MODE `enum CSV_MODE`

Enumerator

| | |
|-------------------|--|
| CSV_MODE_NULL | |
| CSV_MODE_COMMENT | |
| CSV_MODE_VARIABLE | |
| CSV_MODE_THREAD | |
| CSV_MODE_STITCH | |

18.44.4 Function Documentation

18.44.4.1 bcf_difat_create() `bcf_file_difat * bcf_difat_create (`
 `FILE * file,`
 `unsigned int fatSectors,`
 `const unsigned int sectorSize)`
file fatSectors sectorSize

Returns

bcf_file_difat*

18.44.4.2 bcf_directory_free() void bcf_directory_free (
 bcf_directory ** dir)
dir

18.44.4.3 bcf_file_difat_free() void bcf_file_difat_free (
 bcf_file_difat * difat)

18.44.4.4 bcf_file_fat_free() void bcf_file_fat_free (
 bcf_file_fat ** fat)

18.44.4.5 bcf_file_free() void bcf_file_free (
 bcf_file * bcfFile)
bcfFile

18.44.4.6 bcfFile_read() int bcfFile_read (
 FILE * file,
 bcf_file * bcfFile)
file bcfFile

Returns

int

18.44.4.7 bcfFileFat_create() bcf_file_fat * bcfFileFat_create (
 const unsigned int sectorSize)
sectorSize

Returns

bcf_file_fat*

18.44.4.8 bcfFileHeader_isValid() int bcfFileHeader_isValid (
 bcf_file_header header)

18.44.4.9 bcfFileHeader_read() bcf_file_header bcfFileHeader_read (
 FILE * file)
file

Returns

bcf_file_header

18.44.4.10 binaryReadString() void binaryReadString (

```
FILE * file,
char * buffer,
int maxLength )
```

file buffer maxLength

18.44.4.11 binaryReadUnicodeString() void binaryReadUnicodeString (

```
FILE * file,
char * buffer,
const int stringLength )
```

file buffer stringLength

18.44.4.12 binaryWriteInt() void binaryWriteInt (

```
FILE * f,
int data )
```

f data

Todo replace with emblnt_read

18.44.4.13 binaryWriteIntBE() void binaryWriteIntBE (

```
FILE * f,
int data )
```

f data

Todo replace with emblnt_read

18.44.4.14 binaryWriteShort() void binaryWriteShort (

```
FILE * f,
short data )
```

f data

Todo replace with emblnt_read

18.44.4.15 binaryWriteUInt() void binaryWriteUInt (

```
FILE * f,
unsigned int data )
```

f data

Todo replace with emblnt_read

18.44.4.16 binaryWriteUIntBE() void binaryWriteUIntBE (

```
FILE * f,
unsigned int data )
```

f data

Todo replace with emblnt_read

18.44.4.17 binaryWriteUShort() void binaryWriteUShort (FILE * *f*, unsigned short *data*)
f data

Todo replace with emblnt_read

18.44.4.18 binaryWriteUShortBE() void binaryWriteUShortBE (FILE * *f*, unsigned short *data*)
f data

Todo replace with emblnt_read

18.44.4.19 check_header_present() int check_header_present (FILE * *file*, int *minimum_header_length*)
file minimum_header_length

Returns

int

Checks that there are enough bytes to interpret the header, stops possible segfaults when reading in the header bytes.

Returns 0 if there aren't enough, or the length of the file if there are.

18.44.4.20 CompoundFileDirectory() *bcf_directory* * CompoundFileDirectory (const unsigned int *maxNumberOfDirectoryEntries*)
maxNumberOfDirectoryEntries

Returns

*bcf_directory**

18.44.4.21 CompoundFileDirectoryEntry() *bcf_directory_entry* * CompoundFileDirectoryEntry (FILE * *file*)
file

Returns

*bcf_directory_entry**

18.44.4.22 compress_get_bits() int compress_get_bits (compress * *c*, int *length*)
c length Returns .

18.44.4.23 compress_get_position() int compress_get_position (compress * *c*)
c. Returns the position as an int.

18.44.4.24 compress_get_token() int compress_get_token (
 compress * c)

c . Returns the token as an int.

18.44.4.25 compress_load_block() void compress_load_block (
 compress * c)

c . Returns nothing.

18.44.4.26 compress_load_character_huffman() void compress_load_character_huffman (
 compress * c)

Load character table to compress struct c. Returns nothing.

18.44.4.27 compress_load_character_length_huffman() void compress_load_character_length_huffman (
 compress * c)

c . Returns.

18.44.4.28 compress_load_distance_huffman() void compress_load_distance_huffman (
 compress * c)

c . Returns nothing.

18.44.4.29 compress_pop() int compress_pop (
 compress * c,
 int bit_count)

c bit_count . Returns.

18.44.4.30 compress_read_variable_length() int compress_read_variable_length (
 compress * c)

c . Returns.

18.44.4.31 copy_trim() char * copy_trim (
 char const * s)

s

Returns

char*

Todo description

18.44.4.32 create_test_file_1() int create_test_file_1 (
 const char * outf)

18.44.4.33 create_test_file_2() int create_test_file_2 (
 const char * outf)

18.44.4.34 create_test_file_3() int create_test_file_3 (
 const char * outf)

```
18.44.4.35 decode_t01_record() int decode_t01_record (
    unsigned char b[3],
    int * x,
    int * y,
    int * flags )
```

b x y flags .

Todo remove the unused return argument.

```
18.44.4.36 decode_tajima_ternary() void decode_tajima_ternary (
    unsigned char b[3],
    int * x,
    int * y )
```

Decode the signed ternary of the tajima format from *b* to the position values *x* and *y*.
There is no return argument.

```
18.44.4.37 decodeNewStitch() int decodeNewStitch (
    unsigned char value )
```

value

Returns

int

```
18.44.4.38 emb_optOut() char * emb_optOut (
    EmbReal num,
    char * str )
```

Optimizes the number (*num*) for output to a text file and returns it as a string (*str*).
num str

Returns

char*

```
18.44.4.39 emb_readline() int emb_readline (
    FILE * file,
    char * line,
    int maxLength )
```

file line maxLength

Returns

int

```
18.44.4.40 embColor_read() void embColor_read (
    FILE * f,
    EmbColor * c,
    int toRead )
```

f c toRead

```
18.44.4.41 embColor_write() void embColor_write (
    FILE * f,
    EmbColor c,
    int toWrite )
fc toWrite
```

```
18.44.4.42 embInt_read() void embInt_read (
    FILE * f,
    char * label,
    void * b,
    int mode )
f label b mode
```

Read and write system for multiple byte types.

The caller passes the function to read/write from, the memory location as a void pointer and a mode identifier that describes the type. This way we can abstract out the endianness of the system running the library and don't have to maintain many functions, just two.

```
18.44.4.43 embInt_write() void embInt_write (
    FILE * f,
    char * label,
    void * b,
    int mode )
f label b mode
```

```
18.44.4.44 encode_t01_record() void encode_t01_record (
    unsigned char b[3],
    int x,
    int y,
    int flags )
```

Encode into bytes *b* the values of the *x*-position *x*, *y*-position *y* and the *flags*.

```
18.44.4.45 encode_tajima_ternary() int encode_tajima_ternary (
    unsigned char b[3],
    int x,
    int y )
```

Encode the signed ternary of the tajima format into *b* the position values *x* and *y*.

If the values of *x* or *y* fall outside of the valid range of -121 and +121 then it returns 0 and 1.

```
18.44.4.46 entriesInDifatSector() unsigned int entriesInDifatSector (
    bcf_file_difat * fat )
```

fat

Returns

unsigned int

```
18.44.4.47 fpad() void fpad (
    FILE * file,
    char c,
    int n )
```

f

Returns

int

18.44.4.48 fread_int16() short fread_int16 (FILE * *f*)

Returns

short

18.44.4.49 fread_int32_be() int fread_int32_be (FILE * *f*)

Returns

int

Todo replace with emblnt_read

18.44.4.50 fread_uint16() unsigned short fread_uint16 (FILE * *f*)

f

Returns

unsigned short

Todo replace with emblnt_read

18.44.4.51 GetFile() FILE * GetFile (bcf_file * *bcfFile*, FILE * *file*, char * *fileToFind*)

Get the File object.

bcfFile file fileToFind

Returns

FILE*

18.44.4.52 huffman_build_table() void huffman_build_table (huffman * *h*)

These next 2 functions represent the [Huffman](#) class in tartarize's code. *h*

18.44.4.53 huffman_table_lookup() int * huffman_table_lookup (huffman * *h*, int *byte_lookup*, int * *lengths*)

18.44.4.54 hus_compress() int hus_compress (char * *data*, int *length*, char * *output*, int * *output_length*)

data length output output_length. Returns whether it was successful as an int.

This avoids the now unnecessary compression by placing a minimal header of 6 bytes and using only literals in the huffman compressed part (see the sources above).

18.44.4.55 `hus_decompress()` `int hus_decompress (`
 `char * data,`
 `int length,`
 `char * output,`
 `int * output_length)`
data length output output_length. Returns whether the decompression was successful.

18.44.4.56 `loadFatFromSector()` `void loadFatFromSector (`
 `bcf_file_fat * fat,`
 `FILE * file)`
fat file

18.44.4.57 `mitDecodeStitch()` `int mitDecodeStitch (`
 `unsigned char value)`
value
Returns
`int`

18.44.4.58 `mitEncodeStitch()` `unsigned char mitEncodeStitch (`
 `EmbReal value)`
value
Returns
`unsigned char`

18.44.4.59 `numberOfEntriesInDifatSector()` `unsigned int numberOfEntriesInDifatSector (`
 `bcf_file_difat * fat)`

18.44.4.60 `pfaffDecode()` `EmbReal pfaffDecode (`
 `unsigned char a1,`
 `unsigned char a2,`
 `unsigned char a3)`

Decode the bytes *a1*, *a2* and *a3*. Returns the EmbReal floating-point value.

18.44.4.61 `pfaffEncode()` `void pfaffEncode (`
 `FILE * file,`
 `int dx,`
 `int dy,`
 `int flags)`
file dx dy flags

18.44.4.62 `printArcResults()` `void printArcResults (`
 `EmbReal bulge,`
 `EmbArc arc,`
 `EmbReal centerX,`
 `EmbReal centerY,`
 `EmbReal radius,`
 `EmbReal diameter,`
 `EmbReal chord,`
 `EmbReal chordMidX,`

```
    EmbReal chordMidY,  
    EmbReal sagitta,  
    EmbReal apothem,  
    EmbReal incAngle,  
    char clockwise )
```

18.44.4.63 `read100()` `char read100 (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.64 `read10o()` `char read10o (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.65 `readArt()` `char readArt (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.66 `readBmc()` `char readBmc (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.67 `readBro()` `char readBro (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.68 `readCnd()` `char readCnd (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.69 `readCol()` `char readCol (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.70 `readCsd()` `char readCsd (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.71 `readCsv()` `char readCsv (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.72 `readDat()` `char readDat (`
 `EmbPattern * pattern,`
 `FILE * file)`

```
18.44.4.73 readDem() char readDem (
    EmbPattern * pattern,
    FILE * file )
```

```
18.44.4.74 readDescriptions() void readDescriptions (
    FILE * file,
    EmbPattern * pattern )
```

```
18.44.4.75 readDsb() char readDsb (
    EmbPattern * pattern,
    FILE * file )
```

```
18.44.4.76 readDst() char readDst (
    EmbPattern * pattern,
    FILE * file )
```

```
18.44.4.77 readDsz() char readDsz (
    EmbPattern * pattern,
    FILE * file )
```

18.44.4.78 **ZSK USA Embroidery Format (.dsz)** The ZSK USA dsz format is stitch-only.

```
18.44.4.79 readDxf() char readDxf (
    EmbPattern * pattern,
    FILE * file )
```

```
18.44.4.80 readEdr() char readEdr (
    EmbPattern * pattern,
    FILE * file )
```

18.44.4.81 **Embird Embroidery Format (.edr)** Stitch Only Format

```
18.44.4.82 readEmd() char readEmd (
    EmbPattern * pattern,
    FILE * file )
```

```
18.44.4.83 readExp() char readExp (
    EmbPattern * pattern,
    FILE * file )
```

```
18.44.4.84 readExy() char readExy (
    EmbPattern * pattern,
    FILE * file )
```

```
18.44.4.85 readEys() char readEys (
    EmbPattern * pattern,
    FILE * file )
```

18.44.4.86 Sierra Expanded Embroidery Format (.eys) Stitch Only Format.
Smoothie G-Code Embroidery Format (.fxy)?

18.44.4.87 readFeatherPatterns() void readFeatherPatterns (FILE * file, EmbPattern * pattern)

18.44.4.88 readFullSector() unsigned int readFullSector (FILE * file, bcf_file_difat * bcffile, unsigned int * difatEntriesToRead)
file bcffile difatEntriesToRead

Returns

unsigned int

18.44.4.89 readFxy() char readFxy (EmbPattern * pattern, FILE * file)

18.44.4.90 Embroidery Format (.fxy) Stitch Only Format.

18.44.4.91 readGc() char readGc (EmbPattern * pattern, FILE * file)

Smoothie G-Code

Main Reference: Machinery's Handbook Guide A Guide to Tables, Formulas, & More in the 31st Edition by John Milton Amiss, Franklin D. Jones and Henry Ryffel

18.44.4.92 readGnc() char readGnc (EmbPattern * pattern, FILE * file)

18.44.4.93 Great Notions Embroidery Format (.gnc) Stitch Only Format.

18.44.4.94 readGt() char readGt (EmbPattern * pattern, FILE * file)

18.44.4.95 Gold Thread Embroidery Format (.gt) Stitch Only Format.

18.44.4.96 readHoopName() void readHoopName (FILE * file, EmbPattern * pattern)

18.44.4.97 readHus() char readHus (EmbPattern * pattern, FILE * file)

18.44.4.98 `readImageString()` void readImageString (

```
FILE * file,
EmbPattern * pattern )
```

18.44.4.99 `readInb()` char readInb (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.100 `Inbro Embroidery Format (.inb)` Stitch Only Format.

18.44.4.101 `readInf()` char readInf (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.102 `Embroidery Color Format (.inf)` Stitch Only Format.

18.44.4.103 `readJef()` char readJef (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.104 `readKsm()` char readKsm (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.105 `readMax()` char readMax (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.106 `readMit()` char readMit (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.107 `Mitsubishi Embroidery Format (.mit)` Stitch Only Format.

18.44.4.108 `readMotifPatterns()` void readMotifPatterns (

```
FILE * file,
EmbPattern * pattern )
```

18.44.4.109 `readNew()` char readNew (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.110 `Ameco Embroidery Format (.new)` Stitch Only Format.

18.44.4.111 `readNextSector()` void readNextSector (

```
FILE * file,
bcf_directory * dir )
```

file dir

18.44.4.112 readOfm() char readOfm (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.113 readPcd() char readPcd (

```
EmbPattern * pattern,
const char * fileName,
FILE * file )
```

18.44.4.114 Pfaff PCD File Format (.pcd) Stitch Only Format.

The format uses a signed 3 byte-length number type.

See the description here (5) for the overview of the format.

For an example of the format see (11).

18.44.4.115 readPcm() char readPcm (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.116 Pfaff Embroidery Format (.pcm) The Pfaff pcm format is stitch-only.

18.44.4.117 readPcq() char readPcq (

```
EmbPattern * pattern,
const char * fileName,
FILE * file )
```

18.44.4.118 Embroidery Format (.pcq) The Pfaff pcq format is stitch-only.

18.44.4.119 readPcs() char readPcs (

```
EmbPattern * pattern,
const char * fileName,
FILE * file )
```

18.44.4.120 Embroidery Format (.pcq) The Pfaff pcs format is stitch-only.

18.44.4.121 readPec() char readPec (

```
EmbPattern * pattern,
const char * fileName,
FILE * file )
```

18.44.4.122 readPecStitches() void readPecStitches (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.123 Embroidery Format (.pec) The Brother pec format is stitch-only.

18.44.4.124 readPel() char readPel (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.125 Embroidery Format (.pec) The Brother pel format is stitch-only.

18.44.4.126 `readPem()` `char readPem (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.127 Embroidery Format (.pec) The Brother pem format is stitch-only.

18.44.4.128 `readPes()` `char readPes (`
 `EmbPattern * pattern,`
 `const char * fileName,`
 `FILE * file)`

18.44.4.129 `readPESHeaderV10()` `void readPESHeaderV10 (`
 `FILE * file,`
 `EmbPattern * pattern)`

18.44.4.130 `readPESHeaderV5()` `void readPESHeaderV5 (`
 `FILE * file,`
 `EmbPattern * pattern)`

18.44.4.131 `readPESHeaderV6()` `void readPESHeaderV6 (`
 `FILE * file,`
 `EmbPattern * pattern)`

18.44.4.132 `readPESHeaderV7()` `void readPESHeaderV7 (`
 `FILE * file,`
 `EmbPattern * pattern)`

18.44.4.133 `readPESHeaderV8()` `void readPESHeaderV8 (`
 `FILE * file,`
 `EmbPattern * pattern)`

18.44.4.134 `readPESHeaderV9()` `void readPESHeaderV9 (`
 `FILE * file,`
 `EmbPattern * pattern)`

18.44.4.135 `readPhb()` `char readPhb (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.136 Embroidery Format (.pec) The Brother phb format is stitch-only.

18.44.4.137 `readPhc()` `char readPhc (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.138 Embroidery Format (.pec) The Brother phc format is stitch-only.

18.44.4.139 readPlt() char readPlt (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.140 Embroidery Format (.plt) The AutoCAD plt format is stitch-only.

18.44.4.141 readProgrammableFills() void readProgrammableFills (

```
FILE * file,
EmbPattern * pattern )
```

18.44.4.142 readRgb() char readRgb (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.143 Color File (.rgb) The RGB format is a color-only format to act as an external color file for other formats.

18.44.4.144 readSew() char readSew (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.145 readShv() char readShv (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.146 readSst() char readSst (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.147 Embroidery Format (.sst) The Sunstar sst format is stitch-only.

18.44.4.148 readStx() char readStx (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.149 readSvg() char readSvg (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.150 readT01() char readT01 (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.151 Embroidery Format (.pcq) The Pfaff t01 format is stitch-only.

18.44.4.152 readT09() char readT09 (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.152.1 Embroidery Format (.pcq) The Pfaff t09 format is stitch-only.

18.44.4.153 `readTap()` `char readTap (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.154 `readThr()` `char readThr (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.155 Embroidery Format (.thr) The ThreadWorks thr format is stitch-only.

18.44.4.156 `readThreads()` `void readThreads (`
 `FILE * file,`
 `EmbPattern * pattern)`

18.44.4.157 `readTxt()` `char readTxt (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.158 File (.txt) The txt format is stitch-only and isn't associated with a specific company.

18.44.4.159 `readU00()` `char readU00 (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.160 Embroidery Format (.u00) The Barudan u00 format is stitch-only.

18.44.4.161 `readU01()` `char readU01 (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.162 Embroidery Format (.u00) The Barudan u01 format is stitch-only.

18.44.4.163 `readVip()` `char readVip (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.164 `readVp3()` `char readVp3 (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.165 `readXxx()` `char readXxx (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.166 `readZsk()` `char readZsk (`
 `EmbPattern * pattern,`
 `FILE * file)`

```
18.44.4.167 safe_free() void safe_free (
    void * data )
data
```

```
18.44.4.168 stringInArray() int stringInArray (
    const char * s,
    const char ** array )
```

Tests for the presence of a string *s* in the supplied *array*.
The end of the array is marked by an empty string.

Returns

0 if not present 1 if present.

```
18.44.4.169 testEmbCircle() int testEmbCircle (
    void )
```

```
18.44.4.170 testEmbCircle_2() int testEmbCircle_2 (
    void )
```

```
18.44.4.171 testEmbFormat() int testEmbFormat (
    void )
```

```
18.44.4.172 testGeomArc() int testGeomArc (
    void )
```

```
18.44.4.173 testTangentPoints() void testTangentPoints (
    EmbCircle c,
    EmbVector p,
    EmbVector * t0,
    EmbVector * t1 )
```

```
18.44.4.174 testThreadColor() int testThreadColor (
    void )
```

```
18.44.4.175 write100() char write100 (
    EmbPattern * pattern,
    FILE * file )
```

```
18.44.4.176 write10o() char write10o (
    EmbPattern * pattern,
    FILE * file )
```

18.44.4.177 write_24bit() void write_24bit (FILE * file, int x)
file x

18.44.4.178 writeArt() char writeArt (EmbPattern * pattern, FILE * file)

18.44.4.179 writeBmc() char writeBmc (EmbPattern * pattern, FILE * file)

18.44.4.180 writeBro() char writeBro (EmbPattern * pattern, FILE * file)

18.44.4.181 writeCnd() char writeCnd (EmbPattern * pattern, FILE * file)

18.44.4.182 writeCol() char writeCol (EmbPattern * pattern, FILE * file)

18.44.4.183 writeCsd() char writeCsd (EmbPattern * pattern, FILE * file)

18.44.4.184 writeCsv() char writeCsv (EmbPattern * pattern, FILE * file)

18.44.4.185 writeDat() char writeDat (EmbPattern * pattern, FILE * file)

18.44.4.186 writeDem() char writeDem (EmbPattern * pattern, FILE * file)

18.44.4.187 writeDsb() char writeDsb (EmbPattern * pattern, FILE * file)

18.44.4.188 writeDst() char writeDst (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.189 writeDsz() char writeDsz (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.190 writeDxf() char writeDxf (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.191 writeEdr() char writeEdr (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.192 writeEmd() char writeEmd (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.193 writeExp() char writeExp (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.194 writeExy() char writeExy (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.195 writeEys() char writeEys (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.196 writeFxy() char writeFxy (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.197 writeGc() char writeGc (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.198 writeGnc() char writeGnc (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.199 `writeGt()` `char writeGt (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.200 `writeHus()` `char writeHus (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.201 `writelnb()` `char writeInb (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.202 `writelnf()` `char writeInf (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.203 `writeJef()` `char writeJef (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.204 `writeKsm()` `char writeKsm (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.205 `writeMax()` `char writeMax (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.206 `writeMit()` `char writeMit (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.207 `writeNew()` `char writeNew (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.208 `writeOfm()` `char writeOfm (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.209 `writePcd()` `char writePcd (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.44.4.210 writePcm() char writePcm (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.211 writePcq() char writePcq (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.212 writePcs() char writePcs (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.213 writePec() char writePec (

```
EmbPattern * pattern,
const char * fileName,
FILE * file )
```

18.44.4.214 writePecStitches() void writePecStitches (

```
EmbPattern * pattern,
FILE * file,
const char * filename )
```

18.44.4.215 writePel() char writePel (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.216 writePem() char writePem (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.217 writePes() char writePes (

```
EmbPattern * pattern,
const char * fileName,
FILE * file )
```

18.44.4.218 writePhb() char writePhb (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.219 writePhc() char writePhc (

```
EmbPattern * pattern,
FILE * file )
```

```
18.44.4.220 writePlt() char writePlt (
    EmbPattern * pattern,
    FILE * file )
```

```
18.44.4.221 writeRgb() char writeRgb (
    EmbPattern * pattern,
    FILE * file )
```

```
18.44.4.222 writeSew() char writeSew (
    EmbPattern * pattern,
    FILE * file )
```

```
18.44.4.223 writeShv() char writeShv (
    EmbPattern * pattern,
    FILE * file )
```

```
18.44.4.224 writeSst() char writeSst (
    EmbPattern * pattern,
    FILE * file )
```

```
18.44.4.225 writeStx() char writeStx (
    EmbPattern * pattern,
    FILE * file )
```

```
18.44.4.226 writeSvg() char writeSvg (
    EmbPattern * pattern,
    FILE * file )
```

Writes the data from *pattern* to a file with the given *fileName*. Returns `true` if successful, otherwise returns `false`.

```
18.44.4.227 writeT01() char writeT01 (
    EmbPattern * pattern,
    FILE * file )
```

```
18.44.4.228 writeT09() char writeT09 (
    EmbPattern * pattern,
    FILE * file )
```

```
18.44.4.229 writeTap() char writeTap (
    EmbPattern * pattern,
    FILE * file )
```

```
18.44.4.230 writeThr() char writeThr (
    EmbPattern * pattern,
    FILE * file )
```

18.44.4.231 writeTxt() char writeTxt (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.232 writeU00() char writeU00 (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.233 writeU01() char writeU01 (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.234 writeVip() char writeVip (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.235 writeVp3() char writeVp3 (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.236 writeXxx() char writeXxx (

```
EmbPattern * pattern,
FILE * file )
```

18.44.4.237 writeZsk() char writeZsk (

```
EmbPattern * pattern,
FILE * file )
```

18.44.5 Variable Documentation

18.44.5.1 imageWithFrame const char imageWithFrame[38][48] [extern]

18.45 embroidery_internal.h

[Go to the documentation of this file.](#)

```
00001 #ifndef LIBEMBROIDERY_INTERNAL_HEADER__
00002 #define LIBEMBROIDERY_INTERNAL_HEADER__
00003
00004 #include "embroidery.h"
00005
00010 /* For FILE * */
00011 #include <stdio.h>
00012
00016 #define CompoundFileSector_MaxRegSector 0xFFFFFFF
00017 #define CompoundFileSector_DIFAT_Sector 0xFFFFFFF
00018 #define CompoundFileSector_FAT_Sector 0xFFFFFFF
00019 #define CompoundFileSector_EndofChain 0xFFFFFFF
00020 #define CompoundFileSector_FreeSector 0xFFFFFFF
00021
00025 #define ObjectTypeUnknown 0x00
00026 #define ObjectTypeStorage 0x01
00027 #define ObjectTypeStream 0x02
00028 #define ObjectTypeRootEntry 0x05
00033 #define CompoundFileStreamId_MaxRegularStreamId 0xFFFFFFF
00034 #define CompoundFileStreamId_NoStream 0xFFFFFFF
00036 #define ELEMENT_XML 0
```

```

00037 #define ELEMENT_A 1
00038 #define ELEMENT_ANIMATE 2
00039 #define ELEMENT_ANIMATECOLOR 3
00040 #define ELEMENT_ANIMATEMOTION 4
00041 #define ELEMENT_ANIMATETRANSFORM 5
00042 #define ELEMENT_ANIMATION 6
00043 #define ELEMENT_AUDIO 7
00044 #define ELEMENT_CIRCLE 8
00045 #define ELEMENT_DEFS 9
00046 #define ELEMENT_DESC 10
00047 #define ELEMENT_DISCARD 11
00048 #define ELEMENT_ELLIPSE 12
00049 #define ELEMENT_FONT 13
00050 #define ELEMENT_FONT_FACE 14
00051 #define ELEMENT_FONT_FACE_SRC 15
00052 #define ELEMENT_FONT_FACE_URI 16
00053 #define ELEMENT_FOREIGN_OBJECT 17
00054 #define ELEMENT_G 18
00055 #define ELEMENT_GLYPH 19
00056 #define ELEMENT_HANDLER 20
00057 #define ELEMENT_HKERN 21
00058 #define ELEMENT_IMAGE 22
00059 #define ELEMENT_LINE 23
00060 #define ELEMENT_LINEAR_GRADIENT 24
00061 #define ELEMENT_LISTENER 25
00062 #define ELEMENT_METADATA 26
00063 #define ELEMENT_MISSING_GLYPH 27
00064 #define ELEMENT_MPATH 28
00065 #define ELEMENT_PATH 29
00066 #define ELEMENT_POLYGON 30
00067 #define ELEMENT_POLYLINE 31
00068 #define ELEMENT_PREFETCH 32
00069 #define ELEMENT_RADIAL_GRADIENT 33
00070 #define ELEMENT_RECT 34
00071 #define ELEMENT_SCRIPT 35
00072 #define ELEMENT_SET 36
00073 #define ELEMENT_SOLID_COLOR 37
00074 #define ELEMENT_STOP 38
00075 #define ELEMENT_SVG 39
00076 #define ELEMENT_SWITCH 40
00077 #define ELEMENT_TBREAK 41
00078 #define ELEMENT_TEXT 42
00079 #define ELEMENT_TEXT_AREA 43
00080 #define ELEMENT_TITLE 44
00081 #define ELEMENT_TSPAN 45
00082 #define ELEMENT_USE 46
00083 #define ELEMENT_VIDEO 47
00084
00085 /* INTERNAL DEFINES */
00086 #define RED_TERM_COLOR "\x1B[0;31m"
00087 #define GREEN_TERM_COLOR "\x1B[0;32m"
00088 #define YELLOW_TERM_COLOR "\x1B[1;33m"
00089 #define RESET_TERM_COLOR "\033[0m"
00090
00091 #define HOOP_126X110 0
00092 #define HOOP_110X110 1
00093 #define HOOP_50X50 2
00094 #define HOOP_140X200 3
00095 #define HOOP_230X200 4
00096
00097 #define EMB_MIN(A, B) (((A) < (B)) ? (A) : (B))
00098 #define EMB_MAX(A, B) (((A) > (B)) ? (A) : (B))
00099
00100 /* Libembroidery's handling of integer types.
00101 */
00102 #define EMB_BIG_ENDIAN 0
00103 #define EMB_LITTLE_ENDIAN 1
00104
00105 #define ENDIAN_HOST EMB_LITTLE_ENDIAN
00106
00107 #define EMB_INT16_BIG 2
00108 #define EMB_INT16_LITTLE 3
00109 #define EMB_INT32_BIG 4
00110 #define EMB_INT32_LITTLE 5
00111
00112 #define PES0001 0
00113 #define PES0020 1
00114 #define PES0022 2
00115 #define PES0030 3
00116 #define PES0040 4
00117 #define PES0050 5
00118 #define PES0055 6
00119 #define PES0056 7
00120 #define PES0060 8
00121 #define PES0070 9
00122 #define PES0080 10
00123 #define PES0090 11

```

```
00124 #define PES0100          12
00125 #define N_PES_VERSIONS 13
00126
00127 /* DXF Version Identifiers */
00128 #define DXF_VERSION_R10 "AC1006"
00129 #define DXF_VERSION_R11 "AC1009"
00130 #define DXF_VERSION_R12 "AC1009"
00131 #define DXF_VERSION_R13 "AC1012"
00132 #define DXF_VERSION_R14 "AC1014"
00133 #define DXF_VERSION_R15 "AC1015"
00134 #define DXF_VERSION_R18 "AC1018"
00135 #define DXF_VERSION_R21 "AC1021"
00136 #define DXF_VERSION_R24 "AC1024"
00137 #define DXF_VERSION_R27 "AC1027"
00138
00139 #define DXF_VERSION_2000 "AC1015"
00140 #define DXF_VERSION_2002 "AC1015"
00141 #define DXF_VERSION_2004 "AC1018"
00142 #define DXF_VERSION_2006 "AC1018"
00143 #define DXF_VERSION_2007 "AC1021"
00144 #define DXF_VERSION_2009 "AC1021"
00145 #define DXF_VERSION_2010 "AC1024"
00146 #define DXF_VERSION_2013 "AC1027"
00147
00148 #define SVG_CREATOR_NULL          0
00149 #define SVG_CREATOR_EMBROIDERMODDER 1
00150 #define SVG_CREATOR_ILLUSTRATOR    2
00151 #define SVG_CREATOR_INKSCAPE      3
00152
00153 #define SVG_EXPECT_NULL          0
00154 #define SVG_EXPECT_ELEMENT       1
00155 #define SVG_EXPECT_ATTRIBUTE     2
00156 #define SVG_EXPECT_VALUE        3
00157
00158 /* SVG_TYPES
00159 * -----
00160 */
00161 #define SVG_NULL          0
00162 #define SVG_ELEMENT       1
00163 #define SVG_PROPERTY      2
00164 #define SVG_MEDIA_PROPERTY 3
00165 #define SVG_ATTRIBUTE     4
00166 #define SVG_CATCH_ALL      5
00167
00168 /* path flag codes */
00169 #define LINETO          0
00170 #define MOVETO         1
00171 #define BULGETOCONTROL 2
00172 #define BULGETOEND     4
00173 #define ELLIPSETORAD   8
00174 #define ELLIPSETOEND   16
00175 #define CUBICTOCONTROL1 32
00176 #define CUBICTOCONTROL2 64
00177 #define CUBICTOEND     128
00178 #define QUADTOCONTROL 256
00179 #define QUADTOEND      512
00180
00181 /* STRUCTS
00182 *****/
00183
00184 /* double-indirection file allocation table references */
00185
00186 typedef struct _bcf_file_difat
00187 {
00188     unsigned int fatSectorCount;
00189     unsigned int fatSectorEntries[109];
00190     unsigned int sectorSize;
00191 } bcf_file_difat;
00192
00193 typedef struct _bcf_file_fat
00194 {
00195     int           fatEntryCount;
00196     unsigned int fatEntries[255]; /* maybe make this dynamic */
00197     unsigned int numberOfEntriesInFatSector;
00198 } bcf_file_fat;
00199
00200 typedef struct _bcf_directory_entry
00201 {
00202     char          directoryEntryName[32];
00203     unsigned short directoryEntryNameLength;
00204     objectType;
00205     colorFlag;
00206     leftSiblingId;
00207     rightSiblingId;
00208     childId;
00209     CLSID[16];
00210     stateBits;
```

```

00223     EmbTime           creationTime;
00224     EmbTime           modifiedTime;
00225     unsigned int      startingSectorLocation;
00226     unsigned long      streamSize; /* should be long long but in our case we shouldn't need
00227     it, and hard to support on c89 cross platform */
00228     unsigned int      streamSizeHigh; /* store the high int of streamsize */
00229 } bcf_directory_entry;
00230
00231 typedef struct _bcf_directory
00232 {
00233     bcf_directory_entry* dirEntries;
00234     unsigned int      maxNumberOfDirectoryEntries;
00235 } bcf_directory;
00236
00237 typedef struct _bcf_file_header
00238 {
00239     unsigned char    signature[8];
00240     unsigned char    CLSID[16];
00241     unsigned short   minorVersion;
00242     unsigned short   majorVersion;
00243     unsigned short   byteOrder;
00244     unsigned short   sectorShift;
00245     unsigned short   miniSectorShift;
00246     unsigned short   reserved1;
00247     unsigned int     reserved2;
00248     unsigned int     numberofDirectorySectors;
00249     unsigned int     numberofFATSectors;
00250     unsigned int     firstDirectorySectorLocation;
00251     unsigned int     transactionSignatureNumber;
00252     unsigned int     miniStreamCutoffsize;
00253     unsigned int     firstMiniFATSectorLocation;
00254     unsigned int     numberofMiniFatSectors;
00255     unsigned int     firstDifatSectorLocation;
00256     unsigned int     numberofDifatSectors;
00257 } bcf_file_header;
00258
00259 typedef struct _bcf_file
00260 {
00261     bcf_file_header header;
00262     bcf_file_difat* difat;
00263     bcf_file_fat* fat;
00264     bcf_directory* directory;
00265 } bcf_file;
00266
00267 typedef struct _vp3Hoop
00268 {
00269     int right;
00270     int bottom;
00271     int left;
00272     int top;
00273     int threadLength;
00274     char unknown2;
00275     unsigned char numberOfColors;
00276     unsigned short unknown3;
00277     int unknown4;
00278     int numberOfBytesRemaining;
00279
00280     int xOffset;
00281     int yOffset;
00282
00283     unsigned char byte1;
00284     unsigned char byte2;
00285     unsigned char byte3;
00286
00287     /* Centered hoop dimensions */
00288     int right2;
00289     int left2;
00290     int bottom2;
00291     int top2;
00292
00293     int width;
00294     int height;
00295 } vp3Hoop;
00296
00297 typedef struct ThredHeader_ /* thred file header */
00298 {
00299     unsigned int sigVersion; /* signature and version */
00300     unsigned int length; /* length of ThredHeader + length of stitch data */
00301     unsigned short numStitches; /* number of stitches */
00302     unsigned short hoopSize; /* size of hoop */
00303     unsigned short reserved[7]; /* reserved for expansion */
00304 } ThredHeader;
00305
00306 typedef struct ThredExtension_ /* thred v1.0 file header extension */
00307 {
00308     float hoopX; /* hoop size x dimension in 1/6 mm units */
00309
00310
00311
00312
00313
00314
00315
00316
00317
00318
00319
00320
00321
00322
00323
00324
00325
00326
00327
00328
00329
00330
00331
00332
00333

```

```

00334     float hoopY;           /* hoop size y dimension in 1/6 mm units */
00335     float stitchGranularity; /* stitches per millimeter--not implemented */
00336     char creatorName[50];   /* name of the file creator */
00337     char modifierName[50];  /* name of last file modifier */
00338     char auxFormat;        /* auxiliary file format, 0=PCS,1=DST,2=PES */
00339     char reserved[31];     /* reserved for expansion */
00340 } ThredExtension;
00341
00342 typedef struct SubDescriptor_
00343 {
00344     int someNum;
00345     int someInt;
00346     int someOtherInt;
00347     char* colorCode;
00348     char* colorName;
00349 } SubDescriptor;
00350
00351 typedef struct StxThread_
00352 {
00353     char* colorCode;
00354     char* colorName;
00355     char* sectionName;
00356     SubDescriptor* subDescriptors;
00357     EmbColor stxColor;
00358 } StxThread;
00359
00360 typedef struct VipHeader_ {
00361     int magicCode;
00362     int numberofStitches;
00363     int numberofColors;
00364     short positiveXHoopSize;
00365     short positiveYHoopSize;
00366     short negativeXHoopSize;
00367     short negativeYHoopSize;
00368     int attributeOffset;
00369     int xOffset;
00370     int yOffset;
00371     unsigned char stringVal[8];
00372     short unknown;
00373     int colorLength;
00374 } VipHeader;
00375
00376 typedef enum
00377 {
00378     CSV_EXPECT_NULL,
00379     CSV_EXPECT_QUOTE1,
00380     CSV_EXPECT_QUOTE2,
00381     CSV_EXPECT_COMMA
00382 } CSV_EXPECT;
00383
00384 typedef enum
00385 {
00386     CSV_MODE_NULL,
00387     CSV_MODE_COMMENT,
00388     CSV_MODE_VARIABLE,
00389     CSV_MODE_THREAD,
00390     CSV_MODE_STITCH
00391 } CSV_MODE;
00392
00393 typedef struct SvgAttribute_
00394 {
00395     char* name;
00396     char* value;
00397 } SvgAttribute;
00398
00399 typedef struct Huffman {
00400     int default_value;
00401     int lengths[1000];
00402     int nlenghts;
00403     int table[1000];
00404     int table_width;
00405     int ntable;
00406 } huffman;
00407
00408 typedef struct Compress {
00409     int bit_position;
00410     char *input_data;
00411     int input_length;
00412     int bits_total;
00413     int block_elements;
00414     huffman character_length_huffman;
00415     huffman character_huffman;
00416     huffman distance_huffman;
00417 } compress;
00418
00419 /* Function Declarations
00420 *****/

```

```

00453 void huffman_build_table(huffman *h);
00454 int *huffman_table_lookup(huffman *h, int byte_lookup, int *lengths);
00455
00456 int compress_get_bits(compress *c, int length);
00457 int compress_pop(compress *c, int bit_count);
00458 int compress_read_variable_length(compress *c);
00459 void compress_load_character_length_huffman(compress *c);
00460 void compress_load_character_huffman(compress *c);
00461 void compress_load_distance_huffman(compress *c);
00462 void compress_load_block(compress *c);
00463 int compress_get_token(compress *c);
00464 int compress_get_position(compress *c);
00465
00466 void readPecStitches(EmbPattern* pattern, FILE* file);
00467 void writePecStitches(EmbPattern* pattern, FILE* file, const char* filename);
00468
00469 int decodeNewStitch(unsigned char value);
00470
00471 void pfaffEncode(FILE* file, int x, int y, int flags);
00472 EmbReal pfaffDecode(unsigned char a1, unsigned char a2, unsigned char a3);
00473
00474 unsigned char mitEncodeStitch(EmbReal value);
00475 int mitDecodeStitch(unsigned char value);
00476
00477 int encode_tajima_ternary(unsigned char b[3], int x, int y);
00478 void decode_tajima_ternary(unsigned char b[3], int *x, int *y);
00479
00480 void encode_t01_record(unsigned char b[3], int x, int y, int flags);
00481 int decode_t01_record(unsigned char b[3], int *x, int *y, int *flags);
00482 void readPESHeaderV5(FILE* file, EmbPattern* pattern);
00483 void readPESHeaderV6(FILE* file, EmbPattern* pattern);
00484 void readPESHeaderV7(FILE* file, EmbPattern* pattern);
00485 void readPESHeaderV8(FILE* file, EmbPattern* pattern);
00486 void readPESHeaderV9(FILE* file, EmbPattern* pattern);
00487 void readPESHeaderV10(FILE* file, EmbPattern* pattern);
00488
00489 void readDescriptions(FILE* file, EmbPattern* pattern);
00490 void readHoopName(FILE* file, EmbPattern* pattern);
00491 void readImageString(FILE* file, EmbPattern* pattern);
00492 void readProgrammableFills(FILE* file, EmbPattern* pattern);
00493 void readMotifPatterns(FILE* file, EmbPattern* pattern);
00494 void readFeatherPatterns(FILE* file, EmbPattern* pattern);
00495 void readThreads(FILE* file, EmbPattern* pattern);
00496
00497 void embInt_read(FILE* f, char *label, void *b, int mode);
00498 void embInt_write(FILE* f, char *label, void *b, int mode);
00499 int emb_readline(FILE* file, char *line, int maxLength);
00500
00501 int bcfFile_read(FILE* file, bcf_file* bcfFile);
00502 FILE* GetFile(bcf_file* bcfFile, FILE* file, char* fileToFind);
00503 void bcf_file_free(bcf_file* bcfFile);
00504
00505 void binaryReadString(FILE* file, char *buffer, int maxLength);
00506 void binaryReadUnicodeString(FILE* file, char *buffer, const int stringLength);
00507
00508 int stringInArray(const char *s, const char **array);
00509 void fpad(FILE *f, char c, int n);
00510 char *copy_trim(char const *s);
00511 char* emb_optOut(EmbReal num, char* str);
00512
00513 void write_24bit(FILE* file, int);
00514 int check_header_present(FILE* file, int minimum_header_length);
00515
00516 unsigned short fread_uint16(FILE *file);
00517 short fread_int16(FILE* f);
00518 int fread_int32_be(FILE* f);
00519 void safe_free(void *data);
00520 void embInt_read(FILE* f, char *label, void *b, int mode);
00521
00522 void binaryWriteUIntBE(FILE* f, unsigned int data);
00523 void binaryWriteUInt(FILE* f, unsigned int data);
00524 void binaryWriteIntBE(FILE* f, int data);
00525 void binaryWriteInt(FILE* f, int data);
00526 void binaryWriteUShort(FILE* f, unsigned short data);
00527 void binaryWriteUShortBE(FILE* f, unsigned short data);
00528 void binaryWriteShort(FILE* f, short data);
00529
00530 bcf_file_difat* bcf_difat_create(FILE* file, unsigned int fatSectors, const unsigned int sectorSize);
00531 unsigned int readFullSector(FILE* file, bcf_file_difat* bcfFile, unsigned int* numberDifatEntriesStillToRead);
00532 unsigned int numberOFEntriesInDifatSector(bcf_file_difat* fat);
00533 void bcf_file_difat_free(bcf_file_difat* difat);
00534
00535 unsigned int entriesInDifatSector(bcf_file_difat* fat);
00536 bcf_file_fat* bcfFileFat_create(const unsigned int sectorSize);
00537 void loadFatFromSector(bcf_file_fat* fat, FILE* file);
00538 void bcf_file_fat_free(bcf_file_fat** fat);

```

```
00539
00540 bcf_directory_entry* CompoundFileDirectoryEntry(FILE* file);
00541 bcf_directory* CompoundFileDirectory(const unsigned int maxNumberOfDirectoryEntries);
00542 void readNextSector(FILE* file, bcf_directory* dir);
00543 void bcf_directory_free(bcf_directory** dir);
00544
00545 bcf_file_header bcfFileHeader_read(FILE* file);
00546 int bcfFileHeader_isValid(bcf_file_header header);
00547
00548 int hus_compress(char* input, int size, char* output, int *out_size);
00549 int hus_decompress(char* input, int size, char* output, int *out_size);
00550
00551 int encode_tajima_ternary(unsigned char b[3], int x, int y);
00552 void decode_tajima_ternary(unsigned char b[3], int *x, int *y);
00553 void testTangentPoints(EmbCircle c, EmbVector p, EmbVector *t0, EmbVector *t1);
00554 void printArcResults(EmbReal bulge, EmbArc arc,
00555                 EmbReal centerX, EmbReal centerY,
00556                 EmbReal radius, EmbReal diameter,
00557                 EmbReal chord,
00558                 EmbReal chordMidX, EmbReal chordMidY,
00559                 EmbReal sagitta, EmbReal apothem,
00560                 EmbReal incAngle, char clockwise);
00561 int create_test_file_1(const char* outf);
00562 int create_test_file_2(const char* outf);
00563 int create_test_file_3(const char* outf);
00564 int testEmbCircle(void);
00565 int testEmbCircle_2(void);
00566 int testGeomArc(void);
00567 int testThreadColor(void);
00568 int testEmbFormat(void);
00569
00570 void embColor_read(FILE *f, EmbColor *c, int toRead);
00571 void embColor_write(FILE *f, EmbColor c, int toWrite);
00572
00573 char read100(EmbPattern *pattern, FILE* file);
00574 char write100(EmbPattern *pattern, FILE* file);
00575 char read10o(EmbPattern *pattern, FILE* file);
00576 char write10o(EmbPattern *pattern, FILE* file);
00577 char readArt(EmbPattern *pattern, FILE* file);
00578 char writeArt(EmbPattern *pattern, FILE* file);
00579 char readBmc(EmbPattern *pattern, FILE* file);
00580 char writeBmc(EmbPattern *pattern, FILE* file);
00581 char readBro(EmbPattern *pattern, FILE* file);
00582 char writeBro(EmbPattern *pattern, FILE* file);
00583 char readCnd(EmbPattern *pattern, FILE* file);
00584 char writeCnd(EmbPattern *pattern, FILE* file);
00585 char readCol(EmbPattern *pattern, FILE* file);
00586 char writeCol(EmbPattern *pattern, FILE* file);
00587 char readCsd(EmbPattern *pattern, FILE* file);
00588 char writeCsd(EmbPattern *pattern, FILE* file);
00589 char readCsv(EmbPattern *pattern, FILE* file);
00590 char writeCsv(EmbPattern *pattern, FILE* file);
00591 char readDat(EmbPattern *pattern, FILE* file);
00592 char writeDat(EmbPattern *pattern, FILE* file);
00593 char readDem(EmbPattern *pattern, FILE* file);
00594 char writeDem(EmbPattern *pattern, FILE* file);
00595 char readDsb(EmbPattern *pattern, FILE* file);
00596 char writeDsb(EmbPattern *pattern, FILE* file);
00597 char readDst(EmbPattern *pattern, FILE* file);
00598 char writeDst(EmbPattern *pattern, FILE* file);
00599 char readDsz(EmbPattern *pattern, FILE* file);
00600 char writeDsz(EmbPattern *pattern, FILE* file);
00601 char readDxf(EmbPattern *pattern, FILE* file);
00602 char writeDxf(EmbPattern *pattern, FILE* file);
00603 char readEdr(EmbPattern *pattern, FILE* file);
00604 char writeEdr(EmbPattern *pattern, FILE* file);
00605 char readEmd(EmbPattern *pattern, FILE* file);
00606 char writeEmd(EmbPattern *pattern, FILE* file);
00607 char readExp(EmbPattern *pattern, FILE* file);
00608 char writeExp(EmbPattern *pattern, FILE* file);
00609 char readExy(EmbPattern *pattern, FILE* file);
00610 char writeExy(EmbPattern *pattern, FILE* file);
00611 char readEys(EmbPattern *pattern, FILE* file);
00612 char writeEys(EmbPattern *pattern, FILE* file);
00613 char readFxy(EmbPattern *pattern, FILE* file);
00614 char writeFxy(EmbPattern *pattern, FILE* file);
00615 char readGc(EmbPattern *pattern, FILE* file);
00616 char writeGc(EmbPattern *pattern, FILE* file);
00617 char readGnc(EmbPattern *pattern, FILE* file);
00618 char writeGnc(EmbPattern *pattern, FILE* file);
00619 char readGt(EmbPattern *pattern, FILE* file);
00620 char writeGt(EmbPattern *pattern, FILE* file);
00621 char readHus(EmbPattern *pattern, FILE* file);
00622 char writeHus(EmbPattern *pattern, FILE* file);
00623 char readInb(EmbPattern *pattern, FILE* file);
00624 char writeInb(EmbPattern *pattern, FILE* file);
00625 char readInf(EmbPattern *pattern, FILE* file);
```

```

00626 char writeInf(EmbPattern *pattern, FILE* file);
00627 char readJef(EmbPattern *pattern, FILE* file);
00628 char writeJef(EmbPattern *pattern, FILE* file);
00629 char readKsm(EmbPattern *pattern, FILE* file);
00630 char writeKsm(EmbPattern *pattern, FILE* file);
00631 char readMax(EmbPattern *pattern, FILE* file);
00632 char writeMax(EmbPattern *pattern, FILE* file);
00633 char readMit(EmbPattern *pattern, FILE* file);
00634 char writeMit(EmbPattern *pattern, FILE* file);
00635 char readNew(EmbPattern *pattern, FILE* file);
00636 char writeNew(EmbPattern *pattern, FILE* file);
00637 char readOfm(EmbPattern *pattern, FILE* file);
00638 char writeOfm(EmbPattern *pattern, FILE* file);
00639 char readPcd(EmbPattern *pattern, const char *fileName, FILE* file);
00640 char writePcd(EmbPattern *pattern, FILE* file);
00641 char readPcm(EmbPattern *pattern, FILE* file);
00642 char writePcm(EmbPattern *pattern, FILE* file);
00643 char readPcq(EmbPattern *pattern, const char *fileName, FILE* file);
00644 char writePcq(EmbPattern *pattern, FILE* file);
00645 char readPcs(EmbPattern *pattern, const char *fileName, FILE* file);
00646 char writePcs(EmbPattern *pattern, FILE* file);
00647 char readPec(EmbPattern *pattern, const char *fileName, FILE* file);
00648 char writePec(EmbPattern *pattern, const char *fileName, FILE* file);
00649 char readPel(EmbPattern *pattern, FILE *file);
00650 char writePel(EmbPattern *pattern, FILE *file);
00651 char readPem(EmbPattern *pattern, FILE *file);
00652 char writePem(EmbPattern *pattern, FILE *file);
00653 char readPes(EmbPattern *pattern, const char *fileName, FILE* file);
00654 char writePes(EmbPattern *pattern, const char *fileName, FILE* file);
00655 char readPhb(EmbPattern *pattern, FILE* file);
00656 char writePhb(EmbPattern *pattern, FILE *file);
00657 char readPhc(EmbPattern *pattern, FILE* file);
00658 char writePhc(EmbPattern *pattern, FILE *file);
00659 char readPlt(EmbPattern *pattern, FILE* file);
00660 char writePlt(EmbPattern *pattern, FILE* file);
00661 char readRgb(EmbPattern *pattern, FILE* file);
00662 char writeRgb(EmbPattern *pattern, FILE* file);
00663 char readSew(EmbPattern *pattern, FILE* file);
00664 char writeSew(EmbPattern *pattern, FILE* file);
00665 char readShv(EmbPattern *pattern, FILE* file);
00666 char writeShv(EmbPattern *pattern, FILE *file);
00667 char readSst(EmbPattern *pattern, FILE* file);
00668 char writeSst(EmbPattern *pattern, FILE *file);
00669 char readStx(EmbPattern *pattern, FILE* file);
00670 char writeStx(EmbPattern *pattern, FILE *file);
00671 char readSvg(EmbPattern *pattern, FILE* file);
00672 char writeSvg(EmbPattern *pattern, FILE* file);
00673 char readT01(EmbPattern *pattern, FILE* file);
00674 char writeT01(EmbPattern *pattern, FILE* file);
00675 char readT09(EmbPattern *pattern, FILE* file);
00676 char writeT09(EmbPattern *pattern, FILE* file);
00677 char readTap(EmbPattern *pattern, FILE* file);
00678 char writeTap(EmbPattern *pattern, FILE* file);
00679 char readThr(EmbPattern *pattern, FILE* file);
00680 char writeThr(EmbPattern *pattern, FILE* file);
00681 char readTxt(EmbPattern *pattern, FILE* file);
00682 char writeTxt(EmbPattern *pattern, FILE* file);
00683 char readU00(EmbPattern *pattern, FILE* file);
00684 char writeU00(EmbPattern *pattern, FILE *file);
00685 char readU01(EmbPattern *pattern, FILE* file);
00686 char writeU01(EmbPattern *pattern, FILE *file);
00687 char readVip(EmbPattern *pattern, FILE* file);
00688 char writeVip(EmbPattern *pattern, FILE* file);
00689 char readVp3(EmbPattern *pattern, FILE* file);
00690 char writeVp3(EmbPattern *pattern, FILE* file);
00691 char readXxx(EmbPattern *pattern, FILE* file);
00692 char writeXxx(EmbPattern *pattern, FILE* file);
00693 char readZsk(EmbPattern *pattern, FILE* file);
00694 char writeZsk(EmbPattern *pattern, FILE* file);
00695
00696 extern const char imageWithFrame[38][48];
00697
00698 #endif

```

18.46 extern/libembroidery/src/encoding.c File Reference

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include "embroidery_internal.h"

```

Functions

- void `write_24bit` (FILE *file, int
`file` *x*)
- `EmbColor embColor_fromHexStr` (char *val)
Converts a 6 digit hex string (I.E. "00FF00") into an EmbColor and returns it.
- void `reverse_byte_order` (void *b, int bytes)
- int `decode_t01_record` (unsigned char b[3], int *x, int *y, int *flags)
- void `encode_t01_record` (unsigned char b[3], int x, int y, int flags)
- int `encode_tajima_ternary` (unsigned char b[3], int x, int y)
- void `decode_tajima_ternary` (unsigned char b[3], int *x, int *y)
- void `pfaffEncode` (FILE *file, int dx, int dy, int flags)
- `EmbReal pfaffDecode` (unsigned char a1, unsigned char a2, unsigned char a3)
- unsigned char `mitEncodeStitch` (`EmbReal` value)
`value`
- int `mitDecodeStitch` (unsigned char value)
`value`
- int `decodeNewStitch` (unsigned char value)
`value`
- void `emblnt_read` (FILE *f, char *label, void *b, int mode)
- void `emblnt_write` (FILE *f, char *label, void *b, int mode)

18.46.1 Detailed Description

The functions in this file are grouped together to aid the developer's understanding of the similarities between the file formats. This also helps reduce errors between reimplementation of the same idea.

For example: the Tajima ternary encoding of positions is used by at least 4 formats and the only part that changes is the flag encoding.

18.46.2 Function Documentation

18.46.2.1 decode_t01_record() int decode_t01_record (
 unsigned char b[3],
 int * x,
 int * y,
 int * flags)

b x y flags .

Todo remove the unused return argument.

18.46.2.2 decode_tajima_ternary() void decode_tajima_ternary (
 unsigned char b[3],
 int * x,
 int * y)

Decode the signed ternary of the tajima format from *b* to the position values *x* and *y*.
There is no return argument.

18.46.2.3 decodeNewStitch() int decodeNewStitch (
 unsigned char value)
`value`
Returns
 int

18.46.2.4 embColor_fromHexStr() `EmbColor` `embColor_fromHexStr (`
 `char * val)`

Converts a 6 digit hex string (I.E. "00FF00") into an EmbColor and returns it.
val/6 byte code describing the color as a hex string, doesn't require null termination.

Returns

`EmbColor` the same color as our internal type.

18.46.2.5 embInt_read() `void` `embInt_read (`
 `FILE * f,`
 `char * label,`
 `void * b,`
 `int mode)`

f label b mode

Read and write system for multiple byte types.

The caller passes the function to read/write from, the memory location as a void pointer and a mode identifier that describes the type. This way we can abstract out the endianness of the system running the library and don't have to maintain many functions, just two.

18.46.2.6 embInt_write() `void` `embInt_write (`
 `FILE * f,`
 `char * label,`
 `void * b,`
 `int mode)`

f label b mode

18.46.2.7 encode_t01_record() `void` `encode_t01_record (`
 `unsigned char b[3],`
 `int x,`
 `int y,`
 `int flags)`

Encode into bytes *b* the values of the x-position *x*, y-position *y* and the *flags*.

18.46.2.8 encode_tajima_ternary() `int` `encode_tajima_ternary (`
 `unsigned char b[3],`
 `int x,`
 `int y)`

Encode the signed ternary of the tajima format into *b* the position values *x* and *y*.

If the values of *x* or *y* fall outside of the valid range of -121 and +121 then it returns 0 and 1.

18.46.2.9 mitDecodeStitch() `int` `mitDecodeStitch (`
 `unsigned char value)`

value

Returns

`int`

18.46.2.10 mitEncodeStitch() `unsigned char` `mitEncodeStitch (`
 `EmbReal value)`

value

Returns

`unsigned char`

```
18.46.2.11 pfaffDecode() EmbReal pfaffDecode (
    unsigned char a1,
    unsigned char a2,
    unsigned char a3 )
```

Decode the bytes *a1*, *a2* and *a3*. Returns the EmbReal floating-point value.

```
18.46.2.12 pfaffEncode() void pfaffEncode (
    FILE * file,
    int dx,
    int dy,
    int flags )
```

file dx dy flags

```
18.46.2.13 reverse_byte_order() void reverse_byte_order (
    void * b,
    int bytes )
```

Reverses the byte order of *bytes* number of bytes at memory location *b*. Only works for 2 or 4 byte arrays.

```
18.46.2.14 write_24bit() void write_24bit (
    FILE * file,
    int x )
```

file x

18.47 extern/libembroidery/src/fill.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include "embroidery_internal.h"
```

Functions

- int `lindenmayer_system (L_system L, char *state, int iterations, int complete)`
- static void `join_short_stitches (int *points, int *n_points, int width, int tolerance)`
- static int * `threshold_method (EmblImage *image, int *n_points, int subsample_width, int subsample_height, int threshold)`
- static void `greedy_algorithm (int *points, int n_points, int width, EmbReal bias)`
- static void `save_points_to_pattern (EmbPattern *pattern, int *points, int n_points, EmbReal scale, int width, int height)`
- void `embPattern_horizontal_fill (EmbPattern *pattern, EmblImage *image, int threshhold)`
- void `embPattern_crossstitch (EmbPattern *pattern, EmblImage *image, int threshhold)`
- int `hilbert_curve (EmbPattern *pattern, int iterations)`
- void `generate_dragon_curve (char *state, int iterations)`
- int `dragon_curve (int iterations)`
- void `embPolygon_reduceByDistance (EmbArray *vertices, EmbArray *simplified, float distance)`
- void `embPolygon_reduceByNth (EmbArray *vertices, EmbArray *out, int nth)`
- `EmbPattern * embPattern_combine (EmbPattern *p1, EmbPattern *p2)`
- void `embPattern_stitchArc (EmbPattern *p, EmbArc arc, int thread_index, int style)`
- void `embPattern_stitchCircle (EmbPattern *p, EmbCircle circle, int thread_index, int style)`
- void `embPattern_stitchEllipse (EmbPattern *p, EmbEllipse ellipse, int thread_index, int style)`
- void `embPattern_stitchPath (EmbPattern *p, EmbPath path, int thread_index, int style)`
- void `embPattern_stitchPolygon (EmbPattern *p, EmbPolygon polygon, int thread_index, int style)`
- void `embPattern_stitchPolyline (EmbPattern *p, EmbPolyline polyline, int thread_index, int style)`
- void `embPattern_stitchRect (EmbPattern *p, EmbRect rect, int thread_index, int style)`
- void `embPattern_stitchText (EmbPattern *p, EmbRect rect, int thread_index, int style)`
- void `embPattern_convertGeometry (EmbPattern *p)`

Variables

- const char * **rules** [] = {"+BF-AFA-FB+", "-AF+BFB+FA-"}
• **L_system hilbert_curve_l_system**

18.47.1 Function Documentation

18.47.1.1 dragon_curve() int dragon_curve (

int iterations)

Create the dragon curve for *iterations*.

Returns 0 if the number of iterations is greater than 10 and 1 otherwise.

18.47.1.2 embPattern_combine() **EmbPattern** * embPattern_combine (

EmbPattern * *p1*,

EmbPattern * *p2*)

p1 p2

Returns

EmbPattern*

18.47.1.3 embPattern_convertGeometry() void embPattern_convertGeometry (

EmbPattern * *p*)

p

18.47.1.4 embPattern_crossstitch() void embPattern_crossstitch (

EmbPattern * *pattern*,

EmbImage * *image*,

int *threshold*)

pattern image threshold

Uses a *threshold* method to determine where to put crosses in the fill.

To improve this, we can remove the vertical stitches when two crosses neighbour. Currently the simple way to do this is to chain crosses that are neighbours exactly one ahead.

18.47.1.5 embPattern_horizontal_fill() void embPattern_horizontal_fill (

EmbPattern * *pattern*,

EmbImage * *image*,

int *threshold*)

pattern image threshold

Uses a *threshold* method to determine where to put lines in the fill.

Needs to pass a "donut test", i.e. an image with black pixels where: $10 < x*x + y*y < 20$ over the area (-30, 30) x (-30, 30).

Use render then image difference to see how well it passes.

18.47.1.6 embPattern_stitchArc() void embPattern_stitchArc (

EmbPattern * *p*,

EmbArc *arc*,

int *thread_index*,

int *style*)

p arc thread_index style

18.47.1.7 embPattern_stitchCircle() void embPattern_stitchCircle (

```
EmbPattern * p,
EmbCircle circle,
int thread_index,
int style )
```

p circle thread_index style

style determines: stitch density fill pattern outline or fill

For now it's a straight fill of 1000 stitches of the whole object by default.

Consider the intersection of a line in direction "d" that passes through the disc with center "c", radius "r". The start and end points are:

$$(\mathbf{c} - \mathbf{r}(d/|d|), \mathbf{c} + \mathbf{r}(d/|d|))$$

Lines that are above and below this with an even separation $\$s\$$ can be found by taking the point on the line to be $\mathbf{c} + s\mathbf{n}$ where the \mathbf{n} is the unit normal vector to \mathbf{d} and the vector to be \mathbf{d} again. The intersection points are therefore a right angled triangle, with one side r , another s and the third the length to be solved, by Pythagoras we have:

$$(\mathbf{c} + s\mathbf{n} - \sqrt{r^2 - s^2}(d/|d|), \mathbf{c} + s\mathbf{n} + \sqrt{r^2 - s^2}(d/|d|))$$

repeating this process gives us all the end points and the fill only alters these lines by splitting the ones longer than some tolerance.

18.47.1.8 embPattern_stitchEllipse() void embPattern_stitchEllipse (

```
EmbPattern * p,
EmbEllipse ellipse,
int thread_index,
int style )
```

p ellipse thread_index style

Todo finish stitchEllipse

18.47.1.9 embPattern_stitchPath() void embPattern_stitchPath (

```
EmbPattern * p,
EmbPath path,
int thread_index,
int style )
```

p rect thread_index style

Todo finish stitch path

18.47.1.10 embPattern_stitchPolygon() void embPattern_stitchPolygon (

```
EmbPattern * p,
EmbPolygon polygon,
int thread_index,
int style )
```

p rect thread_index style

Todo finish stitch polygon

18.47.1.11 embPattern_stitchPolyline() void embPattern_stitchPolyline (

```
EmbPattern * p,
EmbPolyline polyline,
int thread_index,
int style )
```

p rect thread_index style

Todo finish stitch polyline

18.47.1.12 `embPattern_stitchRect()` `void embPattern_stitchRect (`

```
    EmbPattern * p,
    EmbRect rect,
    int thread_index,
    int style )
```

p rect thread_index style

Here we just stitch the rectangle in the direction of it's longer side.

18.47.1.13 `embPattern_stitchText()` `void embPattern_stitchText (`

```
    EmbPattern * p,
    EmbRect rect,
    int thread_index,
    int style )
```

p rect thread_index style

18.47.1.14 `embPolygon_reduceByDistance()` `void embPolygon_reduceByDistance (`

```
    EmbArray * vertices,
    EmbArray * simplified,
    float distance )
```

vertices simplified distance

Reduces the polygon by distance.

This is a non-destructive function, so the caller is responsible for freeing "vertices" if they choose to keep "simplified".

18.47.1.15 `embPolygon_reduceByNth()` `void embPolygon_reduceByNth (`

```
    EmbArray * vertices,
    EmbArray * out,
    int nth )
```

vertices out nth

Reduces the polygon by removing the Nth vertex in the vertices list. This is a non-destructive function, so the caller is responsible for freeing vertices if they choose to keep out.

18.47.1.16 `generate_dragon_curve()` `void generate_dragon_curve (`

```
    char * state,
    int iterations )
```

state iterations

using the "paper folding" method

Todo find citation for paper folding method

18.47.1.17 `greedy_algorithm()` `static void greedy_algorithm (`

```
    int * points,
    int n_points,
    int width,
    EmbReal bias ) [static]
```

points n_points width bias

18.47.1.18 Greedy Algorithm For each point in the list find the shortest distance to any possible neighbour, then perform a swap to make that neighbour the next item in the list.

To make the stitches lie more on one axis than the other bias the distance operator to prefer horizontal direction.

18.47.1.19 hilbert_curve() `int hilbert_curve (`
`EmbPattern * pattern,`
`int iterations)`
pattern iterations
https://en.wikipedia.org/wiki/Hilbert_curve

Using the Lindenmayer System, so we can save work across different functions.

18.47.1.20 join_short_stitches() `static void join_short_stitches (`
`int * points,`
`int * n_points,`
`int width,`
`int tolerence) [static]`
points n_points width tolerence

Remove points that lie in the middle of two short stitches that could be one longer stitch. Repeat until none are found.

18.47.1.21 lindenmayer_system() `int lindenmayer_system (`
`L_system L,`
`char * state,`
`int iterations,`
`int complete)`
L state iterations complete

Returns

`int`

This is a slow generation algorithm.

18.47.1.22 save_points_to_pattern() `static void save_points_to_pattern (`
`EmbPattern * pattern,`
`int * points,`
`int n_points,`
`EmbReal scale,`
`int width,`
`int height) [static]`
pattern points n_points scale width height

18.47.1.23 threshold_method() `static int * threshold_method (`
`EmbImage * image,`
`int * n_points,`
`int subsample_width,`
`int subsample_height,`
`int threshold) [static]`
image n_points subsample_width subsample_height threshold

Returns

`int*`

Identify darker pixels to put stitches in.

18.47.2 Variable Documentation

18.47.2.1 hilbert_curve_l_system `L_system hilbert_curve_l_system`

Initial value:

```
= {  
    'A', "AB", "F+-", (char**)rules  
}
```

18.47.2.2 rules const char* rules[] = {"+BF-AFA-FB+", "-AF+BFB+FA-"}

18.48 extern/libembroidery/src/formats.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include <ctype.h>
#include "embroidery_internal.h"
```

Functions

- void **safe_free** (void *data)
data
- int **embFormat_getExtension** (const char *fileName, char *ending)
fileName ending
- int **emb_identify_format** (const char *fileName)
fileName
- short **fread_int16** (FILE *f)
f
- unsigned short **fread_uint16** (FILE *f)
f
- int **fread_int32_be** (FILE *f)
f
- void **fpad** (FILE *file, char c, int n)
f
- void **binaryWriteShort** (FILE *f, short data)
f data
- void **binaryWriteUShort** (FILE *f, unsigned short data)
f data
- void **binaryWriteUShortBE** (FILE *f, unsigned short data)
f data
- void **binaryWriteInt** (FILE *f, int data)
f data
- void **binaryWriteIntBE** (FILE *f, int data)
f data
- void **binaryWriteUInt** (FILE *f, unsigned int data)
f data
- void **binaryWriteUIntBE** (FILE *f, unsigned int data)
f data
- char **embPattern_read** (EmbPattern *pattern, const char *fileName, int format)
pattern fileName format
- char **embPattern_write** (EmbPattern *pattern, const char *fileName, int format)
pattern fileName format
- char **embPattern_readAuto** (EmbPattern *pattern, const char *fileName)
pattern fileName
- char **embPattern_writeAuto** (EmbPattern *pattern, const char *fileName)
pattern fileName

Variables

- EmbFormatList formatTable [numberOfFormats]
- const char imageWithFrame [38][48]

18.48.1 Function Documentation

18.48.1.1 binaryWriteInt() void binaryWriteInt (

```
FILE * f,  
      int data )
```

f data

Todo replace with emblnt_read

18.48.1.2 binaryWriteIntBE() void binaryWriteIntBE (

```
FILE * f,  
      int data )
```

f data

Todo replace with emblnt_read

18.48.1.3 binaryWriteShort() void binaryWriteShort (

```
FILE * f,  
      short data )
```

f data

Todo replace with emblnt_read

18.48.1.4 binaryWriteUInt() void binaryWriteUInt (

```
FILE * f,  
      unsigned int data )
```

f data

Todo replace with emblnt_read

18.48.1.5 binaryWriteUIntBE() void binaryWriteUIntBE (

```
FILE * f,  
      unsigned int data )
```

f data

Todo replace with emblnt_read

18.48.1.6 binaryWriteUShort() void binaryWriteUShort (

```
FILE * f,  
      unsigned short data )
```

f data

Todo replace with emblnt_read

18.48.1.7 `binaryWriteUShortBE()` void binaryWriteUShortBE (FILE * *f*, unsigned short *data*)
f data

Todo replace with emblnt_read

18.48.1.8 `emb_identify_format()` int emb_identify_format (const char * *fileName*)
fileName
Returns
int

18.48.1.9 `embFormat_getExtension()` int embFormat_getExtension (const char * *fileName*, char * *ending*)
fileName ending
Returns
int

18.48.1.10 `embPattern_read()` char embPattern_read (EmbPattern * *pattern*, const char * *fileName*, int *format*)
pattern fileName format
Returns
char

18.48.1.11 `embPattern_readAuto()` char embPattern_readAuto (EmbPattern * *pattern*, const char * *fileName*)
pattern fileName
Returns
char

18.48.1.12 `embPattern_write()` char embPattern_write (EmbPattern * *pattern*, const char * *fileName*, int *format*)
pattern fileName format
Returns
char

18.48.1.13 embPattern_writeAuto() `char embPattern_writeAuto (`
 `EmbPattern * pattern,`
 `const char * fileName)`
pattern fileName

Returns

`char`

18.48.1.14 fpad() `void fpad (`
 `FILE * file,`
 `char c,`
 `int n)`
f

Returns

`int`

18.48.1.15 fread_int16() `short fread_int16 (`
 `FILE * f)`
f

Returns

`short`

18.48.1.16 fread_int32_be() `int fread_int32_be (`
 `FILE * f)`
f

Returns

`int`

Todo replace with emblnt_read

18.48.1.17 fread_uint16() `unsigned short fread_uint16 (`
 `FILE * f)`
f

Returns

`unsigned short`

Todo replace with emblnt_read

18.48.1.18 safe_free() `void safe_free (`
 `void * data)`
data

18.48.2 Variable Documentation

18.48.2.1 formatTable `EmbFormatList formatTable[numberOfFormats]`

This file is part of libembroidery.

Copyright 2018-2022 The Embroidermodder Team Licensed under the terms of the zlib license.

This file contains all the read and write functions for the library.

Todo This list needs reviewed in case some stitch formats also can contain object data (EMBFORMAT_↔ STCHANDOBJ). *

18.48.2.2 imageWithFrame `const char imageWithFrame[38][48]`**18.49 extern/libembroidery/src/formats/format_100.c File Reference**

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `read100` (`EmbPattern *pattern, FILE *file`)
- char `write100` (`EmbPattern *pattern, FILE *file`)

18.49.1 Detailed Description

The Toyota Embroidery Format (.100)

The Toyota 100 format is a stitch-only format that uses an external color file.

The stitch encoding is in 3 byte chunks.

18.49.2 Function Documentation**18.49.2.1 read100()** `char read100 (`
`EmbPattern * pattern,`
`FILE * file)`**18.49.2.2 write100()** `char write100 (`
`EmbPattern * pattern,`
`FILE * file)`**18.50 extern/libembroidery/src/formats/format_100.c File Reference**

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `read10o` (`EmbPattern *pattern, FILE *file`)
- char `write10o` (`EmbPattern *pattern, FILE *file`)

18.50.1 Detailed Description

The Toyota Embroidery Format (.100)

The Toyota 100 format is a stitch-only format that uses an external color file.

The stitch encoding is in 4 byte chunks.

18.50.2 Function Documentation

18.50.2.1 read10o() char read10o (

```
    EmbPattern * pattern,
    FILE * file )
```

18.50.2.2 write10o() char write10o (

```
    EmbPattern * pattern,
    FILE * file )
```

18.51 extern/libembroidery/src/formats/format_art.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char **readArt** (EmbPattern *pattern, FILE *file)
- char **writeArt** (EmbPattern *pattern, FILE *file)

18.51.1 Detailed Description

The Bernina Embroidery Format (.art)

We don't know much about this format.

Todo Find a source.

18.51.2 Function Documentation

18.51.2.1 readArt() char readArt (

```
    EmbPattern * pattern,
    FILE * file )
```

18.51.2.2 writeArt() char writeArt (

```
    EmbPattern * pattern,
    FILE * file )
```

18.52 extern/libembroidery/src/formats/format_bmc.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char **readBmc** (EmbPattern *pattern, FILE *file)
- char **writeBmc** (EmbPattern *pattern, FILE *file)

18.52.1 Detailed Description

The Bitmap Cache Embroidery Format (.bmc)
We don't know much about this format.

Todo Find a source.

18.52.2 Function Documentation

```
18.52.2.1 readBmc() char readBmc (
    EmbPattern * pattern,
    FILE * file )
```

```
18.52.2.2 writeBmc() char writeBmc (
    EmbPattern * pattern,
    FILE * file )
```

18.53 extern/libembroidery/src/formats/format_bro.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char **readBro** (EmbPattern *pattern, FILE *file)
- char **writeBro** (EmbPattern *pattern, FILE *file)

18.53.1 Detailed Description

The Bits and Volts Embroidery Format (.bro)
The Bits and Volts bro format is a stitch-only format that uses an external color file.
The header is 256 bytes. There's a series of unknown variables in the header.
The stitch list uses a variable length encoding which is 2 bytes for any stitch.

18.53.2 Function Documentation

```
18.53.2.1 readBro() char readBro (
    EmbPattern * pattern,
    FILE * file )
```

```
18.53.2.2 writeBro() char writeBro (
    EmbPattern * pattern,
    FILE * file )
```

18.54 extern/libembroidery/src/formats/format_cnd.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `readCnd` (`EmbPattern` *pattern, `FILE` *file)
- char `writeCnd` (`EmbPattern` *pattern, `FILE` *file)

18.54.1 Detailed Description

The Melco Embroidery Format (.cnd)

The Melco cnd format is a stitch-only format.

We don't know much about this format.

Todo Find a source.

18.54.2 Function Documentation

18.54.2.1 `readCnd()` char `readCnd`

```
    EmbPattern * pattern,
    FILE * file )
```

18.54.2.2 `writeCnd()` char `writeCnd`

```
    EmbPattern * pattern,
    FILE * file )
```

18.55 extern/libembroidery/src/formats/format_col.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `readCol` (`EmbPattern` *pattern, `FILE` *file)
- char `writeCol` (`EmbPattern` *pattern, `FILE` *file)

18.55.1 Detailed Description

The Embroidery Thread Color Format (.col)

An external color file format for formats that do not record their own colors.

It is a human-readable format that has a header that is a single line containing only the number of threads in decimal followed by the windows line break `\\textbackslash{}r\\textbackslash{}n`.

Then the rest of the file is a comma separated value list of all threads with 4 values per line: the index of the thread then the red, green and blue channels of the color in that order.

18.55.1.0.1 Example If we had a pattern called "example" with four colors: black, red, magenta and cyan in that order then the file is (with the white space written out):

```
example.col
4\r
0,0,0,0\r
1,255,0,0\r
2,0,255,0\r
3,0,0,255\r
```

18.55.2 Function Documentation

```
18.55.2.1 readCol() char readCol (
    EmbPattern * pattern,
    FILE * file )
```

```
18.55.2.2 writeCol() char writeCol (
    EmbPattern * pattern,
    FILE * file )
```

18.56 extern/libembroidery/src/formats/format_csd.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Macros

- #define **CsdSubMaskSize** 479
- #define **CsdXorMaskSize** 501

Functions

- void **BuildDecryptionTable** (int seed)
- unsigned char **DecodeCsdByte** (long fileOffset, unsigned char val, int type)
- char **readCsd** (EmbPattern *pattern, FILE *file)
- char **writeCsd** (EmbPattern *pattern, FILE *file)

Variables

- char **_subMask** [CsdSubMaskSize]
- char **_xorMask** [CsdXorMaskSize]
- const unsigned char **csd_decryptArray** []

18.56.1 Detailed Description

The Singer Embroidery Format (.csd)
Stitch Only Format.

18.56.2 Macro Definition Documentation

```
18.56.2.1 CsdSubMaskSize #define CsdSubMaskSize 479
```

```
18.56.2.2 CsdXorMaskSize #define CsdXorMaskSize 501
```

18.56.3 Function Documentation

```
18.56.3.1 BuildDecryptionTable() void BuildDecryptionTable (
    int seed )
```

```
18.56.3.2 DecodeCsdByte() unsigned char DecodeCsdByte (
    long fileOffset,
    unsigned char val,
    int type )
```

```
18.56.3.3 readCsd() char readCsd (
    EmbPattern * pattern,
    FILE * file )
```

```
18.56.3.4 writeCsd() char writeCsd (
    EmbPattern * pattern,
    FILE * file )
```

18.56.4 Variable Documentation

18.56.4.1 _subMask char _subMask[CsdSubMaskSize]

18.56.4.2 _xorMask char _xorMask[CsdXorMaskSize]

18.56.4.3 csd_decryptArray const unsigned char csd_decryptArray[]

Initial value:

```
= {
    0x43, 0x6E, 0x72, 0x7A, 0x76, 0x6C, 0x61, 0x6F, 0x7C, 0x29, 0x5D, 0x62, 0x60, 0x6E, 0x61, 0x62,
    0x20, 0x41, 0x66, 0x6A, 0x3A, 0x35, 0x5A, 0x63, 0x7C, 0x37, 0x3A, 0x2A, 0x25, 0x24, 0x2A, 0x33,
    0x00, 0x10, 0x14, 0x03, 0x72, 0x4C, 0x48, 0x42, 0x08, 0x7A, 0x5E, 0x0B, 0x6F, 0x45, 0x47, 0x5F,
    0x40, 0x54, 0x5C, 0x57, 0x55, 0x59, 0x3A, 0x32, 0x6F, 0x53, 0x54, 0x50, 0x5C, 0x4A, 0x56,
    0x2F, 0x2F, 0x62, 0x2C, 0x22, 0x65, 0x25, 0x28, 0x38, 0x30, 0x38, 0x22, 0x2B, 0x25, 0x3A, 0x6F,
    0x27, 0x38, 0x3E, 0x3F, 0x74, 0x37, 0x33, 0x77, 0x2E, 0x30, 0x3D, 0x34, 0x2E, 0x32, 0x2B, 0x2C,
    0x0C, 0x18, 0x42, 0x13, 0x16, 0x0A, 0x15, 0x02, 0x0B, 0x1C, 0x1E, 0x0E, 0x08, 0x60, 0x64, 0x0D,
    0x09, 0x51, 0x25, 0x1A, 0x18, 0x16, 0x19, 0x1A, 0x58, 0x10, 0x14, 0x08, 0x15, 0x1B, 0x5F,
    0xD5, 0xD2, 0xAE, 0xA3, 0xC1, 0xF0, 0xF4, 0xE8, 0xF8, 0xEC, 0xA6, 0xAB, 0xCD, 0xF8, 0xFB,
    0xE2, 0xF0, 0xFE, 0xFA, 0xF5, 0xB5, 0xF7, 0xF9, 0xFC, 0xB9, 0xF5, 0xEF, 0xF4, 0xF8, 0xEC, 0xBF,
    0xC3, 0xCE, 0xD7, 0xCD, 0xD0, 0xD7, 0xCF, 0xC2, 0xDB, 0xA4, 0xA0, 0xB0, 0xAF, 0xBE, 0x98, 0xE2,
    0xC2, 0x91, 0xE5, 0xDC, 0xDA, 0xD2, 0x96, 0xC4, 0x98, 0xF8, 0xC9, 0xD2, 0xDD, 0xD3, 0x9E, 0xDE,
    0xAE, 0xA5, 0xE2, 0x8C, 0xB6, 0xAC, 0xA3, 0xA9, 0xBC, 0xA8, 0xA6, 0xEB, 0x8B, 0xBF, 0xA1, 0xAC,
    0xB5, 0xA3, 0xBB, 0xB6, 0xA7, 0xD8, 0xDC, 0x9A, 0xAA, 0xF9, 0x82, 0xFB, 0x9D, 0xB9, 0xAB, 0xB3,
    0x94, 0xC1, 0xA0, 0x8C, 0x8B, 0x8E, 0x95, 0x8F, 0x87, 0x99, 0xE7, 0xE1, 0xA3, 0x83, 0x8B, 0xCF,
    0xA3, 0x85, 0x9D, 0x83, 0xD4, 0xB7, 0x83, 0x84, 0x91, 0x97, 0x9F, 0x88, 0x8F, 0xDD, 0xAD, 0x90
}
```

18.57 extern/libembroidery/src/formats/format_csv.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- `char * csvStitchFlagToStr (int flags)`
- `int csvStrToStitchFlag (const char *str)`

- char `readCsv (EmbPattern *pattern, FILE *file)`
- char `writeCsv (EmbPattern *pattern, FILE *file)`

18.57.1 Detailed Description

Comma Separated Values (.csv)

Comma Separated Values files aren't a universal system, here we aim to offer a broad support. The dialect is detected based on the opening lines, as each manufacturer should label their CSV files there.

18.57.1.0.1 Embroidermodder 2.0 CSV Dialect Our own version has the identifier comment line:
| Control Symbol | Type | Description | —— | # | COMMENT | | | > | VARIABLE | To store records of a pattern's width, height etc. This means that data stored in the header of say a .dst file is preserved. | \$ | THREAD | | * | STITCH | | * | JUMP | | * | COLOR | To change a color: used for trim as well | * | END | To end a pattern. | * | UNKNOWN | For any feature that we can't identify.

18.57.1.0.2 EmBird CSV Dialect

18.57.2 Function Documentation

18.57.2.1 csvStitchFlagToStr() char * csvStitchFlagToStr (
 int flags)

18.57.2.2 csvStrToStitchFlag() int csvStrToStitchFlag (
 const char * str)

18.57.2.3 readCsv() char readCsv (
 EmbPattern * pattern,
 FILE * file)

18.57.2.4 writeCsv() char writeCsv (
 EmbPattern * pattern,
 FILE * file)

18.58 extern/libembroidery/src/formats/format_dat.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `readDat (EmbPattern *pattern, FILE *file)`
- char `writeDat (EmbPattern *pattern, FILE *file)`

18.58.1 Function Documentation

18.58.1.1 readDat() char readDat (
 EmbPattern * pattern,
 FILE * file)

```
18.58.1.2 writeDat() char writeDat (
    EmbPattern * pattern,
    FILE * file )
```

18.59 extern/libembroidery/src/formats/format_dem.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char [readDem](#) (EmbPattern *pattern, FILE *file)
- char [writeDem](#) (EmbPattern *pattern, FILE *file)

18.59.1 Detailed Description

The Melco Embroidery Format (.dem)
Stitch Only Format

18.59.2 Function Documentation

```
18.59.2.1 readDem() char readDem (
    EmbPattern * pattern,
    FILE * file )
```

```
18.59.2.2 writeDem() char writeDem (
    EmbPattern * pattern,
    FILE * file )
```

18.60 extern/libembroidery/src/formats/format_dsb.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char [readDsb](#) (EmbPattern *pattern, FILE *file)
- char [writeDsb](#) (EmbPattern *pattern, FILE *file)

18.60.1 Detailed Description

The Barudan Embroidery Format (.dsb)

- Stitch Only Format.
- [X] Basic Read Support
- [o] Basic Write Support
- [o] Well Tested Read
- [o] Well Tested Write

18.60.2 Function Documentation

```
18.60.2.1 readDsb() char readDsb (
    EmbPattern * pattern,
    FILE * file )
```

```
18.60.2.2 writeDsb() char writeDsb (
    EmbPattern * pattern,
    FILE * file )
```

18.61 extern/libembroidery/src/formats/format_dst.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Macros

- #define **cci**(c1, c2) (c1*256+c2)

Functions

- int **decode_record_flags** (unsigned char b2)
- void **encode_record** (FILE *file, int x, int y, int flags)
- void **set_dst_variable** (EmbPattern *pattern, char *var, char *val)
- char **readDst** (EmbPattern *pattern, FILE *file)
- char **writeDst** (EmbPattern *pattern, FILE *file)

18.61.1 Detailed Description

.DST (Tajima) embroidery file read/write routines Format comments are thanks to tspilman@dalcoathletic.com who's notes appeared at <http://www.wotsit.org> under Tajima Format.

18.61.1.1 Tajima Embroidery Format (.dst) Stitch Only Format. [X] Basic Read Support [X] Basic Write Support [] Well Tested Read [] Well Tested Write

.DST (Tajima) embroidery file read/write routines Format comments are thanks to tspilman@dalcoathletic.com who's notes appeared at <http://www.wotsit.org> under Tajima Format.

Other references: [2], [1].

18.61.1.1.1 Header The header contains general information about the design. It is in lines of ASCII, so if you open a DST file as a text file, it's the only part that's easy to read. The line ending symbol is `0x0D}. The header is necessary for the file to be read by most softwares and hardwares.

The header is 125 bytes of data followed by padding spaces to make it 512 bytes in total.

The lines are as follows.

| Label | Size | Description | Example |
|--------------|-------------|--|----------------|
| LA: | 17 | The design name with no path or extension. The space reserved is 16 characters, but the name must not be longer than 8 and be padded to 16 with spaces (0x20). | "LA:Star " |
| ST: | 8 | The stitch count. An integer in the format %07d, that is: a 7 digit number padded by leading zeros. This is the total accross all possible stitch flags. | |

| Label | Size | Description | Example |
|--------------|-------------|--|----------------|
| CO: | 4 | The number of color changes (not to be confused with thread count, an all black design we would have the record \textbf{000}). An integer in the format %03d, that is: a 3 digit number padded by leading zeros. | |
| +X: | 6 | The extent of the pattern in the positive x direction in millimeters. An integer in the format %05d, that is: a 5 digit number padded by leading zeros. | |
| -X: | 6 | The extent of the pattern in the negative x direction in millimeters. An integer in the format %05d, that is: a 5 digit integer padded by leading zeros. | |
| +Y: | 6 | The extent of the pattern in the positive y direction in millimeters. An integer in the format %05d, that is: a 5 digit integer padded by leading zeros. | |
| -Y: | 6 | The extent of the pattern in the negative y direction in millimeters. An integer in the format %05d, that is: a 5 digit integer padded by leading zeros. | |
| AX: | 7 | The difference of the end from the start in the x direction in 0.1mm, the first char should be the sign, followed by an integer in the format %05d, that is: a 5 digit integer padded by leading zeros. | |
| AY: | 7 | The difference of the end from the start in the y direction in 0.1mm, the first char should be the sign, followed by an integer in the format %05d, that is: a 5 digit integer padded by leading zeros. | |
| MX: | 7 | The x co-ordinate of the last point in the previous file should the design span multiple files. Like AX, it is the sign, followed by a 5 digit integer. If we have a one file design set it to zero. | |
| MY: | 7 | The y co-ordinate of the last point in the previous file should the design span multiple files. Like AY, it is the sign, followed by a 5 digit integer. If we have a one file design set it to zero. | |
| PD: | 10 | Information about multivolume designs. | |

18.61.1.1.2 Stitch Data Uses 3 byte per stitch encoding with the format as follows:

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|----------|--------------|----------|----------|----------|----------|----------|----------|
| Byte 0 | y+1 | y-1 | y+9 | y-9 | x-9 | x+9 | x-1 | x+1 |
| Byte 1 | y+3 | y-3 | y+27 | y-27 | x-27 | x+27 | x-3 | x+3 |
| Byte 2 | jump | color change | y+81 | y-81 | x-81 | x+81 | set | set |

T01 and Tap appear to use Tajima Ternary.

Where the stitch type is determined as:

Normal Stitch 0b00000011 0x03 Jump Stitch 0b10000011 0x83 Stop/Change Color 0b11000011 0x←C3 End Design 0b11110011 0xF3

Inclusive or'ed with the last byte.

Note that the max stitch length is the largest sum of \$1+3+9+27+81=121\$ where the unit length is 0.1mm so 12.←1mm. The coordinate system is right handed.

18.61.2 Macro Definition Documentation

18.61.2.1 cci #define cci(
 c1,
 c2) (c1*256+c2)

18.61.3 Function Documentation

18.61.3.1 decode_record_flags() int decode_record_flags (unsigned char b2)

```
18.61.3.2 encode_record() void encode_record (
    FILE * file,
    int x,
    int y,
    int flags )
```

```
18.61.3.3 readDst() char readDst (
    EmbPattern * pattern,
    FILE * file )
```

```
18.61.3.4 set_dst_variable() void set_dst_variable (
    EmbPattern * pattern,
    char * var,
    char * val )
```

```
18.61.3.5 writeDst() char writeDst (
    EmbPattern * pattern,
    FILE * file )
```

18.62 extern/libembroidery/src/formats/format_dsz.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `readDsz` (`EmbPattern` *`pattern`, `FILE` *`file`)
- char `writeDsz` (`EmbPattern` *`pattern`, `FILE` *`file`)

18.62.1 Function Documentation

```
18.62.1.1 readDsz() char readDsz (
    EmbPattern * pattern,
    FILE * file )
```

18.62.1.2 ZSK USA Embroidery Format (.dsz) The ZSK USA dsz format is stitch-only.

```
18.62.1.3 writeDsz() char writeDsz (
    EmbPattern * pattern,
    FILE * file )
```

18.63 extern/libembroidery/src/formats/format_dxf.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
```

```
#include "../embroidery_internal.h"
```

Functions

- void [readLine](#) (FILE *file, char *str)
- char [readDxf](#) (EmbPattern *pattern, FILE *file)
- char [writeDxf](#) (EmbPattern *pattern, FILE *file)

18.63.1 Function Documentation

18.63.1.1 [readDxf\(\)](#) char [readDxf](#) (
 EmbPattern * pattern,
 FILE * file)

18.63.1.2 [readLine\(\)](#) void [readLine](#) (
 FILE * file,
 char * str)

18.63.1.3 Drawing Exchange Format (.dxf) Graphics format for drawing files designed and used by AutoDesk for their AutoCAD program. [\[dxf_reference\]](#)

18.63.1.4 [writeDxf\(\)](#) char [writeDxf](#) (
 EmbPattern * pattern,
 FILE * file)

18.64 extern/libembroidery/src/formats/format_edr.c File Reference

```
#include <stdio.h>  
#include <math.h>  
#include "../embroidery_internal.h"
```

Functions

- char [readEdr](#) (EmbPattern *pattern, FILE *file)
- char [writeEdr](#) (EmbPattern *pattern, FILE *file)

18.64.1 Function Documentation

18.64.1.1 [readEdr\(\)](#) char [readEdr](#) (
 EmbPattern * pattern,
 FILE * file)

18.64.1.2 Embird Embroidery Format (.edr) Stitch Only Format

18.64.1.3 [writeEdr\(\)](#) char [writeEdr](#) (
 EmbPattern * pattern,
 FILE * file)

18.65 extern/libembroidery/src/formats/format_emd.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `emdDecode` (unsigned char *inputByte*)
- char `readEmd` (`EmbPattern` **pattern*, `FILE` **file*)
- char `writeEmd` (`EmbPattern` **pattern*, `FILE` **file*)

18.65.1 Detailed Description

The Elna Embroidery Format (.emd)
Stitch Only Format.

18.65.2 Function Documentation

18.65.2.1 `emdDecode()` char `emdDecode` (

```
    unsigned char inputByte )
```

18.65.2.2 `readEmd()` char `readEmd` (

```
    EmbPattern * pattern,
    FILE * file )
```

18.65.2.3 `writeEmd()` char `writeEmd` (

```
    EmbPattern * pattern,
    FILE * file )
```

18.66 extern/libembroidery/src/formats/format_exp.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `expDecode` (unsigned char *a1*)
- char `readExp` (`EmbPattern` **pattern*, `FILE` **file*)
- char `writeExp` (`EmbPattern` **pattern*, `FILE` **file*)

18.66.1 Function Documentation

18.66.1.1 `expDecode()` char `expDecode` (

```
    unsigned char a1 )
```

18.66.1.2 Melco Embroidery Format (.exp) Stitch Only Format.

18.66.1.3 `readExp()` char `readExp` (

```
EmbPattern * pattern,
FILE * file )
```

18.66.1.4 `writeExp()` char `writeExp` (

```
EmbPattern * pattern,
FILE * file )
```

18.67 extern/libembroidery/src/formats/format_exy.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- int `decode_exy_flags` (unsigned char `b2`)
- char `readExy` (`EmbPattern` *`pattern`, `FILE` *`file`)
- char `writeExy` (`EmbPattern` *`pattern`, `FILE` *`file`)

18.67.1 Function Documentation

18.67.1.1 `decode_exy_flags()` int `decode_exy_flags` (

```
unsigned char b2 )
```

18.67.1.2 Eltac Embroidery Format (.exy) Stitch Only Format.

18.67.1.3 `readExy()` char `readExy` (

```
EmbPattern * pattern,
FILE * file )
```

18.67.1.4 `writeExy()` char `writeExy` (

```
EmbPattern * pattern,
FILE * file )
```

18.68 extern/libembroidery/src/formats/format_eyc.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `readEys` (`EmbPattern` *`pattern`, `FILE` *`file`)
- char `writeEys` (`EmbPattern` *`pattern`, `FILE` *`file`)

18.68.1 Function Documentation

```
18.68.1.1 readEys() char readEys (
    EmbPattern * pattern,
    FILE * file )
```

18.68.1.2 **Sierra Expanded Embroidery Format (.eys)** Stitch Only Format.
Smoothie G-Code Embroidery Format (.fxy)?

```
18.68.1.3 writeEys() char writeEys (
    EmbPattern * pattern,
    FILE * file )
```

18.69 extern/libembroidery/src/formats/format_fxy.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char **readFxy** (EmbPattern *pattern, FILE *file)
- char **writeFxy** (EmbPattern *pattern, FILE *file)

18.69.1 Function Documentation

```
18.69.1.1 readFxy() char readFxy (
    EmbPattern * pattern,
    FILE * file )
```

18.69.1.2 **Embroidery Format (.fxy)** Stitch Only Format.

```
18.69.1.3 writeFxy() char writeFxy (
    EmbPattern * pattern,
    FILE * file )
```

18.70 extern/libembroidery/src/formats/format_gc.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char **readGc** (EmbPattern *pattern, FILE *file)
- char **writeGc** (EmbPattern *pattern, FILE *file)

18.70.1 Function Documentation

```
18.70.1.1 readGc() char readGc (
    EmbPattern * pattern,
    FILE * file )
```

Smoothie G-Code

Main Reference: Machinery's Handbook Guide A Guide to Tables, Formulas, & More in the 31st Edition by John Milton Amiss, Franklin D. Jones and Henry Ryffel

```
18.70.1.2 writeGc() char writeGc (
    EmbPattern * pattern,
    FILE * file )
```

18.71 extern/libembroidery/src/formats/format_gnc.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char **readGnc** (EmbPattern *pattern, FILE *file)
- char **writeGnc** (EmbPattern *pattern, FILE *file)

18.71.1 Function Documentation

```
18.71.1.1 readGnc() char readGnc (
    EmbPattern * pattern,
    FILE * file )
```

18.71.1.2 Great Notions Embroidery Format (.gnc) Stitch Only Format.

```
18.71.1.3 writeGnc() char writeGnc (
    EmbPattern * pattern,
    FILE * file )
```

18.72 extern/libembroidery/src/formats/format_gt.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char **readGt** (EmbPattern *pattern, FILE *file)
- char **writeGt** (EmbPattern *pattern, FILE *file)

18.72.1 Function Documentation

```
18.72.1.1 readGt() char readGt (
    EmbPattern * pattern,
    FILE * file )
```

18.72.1.2 Gold Thread Embroidery Format (.gt) Stitch Only Format.**18.72.1.3 writeGt()** `char writeGt (`
 `EmbPattern * pattern,`
 `FILE * file)`**18.73 extern/libembroidery/src/formats/format_hus.c File Reference**

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- int `husDecodeStitchType` (unsigned char b)
- unsigned char * `husDecompressData` (unsigned char *input, int compressedInputLength, int decompressedContentLength)
- unsigned char * `husCompressData` (unsigned char *input, int decompressedInputSize, int *compressedSize)
- int `husDecodeByte` (unsigned char b)
- unsigned char `husEncodeByte` (`EmbReal` f)
- unsigned char `husEncodeStitchType` (int st)
- char `readHus` (`EmbPattern` *pattern, `FILE` *file)
- char `writeHus` (`EmbPattern` *pattern, `FILE` *file)

18.73.1 Function Documentation**18.73.1.1 husCompressData()** `unsigned char * husCompressData (`
 `unsigned char * input,`
 `int decompressedInputSize,`
 `int * compressedSize)`**18.73.1.2 husDecodeByte()** `int husDecodeByte (`
 `unsigned char b)`**18.73.1.3 husDecodeStitchType()** `int husDecodeStitchType (`
 `unsigned char b)`**18.73.1.4 Husqvarna Viking Embroidery Format (.hus)** Stitch Only Format.**18.73.1.5 husDecompressData()** `unsigned char * husDecompressData (`
 `unsigned char * input,`
 `int compressedInputLength,`
 `int decompressedContentLength)`**18.73.1.6 husEncodeByte()** `unsigned char husEncodeByte (`
 `EmbReal f)`

18.73.1.7 husEncodeStitchType() `unsigned char husEncodeStitchType (int st)`

18.73.1.8 readHus() `char readHus (EmbPattern * pattern, FILE * file)`

18.73.1.9 writeHus() `char writeHus (EmbPattern * pattern, FILE * file)`

18.74 extern/libembroidery/src/formats/format_inb.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- `char readInb (EmbPattern *pattern, FILE *file)`
- `char writeInb (EmbPattern *pattern, FILE *file)`

18.74.1 Function Documentation

18.74.1.1 readInb() `char readInb (EmbPattern * pattern, FILE * file)`

18.74.1.2 Inbro Embroidery Format (.inb) Stitch Only Format.

18.74.1.3 writeInb() `char writeInb (EmbPattern * pattern, FILE * file)`

18.75 extern/libembroidery/src/formats/format_inf.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- `char readInf (EmbPattern *pattern, FILE *file)`
- `char writeInf (EmbPattern *pattern, FILE *file)`

18.75.1 Function Documentation

```
18.75.1.1 readInf() char readInf (
    EmbPattern * pattern,
    FILE * file )
```

18.75.1.2 **Embroidery Color Format (.inf)** Stitch Only Format.

```
18.75.1.3 writelInf() char writeInf (
    EmbPattern * pattern,
    FILE * file )
```

18.76 extern/libembroidery/src/formats/format_jef.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Classes

- struct `hoop_padding`

Functions

- int `jefGetHoopSize` (int width, int height)
- char `jefDecode` (unsigned char `inputByte`)
- void `jefSetHoopFromId` (`EmbPattern` *`pattern`, int `hoopCode`)
- void `read_hoop` (FILE *`file`, struct `hoop_padding` *`hoop`, char *`label`)
- char `readJef` (`EmbPattern` *`pattern`, FILE *`file`)
- void `jefEncode` (unsigned char *`b`, char `dx`, char `dy`, int `flags`)
- char `writeJef` (`EmbPattern` *`pattern`, FILE *`file`)

18.76.1 Function Documentation

```
18.76.1.1 jefDecode() char jefDecode (
    unsigned char inputByte )
```

```
18.76.1.2 jefEncode() void jefEncode (
    unsigned char * b,
    char dx,
    char dy,
    int flags )
```

```
18.76.1.3 jefGetHoopSize() int jefGetHoopSize (
    int width,
    int height )
```

18.76.1.4 **Janome Embroidery Format (.jef)** Stitch Only Format.

```
18.76.1.5 jefSetHoopFromId() void jefSetHoopFromId (
    EmbPattern * pattern,
    int hoopCode )
```

```
18.76.1.6 read_hoop() void read_hoop (
    FILE * file,
    struct hoop_padding * hoop,
    char * label )
```

```
18.76.1.7 readJef() char readJef (
    EmbPattern * pattern,
    FILE * file )
```

```
18.76.1.8 writeJef() char writeJef (
    EmbPattern * pattern,
    FILE * file )
```

18.77 extern/libembroidery/src/formats/format_ksm.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- void **ksmEncode** (unsigned char *b, char dx, char dy, int flags)
- char **readKsm** (EmbPattern *pattern, FILE *file)
- char **writeKsm** (EmbPattern *pattern, FILE *file)

18.77.1 Function Documentation

```
18.77.1.1 ksmEncode() void ksmEncode (
    unsigned char * b,
    char dx,
    char dy,
    int flags )
```

18.77.1.2 Pfaff professional Design format (.ksm) Stitch Only Format.

```
18.77.1.3 readKsm() char readKsm (
    EmbPattern * pattern,
    FILE * file )
```

```
18.77.1.4 writeKsm() char writeKsm (
    EmbPattern * pattern,
    FILE * file )
```

18.78 extern/libembroidery/src/formats/format_max.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `readMax` (`EmbPattern` *pattern, `FILE` *file)
 - char `writeMax` (`EmbPattern` *pattern, `FILE` *file)

Variables

- const unsigned char max_header []

18.78.1 Function Documentation

18.78.1.1 readMax() char readMax (

```
EmbPattern * pattern,  
FILE * file )
```

18.78.1.2 writeMax() char writeMax (

```
EmbPattern * pattern,  
FILE * file )
```

18.78.2 Variable Documentation

18.78.2.1 max_header const unsigned char max_header[]

Initial value:

18.78.2.2 Pfaff Embroidery Format (.max) Stitch Only Format.

18.79 `extern/libembroidery/src/formats/format mit.c` File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `readMit` (`EmbPattern` *`pattern`, `FILE` *`file`)
 - char `writeMit` (`EmbPattern` *`pattern`, `FILE` *`file`)

18.79.1 Function Documentation

```
18.79.1.1 readMit() char readMit (
    EmbPattern * pattern,
    FILE * file )
```

18.79.1.2 Mitsubishi Embroidery Format (.mit) Stitch Only Format.

```
18.79.1.3 writeMit() char writeMit (
    EmbPattern * pattern,
    FILE * file )
```

18.80 extern/libembroidery/src/formats/format_new.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char **readNew** (EmbPattern *pattern, FILE *file)
- char **writeNew** (EmbPattern *pattern, FILE *file)

18.80.1 Function Documentation

```
18.80.1.1 readNew() char readNew (
    EmbPattern * pattern,
    FILE * file )
```

18.80.1.2 Ameco Embroidery Format (.new) Stitch Only Format.

```
18.80.1.3 writeNew() char writeNew (
    EmbPattern * pattern,
    FILE * file )
```

18.81 extern/libembroidery/src/formats/format_ofm.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char * **ofmReadLibrary** (FILE *file)
- static int **ofmReadClass** (FILE *file)
- void **ofmReadBlockHeader** (FILE *file)
- void **ofmReadColorChange** (FILE *file, EmbPattern *pattern)
- void **ofmReadThreads** (FILE *file, EmbPattern *p)
- **EmbReal ofmDecode** (unsigned char b1, unsigned char b2)
- void **ofmReadExpanded** (FILE *file, EmbPattern *p)
- char **readOfm** (EmbPattern *pattern, FILE *fileCompound)
- char **writeOfm** (EmbPattern *pattern, FILE *file)

18.81.1 Function Documentation

18.81.1.1 ofmDecode() `EmbReal ofmDecode (`
 `unsigned char b1,`
 `unsigned char b2)`

18.81.1.2 ofmReadBlockHeader() `void ofmReadBlockHeader (`
 `FILE * file)`

18.81.1.3 ofmReadClass() `static int ofmReadClass (`
 `FILE * file) [static]`

18.81.1.4 ofmReadColorChange() `void ofmReadColorChange (`
 `FILE * file,`
 `EmbPattern * pattern)`

18.81.1.5 ofmReadExpanded() `void ofmReadExpanded (`
 `FILE * file,`
 `EmbPattern * p)`

18.81.1.6 ofmReadLibrary() `char * ofmReadLibrary (`
 `FILE * file)`

18.81.1.7 Melco Embroidery Format (.ofm) Stitch Only Format.

18.81.1.8 ofmReadThreads() `void ofmReadThreads (`
 `FILE * file,`
 `EmbPattern * p)`

18.81.1.9 readOfm() `char readOfm (`
 `EmbPattern * pattern,`
 `FILE * fileCompound)`

18.81.1.10 writeOfm() `char writeOfm (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.82 extern/libembroidery/src/formats/format_pcd.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `readPcd` (`EmbPattern` *pattern, const char *fileName, FILE *file)
- char `writePcd` (`EmbPattern` *pattern, FILE *file)

18.82.1 Function Documentation

18.82.1.1 `readPcd()` char `readPcd` (

```
    EmbPattern * pattern,
    const char * fileName,
    FILE * file )
```

18.82.1.2 Pfaff PCD File Format (.pcd) Stitch Only Format.

The format uses a signed 3 byte-length number type.

See the description here (5) for the overview of the format.

For an example of the format see (11).

18.82.1.3 `writePcd()` char `writePcd` (

```
    EmbPattern * pattern,
    FILE * file )
```

18.83 extern/libembroidery/src/formats/format_pcm.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `readPcm` (`EmbPattern` *pattern, FILE *file)
- char `writePcm` (`EmbPattern` *pattern, FILE *file)

18.83.1 Function Documentation

18.83.1.1 `readPcm()` char `readPcm` (

```
    EmbPattern * pattern,
    FILE * file )
```

18.83.1.2 Pfaff Embroidery Format (.pcm) The Pfaff pcm format is stitch-only.

18.83.1.3 `writePcm()` char `writePcm` (

```
    EmbPattern * pattern,
    FILE * file )
```

18.84 extern/libembroidery/src/formats/format_pcq.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `readPcq (EmbPattern *pattern, const char *fileName, FILE *file)`
- char `writePcq (EmbPattern *pattern, FILE *file)`

18.84.1 Function Documentation

18.84.1.1 `readPcq()` char `readPcq (`
 `EmbPattern * pattern,`
 `const char * fileName,`
 `FILE * file)`

18.84.1.2 Embroidery Format (.pcq) The Pfaff pcq format is stitch-only.

18.84.1.3 `writePcq()` char `writePcq (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.85 extern/libembroidery/src/formats/format_pcs.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `readPcs (EmbPattern *pattern, const char *fileName, FILE *file)`
- char `writePcs (EmbPattern *pattern, FILE *file)`

18.85.1 Function Documentation

18.85.1.1 `readPcs()` char `readPcs (`
 `EmbPattern * pattern,`
 `const char * fileName,`
 `FILE * file)`

18.85.1.2 Embroidery Format (.pcq) The Pfaff pcs format is stitch-only.

18.85.1.3 `writePcs()` char `writePcs (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.86 extern/libembroidery/src/formats/format_pec.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- void `readPecStitches` (`EmbPattern` **pattern*, `FILE` **file*)
- void `pecEncodeJump` (`FILE` **file*, int *x*, int *types*)
- void `pecEncodeStop` (`FILE` **file*, unsigned char *val*)
- char `readPec` (`EmbPattern` **pattern*, const char **fileName*, `FILE` **file*)
- void `pecEncode` (`FILE` **file*, `EmbPattern` **p*)
- void `writelImage` (`FILE` **file*, unsigned char *image*[][48])
- void `writePecStitches` (`EmbPattern` **pattern*, `FILE` **file*, const char **fileName*)
- char `writePec` (`EmbPattern` **pattern*, const char **fileName*, `FILE` **file*)

18.86.1 Function Documentation

18.86.1.1 `pecEncode()` void `pecEncode` (
 `FILE` * *file*,
 `EmbPattern` * *p*)

18.86.1.2 `pecEncodeJump()` void `pecEncodeJump` (
 `FILE` * *file*,
 int *x*,
 int *types*)

18.86.1.3 `pecEncodeStop()` void `pecEncodeStop` (
 `FILE` * *file*,
 unsigned char *val*)

18.86.1.4 `readPec()` char `readPec` (
 `EmbPattern` * *pattern*,
 const char * *fileName*,
 `FILE` * *file*)

18.86.1.5 `readPecStitches()` void `readPecStitches` (
 `EmbPattern` * *pattern*,
 `FILE` * *file*)

18.86.1.6 Embroidery Format (.pec) The Brother pec format is stitch-only.

18.86.1.7 `writelImage()` void `writelImage` (
 `FILE` * *file*,
 unsigned char *image*[][48])

Write a PES embedded *image* to the given *file* pointer.

18.86.1.8 `writePec()` char `writePec` (
 `EmbPattern` * *pattern*,
 const char * *fileName*,
 `FILE` * *file*)

```
18.86.1.9 writePecStitches() void writePecStitches (
    EmbPattern * pattern,
    FILE * file,
    const char * fileName )
```

18.87 extern/libembroidery/src/formats/format_pel.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char [readPel](#) (EmbPattern *pattern, FILE *file)
- char [writePel](#) (EmbPattern *pattern, FILE *file)

18.87.1 Function Documentation

```
18.87.1.1 readPel() char readPel (
    EmbPattern * pattern,
    FILE * file )
```

18.87.1.2 Embroidery Format (.pec) The Brother pel format is stitch-only.

```
18.87.1.3 writePel() char writePel (
    EmbPattern * pattern,
    FILE * file )
```

18.88 extern/libembroidery/src/formats/format_pem.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char [readPem](#) (EmbPattern *pattern, FILE *file)
- char [writePem](#) (EmbPattern *pattern, FILE *file)

18.88.1 Function Documentation

```
18.88.1.1 readPem() char readPem (
    EmbPattern * pattern,
    FILE * file )
```

18.88.1.2 Embroidery Format (.pec) The Brother pem format is stitch-only.

```
18.88.1.3 writePem() char writePem (
    EmbPattern * pattern,
    FILE * file )
```

18.89 extern/libembroidery/src/formats/format_pes.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `readPes` (`EmbPattern` *pattern, const char *fileName, FILE *file)
- void `readDescriptions` (FILE *file, `EmbPattern` *pattern)
- void `readPESHeaderV5` (FILE *file, `EmbPattern` *pattern)
- void `readPESHeaderV6` (FILE *file, `EmbPattern` *pattern)
- void `readPESHeaderV7` (FILE *file, `EmbPattern` *pattern)
- void `readPESHeaderV8` (FILE *file, `EmbPattern` *pattern)
- void `readPESHeaderV9` (FILE *file, `EmbPattern` *pattern)
- void `readPESHeaderV10` (FILE *file, `EmbPattern` *pattern)
- void `readHoopName` (FILE *file, `EmbPattern` *pattern)
- void `readImageString` (FILE *file, `EmbPattern` *pattern)
- void `readProgrammableFills` (FILE *file, `EmbPattern` *pattern)
- void `readMotifPatterns` (FILE *file, `EmbPattern` *pattern)
- void `readFeatherPatterns` (FILE *file, `EmbPattern` *pattern)
- void `readThreads` (FILE *file, `EmbPattern` *pattern)
- void `pesWriteSewSegSection` (`EmbPattern` *pattern, FILE *file)
- void `pesWriteEmbOneSection` (`EmbPattern` *pattern, FILE *file)
- char `writePes` (`EmbPattern` *pattern, const char *fileName, FILE *file)

Variables

- const char * `pes_version_strings` []
- int `pes_version` = PES0001

18.89.1 Function Documentation

18.89.1.1 pesWriteEmbOneSection() void pesWriteEmbOneSection (

```
    EmbPattern * pattern,
    FILE * file )
```

18.89.1.2 pesWriteSewSegSection() void pesWriteSewSegSection (

```
    EmbPattern * pattern,
    FILE * file )
```

18.89.1.3 readDescriptions() void readDescriptions (

```
    FILE * file,
    EmbPattern * pattern )
```

18.89.1.4 readFeatherPatterns() void readFeatherPatterns (

```
    FILE * file,
    EmbPattern * pattern )
```

```
18.89.1.5 readHoopName() void readHoopName (
    FILE * file,
    EmbPattern * pattern )
```

```
18.89.1.6 readImageString() void readImageString (
    FILE * file,
    EmbPattern * pattern )
```

```
18.89.1.7 readMotifPatterns() void readMotifPatterns (
    FILE * file,
    EmbPattern * pattern )
```

```
18.89.1.8 readPes() char readPes (
    EmbPattern * pattern,
    const char * fileName,
    FILE * file )
```

```
18.89.1.9 readPESHeaderV10() void readPESHeaderV10 (
    FILE * file,
    EmbPattern * pattern )
```

```
18.89.1.10 readPESHeaderV5() void readPESHeaderV5 (
    FILE * file,
    EmbPattern * pattern )
```

```
18.89.1.11 readPESHeaderV6() void readPESHeaderV6 (
    FILE * file,
    EmbPattern * pattern )
```

```
18.89.1.12 readPESHeaderV7() void readPESHeaderV7 (
    FILE * file,
    EmbPattern * pattern )
```

```
18.89.1.13 readPESHeaderV8() void readPESHeaderV8 (
    FILE * file,
    EmbPattern * pattern )
```

```
18.89.1.14 readPESHeaderV9() void readPESHeaderV9 (
    FILE * file,
    EmbPattern * pattern )
```

```
18.89.1.15 readProgrammableFills() void readProgrammableFills (
    FILE * file,
    EmbPattern * pattern )
```

```
18.89.1.16 readThreads() void readThreads (
    FILE * file,
    EmbPattern * pattern )
```

```
18.89.1.17 writePes() char writePes (
    EmbPattern * pattern,
    const char * fileName,
    FILE * file )
```

18.89.2 Variable Documentation

18.89.2.1 **pes_version** int pes_version = **PES0001**

18.89.2.2 **pes_version_strings** const char* pes_version_strings[]

Initial value:

```
= {
    "#PES0001",
    "#PES0020",
    "#PES0022",
    "#PES0030",
    "#PES0040",
    "#PES0050",
    "#PES0055",
    "#PES0056",
    "#PES0060",
    "#PES0070",
    "#PES0080",
    "#PES0090",
    "#PES0100",
}
```

18.89.2.3 Embroidery Format (.pec) The Brother pes format is stitch-only.

18.90 extern/libembroidery/src/formats/format_phb.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char **readPhb** (EmbPattern *pattern, FILE *file)
- char **writePhb** (EmbPattern *pattern, FILE *file)

18.90.1 Function Documentation

```
18.90.1.1 readPhb() char readPhb (
    EmbPattern * pattern,
    FILE * file )
```

18.90.1.2 Embroidery Format (.pec) The Brother phb format is stitch-only.

```
18.90.1.3 writePhb() char writePhb (
    EmbPattern * pattern,
    FILE * file )
```

18.91 extern/libembroidery/src/formats/format_phc.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `readPhc` (`EmbPattern` **pattern*, `FILE` **file*)
- char `writePhc` (`EmbPattern` **pattern*, `FILE` **file*)

18.91.1 Function Documentation

18.91.1.1 `readPhc()` `char readPhc (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.91.1.2 Embroidery Format (.pec) The Brother phc format is stitch-only.

18.91.1.3 `writePhc()` `char writePhc (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.92 extern/libembroidery/src/formats/format_plt.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `readPlt` (`EmbPattern` **pattern*, `FILE` **file*)
- char `writePlt` (`EmbPattern` **pattern*, `FILE` **file*)

18.92.1 Function Documentation

18.92.1.1 `readPlt()` `char readPlt (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.92.1.2 Embroidery Format (.plt) The AutoCAD plt format is stitch-only.

18.92.1.3 `writePlt()` `char writePlt (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.93 extern/libembroidery/src/formats/format_rgb.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char **readRgb** (*EmbPattern* *pattern, FILE *file)
- char **writeRgb** (*EmbPattern* *pattern, FILE *file)

18.93.1 Function Documentation

18.93.1.1 readRgb() char readRgb (

```
    EmbPattern * pattern,
    FILE * file )
```

18.93.1.2 Color File (.rgb) The RGB format is a color-only format to act as an external color file for other formats.

18.93.1.3 writeRgb() char writeRgb (

```
    EmbPattern * pattern,
    FILE * file )
```

18.94 extern/libembroidery/src/formats/format_sew.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char **sewDecode** (unsigned char inputByte)
- char **readSew** (*EmbPattern* *pattern, FILE *file)
- char **writeSew** (*EmbPattern* *pattern, FILE *file)

18.94.1 Function Documentation

18.94.1.1 readSew() char readSew (

```
    EmbPattern * pattern,
    FILE * file )
```

18.94.1.2 sewDecode() char sewDecode (

```
    unsigned char inputByte )
```

18.94.1.3 Embroidery Format (.sew) The Janome sew format is stitch-only.

18.94.1.4 writeSew() char writeSew (

```
    EmbPattern * pattern,
    FILE * file )
```

18.95 extern/libembroidery/src/formats/format_shv.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `shvDecode` (unsigned char *inputByte*)
- short `shvDecodeShort` (unsigned short *inputByte*)
- char `readShv` (`EmbPattern` **pattern*, `FILE` **file*)
- char `writeShv` (`EmbPattern` **pattern*, `FILE` **file*)

18.95.1 Function Documentation

18.95.1.1 `readShv()` char `readShv` (
 `EmbPattern` * *pattern*,
 `FILE` * *file*)

18.95.1.2 `shvDecode()` char `shvDecode` (
 unsigned char *inputByte*)

18.95.1.3 Viking Embroidery Format (.shv) The Husqvarna Viking shv format is stitch-only.

18.95.1.4 `shvDecodeShort()` short `shvDecodeShort` (
 unsigned short *inputByte*)

18.95.1.5 `writeShv()` char `writeShv` (
 `EmbPattern` * *pattern*,
 `FILE` * *file*)

18.96 extern/libembroidery/src/formats/format_sst.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `readSst` (`EmbPattern` **pattern*, `FILE` **file*)
- char `writeSst` (`EmbPattern` **pattern*, `FILE` **file*)

18.96.1 Function Documentation

18.96.1.1 `readSst()` char `readSst` (
 `EmbPattern` * *pattern*,
 `FILE` * *file*)

18.96.1.2 Embroidery Format (.sst) The Sunstar sst format is stitch-only.

18.96.1.3 writeSst() char writeSst (

```
    EmbPattern * pattern,
    FILE * file )
```

18.97 extern/libembroidery/src/formats/format_stx.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- int **stxReadThread** (StxThread *thread, FILE *file)
- char **readStx** (EmbPattern *pattern, FILE *file)
- char **writeStx** (EmbPattern *pattern, FILE *file)

18.97.1 Function Documentation

18.97.1.1 readStx() char readStx (

```
    EmbPattern * pattern,
    FILE * file )
```

18.97.1.2 stxReadThread() int stxReadThread (

```
    StxThread * thread,
    FILE * file )
```

18.97.1.3 Stitch Embroidery Format (.stx) The Data Stitch stx format is stitch-only.

18.97.1.4 writeStx() char writeStx (

```
    EmbPattern * pattern,
    FILE * file )
```

18.98 extern/libembroidery/src/formats/format_svg.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char **readSvg** (EmbPattern *pattern, FILE *file)
- char **writeSvg** (EmbPattern *pattern, FILE *file)

Variables

- int `svgCreator`
- int `svgExpect`
- int `svgMultiValue`
- int `current_element_id`
- `SvgAttribute attributeList [1000]`
- int `n_attributes = 0`
- char `currentAttribute [1000]`
- char `currentValue [1000]`

18.98.1 Function Documentation

18.98.1.1 `readSvg()` char `readSvg (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.98.1.2 `writeSvg()` char `writeSvg (`
 `EmbPattern * pattern,`
 `FILE * file)`

Writes the data from `pattern` to a file with the given `fileName`. Returns `true` if successful, otherwise returns `false`.

18.98.2 Variable Documentation

18.98.2.1 `attributeList` `SvgAttribute attributeList[1000]`

18.98.2.2 `current_element_id` int `current_element_id`

18.98.2.3 `currentAttribute` char `currentAttribute[1000]`

18.98.2.4 `currentValue` char `currentValue[1000]`

18.98.2.5 `n_attributes` int `n_attributes = 0`

18.98.2.6 `svgCreator` int `svgCreator`

18.98.2.7 Vector Graphics (.svg) The scalable vector graphics (SVG) format is a graphics format maintained by ...

18.98.2.8 `svgExpect` int `svgExpect`

18.98.2.9 `svgMultiValue` int `svgMultiValue`

18.99 extern/libembroidery/src/formats/format_t01.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `readT01` (`EmbPattern` *pattern, `FILE` *file)
- char `writeT01` (`EmbPattern` *pattern, `FILE` *file)

18.99.1 Function Documentation

18.99.1.1 `readT01()` char `readT01` (
 `EmbPattern` * *pattern*,
 `FILE` * *file*)

18.99.1.2 Embroidery Format (.pcq) The Pfaff t01 format is stitch-only.

18.99.1.3 `writeT01()` char `writeT01` (
 `EmbPattern` * *pattern*,
 `FILE` * *file*)

18.100 extern/libembroidery/src/formats/format_t09.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `readT09` (`EmbPattern` *pattern, `FILE` *file)
- char `writeT09` (`EmbPattern` *pattern, `FILE` *file)

18.100.1 Function Documentation

18.100.1.1 `readT09()` char `readT09` (
 `EmbPattern` * *pattern*,
 `FILE` * *file*)

18.100.1.1.1 Embroidery Format (.pcq) The Pfaff t09 format is stitch-only.

18.100.1.2 `writeT09()` char `writeT09` (
 `EmbPattern` * *pattern*,
 `FILE` * *file*)

18.101 `extern/libembroidery/src/formats/format_tap.c` File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "../embroidery_internal.h"
```

Functions

- void `encode_tap_record` (FILE *file, int x, int y, int flags)
- int `decode_tap_record_flags` (unsigned char b2)
- char `readTap` (EmbPattern *pattern, FILE *file)
- char `writeTap` (EmbPattern *pattern, FILE *file)

18.101.1 Function Documentation

18.101.1.1 `decode_tap_record_flags()` int decode_tap_record_flags (unsigned char b2)

18.101.1.2 `encode_tap_record()` void encode_tap_record (FILE * file, int x, int y, int flags)

18.101.1.3 Embroidery Format (.tap) The Happy tap format is stitch-only.

18.101.1.4 `readTap()` char readTap (EmbPattern * pattern, FILE * file)

18.101.1.5 `writeTap()` char writeTap (EmbPattern * pattern, FILE * file)

18.102 `extern/libembroidery/src/formats/format_thr.c` File Reference

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char `readThr` (EmbPattern *pattern, FILE *file)
- char `writeThr` (EmbPattern *pattern, FILE *file)

18.102.1 Function Documentation

```
18.102.1.1 readThr() char readThr (
    EmbPattern * pattern,
    FILE * file )
```

18.102.1.2 **Embroidery Format (.thr)** The ThreadWorks thr format is stitch-only.

```
18.102.1.3 writeThr() char writeThr (
    EmbPattern * pattern,
    FILE * file )
```

18.103 extern/libembroidery/src/formats/format_txt.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char **readTxt** (EmbPattern *pattern, FILE *file)
- char **writeTxt** (EmbPattern *pattern, FILE *file)

18.103.1 Function Documentation

```
18.103.1.1 readTxt() char readTxt (
    EmbPattern * pattern,
    FILE * file )
```

18.103.1.2 **File (.txt)** The txt format is stitch-only and isn't associated with a specific company.

```
18.103.1.3 writeTxt() char writeTxt (
    EmbPattern * pattern,
    FILE * file )
```

18.104 extern/libembroidery/src/formats/format_u00.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char **readU00** (EmbPattern *pattern, FILE *file)
- char **writeU00** (EmbPattern *pattern, FILE *file)

18.104.1 Function Documentation

```
18.104.1.1 readU00() char readU00 (
    EmbPattern * pattern,
    FILE * file )
```

18.104.1.2 Embroidery Format (.u00) The Barudan u00 format is stitch-only.

18.104.1.3 writeU00() char writeU00 (

```
EmbPattern * pattern,
FILE * file )
```

18.105 extern/libembroidery/src/formats/format_u01.c File Reference

```
#include <stdio.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- char **readU01** (EmbPattern *pattern, FILE *file)
- char **writeU01** (EmbPattern *pattern, FILE *file)

18.105.1 Function Documentation

18.105.1.1 readU01() char readU01 (

```
EmbPattern * pattern,
FILE * file )
```

18.105.1.2 Embroidery Format (.u00) The Barudan u01 format is stitch-only.

18.105.1.3 writeU01() char writeU01 (

```
EmbPattern * pattern,
FILE * file )
```

18.106 extern/libembroidery/src/formats/format_vip.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- int **vipDecodeByte** (unsigned char b)
- int **vipDecodeStitchType** (unsigned char b)
- unsigned char * **vipDecompressData** (unsigned char *input, int compressedInputLength, int decompressedContentLength)
- char **readVip** (EmbPattern *pattern, FILE *file)
- unsigned char * **vipCompressData** (unsigned char *input, int decompressedInputSize, int *compressedSize)
- unsigned char **vipEncodeByte** (EmbReal f)
- unsigned char **vipEncodeStitchType** (int st)
- char **writeVip** (EmbPattern *pattern, FILE *file)

Variables

- const unsigned char **vipDecodingTable** []

18.106.1 Function Documentation

18.106.1.1 `readVip()` `char readVip (`
`EmbPattern * pattern,`
`FILE * file)`

18.106.1.2 `vipCompressData()` `unsigned char * vipCompressData (`
`unsigned char * input,`
`int decompressedInputSize,`
`int * compressedSize)`

18.106.1.3 `vipDecodeByte()` `int vipDecodeByte (`
`unsigned char b)`

18.106.1.4 `vipDecodeStitchType()` `int vipDecodeStitchType (`
`unsigned char b)`

18.106.1.5 `vipDecompressData()` `unsigned char * vipDecompressData (`
`unsigned char * input,`
`int compressedInputLength,`
`int decompressedContentLength)`

18.106.1.6 `vipEncodeByte()` `unsigned char vipEncodeByte (`
`EmbReal f)`

18.106.1.7 `vipEncodeStitchType()` `unsigned char vipEncodeStitchType (`
`int st)`

18.106.1.8 `writeVip()` `char writeVip (`
`EmbPattern * pattern,`
`FILE * file)`

18.106.2 Variable Documentation

18.106.2.1 `vipDecodingTable` `const unsigned char vipDecodingTable[]`

Initial value:

```
= {  

    0x2E, 0x82, 0xE4, 0x6F, 0x38, 0xA9, 0xDC, 0xC6, 0x7B, 0xB6, 0x28, 0xAC, 0xFD, 0xAA, 0x8A, 0x4E,  

    0x76, 0x2E, 0xF0, 0xE4, 0x25, 0x1B, 0x8A, 0x68, 0x4B, 0x92, 0xB9, 0x24, 0x95, 0xF0, 0x3B, 0xEF,  

    0xF7, 0x40, 0x24, 0x18, 0x39, 0x31, 0xBB, 0xE1, 0x53, 0xA8, 0x1F, 0xB1, 0x3A, 0x07, 0xFB, 0xCB,  

    0xE6, 0x00, 0x81, 0x50, 0x0E, 0x40, 0xE1, 0x2C, 0x73, 0x50, 0xD, 0x91, 0xD6, 0x0A, 0x5D, 0x06,  

    0x8B, 0xB8, 0x62, 0xAE, 0x47, 0x00, 0x53, 0x5A, 0xB7, 0x80, 0xAA, 0x28, 0xF7, 0x5D, 0x70, 0x5E,  

    0x2C, 0x0B, 0x98, 0xE3, 0xA0, 0x98, 0x60, 0x47, 0x89, 0x9B, 0x82, 0xFB, 0x40, 0xC9, 0xB4, 0x00,  

    0x0E, 0x68, 0x6A, 0x1E, 0x09, 0x85, 0xC0, 0x53, 0x81, 0xD1, 0x98, 0x89, 0xAF, 0xE8, 0x85, 0x4F,  

    0xE3, 0x69, 0x89, 0x03, 0xA1, 0x2E, 0x8F, 0xCF, 0xED, 0x91, 0x9F, 0x58, 0x1E, 0xD6, 0x84, 0x3C,  

    0x09, 0x27, 0xBD, 0xF4, 0xC3, 0x90, 0xC0, 0x51, 0x1B, 0x2B, 0x63, 0xBC, 0xB9, 0x3D, 0x40, 0x4D,  

    0x62, 0x6F, 0xE0, 0x8C, 0xF5, 0x5D, 0x08, 0xFD, 0x3D, 0x50, 0x36, 0xD7, 0xC9, 0xC9, 0x43, 0xE4,  

    0x2D, 0xCB, 0x95, 0xB6, 0xF4, 0x0D, 0xEA, 0xC2, 0xFD, 0x66, 0x3F, 0x5E, 0xBD, 0x69, 0x06, 0x2A,  

    0x03, 0x19, 0x47, 0x2B, 0xDF, 0x38, 0xEA, 0x4F, 0x80, 0x49, 0x95, 0xB2, 0xD6, 0xF9, 0x9A, 0x75,  

    0xF4, 0xD8, 0x9B, 0x1D, 0xB0, 0xA4, 0x69, 0xDB, 0xA9, 0x21, 0x79, 0x6F, 0xD8, 0xDE, 0x33, 0xFE,  

}
```

```

0x9F, 0x04, 0xE5, 0x9A, 0x6B, 0x9B, 0x73, 0x83, 0x62, 0x7C, 0xB9, 0x66, 0x76, 0xF2, 0x5B, 0xC9,
0x5E, 0xFC, 0x74, 0xAA, 0x6C, 0xF1, 0xCD, 0x93, 0xCE, 0xE9, 0x80, 0x53, 0x03, 0x3B, 0x97, 0x4B,
0x39, 0x76, 0xC2, 0xC1, 0x56, 0xCB, 0x70, 0xFD, 0x3B, 0x3E, 0x52, 0x57, 0x81, 0x5D, 0x56, 0x8D,
0x51, 0x90, 0xD4, 0x76, 0xD7, 0x55, 0x16, 0x02, 0x6D, 0xF2, 0x4D, 0xE1, 0x0E, 0x96, 0x4F, 0xA1,
0x3A, 0xA0, 0x60, 0x59, 0x64, 0x04, 0x1A, 0xE4, 0x67, 0xB6, 0xED, 0x3F, 0x74, 0x20, 0x55, 0x1F,
0xFB, 0x23, 0x92, 0x91, 0x53, 0xC8, 0x65, 0xAB, 0x9D, 0x51, 0xD6, 0x73, 0xDE, 0x01, 0xB1, 0x80,
0xB7, 0xC0, 0xD6, 0x80, 0x1C, 0x2E, 0x3C, 0x83, 0x63, 0xEE, 0xBC, 0x33, 0x25, 0xE2, 0x0E, 0x7A,
0x67, 0xDE, 0x3F, 0x71, 0x14, 0x49, 0x9C, 0x92, 0x93, 0x0D, 0x26, 0x9A, 0x0E, 0xDA, 0xED, 0x6F,
0xA4, 0x89, 0x0C, 0x1B, 0xF0, 0xA1, 0xDF, 0xE1, 0x9E, 0x3C, 0x04, 0x78, 0xE4, 0xAB, 0x6D, 0xFF,
0x9C, 0xAF, 0xCA, 0xC7, 0x88, 0x17, 0x9C, 0xE5, 0xB7, 0x33, 0x6D, 0xDC, 0xED, 0x8F, 0x6C, 0x18,
0x1D, 0x71, 0x06, 0xB1, 0xC5, 0xE2, 0xCF, 0x13, 0x77, 0x81, 0xC5, 0xB7, 0x14, 0x0A, 0x6B,
0x40, 0x26, 0xA0, 0x88, 0xD1, 0x62, 0x6A, 0xB3, 0x50, 0x12, 0x89, 0x9B, 0xB5, 0x83, 0x9B, 0x37
}

```

18.106.2.2 Embroidery Format (.pcq)

The Pfaff vip format is stitch-only.

18.107 extern/libembroidery/src/formats/format_vp3.c File Reference

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include "../embroidery_internal.h"

```

Functions

- `unsigned char * vp3ReadString (FILE *file)`
- `int vp3Decode (unsigned char inputByte)`
- `short vp3DecodeInt16 (unsigned short inputByte)`
- `vp3Hoop vp3ReadHoopSection (FILE *file)`
- `char readVp3 (EmbPattern *pattern, FILE *file)`
- `void vp3WriteStringLen (FILE *file, const char *str, int len)`
- `void vp3WriteString (FILE *file, const char *str)`
- `void vp3PatchByteCount (FILE *file, int offset, int adjustment)`
- `char writeVp3 (EmbPattern *pattern, FILE *file)`

18.107.1 Function Documentation

18.107.1.1 `readVp3()` `char readVp3 (`

```

    EmbPattern * pattern,
    FILE * file )

```

18.107.1.2 `vp3Decode()` `int vp3Decode (`

```

    unsigned char inputByte )

```

18.107.1.3 `vp3DecodeInt16()` `short vp3DecodeInt16 (`

```

    unsigned short inputByte )

```

18.107.1.4 `vp3PatchByteCount()` `void vp3PatchByteCount (`

```

    FILE * file,
    int offset,
    int adjustment )

```

18.107.1.5 vp3ReadHoopSection() `vp3Hoop vp3ReadHoopSection (FILE * file)`

18.107.1.6 vp3ReadString() `unsigned char * vp3ReadString (FILE * file)`

18.107.1.7 Embroidery Format (.pcq) The Pfaff vp3 format is stitch-only.

18.107.1.8 vp3WriteString() `void vp3WriteString (FILE * file, const char * str)`

18.107.1.9 vp3WriteStringLen() `void vp3WriteStringLen (FILE * file, const char * str, int len)`

18.107.1.10 writeVp3() `char writeVp3 (EmbPattern * pattern, FILE * file)`

18.108 extern/libembroidery/src/formats/format_xxx.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include "../embroidery_internal.h"
```

Functions

- `char xxxDecodeByte (unsigned char inputByte)`
- `char readXxx (EmbPattern *pattern, FILE *file)`
- `void xxxEncodeStop (FILE *file, EmbStitch s)`
- `void xxxEncodeStitch (FILE *file, EmbReal deltaX, EmbReal deltaY, int flags)`
- `void xxxEncodeDesign (FILE *file, EmbPattern *p)`
- `char writeXxx (EmbPattern *pattern, FILE *file)`

18.108.1 Function Documentation

18.108.1.1 readXxx() `char readXxx (EmbPattern * pattern, FILE * file)`

18.108.1.2 writeXxx() `char writeXxx (EmbPattern * pattern, FILE * file)`

18.108.1.3 xxxDecodeByte() `char xxxDecodeByte (unsigned char inputByte)`

18.108.1.4 Embroidery Format (.xxx) The Singer xxx format is stitch-only.

18.108.1.5 `xxxEncodeDesign()` void `xxxEncodeDesign` (

```
FILE * file,
EmbPattern * p )
```

18.108.1.6 `xxxEncodeStitch()` void `xxxEncodeStitch` (

```
FILE * file,
EmbReal deltaX,
EmbReal deltaY,
int flags )
```

18.108.1.7 `xxxEncodeStop()` void `xxxEncodeStop` (

```
FILE * file,
EmbStitch s )
```

18.109 `extern/libembroidery/src/formats/format_zsk.c` File Reference

```
#include <stdio.h>
#include <string.h>
#include "../embroidery_internal.h"
```

Functions

- char `readZsk` (`EmbPattern` *`pattern`, `FILE` *`file`)
- char `writeZsk` (`EmbPattern` *`pattern`, `FILE` *`file`)

18.109.1 Detailed Description

The ZSK USA Embroidery Format (.zsk)

The ZSK USA zsk format is stitch-only.

18.109.2 Function Documentation

18.109.2.1 `readZsk()` char `readZsk` (

```
EmbPattern * pattern,
FILE * file )
```

18.109.2.2 `writeZsk()` char `writeZsk` (

```
EmbPattern * pattern,
FILE * file )
```

18.110 `extern/libembroidery/src/geometry.c` File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "embroidery.h"
```

Functions

- **`EmbGeometry * embGeometry_init (int type_in)`**
Our generic object interface backends to each individual type.
- **`void embGeometry_free (EmbGeometry *obj)`**
Free the memory occupied by a non-stitch geometry object.
- **`void embGeometry_move (EmbGeometry *obj, EmbVector delta)`**
Translate obj by the vector delta.
- **`EmbRect embGeometry_boundingRect (EmbGeometry *obj)`**
Calculate the bounding box of geometry obj based on what kind of geometric object it is.
- **`void embGeometry_vulcanize (EmbGeometry *obj)`**
Toggle the rubber mode of the object.

18.110.1 Function Documentation

18.110.1.1 **`embGeometry_boundingRect()`** `EmbRect embGeometry_boundingRect (` `EmbGeometry * obj)`

Calculate the bounding box of geometry *obj* based on what kind of geometric object it is.

obj A pointer to the geometry memory.

Returns

`EmbRect` The bounding box in the same scale as the input geometry.

In the case of a failure the bounding box returned is always the unit square with top left corner at (0, 0).

18.110.1.2 **`embGeometry_free()`** `void embGeometry_free (` `EmbGeometry * obj)`

Free the memory occupied by a non-stitch geometry object.

obj Pointer to geometry memory.

18.110.1.3 **`embGeometry_init()`** `EmbGeometry * embGeometry_init (` `int type_in)`

Our generic object interface backends to each individual type.

type_in

Returns

`EmbGeometry*`

18.110.1.4 **`embGeometry_move()`** `void embGeometry_move (` `EmbGeometry * obj,` `EmbVector delta)`

Translate *obj* by the vector *delta*.

obj A pointer to the geometry memory. *delta* A vector in the 0.1mm scale to offset the geometry by.

18.110.1.5 **`embGeometry_vulcanize()`** `void embGeometry_vulcanize (` `EmbGeometry * obj)`

Toggle the rubber mode of the object.

obj

Todo Review. This could be controlled by a simple flag.

18.111 extern/libembroidery/src/geometry/arc.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "../embroidery.h"
```

Functions

- `EmbArc embArc_init (void)`
- `char embArc_clockwise (EmbArc arc)`
- `void getArcCenter (EmbArc arc, EmbVector *arcCenter)`
- `char getArcDataFromBulge (EmbReal bulge, EmbArc *arc, EmbReal *arcCenterX, EmbReal *arcCenterY, EmbReal *radius, EmbReal *diameter, EmbReal *chord, EmbReal *chordMidX, EmbReal *chordMidY, EmbReal *sagitta, EmbReal *apothem, EmbReal *incAngleInDegrees, char *clockwise)`
- `char clockwise (EmbGeometry *obj)`
- `void embArc_setCenter (EmbArc *arc, EmbVector point)`
- `void embArc_setRadius (EmbArc *arc, float radius)`
- `void embArc_setStartAngle (EmbArc *arc, float angle)`
- `void embArc_setEndAngle (EmbArc *arc, float angle)`
- `float embArc_startAngle (EmbArc arc)`
- `float embArc_endAngle (EmbArc arc)`
- `float embArc_area (EmbArc arc)`
- `float embArc_arcLength (EmbArc arc)`
- `float embArc_chord (EmbArc arc)`
- `float embArc_includedAngle (EmbArc arc)`
- `char Arc_clockwise ()`
- `void embArc_updatePath (EmbArc arc)`
- `void embArc_paint (void)`
- `void embArc_updateRubber (EmbArc arc, int pattern, int layer, int index)`
- `EmbVector embArc_mouseSnapPoint (EmbArc arc, EmbVector mousePoint)`
- `void embArc_gripEdit (EmbArc *arc, EmbVector before, EmbVector after)`
- `void set_object_color (EmbGeometry *obj, EmbColor color)`
- `void embBaseSetColorRGB (EmbGeometry *obj, unsigned int rgb)`
- `void Base_setLineType (EmbGeometry *obj, int lineType)`
- `void Base_setLineWidth (EmbGeometry *obj, float lineWidth)`
- `EmbVector Base_objectRubberPoint (EmbGeometry *obj, const char *key)`
- `const char * Base_objectRubberText (EmbGeometry *obj, const char *key)`
- `void embCircle_prompt (const char *str)`
- `void embCircle_setArea (EmbCircle *circle, float area)`
- `void embCircle_setCircumference (EmbCircle *circle, float circumference)`
- `void embEllipse_main ()`
- `void embEllipse_click (float x, float y)`
- `EmbVector embRect_bottomLeft (EmbRect rect)`
- `EmbVector embRect_bottomRight (EmbRect rect)`

18.111.1 Function Documentation

18.111.1.1 Arc_clockwise() `char Arc_clockwise ()`

18.111.1.2 Base_objectRubberPoint() `EmbVector` `Base_objectRubberPoint` (

```
EmbGeometry * obj,
const char * key )
```

18.111.1.3 Base_objectRubberText() `const char *` `Base_objectRubberText` (

```
EmbGeometry * obj,
const char * key )
```

18.111.1.4 Base_setLineType() `void` `Base_setLineType` (

```
EmbGeometry * obj,
int lineType )
```

18.111.1.5 Base_setLineWeight() `void` `Base_setLineWeight` (

```
EmbGeometry * obj,
float lineWeight )
```

18.111.1.6 clockwise() `char` `clockwise` (

```
EmbGeometry * obj )
```

18.111.1.7 embArc_arcLength() `float` `embArc_arcLength` (

```
EmbArc arc )
```

18.111.1.8 embArc_area() `float` `embArc_area` (

```
EmbArc arc )
```

18.111.1.9 embArc_chord() `float` `embArc_chord` (

```
EmbArc arc )
```

18.111.1.10 embArc_clockwise() `char` `embArc_clockwise` (

```
EmbArc arc )
```

18.111.1.11 embArc_endAngle() `float` `embArc_endAngle` (

```
EmbArc arc )
```

18.111.1.12 embArc_gripEdit() `void` `embArc_gripEdit` (

```
EmbArc * arc,
EmbVector before,
EmbVector after )
```

18.111.1.13 embArc_includedAngle() `float` `embArc_includedAngle` (

```
EmbArc arc )
```

- 18.111.1.14 `embArc_init()`** `EmbArc` `embArc_init` (
- `void`)
- 18.111.1.15 `embArc_mouseSnapPoint()`** `EmbVector` `embArc_mouseSnapPoint` (
- `EmbArc` `arc`,
- `EmbVector` `mousePoint`)
- 18.111.1.16 `embArc_paint()`** `void` `embArc_paint` (
- `void`)
- 18.111.1.17 `embArc_setCenter()`** `void` `embArc_setCenter` (
- `EmbArc` * `arc`,
- `EmbVector` `point`)
- 18.111.1.18 `embArc_setEndAngle()`** `void` `embArc_setEndAngle` (
- `EmbArc` * `arc`,
- `float` `angle`)
- 18.111.1.19 `embArc_setRadius()`** `void` `embArc_setRadius` (
- `EmbArc` * `arc`,
- `float` `radius`)
- 18.111.1.20 `embArc_setStartAngle()`** `void` `embArc_setStartAngle` (
- `EmbArc` * `arc`,
- `float` `angle`)
- 18.111.1.21 `embArc_startAngle()`** `float` `embArc_startAngle` (
- `EmbArc` `arc`)
- 18.111.1.22 `embArc_updatePath()`** `void` `embArc_updatePath` (
- `EmbArc` `arc`)
- 18.111.1.23 `embArc_updateRubber()`** `void` `embArc_updateRubber` (
- `EmbArc` `arc`,
- `int` `pattern`,
- `int` `layer`,
- `int` `index`)
- 18.111.1.24 `embBaseSetColorRGB()`** `void` `embBaseSetColorRGB` (
- `EmbGeometry` * `obj`,
- `unsigned int` `rgb`)
- 18.111.1.25 `embCircle_prompt()`** `void` `embCircle_prompt` (
- `const char` * `str`)

18.111.1.26 `embCircle_setArea()` void embCircle_setArea (EmbCircle * circle, float area)

18.111.1.27 `embCircle_setCircumference()` void embCircle_setCircumference (EmbCircle * circle, float circumference)

18.111.1.28 `embEllipse_click()` void embEllipse_click (float x, float y)

18.111.1.29 `embEllipse_main()` void embEllipse_main ()

18.111.1.30 `embRect_bottomLeft()` EmbVector embRect_bottomLeft (EmbRect rect)

18.111.1.31 `embRect_bottomRight()` EmbVector embRect_bottomRight (EmbRect rect)

18.111.1.32 `getArcCenter()` void getArcCenter (EmbArc arc, EmbVector * arcCenter)

18.111.1.33 `getArcDataFromBulge()` char getArcDataFromBulge (EmbReal bulge, EmbArc * arc, EmbReal * arcCenterX, EmbReal * arcCenterY, EmbReal * radius, EmbReal * diameter, EmbReal * chord, EmbReal * chordMidX, EmbReal * chordMidY, EmbReal * sagitta, EmbReal * apothem, EmbReal * incAngleInDegrees, char * clockwise)

18.111.1.34 `set_object_color()` void set_object_color (EmbGeometry * obj, EmbColor color)

18.112 *extern/libembroidery/src/geometry/circle.c* File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
```

```
#include "../embroidery.h"
```

Functions

- `EmbCircle embCircle_init (void)`
- `EmbReal embCircle_area (EmbCircle circle)`
- `EmbReal embCircle_circumference (EmbCircle circle)`
- `int getCircleCircleIntersections (EmbCircle c0, EmbCircle c1, EmbVector *p0, EmbVector *p1)`
- `int getCircleTangentPoints (EmbCircle c, EmbVector point, EmbVector *t0, EmbVector *t1)`

18.112.1 Function Documentation

18.112.1.1 `embCircle_area()` `EmbReal embCircle_area (`
`EmbCircle circle)`

18.112.1.2 `embCircle_circumference()` `EmbReal embCircle_circumference (`
`EmbCircle circle)`

18.112.1.3 `embCircle_init()` `EmbCircle embCircle_init (`
`void)`

18.112.1.4 `getCircleCircleIntersections()` `int getCircleCircleIntersections (`
`EmbCircle c0,`
`EmbCircle c1,`
`EmbVector * p0,`
`EmbVector * p1)`

18.112.1.5 `getCircleTangentPoints()` `int getCircleTangentPoints (`
`EmbCircle c,`
`EmbVector point,`
`EmbVector * t0,`
`EmbVector * t1)`

18.113 `extern/libembroidery/src/geometry/ellipse.c` File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "../embroidery.h"
```

Functions

- `EmbEllipse embEllipse_init (void)`
- `EmbReal embEllipse_area (EmbEllipse ellipse)`
- `EmbReal embEllipse_perimeter (EmbEllipse ellipse)`
- `EmbReal embEllipse_diameterX (EmbEllipse ellipse)`
- `EmbReal embEllipse_diameterY (EmbEllipse ellipse)`
- `EmbReal embEllipse_width (EmbEllipse ellipse)`
- `EmbReal embEllipse_height (EmbEllipse ellipse)`

- void `embEllipse_setSize` (float width, float height)
- void `embEllipse_setRadiusMajor` (float radius)
- void `embEllipse_setRadiusMinor` (float radius)
- void `embEllipse_setDiameterMajor` (`EmbEllipse` *ellipse, float diameter)
- void `embEllipse_setDiameterMinor` (`EmbEllipse` *ellipse, float diameter)
- `EmbVector` `ellipse_objectQuadrant0` (`EmbEllipse` *ellipse)
- `EmbVector` `ellipse_objectQuadrant90` (`EmbEllipse` *ellipse)
- `EmbVector` `ellipse_objectQuadrant180` (`EmbEllipse` *ellipse)
- `EmbVector` `ellipse_objectQuadrant270` (`EmbEllipse` *ellipse)
- void `embEllipse_updatePath` ()

18.113.1 Function Documentation

18.113.1.1 `ellipse_objectQuadrant0()` `EmbVector` `ellipse_objectQuadrant0` (
`EmbEllipse` * `ellipse`)

18.113.1.2 `ellipse_objectQuadrant180()` `EmbVector` `ellipse_objectQuadrant180` (
`EmbEllipse` * `ellipse`)

18.113.1.3 `ellipse_objectQuadrant270()` `EmbVector` `ellipse_objectQuadrant270` (
`EmbEllipse` * `ellipse`)

18.113.1.4 `ellipse_objectQuadrant90()` `EmbVector` `ellipse_objectQuadrant90` (
`EmbEllipse` * `ellipse`)

18.113.1.5 `embEllipse_area()` `EmbReal` `embEllipse_area` (
`EmbEllipse` `ellipse`)

18.113.1.6 `embEllipse_diameterX()` `EmbReal` `embEllipse_diameterX` (
`EmbEllipse` `ellipse`)

18.113.1.7 `embEllipse_diameterY()` `EmbReal` `embEllipse_diameterY` (
`EmbEllipse` `ellipse`)

18.113.1.8 `embEllipse_height()` `EmbReal` `embEllipse_height` (
`EmbEllipse` `ellipse`)

18.113.1.9 `embEllipse_init()` `EmbEllipse` `embEllipse_init` (
`void`)

18.113.1.10 `embEllipse_perimeter()` `EmbReal` `embEllipse_perimeter` (
`EmbEllipse` `ellipse`)

```
18.113.1.11 embEllipse_setDiameterMajor() void embEllipse_setDiameterMajor (
    EmbEllipse * ellipse,
    float diameter )
```

```
18.113.1.12 embEllipse_setDiameterMinor() void embEllipse_setDiameterMinor (
    EmbEllipse * ellipse,
    float diameter )
```

```
18.113.1.13 embEllipse_setRadiusMajor() void embEllipse_setRadiusMajor (
    float radius )
```

```
18.113.1.14 embEllipse_setRadiusMinor() void embEllipse_setRadiusMinor (
    float radius )
```

```
18.113.1.15 embEllipse_setSize() void embEllipse_setSize (
    float width,
    float height )
```

```
18.113.1.16 embEllipse_updatePath() void embEllipse_updatePath ( )
```

```
18.113.1.17 embEllipse_width() EmbReal embEllipse_width (
    EmbEllipse ellipse )
```

18.114 extern/libembroidery/src/geometry/functions.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "../embroidery.h"
```

Functions

- int **emb_round** (EmbReal x)
- EmbReal **radians** (EmbReal degree)
- EmbReal **degrees** (EmbReal radian)

18.114.1 Function Documentation

```
18.114.1.1 degrees() EmbReal degrees (
    EmbReal radian ) [inline]
```

```
18.114.1.2 emb_round() int emb_round (
    EmbReal x )
```

18.114.1.3 radians() `EmbReal radians (`
`EmbReal degree) [inline]`

18.115 extern/libembroidery/src/geometry/line.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "../embroidery.h"
```

Functions

- `void embLine_normalVector (EmbLine line, EmbVector *result, int clockwise)`
- `EmbVector embLine_toVector (EmbLine line)`
- `EmbVector embLine_intersectionPoint (EmbLine line1, EmbLine line2)`

18.115.1 Function Documentation

18.115.1.1 embLine_intersectionPoint() `EmbVector embLine_intersectionPoint (`
`EmbLine line1,`
`EmbLine line2)`

18.115.1.2 embLine_normalVector() `void embLine_normalVector (`
`EmbLine line,`
`EmbVector * result,`
`int clockwise)`

Finds the normalized vector perpendicular (clockwise) to the line given by v1->v2 (normal to the line)

18.115.1.3 embLine_toVector() `EmbVector embLine_toVector (`
`EmbLine line)`

18.116 extern/libembroidery/src/geometry/path.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "../embroidery.h"
```

18.117 extern/libembroidery/src/geometry/polygon.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "../embroidery.h"
```

18.118 extern/libembroidery/src/geometry/polyline.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "../embroidery.h"
```

18.119 extern/libembroidery/src/geometry/rect.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "../embroidery.h"
```

Functions

- `EmbRect embRect_init (void)`
- `EmbReal embRect_area (EmbRect rect)`

18.119.1 Function Documentation

18.119.1.1 `embRect_area()` `EmbReal embRect_area (`
`EmbRect rect)`

18.119.1.2 `embRect_init()` `EmbRect embRect_init (`
`void)`

18.120 extern/libembroidery/src/geometry/text.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "../embroidery.h"
```

Functions

- `void textSingle_setTextFont (const char *font)`
- `void textSingle_setJustify (const char *justify)`
- `void textSingle_setTextSize (float size)`
- `void textSingle_setTextStyle (char bold, char italic, char under, char strike, char over)`
- `void textSingle_setTextBold (char val)`
- `void textSingle_setTextItalic (char val)`
- `void textSingle_setTextUnderline (char val)`
- `void textSingle_setTextStrikeOut (char val)`
- `void textSingle_setTextOverline (char val)`
- `void textSingle_setTextBackward (char val)`
- `void textSingle_setTextUpsideDown (char val)`
- `void textSingle_paint ()`
- `void textSingle_updateRubber ()`
- `EmbVector textSingle_mouseSnapPoint (EmbVector mousePoint)`
- `void textSingle_gripEdit (EmbVector before, EmbVector after)`

18.120.1 Function Documentation

18.120.1.1 `textSingle_gripEdit()` `void textSingle_gripEdit (`
`EmbVector before,`
`EmbVector after)`

18.120.1.2 `textSingle_mouseSnapPoint()` `EmbVector` `textSingle_mouseSnapPoint (EmbVector mousePoint)`

18.120.1.3 `textSingle_paint()` `void` `textSingle_paint ()`

18.120.1.4 `textSingle_setJustify()` `void` `textSingle_setJustify (const char * justify)`

18.120.1.5 `textSingle_setTextBackward()` `void` `textSingle_setTextBackward (char val)`

18.120.1.6 `textSingle_setTextBold()` `void` `textSingle_setTextBold (char val)`

18.120.1.7 `textSingle_setTextFont()` `void` `textSingle_setTextFont (const char * font)`

18.120.1.8 `textSingle_setTextItalic()` `void` `textSingle_setTextItalic (char val)`

18.120.1.9 `textSingle_setTextOverline()` `void` `textSingle_setTextOverline (char val)`

18.120.1.10 `textSingle_setTextSize()` `void` `textSingle_setTextSize (float size)`

18.120.1.11 `textSingle_setTextStrikeOut()` `void` `textSingle_setTextStrikeOut (char val)`

18.120.1.12 `textSingle_setTextStyle()` `void` `textSingle_setTextStyle (char bold, char italic, char under, char strike, char over)`

18.120.1.13 `textSingle_setTextUnderline()` `void` `textSingle_setTextUnderline (char val)`

18.120.1.14 `textSingle_setTextUpsideDown()` `void` `textSingle_setTextUpsideDown (char val)`

18.120.1.15 `textSingle_updateRubber()` `void textSingle_updateRubber ()`

18.121 `extern/libembroidery/src/geometry/vector.c` File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "../embroidery.h"
```

Functions

- `void embVector_normalize (EmbVector vector, EmbVector *result)`
- `void embVector_multiply (EmbVector vector, EmbReal magnitude, EmbVector *result)`
- `EmbVector embVector_add (EmbVector a, EmbVector b)`
- `EmbVector embVector_average (EmbVector a, EmbVector b)`
- `EmbVector embVector_subtract (EmbVector v1, EmbVector v2)`
- `EmbReal embVector_dot (EmbVector a, EmbVector b)`
- `EmbReal embVector_cross (EmbVector a, EmbVector b)`
The "cross product" as vectors a and b returned as a real value.
- `void embVector_transpose_product (EmbVector v1, EmbVector v2, EmbVector *result)`
- `EmbReal embVector_length (EmbVector vector)`
- `EmbReal embVector_relativeX (EmbVector a1, EmbVector a2, EmbVector a3)`
- `EmbReal embVector_relativeY (EmbVector a1, EmbVector a2, EmbVector a3)`
- `EmbReal embVector_angle (EmbVector v)`
- `EmbVector embVector_unit (EmbReal alpha)`
- `EmbReal embVector_distance (EmbVector a, EmbVector b)`

18.121.1 Function Documentation

18.121.1.1 `embVector_add()` `EmbVector embVector_add (`

```
    EmbVector a,
    EmbVector b )
```

The sum of vectors *a* and *b* returned as a vector.

Equivalent to:

$$\mathbf{c} = \mathbf{a} + \mathbf{b} = \begin{pmatrix} a_x + b_x \\ a_y + b_y \end{pmatrix}$$

18.121.1.2 `embVector_angle()` `EmbReal embVector_angle (`

```
    EmbVector v )
```

The angle, measured anti-clockwise from the x-axis, of a vector *v*.

18.121.1.3 `embVector_average()` `EmbVector embVector_average (`

```
    EmbVector a,
    EmbVector b )
```

The average of vectors *v1* and *v2* returned as a vector.

Equivalent to:

$$\mathbf{c} = \frac{\mathbf{a} + \mathbf{b}}{2} = \begin{pmatrix} \frac{a_x + b_x}{2} \\ \frac{a_y + b_y}{2} \end{pmatrix}$$

```
18.121.1.4 embVector_cross() EmbReal embVector_cross (
    EmbVector a,
    EmbVector b )
```

The "cross product" as vectors *a* and *b* returned as a real value.

Technically, this is the magnitude of the cross product when the embroidery is placed in the z=0 plane (since the cross product is defined for 3-dimensional vectors). That is:

$$|c| = \left| \begin{pmatrix} a_x \\ a_y \\ 0 \end{pmatrix} \times \begin{pmatrix} b_x \\ b_y \\ 0 \end{pmatrix} \right| = \left| \begin{pmatrix} 0 \\ 0 \\ a_x b_y - a_y b_x \end{pmatrix} \right| = a_x b_y - a_y b_x$$

```
18.121.1.5 embVector_distance() EmbReal embVector_distance (
    EmbVector a,
    EmbVector b )
```

The distance between *a* and *b* returned as a real value.

$$d = |\mathbf{a} - \mathbf{b}| = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$$

```
18.121.1.6 embVector_dot() EmbReal embVector_dot (
    EmbVector a,
    EmbVector b )
```

The dot product as vectors *v1* and *v2* returned as a EmbReal.

Equivalent to:

$$c = \mathbf{a} \cdot \mathbf{b} = a_x b_x + a_y b_y$$

```
18.121.1.7 embVector_length() EmbReal embVector_length (
    EmbVector vector )
```

The length or absolute value of the vector *vector*.

Equivalent to:

$$|v| = \sqrt{v_x^2 + v_y^2}$$

```
18.121.1.8 embVector_multiply() void embVector_multiply (
    EmbVector vector,
    EmbReal magnitude,
    EmbVector * result )
```

The scalar multiple *magnitude* of a vector *vector*. Returned as *result*.

Todo make result return argument.

```
18.121.1.9 embVector_normalize() void embVector_normalize (
    EmbVector vector,
    EmbVector * result )
```

Finds the unit length vector *result* in the same direction as *vector*.

Equivalent to:

$$\mathbf{u} = \frac{\mathbf{v}}{|\mathbf{v}|}$$

Todo make result return argument.

```
18.121.1.10 embVector_relativeX() EmbReal embVector_relativeX (
    EmbVector a1,
    EmbVector a2,
    EmbVector a3 )
```

The x-component of the vector

```
18.121.1.11 embVector_relativeY() EmbReal embVector_relativeY (
    EmbVector a1,
    EmbVector a2,
    EmbVector a3 )
```

The y-component of the vector

```
18.121.1.12 embVector_subtract() EmbVector embVector_subtract (
    EmbVector v1,
    EmbVector v2 )
```

The difference between vectors *v1* and *v2* returned as *result*.

Equivalent to:

$$\mathbf{c} = \mathbf{a} - \mathbf{b} = \begin{pmatrix} a_x - b_x \\ a_y - b_y \end{pmatrix}$$

```
18.121.1.13 embVector_transpose_product() void embVector_transpose_product (
    EmbVector v1,
    EmbVector v2,
    EmbVector * result )
```

Since we aren't using full vector algebra here, all vectors are "vertical". so this is like the product $\mathbf{v1}^T \mathbf{v2}$ for our vectors *v1* and *v2* so a "component-wise product". The result is stored at the pointer *result*.

That is $(1 \ 0) (a) = (xa) (x \ y)(0 \ 1) (b) = (yb)$

```
18.121.1.14 embVector_unit() EmbVector embVector_unit (
    EmbReal alpha )
```

The unit vector in the direction *angle*.

$$\mathbf{a}_\alpha = \begin{pmatrix} \cos(\alpha) \\ \sin(\alpha) \end{pmatrix}$$

18.122 extern/libembroidery/src/image.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "embroidery_internal.h"
```

Functions

- void [writeImage](#) (FILE *file, unsigned char image[][48])
- float [image_diff](#) (unsigned char *a, unsigned char *b, int size)

18.122.1 Detailed Description

This backends to the stb libraries and nanosvg library.

Use Python PEP7 for coding style.

18.122.2 Function Documentation

```
18.122.2.1 image_diff() float image_diff (
    unsigned char * a,
    unsigned char * b,
    int size )
```

The distance between the arrays *a* and *b* of length *size*.

```
18.122.2.2 writelImage() void writeImage (
    FILE * file,
    unsigned char image[ ][48] )
```

Write a PES embedded *image* to the given *file* pointer.

18.123 extern/libembroidery/src/main.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include <time.h>
#include "embroidery_internal.h"
```

Macros

- #define FLAG_TO 0
- #define FLAG_TO_SHORT 1
- #define FLAG_HELP 2
- #define FLAG_HELP_SHORT 3
- #define FLAG_FORMATS 4
- #define FLAG_FORMATS_SHORT 5
- #define FLAG QUIET 6
- #define FLAG_QUIET_SHORT 7
- #define FLAG_VERBOSE 8
- #define FLAG_VERBOSE_SHORT 9
- #define FLAG_VERSION 10
- #define FLAG_VERSION_SHORT 11
- #define FLAG_CIRCLE 12
- #define FLAG_CIRCLE_SHORT 13
- #define FLAG_ELLIPSE 14
- #define FLAG_ELLIPSE_SHORT 15
- #define FLAG_LINE 16
- #define FLAG_LINE_SHORT 17
- #define FLAG_POLYGON 18
- #define FLAG_POLYGON_SHORT 19
- #define FLAG_POLYLINE 20
- #define FLAG_POLYLINE_SHORT 21
- #define FLAG_RENDER 22
- #define FLAG_RENDER_SHORT 23
- #define FLAG_SATIN 24
- #define FLAG_SATIN_SHORT 25
- #define FLAG_STITCH 26
- #define FLAG_STITCH_SHORT 27
- #define FLAG_TEST 28
- #define FLAG_FULL_TEST_SUITE 29
- #define FLAG_HILBERT_CURVE 30
- #define FLAG_SIERSPKI_TRIANGLE 31
- #define FLAG_FILL 32

- #define FLAG_FILL_SHORT 33
- #define FLAG_SIMULATE 34
- #define FLAG_COMBINE 35
- #define FLAG_CROSS_STITCH 36
- #define NUM_FLAGS 37

Functions

- void `embVector_print` (`EmbVector` v, char *label)
v label
- void `embArc_print` (`EmbArc` arc)
arc
- int `check_header_present` (FILE *file, int minimum_header_length)
file minimum_header_length
- unsigned int `sectorSize` (`bcf_file` *bcfFile)
bcfFile
- int `haveExtraDIFATSectors` (`bcf_file` *file)
file
- int `seekToSector` (`bcf_file` *bcfFile, FILE *file, const unsigned int sector)
bcfFile file sector
- void `parseDIFATSectors` (FILE *file, `bcf_file` *bcfFile)
file bcfFile
- int `bcfFile_read` (FILE *file, `bcf_file` *bcfFile)
file bcfFile
- FILE * `GetFile` (`bcf_file` *bcfFile, FILE *file, char *fileToFind)
Get the File object.
- void `bcf_file_free` (`bcf_file` *bcfFile)
bcfFile
- `bcf_file_difat` * `bcf_difat_create` (FILE *file, unsigned int fatSectors, const unsigned int sectorSize)
file fatSectors sectorSize
- unsigned int `entriesInDifatSector` (`bcf_file_difat` *fat)
fat
- unsigned int `readFullSector` (FILE *file, `bcf_file_difat` *bcfFile, unsigned int *difatEntriesToRead)
file bcfFile difatEntriesToRead
- void `parseDirectoryEntryName` (FILE *file, `bcf_directory_entry` *dir)
file dir
- `bcf_directory` * `CompoundFileDirectory` (const unsigned int maxNumberOfDirectoryEntries)
maxNumberOfDirectoryEntries
- `EmbTime` `parseTime` (FILE *file)
file
- `bcf_directory_entry` * `CompoundFileDirectoryEntry` (FILE *file)
file
- void `readNextSector` (FILE *file, `bcf_directory` *dir)
file dir
- void `bcf_directory_free` (`bcf_directory` **dir)
dir
- `bcf_file_fat` * `bcfFileFat_create` (const unsigned int sectorSize)
sectorSize
- void `loadFatFromSector` (`bcf_file_fat` *fat, FILE *file)
fat file
- `bcf_file_header` `bcfFileHeader_read` (FILE *file)

- void `embSatinOutline_generateSatinOutline` (`EmbArray` *`lines`, `EmbReal` `thickness`, `EmbSatinOutline` *`result`)
lines thickness result
- `EmbArray` * `embSatinOutline_renderStitches` (`EmbSatinOutline` *`result`, `EmbReal` `density`)
result density
- void `write_24bit` (`FILE` *`file`, int `x`)
file x
- int `embColor_distance` (`EmbColor` `a`, `EmbColor` `b`)
a b
- void `embColor_read` (`FILE` *`f`, `EmbColor` *`c`, int `toRead`)
f c toRead
- void `embColor_write` (`FILE` *`f`, `EmbColor` `c`, int `toWrite`)
f c toWrite
- int `embThread_findNearestColor` (`EmbColor` `color`, `EmbColor` *`color_list`, int `n_colors`)
- int `embThread_findNearestThread` (`EmbColor` `color`, `EmbThread` *`thread_list`, int `n_threads`)
color thread_list n_threads
- `EmbThread` `embThread_getRandom` (void)
- void `binaryReadString` (`FILE` *`file`, char *`buffer`, int `maxLength`)
file buffer maxLength
- void `binaryReadUnicodeString` (`FILE` *`file`, char *`buffer`, const int `stringLength`)
file buffer stringLength
- int `stringInArray` (const char *`s`, const char **`array`)
- int `emb_readline` (`FILE` *`file`, char *`line`, int `maxLength`)
file line maxLength
- void `get_trim_bounds` (char const *`s`, char const **`firstWord`, char const **`trailingSpace`)
Get the trim bounds object.
- char * `copy_trim` (char const *`s`)
s
- char * `emb_optOut` (`EmbReal` `num`, char *`str`)
Optimizes the number (num) for output to a text file and returns it as a string (str).
- void `embTime_initNow` (`EmbTime` *`t`)
t
- `EmbTime` `embTime_time` (`EmbTime` *`t`)
t

Variables

- `EmbThread black_thread` = { { 0, 0, 0 }, "Black", "Black" }
- int `emb_verbose` = 0
Verbosity level.
- int `emb_error` = 0
Error code storage for optional control flow blocking.
- const `EmbReal embConstantPi` = 3.1415926535
- const unsigned int `difatEntriesInHeader` = 109
- const unsigned int `sizeOfFatEntry` = sizeof(unsigned int)
- const unsigned int `sizeOfDifatEntry` = 4
- const unsigned int `sizeOfChainingEntryAtEndOfDifatSector` = 4
- const unsigned int `sizeOfDirectoryEntry` = 128
- char const `WHITESPACE` [] = " \t\n\r"

18.123.1 Macro Definition Documentation

18.123.1.1 FLAG_CIRCLE #define FLAG_CIRCLE 12

18.123.1.2 FLAG_CIRCLE_SHORT #define FLAG_CIRCLE_SHORT 13

18.123.1.3 FLAG_COMBINE #define FLAG_COMBINE 35

18.123.1.4 FLAG_CROSS_STITCH #define FLAG_CROSS_STITCH 36

18.123.1.5 FLAG_ELLIPSE #define FLAG_ELLIPSE 14

18.123.1.6 FLAG_ELLIPSE_SHORT #define FLAG_ELLIPSE_SHORT 15

18.123.1.7 FLAG_FILL #define FLAG_FILL 32

18.123.1.8 FLAG_FILL_SHORT #define FLAG_FILL_SHORT 33

18.123.1.9 FLAG_FORMATS #define FLAG_FORMATS 4

18.123.1.10 FLAG_FORMATS_SHORT #define FLAG_FORMATS_SHORT 5

18.123.1.11 FLAG_FULL_TEST_SUITE #define FLAG_FULL_TEST_SUITE 29

18.123.1.12 FLAG_HELP #define FLAG_HELP 2

18.123.1.13 FLAG_HELP_SHORT #define FLAG_HELP_SHORT 3

18.123.1.14 FLAG_HILBERT_CURVE #define FLAG_HILBERT_CURVE 30

18.123.1.15 FLAG_LINE #define FLAG_LINE 16

18.123.1.16 FLAG_LINE_SHORT #define FLAG_LINE_SHORT 17

18.123.1.17 FLAG_POLYGON #define FLAG_POLYGON 18

18.123.1.18 FLAG_POLYGON_SHORT #define FLAG_POLYGON_SHORT 19

18.123.1.19 FLAG_POLYLINE #define FLAG_POLYLINE 20

18.123.1.20 FLAG_POLYLINE_SHORT #define FLAG_POLYLINE_SHORT 21

18.123.1.21 FLAG QUIET #define FLAG QUIET 6

18.123.1.22 FLAG QUIET SHORT #define FLAG QUIET SHORT 7

18.123.1.23 FLAG_RENDER #define FLAG_RENDER 22

18.123.1.24 FLAG_RENDER_SHORT #define FLAG_RENDER_SHORT 23

18.123.1.25 FLAG_SATIN #define FLAG_SATIN 24

18.123.1.26 FLAG_SATIN_SHORT #define FLAG_SATIN_SHORT 25

18.123.1.27 FLAG_SIERPINSKI_TRIANGLE #define FLAG_SIERPINSKI_TRIANGLE 31

18.123.1.28 FLAG_SIMULATE #define FLAG_SIMULATE 34

18.123.1.29 FLAG_STITCH #define FLAG_STITCH 26

18.123.1.30 FLAG_STITCH_SHORT #define FLAG_STITCH_SHORT 27

18.123.1.31 FLAG_TEST #define FLAG_TEST 28

18.123.1.32 FLAG_TO #define FLAG_TO 0

18.123.1.33 FLAG_TO_SHORT #define FLAG_TO_SHORT 1

18.123.1.34 FLAG_VERBOSE #define FLAG_VERBOSE 8

18.123.1.35 FLAG_VERBOSE_SHORT #define FLAG_VERBOSE_SHORT 9

18.123.1.36 FLAG_VERSION #define FLAG_VERSION 10

18.123.1.37 FLAG_VERSION_SHORT #define FLAG_VERSION_SHORT 11

18.123.1.38 NUM_FLAGS #define NUM_FLAGS 37

18.123.2 Function Documentation

18.123.2.1 bcf_difat_create() `bcf_file_difat * bcf_difat_create (`
 `FILE * file,`
 `unsigned int fatSectors,`
 `const unsigned int sectorSize)`
file *fatSectors* *sectorSize*

Returns

`bcf_file_difat*`

18.123.2.2 bcf_directory_free() `void bcf_directory_free (`
 `bcf_directory ** dir)`
dir

18.123.2.3 bcf_file_free() `void bcf_file_free (`
 `bcf_file * bcfFile)`
bcfFile

18.123.2.4 bcfFile_read() `int bcfFile_read (`
 `FILE * file,`
 `bcf_file * bcfFile)`
file *bcfFile*

Returns

`int`

18.123.2.5 bcfFileFat_create() `bcf_file_fat * bcfFileFat_create (`
 `const unsigned int sectorSize)`
sectorSize

Returns

`bcf_file_fat*`

18.123.2.6 bcfFileHeader_read() `bcf_file_header bcfFileHeader_read (`
 `FILE * file)`
file

Returns

`bcf_file_header`

18.123.2.7 binaryReadString() void binaryReadString (

```
FILE * file,
char * buffer,
int maxLength )
```

file buffer maxLength

18.123.2.8 binaryReadUnicodeString() void binaryReadUnicodeString (

```
FILE * file,
char * buffer,
const int stringLength )
```

file buffer stringLength

18.123.2.9 check_header_present() int check_header_present (

```
FILE * file,
int minimum_header_length )
```

file minimum_header_length

Returns

int

Checks that there are enough bytes to interpret the header, stops possible segfaults when reading in the header bytes.

Returns 0 if there aren't enough, or the length of the file if there are.

18.123.2.10 CompoundFileDirectory() *bcf_directory* * CompoundFileDirectory (

```
const unsigned int maxNumberOfDirectoryEntries )
```

maxNumberOfDirectoryEntries

Returns

*bcf_directory**

18.123.2.11 CompoundFileDirectoryEntry() *bcf_directory_entry* * CompoundFileDirectoryEntry (

```
FILE * file )
```

file

Returns

*bcf_directory_entry**

18.123.2.12 copy_trim() char * copy_trim (

```
char const * s )
```

s

Returns

char*

Todo decription

```
18.123.2.13 emb_optOut() char * emb_optOut (
    EmbReal num,
    char * str )
```

Optimizes the number (*num*) for output to a text file and returns it as a string (*str*).
num str

Returns

char*

```
18.123.2.14 emb_readline() int emb_readline (
    FILE * file,
    char * line,
    int maxLength )
```

file line maxLength

Returns

int

```
18.123.2.15 embArc_print() void embArc_print (
    EmbArc arc )
```

arc

Todo move to *arc.c*

```
18.123.2.16 embColor_distance() int embColor_distance (
    EmbColor a,
    EmbColor b )
```

a b

Returns

int

```
18.123.2.17 embColor_read() void embColor_read (
    FILE * f,
    EmbColor * c,
    int toRead )
```

f c toRead

```
18.123.2.18 embColor_write() void embColor_write (
    FILE * f,
    EmbColor c,
    int toWrite )
```

f c toWrite

```
18.123.2.19 embSatinOutline_generateSatinOutline() void embSatinOutline_generateSatinOutline (
    EmbArray * lines,
    EmbReal thickness,
    EmbSatinOutline * result )
```

lines thickness result

18.123.2.20 embSatinOutline_renderStitches() `EmbArray * embSatinOutline_renderStitches (EmbSatinOutline * result, EmbReal density)`

result density

Returns

`EmbArray*`

18.123.2.21 embThread_findNearestColor() `int embThread_findNearestColor (EmbColor color, EmbColor * color_list, int n_colors)`

Returns the closest color to the required color based on a list of available threads. The algorithm is a simple least squares search against the list. If the (square of) Euclidean 3-dimensional distance between the points in (red, green, blue) space is smaller then the index is saved and the remaining index is returned to the caller.

color The EmbColor color to match. *colors* The EmbThreadList pointer to start the search at. *mode* Is the argument an array of threads (0) or colors (1)?

Returns

`closestIndex` The entry in the ThreadList that matches.

18.123.2.22 embThread_findNearestThread() `int embThread_findNearestThread (EmbColor color, EmbThread * thread_list, int n_threads)`

color thread_list n_threads

Returns

`int`

18.123.2.23 embThread_getRandom() `EmbThread embThread_getRandom (void)`

Returns a random thread color, useful in filling in cases where the actual color of the thread doesn't matter but one needs to be declared to test or render a pattern.

Returns

`c` The resulting color.

18.123.2.24 embTime_initNow() `void embTime_initNow (EmbTime * t)`

t

18.123.2.25 embTime_time() `EmbTime embTime_time (EmbTime * t)`

t

Returns

`EmbTime`

18.123.2.26 embVector_print() void embVector_print (

```
EmbVector v,
char * label )
```

v label

move to [vector.c](#)

18.123.2.27 entriesInDifatSector() unsigned int entriesInDifatSector (

```
bcf_file_difat * fat )
```

fat

Returns

unsigned int

18.123.2.28 get_trim_bounds() void get_trim_bounds (

```
char const * s,
char const ** firstWord,
char const ** trailingSpace )
```

Get the trim bounds object.

s firstWord trailingSpace

18.123.2.29 GetFile() FILE * GetFile (

```
bcf_file * bcfFile,
FILE * file,
char * fileToFind )
```

Get the File object.

bcfFile file fileToFind

Returns

FILE*

18.123.2.30 haveExtraDIFATSectors() int haveExtraDIFATSectors (

```
bcf_file * file )
```

file

Returns

int

18.123.2.31 loadFatFromSector() void loadFatFromSector (

```
bcf_file_fat * fat,
FILE * file )
```

fat file

18.123.2.32 parseDIFATSectors() void parseDIFATSectors (

```
FILE * file,
bcf_file * bcfFile )
```

file bcfFile

18.123.2.33 parseDirectoryEntryName() void parseDirectoryEntryName (

```
FILE * file,
bcf_directory_entry * dir )
```

file dir

18.123.2.34 parseTime() `EmbTime` `parseTime (`
 `FILE * file)`

file

Returns

`EmbTime`

18.123.2.35 readFullSector() `unsigned int` `readFullSector (`
 `FILE * file,`
 `bcf_file_difat * bcfFile,`
 `unsigned int * difatEntriesToRead)`

file bcfFile difatEntriesToRead

Returns

`unsigned int`

18.123.2.36 readNextSector() `void` `readNextSector (`
 `FILE * file,`
 `bcf_directory * dir)`

file dir

18.123.2.37 sectorSize() `unsigned int` `sectorSize (`
 `bcf_file * bcfFile)`

bcfFile

Returns

`unsigned int`

18.123.2.38 seekToSector() `int` `seekToSector (`
 `bcf_file * bcfFile,`
 `FILE * file,`
 `const unsigned int sector)`

bcfFile file sector

Returns

`int`

18.123.2.39 stringInArray() `int` `stringInArray (`
 `const char * s,`
 `const char ** array)`

Tests for the presence of a string *s* in the supplied *array*.
The end of the array is marked by an empty string.

Returns

0 if not present 1 if present.

```
18.123.2.40 write_24bit() void write_24bit (
    FILE * file,
    int x )
file x
```

18.123.3 Variable Documentation

18.123.3.1 black_thread `EmbThread` `black_thread = { { 0, 0, 0 }, "Black", "Black" }`

18.123.3.2 difatEntriesInHeader `const unsigned int difatEntriesInHeader = 109`

18.123.3.3 emb_error `int emb_error = 0`
Error code storage for optional control flow blocking.

18.123.3.4 emb_verbose `int emb_verbose = 0`
Verbosity level.

18.123.3.5 embConstantPi `const EmbReal embConstantPi = 3.1415926535`

18.123.3.6 sizeOfChainingEntryAtEndOfDifatSector `const unsigned int sizeOfChainingEntryAtEndOfDifatSector = 4`

18.123.3.7 sizeOfDifatEntry `const unsigned int sizeOfDifatEntry = 4`

18.123.3.8 sizeOfDirectoryEntry `const unsigned int sizeOfDirectoryEntry = 128`

18.123.3.9 sizeOfFatEntry `const unsigned int sizeOfFatEntry = sizeof(unsigned int)`

18.123.3.10 WHITESPACE `char const WHITESPACE[] = " \t\n\r"`

18.124 extern/libembroidery/src/pattern.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include "embroidery_internal.h"
```

Functions

- `EmbPattern * embPattern_create (void)`
- `void embPattern_hideStitchesOverLength (EmbPattern *p, int length)`
- `int embPattern_addThread (EmbPattern *pattern, EmbThread thread)`
- `void embPattern_fixColorCount (EmbPattern *p)`

- void `embPattern_copystitch_listToPolylines` (`EmbPattern *p`)
- void `embPattern_copyPolylinesTostitch_list` (`EmbPattern *p`)
- void `embPattern_movestitch_listToPolylines` (`EmbPattern *p`)
- void `embPattern_movePolylinesTostitch_list` (`EmbPattern *p`)
- void `embPattern_addStitchAbs` (`EmbPattern *p, EmbReal x, EmbReal y, int flags, int isAutoColorIndex`)
- void `embPattern_addStitchRel` (`EmbPattern *p, EmbReal dx, EmbReal dy, int flags, int isAutoColorIndex`)
- void `embPattern_changeColor` (`EmbPattern *p, int index`)
- void `embPattern_scale` (`EmbPattern *p, EmbReal scale`)
- `EmbRect` `embPattern_calcBoundingBox` (`EmbPattern *p`)
- void `embPattern_flipHorizontal` (`EmbPattern *p`)
- void `embPattern_flipVertical` (`EmbPattern *p`)
- void `embPattern_flip` (`EmbPattern *p, int horz, int vert`)
- void `embPattern_combineJumpStitches` (`EmbPattern *p`)
- void `embPattern_correctForMaxStitchLength` (`EmbPattern *p, EmbReal maxStitchLength, EmbReal maxJumpLength`)
- void `embPattern_center` (`EmbPattern *p`)
- void `embPattern_loadExternalColorFile` (`EmbPattern *p, const char *fileName`)
- void `embPattern_free` (`EmbPattern *p`)
- void `embPattern_addCircleAbs` (`EmbPattern *p, EmbCircle circle`)
- void `embPattern_addEllipseAbs` (`EmbPattern *p, EmbEllipse ellipse`)
- void `embPattern_addLineAbs` (`EmbPattern *p, EmbLine line`)
- void `embPattern_addPathAbs` (`EmbPattern *p, EmbPath obj`)
- void `embPattern_addPointAbs` (`EmbPattern *p, EmbPoint obj`)
- void `embPattern_addPolygonAbs` (`EmbPattern *p, EmbPolygon obj`)
- void `embPattern_addPolylineObjectAbs` (`EmbPattern *p, EmbPolyline obj`)
- void `embPattern_addRectAbs` (`EmbPattern *p, EmbRect rect`)
- void `embPattern_end` (`EmbPattern *p`)
- int `embPattern_color_count` (`EmbPattern *pattern, EmbColor startColor`)
- void `embPattern_designDetails` (`EmbPattern *pattern`)
- int `convert` (`const char *inf, const char *outf`)
- float `embPattern_totalStitchLength` (`EmbPattern *pattern`)
- float `embPattern_minimumStitchLength` (`EmbPattern *pattern`)
- float `embPattern_maximumStitchLength` (`EmbPattern *pattern`)
- void `embPattern_lengthHistogram` (`EmbPattern *pattern, int *bin, int NUMBINS`)
- int `embPattern_realStitches` (`EmbPattern *pattern`)
- int `embPattern_jumpStitches` (`EmbPattern *pattern`)
- int `embPattern_trimStitches` (`EmbPattern *pattern`)

18.124.1 Detailed Description

The file is for the management of the main struct: `EmbPattern`.

18.124.2 Function Documentation

```
18.124.2.1 convert() int convert (
    const char * inf,
    const char * outf )
```

```
18.124.2.2 embPattern_addCircleAbs() void embPattern_addCircleAbs (
    EmbPattern * p,
    EmbCircle circle )
```

Adds a circle object to pattern (`p`) with its center at the absolute position (`cx, cy`) with a radius of (`r`). Positive y is up. Units are in millimeters.

```
18.124.2.3 embPattern_addEllipseAbs() void embPattern_addEllipseAbs (
    EmbPattern * p,
    EmbEllipse ellipse )
```

Adds an ellipse object to pattern (*p*) with its center at the absolute position (*cx,cy*) with radii of (*rx,ry*). Positive y is up. Units are in millimeters.

```
18.124.2.4 embPattern_addLineAbs() void embPattern_addLineAbs (
    EmbPattern * p,
    EmbLine line )
```

Adds a line object to pattern (*p*) starting at the absolute position (*x1,y1*) and ending at the absolute position (*x2,y2*). Positive y is up. Units are in millimeters.

```
18.124.2.5 embPattern_addPathAbs() void embPattern_addPathAbs (
    EmbPattern * p,
    EmbPath obj )
```

```
18.124.2.6 embPattern_addPointAbs() void embPattern_addPointAbs (
    EmbPattern * p,
    EmbPoint obj )
```

Adds a point object to pattern (*p*) at the absolute position (*x,y*). Positive y is up. Units are in millimeters.

```
18.124.2.7 embPattern_addPolygonAbs() void embPattern_addPolygonAbs (
    EmbPattern * p,
    EmbPolygon obj )
```

```
18.124.2.8 embPattern_addPolylineObjectAbs() void embPattern_addPolylineObjectAbs (
    EmbPattern * p,
    EmbPolyline obj )
```

```
18.124.2.9 embPattern_addRectAbs() void embPattern_addRectAbs (
    EmbPattern * p,
    EmbRect rect )
```

Adds a rectangle object to pattern (*p*) at the absolute position (*x,y*) with a width of (*w*) and a height of (*h*). Positive y is up. Units are in millimeters.

```
18.124.2.10 embPattern_addStitchAbs() void embPattern_addStitchAbs (
    EmbPattern * p,
    EmbReal x,
    EmbReal y,
    int flags,
    int isAutoColorIndex )
```

Adds a stitch to the pattern (*p*) at the absolute position (*x,y*). Positive y is up. Units are in millimeters.

```
18.124.2.11 embPattern_addStitchRel() void embPattern_addStitchRel (
    EmbPattern * p,
    EmbReal dx,
    EmbReal dy,
    int flags,
    int isAutoColorIndex )
```

Adds a stitch to the pattern (*p*) at the relative position (*dx,dy*) to the previous stitch. Positive y is up. Units are in millimeters.

18.124.2.12 embPattern_addThread() `int embPattern_addThread (`
`EmbPattern * pattern,`
`EmbThread thread)`

pattern thread

Returns

`int`

18.124.2.13 embPattern_calcBoundingBox() `EmbRect embPattern_calcBoundingBox (`
`EmbPattern * p)`

Returns an EmbRect that encapsulates all stitches and objects in the pattern (*p*).

18.124.2.14 embPattern_center() `void embPattern_center (`
`EmbPattern * p)`

Center the pattern *p*.

18.124.2.15 embPattern_changeColor() `void embPattern_changeColor (`
`EmbPattern * p,`
`int index)`

Change the currentColorIndex of pattern *p* to *index*.

18.124.2.16 embPattern_color_count() `int embPattern_color_count (`
`EmbPattern * pattern,`
`EmbColor startColor)`

18.124.2.17 embPattern_combineJumpStitches() `void embPattern_combineJumpStitches (`
`EmbPattern * p)`

p

18.124.2.18 embPattern_copyPolylinesToStitch_list() `void embPattern_copyPolylinesToStitch_list (`
`EmbPattern * p)`

Copies all of the EmbPolylineObjectList data to Embstitch_list data for pattern (*p*).

18.124.2.19 embPattern_copystitch_listToPolylines() `void embPattern_copystitch_listToPolylines (`
`EmbPattern * p)`

Copies all of the Embstitch_list data to EmbPolylineObjectList data for pattern (*p*).

18.124.2.20 embPattern_correctForMaxStitchLength() `void embPattern_correctForMaxStitchLength (`
`EmbPattern * p,`
`EmbReal maxStitchLength,`
`EmbReal maxJumpLength)`

Todo The params determine the max XY movement rather than the length. They need renamed or clarified further.

18.124.2.21 embPattern_create() `EmbPattern * embPattern_create (`
`void)`

Returns a pointer to an EmbPattern. It is created on the heap. The caller is responsible for freeing the allocated memory with `embPattern_free()`.

Returns

`EmbPattern*`

18.124.2.22 `embPattern_designDetails()` void embPattern_designDetails (EmbPattern * pattern)

18.124.2.23 `embPattern_end()` void embPattern_end (EmbPattern * p)

18.124.2.24 `embPattern_fixColorCount()` void embPattern_fixColorCount (EmbPattern * p)
p

18.124.2.25 `embPattern_flip()` void embPattern_flip (EmbPattern * p,
int horz,
int vert)

Flips the entire pattern (*p*) horizontally about the x-axis if (*horz*) is true. Flips the entire pattern (*p*) vertically about the y-axis if (*vert*) is true.

18.124.2.26 `embPattern_flipHorizontal()` void embPattern_flipHorizontal (EmbPattern * p)

Flips the entire pattern (*p*) horizontally about the y-axis.

18.124.2.27 `embPattern_flipVertical()` void embPattern_flipVertical (EmbPattern * p)

Flips the entire pattern (*p*) vertically about the x-axis.

18.124.2.28 `embPattern_free()` void embPattern_free (EmbPattern * p)

Frees all memory allocated in the pattern (*p*).

18.124.2.29 `embPattern_hideStitchesOverLength()` void embPattern_hideStitchesOverLength (EmbPattern * p,
int length)

p length

18.124.2.30 `embPattern_jumpStitches()` int embPattern_jumpStitches (EmbPattern * pattern)

18.124.2.31 `embPattern_lengthHistogram()` void embPattern_lengthHistogram (EmbPattern * pattern,
int * bin,
int NUMBINS)

18.124.2.32 `embPattern_loadExternalColorFile()` void embPattern_loadExternalColorFile (EmbPattern * p,
const char * fileName)

TODO: Description needed.

18.124.2.33 `embPattern_maximumStitchLength()` float embPattern_maximumStitchLength (EmbPattern * pattern)

18.124.2.34 embPattern_minimumStitchLength() float embPattern_minimumStitchLength (EmbPattern * pattern)

18.124.2.35 embPattern_movePolylineObjectListToStitchList() void embPattern_movePolylineObjectListToStitchList (EmbPattern * p)

Moves all of the EmbPolylineObjectList data to Embstitch_list data for pattern (p).

18.124.2.36 embPattern_movingStitchListToPolylineObjectList() void embPattern_movingStitchListToPolylineObjectList (EmbPattern * p)

Moves all of the Embstitch_list data to EmbPolylineObjectList data for pattern (p).

18.124.2.37 embPattern_realStitches() int embPattern_realStitches (EmbPattern * pattern)

18.124.2.38 embPattern_scale() void embPattern_scale (EmbPattern * p, EmbReal scale)

Very simple scaling of the x and y axis for every point. Doesn't insert or delete stitches to preserve density.

18.124.2.39 embPattern_totalStitchLength() float embPattern_totalStitchLength (EmbPattern * pattern)

pattern

Returns

float

18.124.2.40 embPattern_trimStitches() int embPattern_trimStitches (EmbPattern * pattern)

18.125 extern/libembroidery/src/thread-color.c File Reference

```
#include <stdio.h>
#include <string.h>
#include "embroidery_internal.h"
```

Functions

- int **threadColor** (const char *name, int brand)
- int **threadColorNum** (unsigned int color, int brand)
- const char * **threadColorName** (unsigned int color, int brand)

Variables

- const unsigned char **_dxfColorTable** [][3] = {{ 0, 0, 0 }}
- const **EmbThread** **husThreads** [] = {{{ 0, 0, 0 }, "END", "END"}}
- const **EmbThread** **jefThreads** [] = {{{ 0, 0, 0 }, "END", "END"}}
- const **EmbThread** **shvThreads** [] = {{{ 0, 0, 0 }, "END", "END"}}
- const **EmbThread** **pcmThreads** [] = {{{ 0, 0, 0 }, "END", "END"}}
- const **EmbThread** **pecThreads** [] = {{{ 0, 0, 0 }, "END", "END"}}
- const int **shvThreadCount** = 42
- const int **pecThreadCount** = 65
- **thread_color** * **brand_codes** []
- const char * **brand_codes_files** []

18.125.1 Function Documentation

18.125.1.1 `threadColor()` int threadColor (
 const char * name,
 int brand)

18.125.1.2 `threadColorName()` const char * threadColorName (
 unsigned int color,
 int brand)

18.125.1.3 `threadColorNum()` int threadColorNum (
 unsigned int color,
 int brand)

18.125.2 Variable Documentation

18.125.2.1 `_dxfColorTable` const unsigned char _dxfColorTable[][3] = {{ 0, 0, 0 }}

18.125.2.2 `brand_codes` `thread_color*` brand_codes[]

18.125.2.3 `brand_codes_files` const char* brand_codes_files[]

Initial value:

```
= {  
    "arc_polyester_colors.csv",  
    "arc_rayon_colors.csv",  
    "coats_and_clark_rayon_colors.csv",  
    "exquisite_polyester_colors.csv",  
    "fufu_Polyester_colors.csv",  
    "fufu_Rayon_colors.csv",  
    "Hemingworth_Polyester_colors.csv",  
    "Isacord_Polyester_colors.csv",  
    "Isafil_Rayon_colors.csv",  
    "Marathon_Polyester_colors.csv",  
    "Marathon_Rayon_colors.csv",  
    "Madeira_Polyester_colors.csv",  
    "Madeira_Rayon_colors.csv",  
    "Metro_Polyester_colors.csv",  
    "Pantone_colors.csv",  
    "RobisonAnton_Polyester_colors.csv",  
    "RobisonAnton_Rayon_colors.csv",  
    "Sigma_Polyester_colors.csv",  
    "Sulky_Rayon_colors.csv",  
    "ThreadArt_Rayon_colors.csv",  
    "ThreadArt_Polyester_colors.csv",  
    "ThreadDelight_Polyester_colors.csv",  
    "Z102_Isacord_Polyester_colors.csv",  
    "svg_color_colors.csv"  
}
```

18.125.2.4 `husThreads` const `EmbThread` husThreads[] = {{{ 0, 0, 0 }, "END", "END"}}

18.125.2.5 `jefThreads` const `EmbThread` jefThreads[] = {{{ 0, 0, 0 }, "END", "END"}}

18.125.2.6 `pcmThreads` const `EmbThread` pcmThreads[] = {{{ 0, 0, 0 }, "END", "END"}}

18.125.2.7 pecThreadCount const int pecThreadCount = 65

18.125.2.8 pecThreads const EmbThread pecThreads[] = {{{ 0, 0, 0 }, "END", "END"}}

18.125.2.9 shvThreadCount const int shvThreadCount = 42

18.125.2.10 shvThreads const EmbThread shvThreads[] = {{{ 0, 0, 0 }, "END", "END"}}

18.126 **privacy_policy.md** File Reference

References

- [1] acatina. Technical info. [540](#)
- [2] KDE Community. Projects/liberty/file formats/tajima ternary - kde community wiki. [540](#)
- [3] G. van Rossum and B. Warsaw. Python pep 7. [19](#)

Index

_appName
 embroidermodder.cpp, 357
_appVer
 embroidermodder.cpp, 357
_bcf_directory, 48
 dirEntries, 49
 maxNumberOfDirectoryEntries, 49
_bcf_directory_entry, 49
 childId, 49
 CLSID, 49
 colorFlag, 49
 creationTime, 50
 directoryEntryName, 50
 directoryEntryNameLength, 50
 leftSiblingId, 50
 modifiedTime, 50
 next, 50
 objectType, 50
 rightSiblingId, 50
 startingSectorLocation, 50
 stateBits, 50
 streamSize, 50
 streamSizeHigh, 51
_bcf_file, 51
 difat, 51
 directory, 51
 fat, 51
 header, 51
_bcf_file_difat, 52
 fatSectorCount, 52
 fatSectorEntries, 52
 sectorSize, 52
_bcf_file_fat, 52
 fatEntries, 52
 fatEntryCount, 53
 numberOfEntriesInFatSector, 53
_bcf_file_header, 53
 byteOrder, 53
 CLSID, 54
 firstDifatSectorLocation, 54
 firstDirectorySectorLocation, 54
 firstMiniFATSectorLocation, 54
 majorVersion, 54
 miniSectorShift, 54
 miniStreamCutoffSize, 54
 minorVersion, 54
 numberOfDifatSectors, 54
 numberOfDirectorySectors, 54
 numberOfFATSectors, 54
 numberOfMiniFatSectors, 55
 reserved1, 55
 reserved2, 55
 sectorShift, 55
 signature, 55
 transactionSignatureNumber, 55
_dxIColorTable
 embroidery.h, 465
 thread-color.c, 606
_mainWin
 Application, 60
 mainwindow.cpp, 409
_subMask
 format_csd.c, 537
_vp3Hoop, 55
 bottom, 56
 bottom2, 56
 byte1, 56
 byte2, 56
 byte3, 56
 height, 56
 left, 56
 left2, 56
 numberOfBytesRemaining, 56
 numberOfColors, 56
 right, 56
 right2, 57
 threadLength, 57
 top, 57
 top2, 57
 unknown2, 57
 unknown3, 57
 unknown4, 57
 width, 57
 xOffset, 57
 yOffset, 57
_xorMask
 format_csd.c, 537
~ArcObject
 ArcObject, 65
~BaseObject
 BaseObject, 74
~CircleObject
 CircleObject, 82
~CmdPrompt
 CmdPrompt, 87
~CmdPromptHandle
 CmdPromptHandle, 94
~CmdPromptHistory
 CmdPromptHistory, 96
~CmdPromptInput
 CmdPromptInput, 99
~CmdPromptSplitter
 CmdPromptSplitter, 105
~DimLeaderObject
 DimLeaderObject, 111
~EllipseObject
 EllipseObject, 117
~EmbDetailsDialog
 EmbDetailsDialog, 127
~ImageObject

ImageObject, 154
 ~ImageWidget
 ImageWidget, 158
 ~LayerManager
 LayerManager, 160
 ~LineObject
 LineObject, 163
 ~MainWindow
 MainWindow, 177
 ~MdiArea
 MdiArea, 222
 ~MdiWindow
 MdiWindow, 227
 ~PathObject
 PathObject, 238
 ~PointObject
 PointObject, 243
 ~PolygonObject
 PolygonObject, 248
 ~PolylineObject
 PolylineObject, 253
 ~PreviewDialog
 PreviewDialog, 257
 ~PropertyEditor
 PropertyEditor, 260
 ~RectObject
 RectObject, 273
 ~SaveObject
 SaveObject, 277
 ~Settings_Dialog
 Settings_Dialog, 299
 ~TextSingleObject
 TextSingleObject, 324
 ~UndoEditor
 UndoEditor, 340
 ~View
 View, 345
 10o, 16, 532
 100, 16, 532
 about
 MainWindow, 177
 accept_display_bg_color
 Settings_Dialog, 308
 accept_display_crosshair_color
 Settings_Dialog, 308
 accept_display_selectbox_left_color
 Settings_Dialog, 308
 accept_display_selectbox_left_fill
 Settings_Dialog, 308
 accept_display_selectbox_right_color
 Settings_Dialog, 308
 accept_display_selectbox_right_fill
 Settings_Dialog, 308
 accept_general_mdi_bg_color
 Settings_Dialog, 308
 accept_general_mdi_bg_logo
 Settings_Dialog, 308
 accept_general_mdi_bg_texture
 Settings_Dialog, 308
 accept_grid_color
 Settings_Dialog, 308
 accept_prompt_bg_color
 Settings_Dialog, 309
 accept_prompt_text_color
 Settings_Dialog, 309
 accept_ruler_color
 Settings_Dialog, 309
 acceptChanges
 Settings_Dialog, 299
 Action
 embroidermodder.h, 361
 Action___, 58
 aliases, 58
 hash, 58
 icon, 58
 menu_name, 58
 menu_position, 58
 script, 58
 shortcut, 58
 statustip, 59
 toolbar_name, 59
 toolbar_position, 59
 tooltip, 59
 action_labels
 mainwindow.cpp, 409
 action_table
 embroidermodder.h, 367
 mainwindow.cpp, 409
 actionHash
 MainWindow, 207
 activeCommand
 CmdPrompt, 87
 MainWindow, 177
 activeMdiWindow
 MainWindow, 177
 activeScene
 MainWindow, 177
 activeUndoStack
 MainWindow, 177
 activeView
 MainWindow, 177
 actuator
 MainWindow, 177
 addArc
 SaveObject, 277
 addBlock
 SaveObject, 277
 addCircle
 SaveObject, 277
 addColorsToComboBox
 Settings_Dialog, 299
 addCommand
 CmdPrompt, 87
 CmdPromptInput, 99
 addDimAligned

SaveObject, 278
addDimAngular
 SaveObject, 278
addDimArcLength
 SaveObject, 278
addDimDiameter
 SaveObject, 279
addDimLeader
 SaveObject, 279
addDimLinear
 SaveObject, 279
addDimOrdinate
 SaveObject, 279
addDimRadius
 SaveObject, 279
addEllipse
 SaveObject, 279
addEllipseArc
 SaveObject, 279
addGrid
 SaveObject, 280
addHatch
 SaveObject, 280
addImage
 SaveObject, 280
addInfiniteLine
 SaveObject, 280
addLayer
 LayerManager, 160
addLine
 SaveObject, 280
addObject
 View, 345
addPath
 SaveObject, 280
addPoint
 SaveObject, 280
addPolygon
 SaveObject, 280
addPolyline
 SaveObject, 281
addRay
 SaveObject, 281
addRectangle
 SaveObject, 281
addSlot
 SaveObject, 281
addSpline
 SaveObject, 281
addStack
 UndoEditor, 340
addTextMulti
 SaveObject, 281
addTextSingle
 SaveObject, 281
addToRubberRoom
 View, 345
after
 UndoableGripEditCommand, 333
alert
 CmdPrompt, 87
aliases
 Action__, 58
aliasHash
 CmdPromptInput, 104
alignScenePointWithViewPoint
 View, 345
allGripPoints
 ArcObject, 65
 BaseObject, 74
 CircleObject, 82
 DimLeaderObject, 111
 EllipseObject, 118
 ImageObject, 155
 LineObject, 164
 PathObject, 238
 PointObject, 243
 PolygonObject, 248
 PolylineObject, 254
 RectObject, 273
 TextSingleObject, 324
allowRubber
 View, 345
allowZoomIn
 View, 345
allowZoomOut
 View, 345
alpha
 SelectBox, 284
alphabet
 LSYSTEM, 167
Ameco, 502, 553
angle
 UndoableRotateCommand, 338
appendHistory
 CmdPrompt, 87
 CmdPromptHistory, 96
 CmdPromptInput, 100
appendTheHistory
 CmdPrompt, 87
Application, 59
 _mainWin, 60
 Application, 59
 event, 60
 setMainWin, 60
applyFormatting
 CmdPromptHistory, 97
 CmdPromptInput, 100
arc
 EmbGeometry__, 131
arc.c
 Arc_clockwise, 576
 Base_objectRubberPoint, 576
 Base_objectRubberText, 577
 Base_setLineType, 577
 Base_setLineWidth, 577

clockwise, 577
 embArc_arcLength, 577
 embArc_area, 577
 embArc_chord, 577
 embArc_clockwise, 577
 embArc_endAngle, 577
 embArc_gripEdit, 577
 embArc_includedAngle, 577
 embArc_init, 577
 embArc_mouseSnapPoint, 578
 embArc_paint, 578
 embArc_setCenter, 578
 embArc_setEndAngle, 578
 embArc_setRadius, 578
 embArc_setStartAngle, 578
 embArc_startAngle, 578
 embArc_updatePath, 578
 embArc_updateRubber, 578
 embBaseSetColorRGB, 578
 embCircle_prompt, 578
 embCircle_setArea, 578
 embCircle_setCircumference, 579
 embEllipse_click, 579
 embEllipse_main, 579
 embRect_bottomLeft, 579
 embRect_bottomRight, 579
 getArcCenter, 579
 getArcDataFromBulge, 579
 set_object_color, 579
 Arc_clockwise
 arc.c, 576
 Arc_Polyester
 embroidery.h, 440
 Arc_Rayon
 embroidery.h, 440
 arcEndPoint
 ArcObject, 72
 arcMidPoint
 ArcObject, 72
 ArcObject, 60
 ~ArcObject, 65
 allGripPoints, 65
 arcEndPoint, 72
 arcMidPoint, 72
 ArcObject, 64
 arcStartPoint, 72
 calculateArcData, 65
 gripEdit, 66
 init, 66
 mouseSnapPoint, 66
 objectArcLength, 67
 objectArea, 67
 objectChord, 67
 objectClockwise, 67
 objectEndAngle, 67
 objectEndPoint, 68
 objectEndX, 68
 objectEndY, 68
 objectIncludedAngle, 68
 objectMidPoint, 68
 objectMidX, 69
 objectMidY, 69
 objectRadius, 69
 objectStartAngle, 69
 objectStartPoint, 69
 objectStartX, 69
 objectStartY, 70
 paint, 70
 setObjectEndAngle, 70
 setObjectEndPoint, 70, 71
 setObjectMidPoint, 71
 setObjectRadius, 71
 setObjectStartAngle, 71
 setObjectStartPoint, 71
 Type, 64
 type, 71
 updateArcRect, 71
 updatePath, 72
 updateRubber, 72
 vulcanize, 72
 arc startPoint
 ArcObject, 72
 array.c
 embArray_addArc, 429
 embArray_addCircle, 429
 embArray_addEllipse, 429
 embArray_addFlag, 430
 embArray_addLine, 430
 embArray_addPath, 430
 embArray_addPoint, 430
 embArray_addPolygon, 430
 embArray_addPolyline, 430
 embArray_addRect, 430
 embArray_addStitch, 430
 embArray_addVector, 430
 embArray_copy, 430
 embArray_create, 430
 embArray_free, 431
 embArray_resize, 431
 ArrowStyle
 DimLeaderObject, 110
 arrowStyleAngle
 DimLeaderObject, 114
 arrowStyleLength
 DimLeaderObject, 114
 arrowStylePath
 DimLeaderObject, 114
 art, 16, 533
 assets_dir
 Settings_, 287
 attributeList
 format_svg.c, 566
 attributeOffset
 VipHeader_, 355
 AutoCAD, 505, 543, 562
 AutoDesk, 543

auxFormat
 ThredExtension_, 327

axiom
 LSYSTEM, 167

b
 EmbColor_, 126

Barudan, 506, 539, 570

Base_objectRubberPoint
 arc.c, 576

Base_objectRubberText
 arc.c, 577

Base_setLineType
 arc.c, 577

Base_setLineWeight
 arc.c, 577

BaseObject, 73
 ~BaseObject, 74
 allGripPoints, 74
 BaseObject, 74
 boundingRect, 74
 drawRubberLine, 75
 gripEdit, 75
 line, 75
 lineWeightPen, 75
 lwtPen, 79
 mouseSnapPoint, 75
 objectCenter, 75
 objectCenterX, 75
 objectCenterY, 75
 objectColor, 75
 objectColorRGB, 76
 objectID, 76
 objectLineType, 76
 objectLineWeight, 76
 objectPath, 76
 objectPen, 76
 objectRubberMode, 76
 objectRubberPoint, 76
 objectRubberText, 76
 objID, 79
 objLine, 79
 objPen, 79
 objRubberMode, 79
 objRubberPoints, 79
 objRubberTexts, 79
 realRender, 76
 rect, 76
 setLine, 77
 setObjectCenter, 77
 setObjectCenterX, 77
 setObjectCenterY, 77
 setObjectColor, 77
 setObjectColorRGB, 77
 setObjectLineType, 77
 setObjectLineWeight, 77
 setObjectPath, 78
 setObjectRubberMode, 78
 setObjectRubberPoint, 78
 setObjectRubberText, 78
 setRect, 78
 shape, 78
 Type, 74
 type, 78
 vulcanize, 78

bcf_difat_create
 embroidery_internal.h, 490
 main.c, 594

bcf_directory
 embroidery_internal.h, 489

bcf_directory_entry
 embroidery_internal.h, 489

bcf_directory_free
 embroidery_internal.h, 491
 main.c, 594

bcf_file
 embroidery_internal.h, 489

bcf_file_difat
 embroidery_internal.h, 489

bcf_file_difat_free
 embroidery_internal.h, 491

bcf_file_fat
 embroidery_internal.h, 489

bcf_file_fat_free
 embroidery_internal.h, 491

bcf_file_free
 embroidery_internal.h, 491
 main.c, 594

bcf_file_header
 embroidery_internal.h, 489

bcfFile_read
 embroidery_internal.h, 491
 main.c, 594

bcfFileFat_create
 embroidery_internal.h, 491
 main.c, 594

bcfFileHeader_isValid
 embroidery_internal.h, 491

bcfFileHeader_read
 embroidery_internal.h, 491
 main.c, 594

before
 UndoableGripEditCommand, 333

Bernina, 533

beziers
 EmbSpline_, 142

bgColor
 MdiArea, 225

bgLogo
 MdiArea, 225

bgTexture
 MdiArea, 225

binaryReadString
 embroidery_internal.h, 491
 main.c, 594

binaryReadUnicodeString
 embroidery_internal.h, 492

main.c, 595
binaryWriteInt
 embroidery_internal.h, 492
 formats.c, 529
binaryWriteIntBE
 embroidery_internal.h, 492
 formats.c, 529
binaryWriteShort
 embroidery_internal.h, 492
 formats.c, 529
binaryWriteUInt
 embroidery_internal.h, 492
 formats.c, 529
binaryWriteUIntBE
 embroidery_internal.h, 492
 formats.c, 529
binaryWriteUShort
 embroidery_internal.h, 492
 formats.c, 529
binaryWriteUShortBE
 embroidery_internal.h, 493
 formats.c, 529
bit_position
 Compress, 106
Bitmap Cache, 534
Bits and Volts, 534
bits_total
 Compress, 106
black_thread
 embroidery.h, 465
 main.c, 600
blink
 CmdPrompt, 87
blinkState
 CmdPrompt, 93
blinkTimer
 CmdPrompt, 93
block_elements
 Compress, 107
bmc, 534
bottom
 _vp3Hoop, 56
 EmbRect_, 140
 hoop_padding, 150
bottom2
 _vp3Hoop, 56
boundingRect
 BaseObject, 74
 EmbDetailsDialog, 127
Box
 DimLeaderObject, 110
boxDir
 SelectBox, 284
brand_codes
 thread-color.c, 606
brand_codes_files
 thread-color.c, 606
bro, 16, 534
Brother, 503, 504, 557, 558, 561, 562
BuildDecryptionTable
 format_csd.c, 536
BULGETOCONTROL
 embroidery_internal.h, 481
BULGETOEND
 embroidery_internal.h, 481
buttonBox
 EmbDetailsDialog, 127
 Settings_Dialog, 309
buttonCustomFilterClearAll
 Settings_Dialog, 299
buttonCustomFilterClearAllClicked
 Settings_Dialog, 299
buttonCustomFilterSelectAll
 Settings_Dialog, 299
buttonCustomFilterSelectAllClicked
 Settings_Dialog, 299
buttonQSnapClearAll
 Settings_Dialog, 300
buttonQSnapClearAllClicked
 Settings_Dialog, 300
buttonQSnapSelectAll
 Settings_Dialog, 300
buttonQSnapSelectAllClicked
 Settings_Dialog, 300
buttonTipOfTheDayClicked
 MainWindow, 178
byte1
 _vp3Hoop, 56
byte2
 _vp3Hoop, 56
byte3
 _vp3Hoop, 56
byteOrder
 _bcf_file_header, 53
calculateArcData
 ArcObject, 65
canRedo
 UndoEditor, 340
canUndo
 UndoEditor, 340
cascade
 MdiArea, 222
catalogNumber
 EmbThread_, 144
cci
 format_dst.c, 541
center
 EmbCircle_, 125
 EmbEllipse_, 129
 UiObject_, 329
 View, 345
centerAt
 View, 345
changeFormatting
 CmdPromptInput, 100
changelog

MainWindow, 178
character_huffman
 Compress, 107
character_length_huffman
 Compress, 107
check_for_color_file
 EmbFormatList_, 129
check_header_present
 embroidery_internal.h, 493
 main.c, 595
checkBoxCustomFilterStateChanged
 Settings_Dialog, 300
checkBoxDisableBGStateChanged
 Settings_Dialog, 300
checkBoxGeneralMdiBGUseColorStateChanged
 Settings_Dialog, 300
checkBoxGeneralMdiBGUseLogoStateChanged
 Settings_Dialog, 300
checkBoxGeneralMdiBGUseTextureStateChanged
 Settings_Dialog, 300
checkBoxGridCenterOnOriginStateChanged
 Settings_Dialog, 300
checkBoxGridColumnMatchCrossHairStateChanged
 Settings_Dialog, 300
checkBoxGridLoadFromFileStateChanged
 Settings_Dialog, 300
checkBoxGridShowOnLoadStateChanged
 Settings_Dialog, 300
checkBoxGridShowOriginStateChanged
 Settings_Dialog, 301
checkBoxLwtRealRenderStateChanged
 Settings_Dialog, 301
checkBoxLwtShowLwtStateChanged
 Settings_Dialog, 301
checkBoxPromptSaveHistoryAsHtmlStateChanged
 Settings_Dialog, 301
checkBoxPromptSaveHistoryStateChanged
 Settings_Dialog, 301
checkBoxQSnapApparentStateChanged
 Settings_Dialog, 301
checkBoxQSnapCenterStateChanged
 Settings_Dialog, 301
checkBoxQSnapEndPointStateChanged
 Settings_Dialog, 301
checkBoxQSnapExtensionStateChanged
 Settings_Dialog, 301
checkBoxQSnapInsertionStateChanged
 Settings_Dialog, 301
checkBoxQSnapIntersectionStateChanged
 Settings_Dialog, 301
checkBoxQSnapMidPointStateChanged
 Settings_Dialog, 301
checkBoxQSnapNearestStateChanged
 Settings_Dialog, 301
checkBoxQSnapNodeStateChanged
 Settings_Dialog, 302
checkBoxQSnapParallelStateChanged
 Settings_Dialog, 302
checkBoxQSnapPerpendicularStateChanged
 Settings_Dialog, 302
checkBoxQSnapQuadrantStateChanged
 Settings_Dialog, 302
checkBoxQSnapTangentStateChanged
 Settings_Dialog, 302
checkBoxRenderHintAAStateChanged
 Settings_Dialog, 302
checkBoxRenderHintHighAAStateChanged
 Settings_Dialog, 302
checkBoxRenderHintNonCosmeticStateChanged
 Settings_Dialog, 302
checkBoxRenderHintSmoothPixStateChanged
 Settings_Dialog, 302
checkBoxRenderHintTextAAStateChanged
 Settings_Dialog, 302
checkBoxRulerShowOnLoadStateChanged
 Settings_Dialog, 302
checkBoxSelectionModePickAddStateChanged
 Settings_Dialog, 302
checkBoxSelectionModePickDragStateChanged
 Settings_Dialog, 302
checkBoxSelectionModePickFirstStateChanged
 Settings_Dialog, 303
checkBoxShowScrollBarsStateChanged
 Settings_Dialog, 303
checkBoxTipOfDay
 MainWindow, 207
checkBoxTipOfDayStateChanged
 MainWindow, 178
 Settings_Dialog, 303
checkBoxUseOpenGLStateChanged
 Settings_Dialog, 303
checkChangedText
 CmdPromptInput, 100
checkCursorPosition
 CmdPromptInput, 100
checkEditedText
 CmdPromptInput, 100
checkForUpdates
 MainWindow, 178
checkSelection
 CmdPromptInput, 100
childId
 _bcf_directory_entry, 49
chooseDisplayBackgroundColor
 Settings_Dialog, 303
chooseDisplayCrossHairColor
 Settings_Dialog, 303
chooseDisplaySelectBoxLeftColor
 Settings_Dialog, 303
chooseDisplaySelectBoxLeftFill
 Settings_Dialog, 303
chooseDisplaySelectBoxRightColor
 Settings_Dialog, 303
chooseDisplaySelectBoxRightFill
 Settings_Dialog, 303
chooseGeneralMdiBackgroundColor

Settings_Dialog, 303
 chooseGeneralMdiBackgroundLogo
 Settings_Dialog, 303
 chooseGeneralMdiBackgroundTexture
 Settings_Dialog, 303
 chooseGridColor
 Settings_Dialog, 304
 choosePromptBackgroundColor
 Settings_Dialog, 304
 choosePromptTextColor
 Settings_Dialog, 304
 chooseRulerColor
 Settings_Dialog, 304
 CHUNK_SIZE
 embroidery.h, 440
 circle
 EmbGeometry_, 131
 circle.c
 embCircle_area, 580
 embCircle_circumference, 580
 embCircle_init, 580
 getCircleCircleIntersections, 580
 getCircleTangentPoints, 580
 CIRCLE_MODE_1P_DIA
 brodermodder.h, 366
 CIRCLE_MODE_1P_DIA_
 mainwindow.cpp, 409
 CIRCLE_MODE_1P_RAD
 brodermodder.h, 366
 CIRCLE_MODE_1P_RAD_
 mainwindow.cpp, 409
 CIRCLE_MODE_2P
 brodermodder.h, 366
 CIRCLE_MODE_2P_
 mainwindow.cpp, 409
 CIRCLE_MODE_3P
 brodermodder.h, 366
 CIRCLE_MODE_3P_
 mainwindow.cpp, 410
 CIRCLE_MODE_TTR
 brodermodder.h, 366
 CIRCLE_MODE_TTR_
 mainwindow.cpp, 410
 CircleObject, 79
 ~CircleObject, 82
 allGripPoints, 82
 CircleObject, 82
 gripEdit, 82
 init, 82
 mouseSnapPoint, 83
 objectArea, 83
 objectCircumference, 83
 objectDiameter, 83
 objectQuadrant0, 83
 objectQuadrant180, 83
 objectQuadrant270, 83
 objectQuadrant90, 83
 objectRadius, 83
 objectSavePath, 83
 paint, 84
 setObjectArea, 84
 setObjectCircumference, 84
 setObjectDiameter, 84
 setObjectRadius, 84
 Type, 82
 type, 84
 updatePath, 84
 updateRubber, 84
 vulcanize, 84
 clearAllFields
 PropertyEditor, 260
 clearFormatting
 CmdPromptInput, 100
 clearRubberRoom
 View, 345
 clearSelection
 View, 345
 clockwise
 arc.c, 577
 Closed
 DimLeaderObject, 110
 closeEvent
 MainWindow, 178
 MdiWindow, 228
 closeToolBar
 MainWindow, 178
 CLSID
 _bcf_directory_entry, 49
 _bcf_file_header, 54
 cmdActive
 CmdPromptInput, 104
 CmdPrompt, 85
 ~CmdPrompt, 87
 activeCommand, 87
 addCommand, 87
 alert, 87
 appendHistory, 87
 appendTheHistory, 87
 blink, 87
 blinkState, 93
 blinkTimer, 93
 CmdPrompt, 87
 copyPressed, 87
 cutPressed, 88
 deletePressed, 88
 disableRapidFire, 88
 downPressed, 88
 enableRapidFire, 88
 endCommand, 88
 escapePressed, 88
 F10Pressed, 88
 F11Pressed, 88
 F12Pressed, 88
 F1Pressed, 88
 F2Pressed, 89
 F3Pressed, 89

F4Pressed, 89
F5Pressed, 89
F6Pressed, 89
F7Pressed, 89
F8Pressed, 89
F9Pressed, 89
floatingChanged, 89
getCurrentText, 89
getHistory, 89
getPrefix, 90
historyAppended, 90
isCommandActive, 90
isRapidFireEnabled, 90
lastCommand, 90
pastePressed, 90
processInput, 90
promptDivider, 93
promptHistory, 93
promptInput, 93
promptSplitter, 93
promptVBoxLayout, 93
redoPressed, 90
resizeTheHistory, 90
runCommand, 90
saveHistory, 90
selectAllPressed, 91
setCurrentText, 91
setHistory, 91
setPrefix, 91
setPromptBackgroundColor, 91
setPromptFontFamily, 91
setPromptFontSize, 91
setPromptFontStyle, 91
setPromptTextColor, 91
shiftPressed, 91
shiftReleased, 92
showSettings, 92
startBlinking, 92
startCommand, 92
startResizingTheHistory, 92
stopBlinking, 92
stopResizingTheHistory, 92
styleHash, 93
tabPressed, 92
undoPressed, 92
updateStyle, 92
upPressed, 92
CmdPromptHandle, 93
~CmdPromptHandle, 94
CmdPromptHandle, 94
handleMoved, 94
handlePressed, 94
handleReleased, 95
mouseMoveEvent, 95
mousePressEvent, 95
mouseReleaseEvent, 95
moveY, 95
pressY, 95
releaseY, 95
CmdPromptHistory, 95
~CmdPromptHistory, 96
appendHistory, 96
applyFormatting, 97
CmdPromptHistory, 96
contextMenuEvent, 97
historyAppended, 97
resizeHistory, 97
startResizeHistory, 97
stopResizeHistory, 97
tmpHeight, 97
CmdPromptInput, 98
~CmdPromptInput, 99
addCommand, 99
aliasHash, 104
appendHistory, 100
applyFormatting, 100
changeFormatting, 100
checkChangedText, 100
checkCursorPosition, 100
checkEditedText, 100
checkSelection, 100
clearFormatting, 100
cmdActive, 104
CmdPromptInput, 99
contextMenuEvent, 100
copyClip, 100
copyPressed, 101
curCmd, 104
curText, 104
cutPressed, 101
defaultPrefix, 104
deletePressed, 101
downPressed, 101
endCommand, 101
escapePressed, 101
eventFilter, 101
F10Pressed, 101
F11Pressed, 101
F12Pressed, 101
F1Pressed, 101
F2Pressed, 102
F3Pressed, 102
F4Pressed, 102
F5Pressed, 102
F6Pressed, 102
F7Pressed, 102
F8Pressed, 102
F9Pressed, 102
isBlinking, 104
lastCmd, 104
pasteClip, 102
pastePressed, 102
prefix, 104
processInput, 102
rapidFireEnabled, 104
redoPressed, 103

runCommand, 103
 selectAllPressed, 103
 shiftPressed, 103
 shiftReleased, 103
 showSettings, 103
 startCommand, 103
 stopBlinking, 103
 tabPressed, 103
 undoPressed, 103
 updateCurrentText, 103
 upPressed, 104
 CmdPromptSplitter, 105
 ~CmdPromptSplitter, 105
 CmdPromptSplitter, 105
 createHandle, 105
 moveResizeHistory, 106
 pressResizeHistory, 106
 releaseResizeHistory, 106
 cnd, 16, 535
 CoatsAndClark_Rayon
 embroidery.h, 440
 CODE_OF_CONDUCT.md, 356
 col, 16, 535
 color
 EmbGeometry_, 131
 EmbLine_, 135
 EmbPath_, 137
 EmbPoint_, 139
 EmbStitch_, 142
 EmbThread_, 144
 UiObject_, 330
 color_only
 EmbFormatList_, 129
 colorChanges
 EmbDetailsDialog, 127
 colorCode
 StxThread_, 320
 SubDescriptor_, 320
 colorFlag
 _bcf_directory_entry, 49
 colorLength
 VipHeader_, 355
 colorName
 StxThread_, 320
 SubDescriptor_, 320
 colorSelector
 MainWindow, 207
 colorSelectorIndexChanged
 MainWindow, 178
 colorTotal
 EmbDetailsDialog, 127
 comboBoxArcClockwise
 property-editor.cpp, 416
 comboBoxes
 property-editor.cpp, 416
 comboBoxGeneralLineType
 property-editor.cpp, 416
 comboBoxGeneralLineWidth
 property-editor.cpp, 416
 comboBoxGridTypeCurrentIndexChanged
 Settings_Dialog, 304
 comboBoxIconSizeCurrentIndexChanged
 Settings_Dialog, 304
 comboBoxIconThemeCurrentIndexChanged
 Settings_Dialog, 304
 comboBoxLanguageCurrentIndexChanged
 Settings_Dialog, 304
 comboBoxPathClosed
 property-editor.cpp, 417
 comboBoxPathVertexNum
 property-editor.cpp, 417
 comboBoxPolylineClosed
 property-editor.cpp, 417
 comboBoxPolylineVertexNum
 property-editor.cpp, 417
 comboBoxPromptFontFamilyCurrentIndexChanged
 Settings_Dialog, 304
 comboBoxPromptFontSizeCurrentIndexChanged
 Settings_Dialog, 304
 comboBoxQSnapLocatorColorCurrentIndexChanged
 Settings_Dialog, 304
 comboBoxRulerMetricCurrentIndexChanged
 Settings_Dialog, 304
 comboBoxScrollBarWidgetCurrentIndexChanged
 Settings_Dialog, 304
 comboBoxSelected
 PropertyEditor, 266
 comboBoxSelectionCoolGripColorCurrentIndexChanged
 Settings_Dialog, 305
 comboBoxSelectionHotGripColorCurrentIndexChanged
 Settings_Dialog, 305
 comboBoxTextSingleBackward
 property-editor.cpp, 417
 comboBoxTextSingleFont
 property-editor.cpp, 417
 comboBoxTextSingleJustify
 property-editor.cpp, 417
 comboBoxTextSingleUpsideDown
 property-editor.cpp, 417
 command
 UiObject_, 330
 CompoundFileDirectory
 embroidery_internal.h, 493
 main.c, 595
 CompoundFileDirectoryEntry
 embroidery_internal.h, 493
 main.c, 595
 CompoundFileSector_DIFAT_Sector
 embroidery_internal.h, 481
 CompoundFileSector_EndOfChain
 embroidery_internal.h, 481
 CompoundFileSector_FAT_Sector
 embroidery_internal.h, 482
 CompoundFileSector_FreeSector
 embroidery_internal.h, 482
 CompoundFileSector_MaxRegSector

embroidery_internal.h, 482
CompoundFileStreamId_MaxRegularStreamId
 embroidery_internal.h, 482
CompoundFileStreamId_NoStream
 embroidery_internal.h, 482
Compress, 106
 bit_position, 106
 bits_total, 106
 block_elements, 107
 character_huffman, 107
 character_length_huffman, 107
 distance_huffman, 107
 input_data, 107
 input_length, 107
compress
 embroidery_internal.h, 489
compress.c
 compress_get_bits, 431
 compress_get_position, 432
 compress_get_token, 432
 compress_init, 432
 compress_load_block, 432
 compress_load_character_huffman, 432
 compress_load_character_length_huffman, 432
 compress_load_distance_huffman, 432
 compress_peek, 432
 compress_pop, 432
 compress_read_variable_length, 432
 huffman_build_table, 432
 huffman_lookup, 432
 huffman_lookup_data, 433
 hus_compress, 432
 hus_decompress, 433
compress_get_bits
 compress.c, 431
 embroidery_internal.h, 493
compress_get_position
 compress.c, 432
 embroidery_internal.h, 493
compress_get_token
 compress.c, 432
 embroidery_internal.h, 493
compress_init
 compress.c, 432
compress_load_block
 compress.c, 432
 embroidery_internal.h, 494
compress_load_character_huffman
 compress.c, 432
 embroidery_internal.h, 494
compress_load_character_length_huffman
 compress.c, 432
 embroidery_internal.h, 494
compress_load_distance_huffman
 compress.c, 432
 embroidery_internal.h, 494
compress_peek
 compress.c, 432
compress_pop
 compress.c, 432
compress_read_variable_length
 compress.c, 432
compress_write_variable_length
 compress.c, 432
constants
 LSYSTEM, 167
contextMenuEvent
 CmdPromptHistory, 97
 CmdPromptInput, 100
 StatusBarButton, 318
 View, 345
control1
 EmbBezier_, 124
control2
 EmbBezier_, 124
controlPointLabels
 UiObject_, 330
controlPoints
 UiObject_, 330
convert
 embroidery.h, 450
 pattern.c, 601
convert_args_to_type
 mainwindow.cpp, 408
copy
 MainWindow, 179
 View, 345
copy_trim
 embroidery_internal.h, 494
 main.c, 595
copyClip
 CmdPromptInput, 100
copyPressed
 CmdPrompt, 87
 CmdPromptInput, 101
copySelected
 View, 345
cornerButtonClicked
 View, 346
count
 EmbArray_, 123
create_icon
 MainWindow, 179
create_test_file_1
 embroidery_internal.h, 494
create_test_file_2
 embroidery_internal.h, 494
create_test_file_3
 embroidery_internal.h, 494
create_toolbar
 MainWindow, 179
createAllActions
 MainWindow, 179
createAllMenus
 MainWindow, 179
createAllToolbars

MainWindow, 179
createComboBox
 PropertyEditor, 260
createComboBoxSelected
 PropertyEditor, 260
createEditMenu
 MainWindow, 179
createFileMenu
 MainWindow, 179
createFontComboBox
 PropertyEditor, 260
createGrid
 View, 346
createGridIso
 View, 346
createGridPolar
 View, 346
createGridRect
 View, 346
createGroupBoxGeneral
 PropertyEditor, 260
createGroupBoxGeometryArc
 PropertyEditor, 260
createGroupBoxGeometryBlock
 PropertyEditor, 261
createGroupBoxGeometryCircle
 PropertyEditor, 261
createGroupBoxGeometryDimAligned
 PropertyEditor, 261
createGroupBoxGeometryDimAngular
 PropertyEditor, 261
createGroupBoxGeometryDimArcLength
 PropertyEditor, 261
createGroupBoxGeometryDimDiameter
 PropertyEditor, 261
createGroupBoxGeometryDimLeader
 PropertyEditor, 261
createGroupBoxGeometryDimLinear
 PropertyEditor, 261
createGroupBoxGeometryDimOrdinate
 PropertyEditor, 262
createGroupBoxGeometryDimRadius
 PropertyEditor, 262
createGroupBoxGeometryEllipse
 PropertyEditor, 262
createGroupBoxGeometryImage
 PropertyEditor, 262
createGroupBoxGeometryInfiniteLine
 PropertyEditor, 262
createGroupBoxGeometryLine
 PropertyEditor, 262
createGroupBoxGeometryPath
 PropertyEditor, 262
createGroupBoxGeometryPoint
 PropertyEditor, 262
createGroupBoxGeometryPolygon
 PropertyEditor, 263
createGroupBoxGeometryPolyline
 PropertyEditor, 263
createGroupBoxGeometryRay
 PropertyEditor, 263
createGroupBoxGeometryRectangle
 PropertyEditor, 263
createGroupBoxGeometryTextMulti
 PropertyEditor, 263
createGroupBoxGeometryTextSingle
 PropertyEditor, 263
createGroupBoxMiscArc
 PropertyEditor, 263
createGroupBoxMiscImage
 PropertyEditor, 263
createGroupBoxMiscPath
 PropertyEditor, 263
createGroupBoxMiscPolyline
 PropertyEditor, 264
createGroupBoxMiscTextSingle
 PropertyEditor, 264
createGroupBoxTextTextSingle
 PropertyEditor, 264
createHandle
 CmdPromptSplitter, 105
createHelpMenu
 MainWindow, 180
createHelpToolbar
 MainWindow, 180
createHistogram
 EmbDetailsDialog, 127
createIconToolbar
 MainWindow, 180
createLayerToolbar
 MainWindow, 180
createLineEdit
 PropertyEditor, 264
createMainWidget
 EmbDetailsDialog, 127
createObjectList
 View, 346
createOrigin
 View, 346
createPanToolbar
 MainWindow, 180
createPromptToolbar
 MainWindow, 180
createPropertiesToolbar
 MainWindow, 180
createRulerTextPath
 View, 346
createSettingsMenu
 MainWindow, 180
createTabDisplay
 Settings_Dialog, 305
createTabFilesPaths
 Settings_Dialog, 305
createTabGeneral
 Settings_Dialog, 305
createTabGridRuler

Settings_Dialog, 305
createTabLineWeight
 Settings_Dialog, 305
createTabOpenSave
 Settings_Dialog, 305
createTabOrthoPolar
 Settings_Dialog, 305
createTabPrinting
 Settings_Dialog, 305
createTabPrompt
 Settings_Dialog, 305
createTabQuickSnap
 Settings_Dialog, 305
createTabQuickTrack
 Settings_Dialog, 305
createTabSelection
 Settings_Dialog, 305
createTabSnap
 Settings_Dialog, 305
createTextToolbar
 MainWindow, 180
createToolButton
 PropertyEditor, 264
createToolButtonPickAdd
 PropertyEditor, 264
createToolButtonQSelect
 PropertyEditor, 264
createViewMenu
 MainWindow, 180
createWindowMenu
 MainWindow, 181
creationTime
 _bcf_directory_entry, 50
creatorName
 ThredExtension_, 328
crosshairColor
 View, 352
crosshairSize
 View, 352
csd, 16, 536
csd_decryptArray
 format_csd.c, 537
CsdSubMaskSize
 format_csd.c, 536
CsdXorMaskSize
 format_csd.c, 536
csv, 538
CSV_EXPECT
 embroidery_internal.h, 490
CSV_EXPECT_COMMA
 embroidery_internal.h, 490
CSV_EXPECT_NULL
 embroidery_internal.h, 490
CSV_EXPECT_QUOTE1
 embroidery_internal.h, 490
CSV_EXPECT_QUOTE2
 embroidery_internal.h, 490
CSV_MODE
 embroidery_internal.h, 490
CSV_MODE_COMMENT
 embroidery_internal.h, 490
CSV_MODE_NULL
 embroidery_internal.h, 490
CSV_MODE_STITCH
 embroidery_internal.h, 490
CSV_MODE_THREAD
 embroidery_internal.h, 490
CSV_MODE_VARIABLE
 embroidery_internal.h, 490
csvStitchFlagToStr
 format_csv.c, 538
csvStrToStitchFlag
 format_csv.c, 538
CUBICTOCONTROL1
 embroidery_internal.h, 482
CUBICTOCONTROL2
 embroidery_internal.h, 482
CUBICTOEND
 embroidery_internal.h, 482
curCmd
 CmdPromptInput, 104
curColor
 MdiWindow, 234
curFile
 MdiWindow, 234
curLayer
 MdiWindow, 234
curLineType
 MdiWindow, 234
curLineWeight
 MdiWindow, 234
current_directory
 Settings_, 287
current_element_id
 format_svg.c, 566
currentAttribute
 format_svg.c, 566
currentColorChanged
 MdiWindow, 228
currentColorIndex
 EmbPattern_, 138
currentDisplayBackgroundColorChanged
 Settings_Dialog, 305
currentDisplayCrossHairColorChanged
 Settings_Dialog, 306
currentDisplaySelectBoxLeftColorChanged
 Settings_Dialog, 306
currentDisplaySelectBoxLeftFillChanged
 Settings_Dialog, 306
currentDisplaySelectBoxRightColorChanged
 Settings_Dialog, 306
currentDisplaySelectBoxRightFillChanged
 Settings_Dialog, 306
currentGeneralMdiBackgroundColorChanged
 Settings_Dialog, 306
currentGridColorChanged

Settings_Dialog, 306
 currentLayerChanged
 MdiWindow, 228
 currentLinetypeChanged
 MdiWindow, 228
 currentLineweightChanged
 MdiWindow, 228
 currentPromptBackgroundColorChanged
 Settings_Dialog, 306
 currentPromptTextColorChanged
 Settings_Dialog, 306
 currentRulerColorChanged
 Settings_Dialog, 306
 currentValue
 format_svg.c, 566
 curText
 CmdPromptInput, 104
 curved
 DimLeaderObject, 114
 cut
 MainWindow, 181
 View, 346
 cutCopyMousePoint
 View, 352
 cutCopyObjectList
 MainWindow, 207
 cutPressed
 CmdPrompt, 88
 CmdPromptInput, 101

 dat, 16
 data
 EmblImage_, 133
 UndoHistory_, 341
 day
 EmbTime_, 145
 dayVision
 MainWindow, 181
 debug_mode
 Settings_, 287
 decode_exy_flags
 format_exy.c, 545
 decode_record_flags
 format_dst.c, 541
 decode_t01_record
 embroidery_internal.h, 494
 encoding.c, 521
 decode_tajima_ternary
 embroidery_internal.h, 495
 encoding.c, 521
 decode_tap_record_flags
 format_tap.c, 568
 DecodeCsdByte
 format_csd.c, 537
 decodeNewStitch
 embroidery_internal.h, 495
 encoding.c, 521
 DEFAULT_MODE
 embroidermodder.h, 366

 default_value
 Huffman, 151
 defaultPrefix
 CmdPromptInput, 104
 degrees
 embroidermodder.h, 367
 embroidery.h, 450
 functions.c, 582
 deleteObject
 View, 346
 deletePressed
 CmdPrompt, 88
 CmdPromptInput, 101
 MainWindow, 181
 MdiWindow, 229
 View, 346
 deleteSelected
 View, 346
 dem, 16, 539
 description
 EmbFormatList_, 130
 EmbThread_, 144
 designDetails
 MainWindow, 181
 MdiWindow, 229
 dialog
 embroidermodder.h, 367
 mainwindow.cpp, 410
 dialog_display_bg_color
 Settings_Dialog, 309
 dialog_display_crosshair_color
 Settings_Dialog, 309
 dialog_display_crosshair_percent
 Settings_Dialog, 309
 dialog_display_renderhint_aa
 Settings_Dialog, 309
 dialog_display_renderhint_high_aa
 Settings_Dialog, 309
 dialog_display_renderhint_noncosmetic
 Settings_Dialog, 309
 dialog_display_renderhint_smooth_pix
 Settings_Dialog, 309
 dialog_display_renderhint_text_aa
 Settings_Dialog, 309
 dialog_display_scrollbar_widget_num
 Settings_Dialog, 309
 dialog_display_selectbox_alpha
 Settings_Dialog, 309
 dialog_display_selectbox_left_color
 Settings_Dialog, 309
 dialog_display_selectbox_left_fill
 Settings_Dialog, 309
 dialog_display_selectbox_right_color
 Settings_Dialog, 309
 dialog_display_selectbox_right_fill
 Settings_Dialog, 309
 dialog_display_show_scrollbars
 Settings_Dialog, 310

dialog_display_units
 Settings_Dialog, 310
dialog_display_use_opengl
 Settings_Dialog, 310
dialog_display_zoomscale_in
 Settings_Dialog, 310
dialog_display_zoomscale_out
 Settings_Dialog, 310
dialog_general_icon_size
 Settings_Dialog, 310
dialog_general_icon_theme
 Settings_Dialog, 310
dialog_general_language
 Settings_Dialog, 310
dialog_general_mdi_bg_color
 Settings_Dialog, 310
dialog_general_mdi_bg_logo
 Settings_Dialog, 310
dialog_general_mdi_bg_texture
 Settings_Dialog, 310
dialog_general_mdi_bg_use_color
 Settings_Dialog, 310
dialog_general_mdi_bg_use_logo
 Settings_Dialog, 310
dialog_general_mdi_bg_use_texture
 Settings_Dialog, 310
dialog_general_system_help_browser
 Settings_Dialog, 310
dialog_general_tip_of_the_day
 Settings_Dialog, 310
dialog_grid_center_on_origin
 Settings_Dialog, 310
dialog_grid_center_x
 Settings_Dialog, 310
dialog_grid_center_y
 Settings_Dialog, 311
dialog_grid_color
 Settings_Dialog, 311
dialog_grid_color_match_crosshair
 Settings_Dialog, 311
dialog_grid_load_from_file
 Settings_Dialog, 311
dialog_grid_show_on_load
 Settings_Dialog, 311
dialog_grid_show_origin
 Settings_Dialog, 311
dialog_grid_size_radius
 Settings_Dialog, 311
dialog_grid_size_x
 Settings_Dialog, 311
dialog_grid_size_y
 Settings_Dialog, 311
dialog_grid_spacing_angle
 Settings_Dialog, 311
dialog_grid_spacing_radius
 Settings_Dialog, 311
dialog_grid_spacing_x
 Settings_Dialog, 311
dialog_grid_spacing_y
 Settings_Dialog, 311
dialog_grid_type
 Settings_Dialog, 311
dialog_lwt_default_lwt
 Settings_Dialog, 311
dialog_lwt_real_render
 Settings_Dialog, 311
dialog_lwt_show_lwt
 Settings_Dialog, 311
dialog_opensave_custom_filter
 Settings_Dialog, 311
dialog_opensave_open_format
 Settings_Dialog, 312
dialog_opensave_open_thumbnail
 Settings_Dialog, 312
dialog_opensave_recent_max_files
 Settings_Dialog, 312
dialog_opensave_save_format
 Settings_Dialog, 312
dialog_opensave_save_thumbnail
 Settings_Dialog, 312
dialog_opensave_trim_dst_num_jumps
 Settings_Dialog, 312
dialog_printing_default_device
 Settings_Dialog, 312
dialog_printing_disable_bg
 Settings_Dialog, 312
dialog_printing_use_last_device
 Settings_Dialog, 312
dialog_prompt_bg_color
 Settings_Dialog, 312
dialog_prompt_font_family
 Settings_Dialog, 312
dialog_prompt_font_size
 Settings_Dialog, 312
dialog_prompt_font_style
 Settings_Dialog, 312
dialog_prompt_save_history
 Settings_Dialog, 312
dialog_prompt_save_history_as_html
 Settings_Dialog, 312
dialog_prompt_save_history_filename
 Settings_Dialog, 312
dialog_prompt_text_color
 Settings_Dialog, 312
dialog_qsnap_aperture_size
 Settings_Dialog, 312
dialog_qsnap_apparent
 Settings_Dialog, 313
dialog_qsnap_center
 Settings_Dialog, 313
dialog_qsnap_enabled
 Settings_Dialog, 313
dialog_qsnap_endpoint
 Settings_Dialog, 313
dialog_qsnap_extension
 Settings_Dialog, 313

dialog_qsnap_insertion
 Settings_Dialog, 313
 dialog_qsnap_intersection
 Settings_Dialog, 313
 dialog_qsnap_locator_color
 Settings_Dialog, 313
 dialog_qsnap_locator_size
 Settings_Dialog, 313
 dialog_qsnap_midpoint
 Settings_Dialog, 313
 dialog_qsnap_nearest
 Settings_Dialog, 313
 dialog_qsnap_node
 Settings_Dialog, 313
 dialog_qsnap_parallel
 Settings_Dialog, 313
 dialog_qsnap_perpendicular
 Settings_Dialog, 313
 dialog_qsnap_quadrant
 Settings_Dialog, 313
 dialog_qsnap_tangent
 Settings_Dialog, 313
 dialog_ruler_color
 Settings_Dialog, 313
 dialog_ruler_metric
 Settings_Dialog, 313
 dialog_ruler_pixel_size
 Settings_Dialog, 314
 dialog_ruler_show_on_load
 Settings_Dialog, 314
 dialog_selection_coolgrip_color
 Settings_Dialog, 314
 dialog_selection_grip_size
 Settings_Dialog, 314
 dialog_selection_hotgrip_color
 Settings_Dialog, 314
 dialog_selection_mode_pickadd
 Settings_Dialog, 314
 dialog_selection_mode_pickdrag
 Settings_Dialog, 314
 dialog_selection_mode_pickfirst
 Settings_Dialog, 314
 dialog_selection_pickbox_size
 Settings_Dialog, 314
 Dictionary
 embroidermodder.h, 361
 difat
 _bcf_file, 51
 difatEntriesInHeader
 main.c, 600
 dimensions
 EmblImage_, 133
 DimLeaderObject, 107
 ~DimLeaderObject, 111
 allGripPoints, 111
 ArrowStyle, 110
 arrowStyleAngle, 114
 arrowStyleLength, 114
 arrowStylePath, 114
 Box, 110
 Closed, 110
 curved, 114
 DimLeaderObject, 110, 111
 Dot, 110
 filled, 114
 Flared, 110
 Fletching, 110
 gripEdit, 111
 init, 111
 lineStyle, 110
 lineStyleAngle, 114
 lineStyleLength, 114
 lineStylePath, 115
 mouseSnapPoint, 111
 NoArrow, 110
 NoLine, 110
 objectAngle, 111
 objectDeltaX, 112
 objectDeltaY, 112
 objectEndPoint1, 112
 objectEndPoint2, 112
 objectLength, 112
 objectMidPoint, 112
 objectX1, 112
 objectX2, 112
 objectY1, 112
 objectY2, 112
 Open, 110
 paint, 112
 setObjectEndPoint1, 113
 setObjectEndPoint2, 113
 setObjectX1, 113
 setObjectX2, 113
 setObjectY1, 113
 setObjectY2, 113
 Tick, 110
 Type, 110
 type, 113
 updateLeader, 114
 updateRubber, 114
 vulcanize, 114
 dirBrush
 SelectBox, 284
 directory
 _bcf_file, 51
 directoryEntryName
 _bcf_directory_entry, 50
 directoryEntryNameLength
 _bcf_directory_entry, 50
 dirEntries
 _bcf_directory, 49
 dirPen
 SelectBox, 284
 disableLwt
 StatusBarButton, 318
 disableMoveRapidFire

MainWindow, 181
View, 346
disablePromptRapidFire
 MainWindow, 181
disableRapidFire
 CmdPrompt, 88
disableReal
 StatusBarButton, 318
display_bg_color
 Settings_, 287
display_crosshair_color
 Settings_, 287
display_crosshair_percent
 Settings_, 287
display_renderhint_aa
 Settings_, 287
display_renderhint_high_aa
 Settings_, 287
display_renderhint_noncosmetic
 Settings_, 287
display_renderhint_smooth_pix
 Settings_, 288
display_renderhint_text_aa
 Settings_, 288
display_scrollbar_widget_num
 Settings_, 288
display_selectbox_alpha
 Settings_, 288
display_selectbox_left_color
 Settings_, 288
display_selectbox_left_fill
 Settings_, 288
display_selectbox_right_color
 Settings_, 288
display_selectbox_right_fill
 Settings_, 288
display_show_scrollbars
 Settings_, 288
display_units
 Settings_, 288
display_use_opengl
 Settings_, 288
display_zoomscale_in
 Settings_, 288
display_zoomscale_out
 Settings_, 288
distance_huffman
 Compress, 107
docIndex
 MainWindow, 207
dockPropEdit
 MainWindow, 207
dockUndoEdit
 MainWindow, 207
DOLPHIN_MODE_NUM_POINTS
 embroidermodder.h, 366
DOLPHIN_MODE_NUM_POINTS_
 mainwindow.cpp, 410
DOLPHIN_MODE_XSCALE
 embroidermodder.h, 366
DOLPHIN_MODE_XSCALE_
 mainwindow.cpp, 410
DOLPHIN_MODE_YSCALE
 embroidermodder.h, 366
DOLPHIN_MODE_YSCALE_
 mainwindow.cpp, 410
done
 UndoableNavCommand, 336
doNothing
 MainWindow, 181
Dot
 DimLeaderObject, 110
downPressed
 CmdPrompt, 88
 CmdPromptInput, 101
dragon_curve
 fill.c, 524
drawBackground
 View, 346
drawForeground
 View, 346
drawRubberLine
 BaseObject, 75
dsb, 16, 539
dst, 16, 540
dstJumpsPerTrim
 EmbPattern_, 138
dsz, 16, 500, 542
dx
 UndoableMoveCommand, 335
 UndoableScaleCommand, 339
dxf, 16, 543
dxf_color
 embroidery.h, 440
DXF_VERSION_2000
 embroidery_internal.h, 482
DXF_VERSION_2002
 embroidery_internal.h, 482
DXF_VERSION_2004
 embroidery_internal.h, 482
DXF_VERSION_2006
 embroidery_internal.h, 482
DXF_VERSION_2007
 embroidery_internal.h, 482
DXF_VERSION_2009
 embroidery_internal.h, 482
DXF_VERSION_2010
 embroidery_internal.h, 482
DXF_VERSION_2013
 embroidery_internal.h, 482
DXF_VERSION_R10
 embroidery_internal.h, 482
DXF_VERSION_R11
 embroidery_internal.h, 483
DXF_VERSION_R12
 embroidery_internal.h, 483

DXF_VERSION_R13
embroidery_internal.h, 483

DXF_VERSION_R14
embroidery_internal.h, 483

DXF_VERSION_R15
embroidery_internal.h, 483

DXF_VERSION_R18
embroidery_internal.h, 483

DXF_VERSION_R21
embroidery_internal.h, 483

DXF_VERSION_R24
embroidery_internal.h, 483

DXF_VERSION_R27
embroidery_internal.h, 483

dy
UndoableMoveCommand, 335
UndoableScaleCommand, 339

edit_toolbar
brodermodder.h, 368
mainwindow-toolbars.cpp, 406

editMenu
MainWindow, 207

edr, 16, 500, 543

ELEMENT_A
embroidery_internal.h, 483

ELEMENT_ANIMATE
embroidery_internal.h, 483

ELEMENT_ANIMATECOLOR
embroidery_internal.h, 483

ELEMENT_ANIMATEMOTION
embroidery_internal.h, 483

ELEMENT_ANIMATETRANSFORM
embroidery_internal.h, 483

ELEMENT_ANIMATION
embroidery_internal.h, 483

ELEMENT_AUDIO
embroidery_internal.h, 483

ELEMENT_CIRCLE
embroidery_internal.h, 483

ELEMENT_DEFS
embroidery_internal.h, 483

ELEMENT_DESC
embroidery_internal.h, 484

ELEMENT_DISCARD
embroidery_internal.h, 484

ELEMENT_ELLIPSE
embroidery_internal.h, 484

ELEMENT_FONT
embroidery_internal.h, 484

ELEMENT_FONT_FACE
embroidery_internal.h, 484

ELEMENT_FONT_FACE_SRC
embroidery_internal.h, 484

ELEMENT_FONT_FACE_URI
embroidery_internal.h, 484

ELEMENT_FOREIGN_OBJECT
embroidery_internal.h, 484

ELEMENT_G
embroidery_internal.h, 484

ELEMENT_GLYPH
embroidery_internal.h, 484

ELEMENT_HANDLER
embroidery_internal.h, 484

ELEMENT_HKERN
embroidery_internal.h, 484

ELEMENT_IMAGE
embroidery_internal.h, 484

ELEMENT_LINE
embroidery_internal.h, 484

ELEMENT_LINEAR_GRADIENT
embroidery_internal.h, 484

ELEMENT_LISTENER
embroidery_internal.h, 484

ELEMENT_METADATA
embroidery_internal.h, 484

ELEMENT_MISSING_GLYPH
embroidery_internal.h, 484

ELEMENT_MPATH
embroidery_internal.h, 485

ELEMENT_PATH
embroidery_internal.h, 485

ELEMENT_POLYGON
embroidery_internal.h, 485

ELEMENT_POLYLINE
embroidery_internal.h, 485

ELEMENT_PREFETCH
embroidery_internal.h, 485

ELEMENT_RADIAL_GRADIENT
embroidery_internal.h, 485

ELEMENT_RECT
embroidery_internal.h, 485

ELEMENT_SCRIPT
embroidery_internal.h, 485

ELEMENT_SET
embroidery_internal.h, 485

ELEMENT_SOLID_COLOR
embroidery_internal.h, 485

ELEMENT_STOP
embroidery_internal.h, 485

ELEMENT_SVG
embroidery_internal.h, 485

ELEMENT_SWITCH
embroidery_internal.h, 485

ELEMENT_TBREAK
embroidery_internal.h, 485

ELEMENT_TEXT
embroidery_internal.h, 485

ELEMENT_TEXT_AREA
embroidery_internal.h, 485

ELEMENT_TITLE
embroidery_internal.h, 485

ELEMENT_TSPAN
embroidery_internal.h, 485

ELEMENT_USE
embroidery_internal.h, 486

ELEMENT_VIDEO

embroidery_internal.h, 486
ELEMENT_XML
 embroidery_internal.h, 486
ellipse
 EmbGeometry_, 131
ellipse.c
 ellipse_objectQuadrant0, 581
 ellipse_objectQuadrant180, 581
 ellipse_objectQuadrant270, 581
 ellipse_objectQuadrant90, 581
 embEllipse_area, 581
 embEllipse_diameterX, 581
 embEllipse_diameterY, 581
 embEllipse_height, 581
 embEllipse_init, 581
 embEllipse_perimeter, 581
 embEllipse_setDiameterMajor, 581
 embEllipse_setDiameterMinor, 582
 embEllipse_setRadiusMajor, 582
 embEllipse_setRadiusMinor, 582
 embEllipse_setSize, 582
 embEllipse_updatePath, 582
 embEllipse_width, 582
ELLIPSE_MODE_ELLIPSE_ROTATION
 embroidermodder.h, 366
ELLIPSE_MODE_MAJORDIAMETER_MINORRADIUS
 embroidermodder.h, 366
ELLIPSE_MODE_MAJORRADIUS_MINORRADIUS
 embroidermodder.h, 366
ellipse_objectQuadrant0
 ellipse.c, 581
ellipse_objectQuadrant180
 ellipse.c, 581
ellipse_objectQuadrant270
 ellipse.c, 581
ellipse_objectQuadrant90
 ellipse.c, 581
EllipseObject, 115
 ~EllipseObject, 117
 allGripPoints, 118
 EllipseObject, 117
 gripEdit, 118
 init, 118
 mouseSnapPoint, 118
 objectDiameterMajor, 118
 objectDiameterMinor, 118
 objectHeight, 119
 objectQuadrant0, 119
 objectQuadrant180, 119
 objectQuadrant270, 119
 objectQuadrant90, 119
 objectRadiusMajor, 119
 objectRadiusMinor, 119
 objectSavePath, 119
 objectWidth, 119
 paint, 119
 setObjectDiameterMajor, 119
 setObjectDiameterMinor, 120
setObjectRadiusMajor, 120
setObjectRadiusMinor, 120
setObjectSize, 120
Type, 117
type, 120
updatePath, 120
updateRubber, 120
vulcanize, 120
ELLIPSETOEND
 embroidery_internal.h, 486
ELLIPSETORAD
 embroidery_internal.h, 486
Elna, 544
Eltac, 545
EMB_ARC
 embroidery.h, 440
EMB_ARRAY
 embroidery.h, 441
EMB_BIG_ENDIAN
 embroidery_internal.h, 486
EMB_CIRCLE
 embroidery.h, 441
emb_constant_pi
 embroidermodder.h, 368
EMB_DIM_DIAMETER
 embroidery.h, 441
EMB_DIM_LEADER
 embroidery.h, 441
EMB_ELLIPSE
 embroidery.h, 441
emb_error
 embroidery.h, 465
 main.c, 600
EMB_FLAG
 embroidery.h, 441
EMB_FORMAT_100
 embroidery.h, 441
EMB_FORMAT_10O
 embroidery.h, 441
EMB_FORMAT_ART
 embroidery.h, 441
EMB_FORMAT_BMC
 embroidery.h, 441
EMB_FORMAT_BRO
 embroidery.h, 441
EMB_FORMAT_CND
 embroidery.h, 441
EMB_FORMAT_COL
 embroidery.h, 441
EMB_FORMAT_CSD
 embroidery.h, 441
EMB_FORMAT_CSV
 embroidery.h, 441
EMB_FORMAT_DAT
 embroidery.h, 441
EMB_FORMAT DEM
 embroidery.h, 441
EMB_FORMAT_DSB

embroidery.h, 441
EMB_FORMAT_DST
 embroidery.h, 442
EMB_FORMAT_DSZ
 embroidery.h, 442
EMB_FORMAT_DXF
 embroidery.h, 442
EMB_FORMAT_EDR
 embroidery.h, 442
EMB_FORMAT_EMD
 embroidery.h, 442
EMB_FORMAT_EXP
 embroidery.h, 442
EMB_FORMAT_EXY
 embroidery.h, 442
EMB_FORMAT_EYS
 embroidery.h, 442
EMB_FORMAT_FXY
 embroidery.h, 442
EMB_FORMAT_GC
 embroidery.h, 442
EMB_FORMAT_GNC
 embroidery.h, 442
EMB_FORMAT_GT
 embroidery.h, 442
EMB_FORMAT_HUS
 embroidery.h, 442
EMB_FORMAT_INB
 embroidery.h, 442
EMB_FORMAT_INF
 embroidery.h, 442
EMB_FORMAT_JEF
 embroidery.h, 442
EMB_FORMAT_KSM
 embroidery.h, 442
EMB_FORMAT_MAX
 embroidery.h, 442
EMB_FORMAT_MIT
 embroidery.h, 443
EMB_FORMAT_NEW
 embroidery.h, 443
EMB_FORMAT_OFM
 embroidery.h, 443
EMB_FORMAT_PCD
 embroidery.h, 443
EMB_FORMAT_PCM
 embroidery.h, 443
EMB_FORMAT_PCQ
 embroidery.h, 443
EMB_FORMAT_PCS
 embroidery.h, 443
EMB_FORMAT_PEC
 embroidery.h, 443
EMB_FORMAT_PEL
 embroidery.h, 443
EMB_FORMAT_PEM
 embroidery.h, 443
EMB_FORMAT_PES
 embroidery.h, 443
EMB_FORMAT_PHB
 embroidery.h, 443
EMB_FORMAT_PHC
 embroidery.h, 443
EMB_FORMAT_PLT
 embroidery.h, 443
EMB_FORMAT_RGB
 embroidery.h, 443
EMB_FORMAT_SEW
 embroidery.h, 443
EMB_FORMAT_SHV
 embroidery.h, 443
EMB_FORMAT_SST
 embroidery.h, 443
EMB_FORMAT_STX
 embroidery.h, 444
EMB_FORMAT_SVG
 embroidery.h, 444
EMB_FORMAT_T01
 embroidery.h, 444
EMB_FORMAT_T09
 embroidery.h, 444
EMB_FORMAT_TAP
 embroidery.h, 444
EMB_FORMAT_THR
 embroidery.h, 444
EMB_FORMAT_TXT
 embroidery.h, 444
EMB_FORMAT_U00
 embroidery.h, 444
EMB_FORMAT_U01
 embroidery.h, 444
EMB_FORMAT_VIP
 embroidery.h, 444
EMB_FORMAT_VP3
 embroidery.h, 444
EMB_FORMAT_XXX
 embroidery.h, 444
EMB_FORMAT_ZSK
 embroidery.h, 444
emb_identify_format
 embroidery.h, 450
 formats.c, 530
EMB_IMAGE
 embroidery.h, 444
EMB_INT16_BIG
 embroidery_internal.h, 486
EMB_INT16_LITTLE
 embroidery_internal.h, 486
EMB_INT32_BIG
 embroidery_internal.h, 486
EMB_INT32_LITTLE
 embroidery_internal.h, 486
EMB_LINE
 embroidery.h, 444
EMB_LITTLE_ENDIAN
 embroidery_internal.h, 486

EMB_MAX
embroidery_internal.h, 486

EMB_MAX_LAYERS
embroidery.h, 444

EMB_MIN
embroidery_internal.h, 486

emb_optOut
embroidery_internal.h, 495
main.c, 595

EMB_PATH
embroidery.h, 444

EMB_POINT
embroidery.h, 444

EMB_POLYGON
embroidery.h, 445

EMB_POLYLINE
embroidery.h, 445

EMB_PUBLIC
embroidery.h, 445

emb_readline
embroidery_internal.h, 495
main.c, 596

EMB_RECT
embroidery.h, 445

emb_round
embroidery.h, 450
functions.c, 582

EMB_SPLINE
embroidery.h, 445

EMB_STITCH
embroidery.h, 445

EMB_TEXT_MULTI
embroidery.h, 445

EMB_TEXT_SINGLE
embroidery.h, 445

EMB_THREAD
embroidery.h, 445

EMB_VECTOR
embroidery.h, 445

emb_verbose
embroidery.h, 465
main.c, 600

EmbAlignedDim
embroidery.h, 447

EmbAlignedDim_, 121
position, 121

EmbAngularDim
embroidery.h, 448

EmbAngularDim_, 121
position, 121

EmbArc
embroidery.h, 448

EmbArc_, 121
end, 122
mid, 122
start, 122

embArc_arcLength
arc.c, 577

embArc_area
arc.c, 577

embArc_chord
arc.c, 577

embArc_clockwise
arc.c, 577
embroidery.h, 450

embArc_endAngle
arc.c, 577

embArc_gripEdit
arc.c, 577

embArc_includedAngle
arc.c, 577

embArc_init
arc.c, 577
embroidery.h, 450

embArc_mouseSnapPoint
arc.c, 578

embArc_paint
arc.c, 578

embArc_print
main.c, 596

embArc_setCenter
arc.c, 578

embArc_setEndAngle
arc.c, 578

embArc_setRadius
arc.c, 578

embArc_setStartAngle
arc.c, 578

embArc_startAngle
arc.c, 578

embArc_updatePath
arc.c, 578

embArc_updateRubber
arc.c, 578

EmbArcLengthDim
embroidery.h, 448

EmbArcLengthDim_, 122
position, 122

EmbArray
embroidery.h, 448

EmbArray_, 123
count, 123
geometry, 123
length, 123
stitch, 123
thread, 123
type, 123

embArray_addArc
array.c, 429
embroidery.h, 450

embArray_addCircle
array.c, 429
embroidery.h, 450

embArray_addEllipse
array.c, 429
embroidery.h, 450

embArray_addFlag
 array.c, 430
 embroidery.h, 451

embArray_addLine
 array.c, 430
 embroidery.h, 451

embArray_addPath
 array.c, 430
 embroidery.h, 451

embArray_addPoint
 array.c, 430
 embroidery.h, 451

embArray_addPolygon
 array.c, 430
 embroidery.h, 451

embArray_addPolyline
 array.c, 430
 embroidery.h, 451

embArray_addRect
 array.c, 430
 embroidery.h, 451

embArray_addStitch
 array.c, 430
 embroidery.h, 451

embArray_addThread
 embroidery.h, 451

embArray_addVector
 array.c, 430
 embroidery.h, 451

embArray_copy
 array.c, 430
 embroidery.h, 451

embArray_create
 array.c, 430
 embroidery.h, 452

embArray_free
 array.c, 431
 embroidery.h, 452

embArray_resize
 array.c, 431
 embroidery.h, 452

embBaseSetColorRGB
 arc.c, 578

EmbBezier
 embroidery.h, 448

EmbBezier_, 124
 control1, 124
 control2, 124
 end, 124
 start, 124

EmbBlock
 embroidery.h, 448

EmbBlock_, 124
 position, 124

EmbCircle
 embroidery.h, 448

EmbCircle_, 125
 center, 125

radius, 125

embCircle_area
 circle.c, 580

embCircle_circumference
 circle.c, 580

embCircle_init
 circle.c, 580
 embroidery.h, 452

embCircle_prompt
 arc.c, 578

embCircle_setArea
 arc.c, 578

embCircle_setCircumference
 arc.c, 579

EmbColor
 embroidery.h, 448

EmbColor_, 125
 b, 126
 g, 126
 r, 126

embColor_create
 embroidery.h, 452

embColor_distance
 embroidery.h, 452
 main.c, 596

embColor_fromHexStr
 embroidery.h, 452
 encoding.c, 521

embColor_make
 embroidery.h, 452

embColor_read
 embroidery_internal.h, 495
 main.c, 596

embColor_write
 embroidery_internal.h, 495
 main.c, 596

embConstantPi
 embroidery.h, 465
 main.c, 600

EmbDetailsDialog, 126
 ~EmbDetailsDialog, 127
 boundingRect, 127
 buttonBox, 127
 colorChanges, 127
 colorTotal, 127
 createHistogram, 127
 createMainWidget, 127
 EmbDetailsDialog, 126
 getInfo, 127
 mainWidget, 127
 stitchesJump, 128
 stitchesReal, 128
 stitchesTotal, 128
 stitchesTrim, 128

EmbDiameterDim
 embroidery.h, 448

EmbDiameterDim_, 128
 position, 128

EmbEllipse
embroidery.h, 448
EmbEllipse_, 128
center, 129
radius, 129
rotation, 129
embEllipse_area
ellipse.c, 581
embroidery.h, 452
embEllipse_click
arc.c, 579
embEllipse_diameterX
ellipse.c, 581
embroidery.h, 452
embEllipse_diameterY
ellipse.c, 581
embroidery.h, 453
embEllipse_height
ellipse.c, 581
embroidery.h, 453
embEllipse_init
ellipse.c, 581
embroidery.h, 453
embEllipse_main
arc.c, 579
embEllipse_make
embroidery.h, 453
embEllipse_perimeter
ellipse.c, 581
embroidery.h, 453
embEllipse_setDiameterMajor
ellipse.c, 581
embEllipse_setDiameterMinor
ellipse.c, 582
embEllipse_setRadiusMajor
ellipse.c, 582
embEllipse_setRadiusMinor
ellipse.c, 582
embEllipse_setSize
ellipse.c, 582
embEllipse_updatePath
ellipse.c, 582
embEllipse_width
ellipse.c, 582
embroidery.h, 453
EmbFlag
embroidery.h, 448
embFormat_getExtension
formats.c, 530
EMBFORMAT_MAXDESC
embroidery.h, 445
EMBFORMAT_MAXEXT
embroidery.h, 445
EMBFORMAT_OBJECTONLY
embroidery.h, 445
EMBFORMAT_STCHANDOBJ
embroidery.h, 445
EMBFORMAT_STITCHONLY
embroidery.h, 445
EMBFORMAT_UNSUPPORTED
embroidery.h, 445
EmbFormatList
embroidery.h, 448
EmbFormatList_, 129
check_for_color_file, 129
color_only, 129
description, 130
extension, 130
reader_state, 130
type, 130
write_external_color_file, 130
writer_state, 130
EmbGeometry
embroidery.h, 448
EmbGeometry_, 130
arc, 131
circle, 131
color, 131
ellipse, 131
flag, 131
line, 131
lineType, 131
object, 131
path, 131
point, 131
polygon, 131
polyline, 132
rect, 132
spline, 132
stitch, 132
thread, 132
type, 132
vector, 132
embGeometry_boundingRect
embroidery.h, 453
geometry.c, 575
embGeometry_free
embroidery.h, 453
geometry.c, 575
embGeometry_init
embroidery.h, 453
geometry.c, 575
embGeometry_move
embroidery.h, 453
geometry.c, 575
embGeometry_vulcanize
embroidery.h, 454
geometry.c, 575
EmblImage
embroidery.h, 448
EmblImage_, 132
data, 133
dimensions, 133
height, 133
name, 133
path, 133

position, 133
 width, 133
emblImage_create
 embroidery.h, 454
emblImage_free
 embroidery.h, 454
emblImage_read
 embroidery.h, 454
emblImage_write
 embroidery.h, 454
EmbInfiniteLine
 embroidery.h, 448
EmbInfiniteLine_, 133
 position, 134
embInt_read
 embroidery_internal.h, 496
 encoding.c, 522
embInt_write
 embroidery_internal.h, 496
 encoding.c, 522
Embird, 500, 538, 543
EmbLayer
 embroidery.h, 448
EmbLayer_, 134
 geometry, 134
 name, 134
EmbLeaderDim
 embroidery.h, 448
EmbLeaderDim_, 134
 position, 135
EmbLine
 embroidery.h, 448
EmbLine_, 135
 color, 135
 end, 135
 lineType, 135
 start, 135
embLine_intersectionPoint
 embroidery.h, 454
 line.c, 583
embLine_make
 embroidery.h, 454
embLine_normalVector
 embroidery.h, 454
 line.c, 583
embLine_toVector
 line.c, 583
EmbLinearDim
 embroidery.h, 449
EmbLinearDim_, 136
 position, 136
EmbOrdinateDim
 embroidery.h, 449
EmbOrdinateDim_, 136
 position, 136
EmbPath
 embroidery.h, 449
EmbPath_, 136
 color, 137
 flagList, 137
 lineType, 137
 pointList, 137
EmbPattern
 embroidery.h, 449
EmbPattern_, 137
 currentColorIndex, 138
 dstJumpsPerTrim, 138
 geometry, 138
 home, 138
 hoop_height, 138
 hoop_width, 138
 layer, 138
 stitch_list, 138
 thread_list, 138
embPattern_addCircleAbs
 embroidery.h, 454
 pattern.c, 601
embPattern_addEllipseAbs
 embroidery.h, 454
 pattern.c, 601
embPattern_addLineAbs
 embroidery.h, 455
 pattern.c, 602
embPattern_addPathAbs
 embroidery.h, 455
 pattern.c, 602
embPattern_addPointAbs
 embroidery.h, 455
 pattern.c, 602
embPattern_addPolygonAbs
 embroidery.h, 455
 pattern.c, 602
embPattern_addPolylineAbs
 embroidery.h, 455
embPattern_addPolylineObjectAbs
 pattern.c, 602
embPattern_addRectAbs
 embroidery.h, 455
 pattern.c, 602
embPattern_addStitchAbs
 embroidery.h, 455
 pattern.c, 602
embPattern_addStitchRel
 embroidery.h, 455
 pattern.c, 602
embPattern_addThread
 embroidery.h, 455
 pattern.c, 602
embPattern_calcBoundingBox
 embroidery.h, 456
 pattern.c, 603
embPattern_center
 embroidery.h, 456
 pattern.c, 603
embPattern_changeColor
 embroidery.h, 456

pattern.c, 603
embPattern_color_count
 embroidery.h, 456
 pattern.c, 603
embPattern_combine
 embroidery.h, 456
 fill.c, 524
embPattern_combineJumpStitches
 embroidery.h, 456
 pattern.c, 603
embPattern_convertGeometry
 embroidery.h, 456
 fill.c, 524
embPattern_copyPolylinesToStitch_list
 pattern.c, 603
embPattern_copyPolylinesToStitchList
 embroidery.h, 456
embPattern_copystitch_listToPolylines
 pattern.c, 603
embPattern_copyStitchListToPolylines
 embroidery.h, 456
embPattern_correctForMaxStitchLength
 embroidery.h, 456
 pattern.c, 603
embPattern_create
 embroidery.h, 457
 pattern.c, 603
embPattern_crossstitch
 embroidery.h, 457
 fill.c, 524
embPattern_designDetails
 embroidery.h, 457
 pattern.c, 603
embPattern_end
 embroidery.h, 457
 pattern.c, 604
embPattern_fixColorCount
 embroidery.h, 457
 pattern.c, 604
embPattern_flip
 embroidery.h, 457
 pattern.c, 604
embPattern_flipHorizontal
 embroidery.h, 457
 pattern.c, 604
embPattern_flipVertical
 embroidery.h, 457
 pattern.c, 604
embPattern_free
 embroidery.h, 457
 pattern.c, 604
embPattern_hideStitchesOverLength
 embroidery.h, 458
 pattern.c, 604
embPattern_horizontal_fill
 embroidery.h, 458
 fill.c, 524
embPattern_jumpStitches
 embroidery.h, 458
 pattern.c, 604
embPattern_lengthHistogram
 embroidery.h, 458
 pattern.c, 604
embPattern_loadExternalColorFile
 embroidery.h, 458
 pattern.c, 604
embPattern_maximumStitchLength
 embroidery.h, 458
 pattern.c, 604
embPattern_minimumStitchLength
 embroidery.h, 458
 pattern.c, 604
embPattern_movePolylinesToStitch_list
 pattern.c, 605
embPattern_movePolylinesToStitchList
 embroidery.h, 458
embPattern_movestitch_listToPolylines
 pattern.c, 605
embPattern_moveStitchListToPolylines
 embroidery.h, 458
embPattern_read
 embroidery.h, 458
 formats.c, 530
embPattern_readAuto
 embroidery.h, 459
 formats.c, 530
embPattern_realStitches
 embroidery.h, 459
 pattern.c, 605
embPattern_render
 embroidery.h, 459
embPattern_scale
 embroidery.h, 459
 pattern.c, 605
embPattern_simulate
 embroidery.h, 459
embPattern_stitchArc
 fill.c, 524
embPattern_stitchCircle
 fill.c, 524
embPattern_stitchEllipse
 fill.c, 525
embPattern_stitchPath
 fill.c, 525
embPattern_stitchPolygon
 fill.c, 525
embPattern_stitchPolyline
 fill.c, 525
embPattern_stitchRect
 fill.c, 526
embPattern_stitchText
 fill.c, 526
embPattern_totalStitchLength
 embroidery.h, 459
 pattern.c, 605
embPattern_trimStitches

embroidery.h, 459
 pattern.c, 605
 embPattern_write
 embroidery.h, 459
 formats.c, 530
 embPattern_writeAuto
 embroidery.h, 460
 formats.c, 530
 EmbPoint
 embroidery.h, 449
 EmbPoint_, 138
 color, 139
 lineType, 139
 position, 139
 EmbPolygon
 embroidery.h, 449
 embPolygon_reduceByDistance
 fill.c, 526
 embPolygon_reduceByNth
 fill.c, 526
 EmbPolyline
 embroidery.h, 449
 EmbRadiusDim
 embroidery.h, 449
 EmbRadiusDim_, 139
 position, 139
 EmbRay
 embroidery.h, 449
 EmbRay_, 140
 position, 140
 EmbReal
 embroidery.h, 449
 EmbRect
 embroidery.h, 449
 EmbRect_, 140
 bottom, 140
 left, 140
 radius, 140
 right, 141
 rotation, 141
 top, 141
 embRect_area
 embroidery.h, 460
 rect.c, 584
 embRect_bottomLeft
 arc.c, 579
 embRect_bottomRight
 arc.c, 579
 embRect_init
 embroidery.h, 460
 rect.c, 584
 embroidermodder.cpp
 appName, 357
 appVer, 357
 exitApp, 357
 main, 357
 usage, 357
 version, 357
 embroidermodder.h
 Action, 361
 action_table, 367
 CIRCLE_MODE_1P_DIA, 366
 CIRCLE_MODE_1P_RAD, 366
 CIRCLE_MODE_2P, 366
 CIRCLE_MODE_3P, 366
 CIRCLE_MODE_TTR, 366
 DEFAULT_MODE, 366
 degrees, 367
 dialog, 367
 Dictionary, 361
 DOLPHIN_MODE_NUM_POINTS, 366
 DOLPHIN_MODE_XSCALE, 366
 DOLPHIN_MODE_YSCALE, 366
 edit_toolbar, 368
 ELLIPSE_MODE_ELLIPSE_ROTATION, 366
 ELLIPSE_MODE_MAJORDIAMETER_MINORRADIUS, 366
 ELLIPSE_MODE_MAJORRADIUS_MINORRADIUS, 366
 emb_constant_pi, 368
 EmbView, 361
 file_toolbar, 368
 get_action_index, 367
 HEART_MODE_NUM_POINTS, 366
 HEART_MODE_STYLE, 366
 HEART_MODE_XSCALE, 366
 HEART_MODE_YSCALE, 366
 Index, 361
 mainWin, 367
 OBJ_COLOR, 362
 OBJ_KEYS, 362
 OBJ_LAYER, 362
 OBJ_LTYPe, 362
 OBJ_LTYPe_CENTER, 362
 OBJ_LTYPe_CONT, 362
 OBJ_LTYPe_DOT, 362
 OBJ_LTYPe_FISHBONE, 363
 OBJ_LTYPe_HIDDEN, 362
 OBJ_LTYPe_PHANTOM, 362
 OBJ_LTYPe_RUNNING, 363
 OBJ_LTYPe_SATIN, 363
 OBJ_LTYPe_VALUES, 362
 OBJ_LTYPe_ZIGZAG, 363
 OBJ_LWT, 362
 OBJ_LWT_01, 363
 OBJ_LWT_02, 363
 OBJ_LWT_03, 363
 OBJ_LWT_04, 363
 OBJ_LWT_05, 363
 OBJ_LWT_06, 363
 OBJ_LWT_07, 363
 OBJ_LWT_08, 363
 OBJ_LWT_09, 363
 OBJ_LWT_10, 363
 OBJ_LWT_11, 363
 OBJ_LWT_12, 363

OBJ_LWT_13, 363
OBJ_LWT_14, 363
OBJ_LWT_15, 363
OBJ_LWT_16, 363
OBJ_LWT_17, 363
OBJ_LWT_18, 363
OBJ_LWT_19, 363
OBJ_LWT_20, 363
OBJ_LWT_21, 363
OBJ_LWT_22, 363
OBJ_LWT_23, 363
OBJ_LWT_24, 363
OBJ_LWT_BYBLOCK, 363
OBJ_LWT_BYLAYER, 363
OBJ_LWT_DEFAULT, 363
OBJ_LWT_VALUES, 363
OBJ_NAME, 362
OBJ_RUBBER, 362
OBJ_RUBBER_CIRCLE_1P_DIA, 364
OBJ_RUBBER_CIRCLE_1P_RAD, 363
OBJ_RUBBER_CIRCLE_2P, 364
OBJ_RUBBER_CIRCLE_3P, 364
OBJ_RUBBER_CIRCLE_TTR, 364
OBJ_RUBBER_CIRCLE_TTT, 364
OBJ_RUBBER_DIMLEADER_LINE, 364
OBJ_RUBBER_ELLIPSE_LINE, 364
OBJ_RUBBER_ELLIPSE_MAJOR_DIAMETER_MINOR_RADIUS, 364
OBJ_RUBBER_ELLIPSE_MAJOR_RADIUS_MINOR_RADIUS, 364
OBJ_RUBBER_ELLIPSE_ROTATION, 364
OBJ_RUBBER_GRIP, 364
OBJ_RUBBER_IMAGE, 364
OBJ_RUBBER_LINE, 364
OBJ_RUBBER_OFF, 363
OBJ_RUBBER_ON, 363
OBJ_RUBBER_POLYGON, 364
OBJ_RUBBER_POLYGON_CIRCUMSCRIBE, 364
OBJ_RUBBER_POLYGON_INSCRIBE, 364
OBJ_RUBBER_POLYLINE, 364
OBJ_RUBBER_RECTANGLE, 364
OBJ_RUBBER_TEXTSINGLE, 364
OBJ_RUBBER_VALUES, 363
OBJ_SNAP_APPINTERSECTION, 364
OBJ_SNAP_CENTER, 364
OBJ_SNAP_ENDPOINT, 364
OBJ_SNAP_EXTENSION, 364
OBJ_SNAP_INSERTION, 364
OBJ_SNAP_INTERSECTION, 364
OBJ_SNAP_MIDPOINT, 364
OBJ_SNAP_NEAREST, 364
OBJ_SNAP_NODE, 364
OBJ_SNAP_NULL, 364
OBJ_SNAP_PARALLEL, 364
OBJ_SNAP_PERPENDICULAR, 364
OBJ_SNAP_QUADRANT, 364
OBJ_SNAP_TANGENT, 364
OBJ_SNAP_VALUES, 364
OBJ_TYPE, 362
OBJ_TYPE_ARC, 365
OBJ_TYPE_BASE, 365
OBJ_TYPE_BLOCK, 365
OBJ_TYPE_CIRCLE, 365
OBJ_TYPE_DIMALIGNED, 365
OBJ_TYPE_DIMANGULAR, 365
OBJ_TYPE_DIMARCLENGTH, 365
OBJ_TYPE_DIMDIAMETER, 365
OBJ_TYPE_DIMLEADER, 365
OBJ_TYPE_DIMLINEAR, 365
OBJ_TYPE_DIMORDINATE, 365
OBJ_TYPE_DIMRADIUS, 365
OBJ_TYPE_ELLIPSE, 365
OBJ_TYPE_ELLIPSEARC, 365
OBJ_TYPE_GRID, 365
OBJ_TYPE_HATCH, 365
OBJ_TYPE_IMAGE, 365
OBJ_TYPE_INFINITELINE, 365
OBJ_TYPE_LINE, 365
OBJ_TYPE_NULL, 365
OBJ_TYPE_PATH, 365
OBJ_TYPE_POINT, 365
OBJ_TYPE_POLYGON, 365
OBJ_TYPE_POLYLINE, 365
OBJ_TYPE_RAY, 365
OBJ_TYPE_RECTANGLE, 365
OBJ_TYPE_RUBBER, 365
OBJ_TYPE_SLOT, 365
OBJ_TYPE_SPLINE, 365
OBJ_TYPE_TEXTMULTI, 365
OBJ_TYPE_TEXTSINGLE, 365
OBJ_TYPE_VALUES, 364
operator+, 367
operator-, 367
PREVIEW_CLONE_NULL, 365
PREVIEW_CLONE_RUBBER, 365
PREVIEW_CLONE_SELECTED, 365
PREVIEW_CLONE_VALUES, 365
PREVIEW_MODE_MOVE, 365
PREVIEW_MODE_NULL, 365
PREVIEW_MODE_ROTATE, 365
PREVIEW_MODE_SCALE, 366
PREVIEW_MODE_VALUES, 365
radians, 367
read_settings, 367
ROTATE_MODE_NORMAL, 366
ROTATE_MODE_REFERENCE, 366
SCALE_MODE_NORMAL, 366
SCALE_MODE_REFERENCE, 366
Settings, 361
settings, 368
SINGLE_LINE_TEXT_MODE_JUSTIFY, 366
SINGLE_LINE_TEXT_MODE_RAPID, 366
SINGLE_LINE_TEXT_MODE_SETFONT, 366
SINGLE_LINE_TEXT_MODE_SETGEOM, 366
SNOWFLAKE_MODE_NUM_POINTS, 366
SNOWFLAKE_MODE_XSCALE, 366

SNOWFLAKE_MODE_YSCALE, 366
SPARE_RUBBER_OFF, 366
SPARE_RUBBER_PATH, 366
SPARE_RUBBER_POLYGON, 366
SPARE_RUBBER_POLYLINE, 366
SPARE_RUBBER_VALUES, 366
STAR_MODE_CENTER_PT, 366
STAR_MODE_NUM_POINTS, 366
STAR_MODE_RAD_INNER, 366
STAR_MODE_RAD_OUTER, 366
to_EmbVector, 367
to_QPointF, 367
UiMode, 366
UiObject, 362
UndoHistory, 362
view_toolbar, 368
write_settings, 367
zoom_toolbar, 368
embroidermodder2/cmdprompt.cpp, 356
embroidermodder2/embdetails-dialog.cpp, 356
embroidermodder2/embroidermodder.cpp, 356
embroidermodder2/embroidermodder.h, 357, 368
embroidermodder2/imagewidget.cpp, 404
embroidermodder2/layer-manager.cpp, 404
embroidermodder2/mainwindow-commands.cpp, 404
embroidermodder2/mainwindow-menus.cpp, 404
embroidermodder2/mainwindow-settings.cpp, 404
embroidermodder2/mainwindow-toolbars.cpp, 406
embroidermodder2/mainwindow.cpp, 407
embroidermodder2/mdiarea.cpp, 411
embroidermodder2/mdiwindow.cpp, 411
embroidermodder2/object-arc.cpp, 411
embroidermodder2/object-base.cpp, 411
embroidermodder2/object-circle.cpp, 411
embroidermodder2/object-dimleader.cpp, 411
embroidermodder2/object-ellipse.cpp, 411
embroidermodder2/object-image.cpp, 411
embroidermodder2/object-line.cpp, 411
embroidermodder2/object-path.cpp, 412
embroidermodder2/object-point.cpp, 412
embroidermodder2/object-polygon.cpp, 412
embroidermodder2/object-polyline.cpp, 412
embroidermodder2/object-rect.cpp, 412
embroidermodder2/object-save.cpp, 412
embroidermodder2/object-textsingle.cpp, 412
embroidermodder2/preview-dialog.cpp, 412
embroidermodder2/property-editor.cpp, 412
embroidermodder2/README.md, 428
embroidermodder2/selectbox.cpp, 428
embroidermodder2/settings-dialog.cpp, 428
embroidermodder2/statusbar-button.cpp, 428
embroidermodder2/statusbar.cpp, 428
embroidermodder2/undo-commands.cpp, 428
embroidermodder2/undo-editor.cpp, 428
embroidermodder2/view.cpp, 429
embroidery.h
 _dxsetColorTable, 465
 Arc_Polyester, 440
Arc_Rayon, 440
black_thread, 465
CHUNK_SIZE, 440
CoatsAndClark_Rayon, 440
convert, 450
degrees, 450
dxf_color, 440
EMB_ARC, 440
EMB_ARRAY, 441
EMB_CIRCLE, 441
EMB_DIM_DIAMETER, 441
EMB_DIM_LEADER, 441
EMB_ELLIPSE, 441
emb_error, 465
EMB_FLAG, 441
EMB_FORMAT_100, 441
EMB_FORMAT_10O, 441
EMB_FORMAT_ART, 441
EMB_FORMAT_BMC, 441
EMB_FORMAT_BRO, 441
EMB_FORMAT_CND, 441
EMB_FORMAT_COL, 441
EMB_FORMAT_CSD, 441
EMB_FORMAT_CSV, 441
EMB_FORMAT_DAT, 441
EMB_FORMAT_DEM, 441
EMB_FORMAT_DSB, 441
EMB_FORMAT_DST, 442
EMB_FORMAT_DSZ, 442
EMB_FORMAT_DXF, 442
EMB_FORMAT_EDR, 442
EMB_FORMAT_EMD, 442
EMB_FORMAT_EXP, 442
EMB_FORMAT_EXY, 442
EMB_FORMAT_EYS, 442
EMB_FORMAT_FXY, 442
EMB_FORMAT_GC, 442
EMB_FORMAT_GNC, 442
EMB_FORMAT_GT, 442
EMB_FORMAT_HUS, 442
EMB_FORMAT_INB, 442
EMB_FORMAT_INF, 442
EMB_FORMAT_JEF, 442
EMB_FORMAT_KSM, 442
EMB_FORMAT_MAX, 442
EMB_FORMAT_MIT, 443
EMB_FORMAT_NEW, 443
EMB_FORMAT_OFM, 443
EMB_FORMAT_PCD, 443
EMB_FORMAT_PCM, 443
EMB_FORMAT_PCQ, 443
EMB_FORMAT_PCS, 443
EMB_FORMAT_PEC, 443
EMB_FORMAT_PEL, 443
EMB_FORMAT_PEM, 443
EMB_FORMAT_PES, 443
EMB_FORMAT_PHB, 443
EMB_FORMAT_PHC, 443

EMB_FORMAT_PLT, 443
EMB_FORMAT_RGB, 443
EMB_FORMAT_SEW, 443
EMB_FORMAT_SHV, 443
EMB_FORMAT_SST, 443
EMB_FORMAT_STX, 444
EMB_FORMAT_SVG, 444
EMB_FORMAT_T01, 444
EMB_FORMAT_T09, 444
EMB_FORMAT_TAP, 444
EMB_FORMAT_THR, 444
EMB_FORMAT_TXT, 444
EMB_FORMAT_U00, 444
EMB_FORMAT_U01, 444
EMB_FORMAT_VIP, 444
EMB_FORMAT_VP3, 444
EMB_FORMAT_XXX, 444
EMB_FORMAT_ZSK, 444
emb_identify_format, 450
EMB_IMAGE, 444
EMB_LINE, 444
EMB_MAX_LAYERS, 444
EMB_PATH, 444
EMB_POINT, 444
EMB_POLYGON, 445
EMB_POLYLINE, 445
EMB_PUBLIC, 445
EMB_RECT, 445
emb_round, 450
EMB_SPLINE, 445
EMB_STITCH, 445
EMB_TEXT_MULTI, 445
EMB_TEXT_SINGLE, 445
EMB_THREAD, 445
EMB_VECTOR, 445
emb_verbose, 465
EmbAlignedDim, 447
EmbAngularDim, 448
EmbArc, 448
embArc_clockwise, 450
embArc_init, 450
EmbArcLengthDim, 448
EmbArray, 448
embArray_addArc, 450
embArray_addCircle, 450
embArray_addEllipse, 450
embArray_addFlag, 451
embArray_addLine, 451
embArray_addPath, 451
embArray_addPoint, 451
embArray_addPolygon, 451
embArray_addPolyline, 451
embArray_addRect, 451
embArray_addStitch, 451
embArray_addThread, 451
embArray_addVector, 451
embArray_copy, 451
embArray_create, 452
embArray_free, 452
embArray_resize, 452
EmbBezier, 448
EmbBlock, 448
EmbCircle, 448
embCircle_init, 452
EmbColor, 448
embColor_create, 452
embColor_distance, 452
embColor_fromHexStr, 452
embColor_make, 452
embConstantPi, 465
EmbDiameterDim, 448
EmbEllipse, 448
embEllipse_area, 452
embEllipse_diameterX, 452
embEllipse_diameterY, 453
embEllipse_height, 453
embEllipse_init, 453
embEllipse_make, 453
embEllipse_perimeter, 453
embEllipse_width, 453
EmbFlag, 448
EMBFORMAT_MAXDESC, 445
EMBFORMAT_MAXEXT, 445
EMBFORMAT_OBJECTONLY, 445
EMBFORMAT_STCHANDOBJ, 445
EMBFORMAT_STITCHONLY, 445
EMBFORMAT_UNSUPPORTED, 445
EmbFormatList, 448
EmbGeometry, 448
embGeometry_boundingRect, 453
embGeometry_free, 453
embGeometry_init, 453
embGeometry_move, 453
embGeometry_vulcanize, 454
EmblImage, 448
emblImage_create, 454
emblImage_free, 454
emblImage_read, 454
emblImage_write, 454
EmblInfiniteLine, 448
EmbLayer, 448
EmbLeaderDim, 448
EmbLine, 448
embLine_intersectionPoint, 454
embLine_make, 454
embLine_normalVector, 454
EmbLinearDim, 449
EmbOrdinateDim, 449
EmbPath, 449
EmbPattern, 449
embPattern_addCircleAbs, 454
embPattern_addEllipseAbs, 454
embPattern_addLineAbs, 455
embPattern_addPathAbs, 455
embPattern_addPointAbs, 455
embPattern_addPolygonAbs, 455

embPattern_addPolylineAbs, 455
 embPattern_addRectAbs, 455
 embPattern_addStitchAbs, 455
 embPattern_addStitchRel, 455
 embPattern_addThread, 455
 embPattern_calcBoundingBox, 456
 embPattern_center, 456
 embPattern_changeColor, 456
 embPattern_color_count, 456
 embPattern_combine, 456
 embPattern_combineJumpStitches, 456
 embPattern_convertGeometry, 456
 embPattern_copyPolylinesToStitchList, 456
 embPattern_copyStitchListToPolylines, 456
 embPattern_correctForMaxStitchLength, 456
 embPattern_create, 457
 embPattern_crossstitch, 457
 embPattern_designDetails, 457
 embPattern_end, 457
 embPattern_fixColorCount, 457
 embPattern_flip, 457
 embPattern_flipHorizontal, 457
 embPattern_flipVertical, 457
 embPattern_free, 457
 embPattern_hideStitchesOverLength, 458
 embPattern_horizontal_fill, 458
 embPattern_jumpStitches, 458
 embPattern_lengthHistogram, 458
 embPattern_loadExternalColorFile, 458
 embPattern_maximumStitchLength, 458
 embPattern_minimumStitchLength, 458
 embPattern_movePolylinesToStitchList, 458
 embPattern_moveStitchListToPolylines, 458
 embPattern_read, 458
 embPattern_readAuto, 459
 embPattern_realStitches, 459
 embPattern_render, 459
 embPattern_scale, 459
 embPattern_simulate, 459
 embPattern_totalStitchLength, 459
 embPattern_trimStitches, 459
 embPattern_write, 459
 embPattern_writeAuto, 460
 EmbPoint, 449
 EmbPolygon, 449
 EmbPolyline, 449
 EmbRadiusDim, 449
 EmbRay, 449
 EmbReal, 449
 EmbRect, 449
 embRect_area, 460
 embRect_init, 460
 EmbSatinOutline, 449
 embSatinOutline_generateSatinOutline, 460
 embSatinOutline_renderStitches, 460
 EmbSpline, 449
 EmbStitch, 449
 EmbTextMulti, 449
 EmbTextSingle, 449
 EmbThread, 449
 embThread_findNearestColor, 460
 embThread_findNearestThread, 461
 embThread_getRandom, 461
 EmbTime, 449
 embTime_initNow, 461
 embTime_time, 461
 EmbVector, 450
 embVector_add, 461
 embVector_angle, 461
 embVector_average, 461
 embVector_cross, 462
 embVector_distance, 462
 embVector_dot, 462
 embVector_length, 462
 embVector_multiply, 462
 embVector_normalize, 462
 embVector_relativeX, 463
 embVector_relativeY, 463
 embVector_subtract, 463
 embVector_transpose_product, 463
 embVector_unit, 463
 END, 445
 Exquisite_Polyester, 445
 formatTable, 465
 Fufu_Polyester, 446
 Fufu_Rayon, 446
 full_test_matrix, 463
 getArcCenter, 463
 getArcDataFromBulge, 463
 getCircleCircleIntersections, 464
 getCircleTangentPoints, 464
 Hemingworth_Polyester, 446
 hilbert_curve, 464
 hus_thread, 446
 husThreads, 465
 Isacord_Polyester, 446
 Isafil_Rayon, 446
 jef_thread, 446
 jefThreads, 465
 JUMP, 446
 L_system, 450
 LIBEMBROIDERY_EMBEDDED_VERSION, 446
 lindenmayer_system, 464
 Madeira_Polyester, 446
 Madeira_Rayon, 446
 Marathon_Polyester, 446
 Marathon_Rayon, 446
 MAX_STITCHES, 446
 MAX_THREADS, 446
 Metro_Polyester, 446
 NORMAL, 446
 numberFormats, 446
 Pantone, 447
 pcm_thread, 447
 pcmThreads, 465
 pec_thread, 447

pecThreadCount, 465
pecThreads, 466
radians, 464
report, 464
RobisonAnton_Polyester, 447
RobisonAnton_Rayon, 447
SEQUIN, 447
shv_thread, 447
shvThreadCount, 466
shvThreads, 466
Sigma_Polyester, 447
STOP, 447
Sulky_Rayon, 447
SVG_Colors, 447
testMain, 464
thread_color, 450
ThreadArt_Polyester, 447
ThreadArt_Rayon, 447
threadColor, 465
threadColorName, 465
threadColorNum, 465
ThreaDelight_Polyester, 447
TRIM, 447
vipDecodingTable, 466
Z102_Isacord_Polyester, 447
embroidery_internal.h
 bcf_difat_create, 490
 bcf_directory, 489
 bcf_directory_entry, 489
 bcf_directory_free, 491
 bcf_file, 489
 bcf_file_difat, 489
 bcf_file_difat_free, 491
 bcf_file_fat, 489
 bcf_file_fat_free, 491
 bcf_file_free, 491
 bcf_file_header, 489
 bcfFile_read, 491
 bcfFileFat_create, 491
 bcfFileHeader_isValid, 491
 bcfFileHeader_read, 491
 binaryReadString, 491
 binaryReadUnicodeString, 492
 binaryWriteInt, 492
 binaryWriteIntBE, 492
 binaryWriteShort, 492
 binaryWriteUInt, 492
 binaryWriteUIntBE, 492
 binaryWriteUShort, 492
 binaryWriteUShortBE, 493
 BULGETOCONTROL, 481
 BULGETOEND, 481
 check_header_present, 493
 CompoundFileDirectory, 493
 CompoundFileDirectoryEntry, 493
 CompoundFileSector_DIFAT_Sector, 481
 CompoundFileSector_EndOfChain, 481
 CompoundFileSector_FAT_Sector, 482
 CompoundFileSector_FreeSector, 482
 CompoundFileSector_MaxRegSector, 482
 CompoundFileStreamId_MaxRegularStreamId, 482
 CompoundFileStreamId_NoStream, 482
 compress, 489
 compress_get_bits, 493
 compress_get_position, 493
 compress_get_token, 493
 compress_load_block, 494
 compress_load_character_huffman, 494
 compress_load_character_length_huffman, 494
 compress_load_distance_huffman, 494
 compress_pop, 494
 compress_read_variable_length, 494
 copy_trim, 494
 create_test_file_1, 494
 create_test_file_2, 494
 create_test_file_3, 494
 CSV_EXPECT, 490
 CSV_EXPECT_COMMA, 490
 CSV_EXPECT_NULL, 490
 CSV_EXPECT_QUOTE1, 490
 CSV_EXPECT_QUOTE2, 490
 CSV_MODE, 490
 CSV_MODE_COMMENT, 490
 CSV_MODE_NULL, 490
 CSV_MODE_STITCH, 490
 CSV_MODE_THREAD, 490
 CSV_MODE_VARIABLE, 490
 CUBICTOCONTROL1, 482
 CUBICTOCONTROL2, 482
 CUBICTOEND, 482
 decode_t01_record, 494
 decode_tajima_ternary, 495
 decodeNewStitch, 495
 DXF_VERSION_2000, 482
 DXF_VERSION_2002, 482
 DXF_VERSION_2004, 482
 DXF_VERSION_2006, 482
 DXF_VERSION_2007, 482
 DXF_VERSION_2009, 482
 DXF_VERSION_2010, 482
 DXF_VERSION_2013, 482
 DXF_VERSION_R10, 482
 DXF_VERSION_R11, 483
 DXF_VERSION_R12, 483
 DXF_VERSION_R13, 483
 DXF_VERSION_R14, 483
 DXF_VERSION_R15, 483
 DXF_VERSION_R18, 483
 DXF_VERSION_R21, 483
 DXF_VERSION_R24, 483
 DXF_VERSION_R27, 483
 ELEMENT_A, 483
 ELEMENT_ANIMATE, 483
 ELEMENT_ANIMATECOLOR, 483
 ELEMENT_ANIMATEMOTION, 483

ELEMENT_ANIMATETRANSFORM, 483
 ELEMENT_ANIMATION, 483
 ELEMENT_AUDIO, 483
 ELEMENT_CIRCLE, 483
 ELEMENT_DEFS, 483
 ELEMENT_DESC, 484
 ELEMENT_DISCARD, 484
 ELEMENT_ELLIPSE, 484
 ELEMENT_FONT, 484
 ELEMENT_FONT_FACE, 484
 ELEMENT_FONT_FACE_SRC, 484
 ELEMENT_FONT_FACE_URI, 484
 ELEMENT_FOREIGN_OBJECT, 484
 ELEMENT_G, 484
 ELEMENT_GLYPH, 484
 ELEMENT_HANDLER, 484
 ELEMENT_HKERN, 484
 ELEMENT_IMAGE, 484
 ELEMENT_LINE, 484
 ELEMENT_LINEAR_GRADIENT, 484
 ELEMENT_LISTENER, 484
 ELEMENT_METADATA, 484
 ELEMENT_MISSING_GLYPH, 484
 ELEMENT_MPATH, 485
 ELEMENT_PATH, 485
 ELEMENT_POLYGON, 485
 ELEMENT_POLYLINE, 485
 ELEMENT_PREFETCH, 485
 ELEMENT_RADIAL_GRADIENT, 485
 ELEMENT_RECT, 485
 ELEMENT_SCRIPT, 485
 ELEMENT_SET, 485
 ELEMENT_SOLID_COLOR, 485
 ELEMENT_STOP, 485
 ELEMENT_SVG, 485
 ELEMENT_SWITCH, 485
 ELEMENT_TBREAK, 485
 ELEMENT_TEXT, 485
 ELEMENT_TEXT_AREA, 485
 ELEMENT_TITLE, 485
 ELEMENT_TSPAN, 485
 ELEMENT_USE, 486
 ELEMENT_VIDEO, 486
 ELEMENT_XML, 486
 ELLIPSETOEND, 486
 ELLIPSETORAD, 486
 EMB_BIG_ENDIAN, 486
 EMB_INT16_BIG, 486
 EMB_INT16_LITTLE, 486
 EMB_INT32_BIG, 486
 EMB_INT32_LITTLE, 486
 EMB_LITTLE_ENDIAN, 486
 EMB_MAX, 486
 EMB_MIN, 486
 emb_optOut, 495
 emb_readline, 495
 embColor_read, 495
 embColor_write, 495
 emblInt_read, 496
 emblInt_write, 496
 encode_t01_record, 496
 encode_tajima_ternary, 496
 ENDIAN_HOST, 486
 entriesInDifatSector, 496
 fpad, 496
 fread_int16, 496
 fread_int32_be, 497
 fread_uint16, 497
 GetFile, 497
 GREEN_TERM_COLOR, 486
 HOOP_110X110, 486
 HOOP_126X110, 486
 HOOP_140X200, 487
 HOOP_230X200, 487
 HOOP_50X50, 487
 huffman, 489
 huffman_build_table, 497
 huffman_table_lookup, 497
 hus_compress, 497
 hus_decompress, 497
 imageWithFrame, 513
 LINETO, 487
 loadFatFromSector, 498
 mitDecodeStitch, 498
 mitEncodeStitch, 498
 MOVETO, 487
 N_PES VERSIONS, 487
 numberEntriesInDifatSector, 498
 ObjectTypeRootEntry, 487
 ObjectTypeStorage, 487
 ObjectTypeStream, 487
 ObjectTypeUnknown, 487
 PES0001, 487
 PES0020, 487
 PES0022, 487
 PES0030, 487
 PES0040, 487
 PES0050, 487
 PES0055, 487
 PES0056, 488
 PES0060, 488
 PES0070, 488
 PES0080, 488
 PES0090, 488
 PES0100, 488
 pfaffDecode, 498
 pfaffEncode, 498
 printArcResults, 498
 QUADTOCONTROL, 488
 QUADTOEND, 488
 read100, 499
 read10o, 499
 readArt, 499
 readBmc, 499
 readBro, 499
 readCnd, 499

readCol, 499
readCsd, 499
readCsv, 499
readDat, 499
readDem, 499
readDescriptions, 500
readDsb, 500
readDst, 500
readDsz, 500
readDxf, 500
readEdr, 500
readEmd, 500
readExp, 500
readExy, 500
readEys, 500
readFeatherPatterns, 501
readFullSector, 501
readFxy, 501
readGc, 501
readGnc, 501
readGt, 501
readHoopName, 501
readHus, 501
readImageString, 501
readInb, 502
readInf, 502
readJef, 502
readKsm, 502
readMax, 502
readMit, 502
readMotifPatterns, 502
readNew, 502
readNextSector, 502
readOfm, 502
readPcd, 503
readPcm, 503
readPcq, 503
readPcs, 503
readPec, 503
readPecStitches, 503
readPel, 503
readPem, 503
readPes, 504
readPESHeaderV10, 504
readPESHeaderV5, 504
readPESHeaderV6, 504
readPESHeaderV7, 504
readPESHeaderV8, 504
readPESHeaderV9, 504
readPhb, 504
readPhc, 504
readPlt, 504
readProgrammableFills, 505
readRgb, 505
readSew, 505
readShv, 505
readSst, 505
readStx, 505
readSvg, 505
readT01, 505
readT09, 505
readTap, 505
readThr, 506
readThreads, 506
readTxt, 506
readU00, 506
readU01, 506
readVip, 506
readVp3, 506
readXxx, 506
readZsk, 506
RED_TERM_COLOR, 488
RESET_TERM_COLOR, 488
safe_free, 506
stringInArray, 507
StxThread, 489
SubDescriptor, 490
SVG_ATTRIBUTE, 488
SVG_CATCH_ALL, 488
SVG_CREATOR_EMBROIDERMODDER, 488
SVG_CREATOR_ILLUSTRATOR, 488
SVG_CREATOR_INKSCAPE, 488
SVG_CREATOR_NULL, 488
SVG_ELEMENT, 488
SVG_EXPECT_ATTRIBUTE, 488
SVG_EXPECT_ELEMENT, 489
SVG_EXPECT_NULL, 489
SVG_EXPECT_VALUE, 489
SVG_MEDIA_PROPERTY, 489
SVG_NULL, 489
SVG_PROPERTY, 489
SvgAttribute, 490
testEmbCircle, 507
testEmbCircle_2, 507
testEmbFormat, 507
testGeomArc, 507
testTangentPoints, 507
testThreadColor, 507
ThredExtension, 490
ThredHeader, 490
VipHeader, 490
vp3Hoop, 490
write100, 507
write10o, 507
write_24bit, 507
writeArt, 508
writeBmc, 508
writeBro, 508
writeCnd, 508
writeCol, 508
writeCsd, 508
writeCsv, 508
writeDat, 508
writeDem, 508
writeDsb, 508
writeDst, 508

writeDsz, 509
 writeDxf, 509
 writeEdr, 509
 writeEmd, 509
 writeExp, 509
 writeExy, 509
 writeEys, 509
 writeFxy, 509
 writeGc, 509
 writeGnc, 509
 writeGt, 509
 writeHus, 510
 writeInb, 510
 writeInf, 510
 writeJef, 510
 writeKsm, 510
 writeMax, 510
 writeMit, 510
 writeNew, 510
 writeOfm, 510
 writePcd, 510
 writePcm, 510
 writePcq, 511
 writePcs, 511
 writePec, 511
 writePecStitches, 511
 writePel, 511
 writePem, 511
 writePes, 511
 writePhb, 511
 writePhc, 511
 writePlt, 511
 writeRgb, 512
 writeSew, 512
 writeShv, 512
 writeSst, 512
 writeStx, 512
 writeSvg, 512
 writeT01, 512
 writeT09, 512
 writeTap, 512
 writeThr, 512
 writeTxt, 512
 writeU00, 513
 writeU01, 513
 writeVip, 513
 writeVp3, 513
 writeXxx, 513
 writeZsk, 513
 YELLOW_TERM_COLOR, 489
EmbSatinOutline
 embroidery.h, 449
EmbSatinOutline_, 141
 length, 141
 side1, 141
 side2, 141
embSatinOutline_generateSatinOutline
 embroidery.h, 460
 main.c, 596
embSatinOutline_renderStitches
 embroidery.h, 460
 main.c, 596
EmbSpline
 embroidery.h, 449
EmbSpline_, 142
 beziers, 142
EmbStitch
 embroidery.h, 449
EmbStitch_, 142
 color, 142
 flags, 142
 x, 142
 y, 143
EmbTextMulti
 embroidery.h, 449
EmbTextMulti_, 143
 position, 143
 text, 143
EmbTextSingle
 embroidery.h, 449
EmbTextSingle_, 143
 position, 144
 text, 144
EmbThread
 embroidery.h, 449
EmbThread_, 144
 catalogNumber, 144
 color, 144
 description, 144
embThread_findNearestColor
 embroidery.h, 460
 main.c, 597
embThread_findNearestThread
 embroidery.h, 461
 main.c, 597
embThread_getRandom
 embroidery.h, 461
 main.c, 597
EmbTime
 embroidery.h, 449
EmbTime_, 145
 day, 145
 hour, 145
 minute, 145
 month, 145
 second, 145
 year, 145
embTime_initNow
 embroidery.h, 461
 main.c, 597
embTime_time
 embroidery.h, 461
 main.c, 597
EmbVector
 embroidery.h, 450
EmbVector_, 146

x, 146
y, 146
embVector_add
 embroidery.h, 461
 vector.c, 586
embVector_angle
 embroidery.h, 461
 vector.c, 586
embVector_average
 embroidery.h, 461
 vector.c, 586
embVector_cross
 embroidery.h, 462
 vector.c, 586
embVector_distance
 embroidery.h, 462
 vector.c, 587
embVector_dot
 embroidery.h, 462
 vector.c, 587
embVector_length
 embroidery.h, 462
 vector.c, 587
embVector_multiply
 embroidery.h, 462
 vector.c, 587
embVector_normalize
 embroidery.h, 462
 vector.c, 587
embVector_print
 main.c, 597
embVector_relativeX
 embroidery.h, 463
 vector.c, 587
embVector_relativeY
 embroidery.h, 463
 vector.c, 588
embVector_subtract
 embroidery.h, 463
 vector.c, 588
embVector_transpose_product
 embroidery.h, 463
 vector.c, 588
embVector_unit
 embroidery.h, 463
 vector.c, 588
EmbView
 embroidermodder.h, 361
EmbView_, 146
 filename, 147
 grid_mode, 147
 grid_type, 147
 lwt_mode, 148
 metric, 148
 n_selected, 148
 origin, 148
 ortho_mode, 148
 pattern, 148
polar_mode, 148
qsnap_mode, 148
qtrack_mode, 148
real_render, 148
rubber_mode, 148
ruler_mode, 149
scale, 149
selected, 149
simulate, 149
simulation_start, 149
snap_mode, 149
text_angle, 149
text_font, 149
text_size, 149
text_style_bold, 149
text_style_italic, 149
text_style_overline, 150
text_style_strikeout, 150
text_style_underline, 150
ui_mode, 150
undo_history, 150
emd, 16, 544
emdDecode
 format_emd.c, 544
enableLwt
 StatusBarButton, 318
enableMoveRapidFire
 MainWindow, 181
 View, 347
enablePromptRapidFire
 MainWindow, 181
enableRapidFire
 CmdPrompt, 88
enableReal
 StatusBarButton, 318
encode_record
 format_dst.c, 542
encode_t01_record
 embroidery_internal.h, 496
 encoding.c, 522
encode_tajima_ternary
 embroidery_internal.h, 496
 encoding.c, 522
encode_tap_record
 format_tap.c, 568
encoding.c
 decode_t01_record, 521
 decode_tajima_ternary, 521
 decodeNewStitch, 521
 embColor_fromHexStr, 521
 emblnt_read, 522
 emblnt_write, 522
 encode_t01_record, 522
 encode_tajima_ternary, 522
 mitDecodeStitch, 522
 mitEncodeStitch, 522
 pfaffDecode, 522
 pfaffEncode, 523

reverse_byte_order, 523
 write_24bit, 523
END
 embroidery.h, 445
end
 EmbArc_, 122
 EmbBezier_, 124
 EmbLine_, 135
endCommand
 CmdPrompt, 88
 CmdPromptInput, 101
ENDIAN_HOST
 embroidery_internal.h, 486
enterEvent
 View, 347
entriesInDifatSector
 embroidery_internal.h, 496
 main.c, 598
Error
 mainwindow.cpp, 409
escapePressed
 CmdPrompt, 88
 CmdPromptInput, 101
 MainWindow, 181
 MdiWindow, 229
 View, 347
event
 Application, 60
eventFilter
 CmdPromptInput, 101
 PropertyEditor, 264
exit
 MainWindow, 181
exitApp
 embroidermodder.cpp, 357
exp, 16, 544
expDecode
 format_exp.c, 544
Exquisite_Polyester
 embroidery.h, 445
extension
 EmbFormatList_, 130
 extern/libembroidery/src/array.c, 429
 extern/libembroidery/src/compress.c, 431
 extern/libembroidery/src/embroidery.h, 433, 466
 extern/libembroidery/src/embroidery_internal.h, 473,
 513
 extern/libembroidery/src/encoding.c, 520
 extern/libembroidery/src/fill.c, 523
 extern/libembroidery/src/formats.c, 528
 extern/libembroidery/src/formats/format_100.c, 532
 extern/libembroidery/src/formats/format_10o.c, 532
 extern/libembroidery/src/formats/format_art.c, 533
 extern/libembroidery/src/formats/format_bmc.c, 533
 extern/libembroidery/src/formats/format_bro.c, 534
 extern/libembroidery/src/formats/format_cnd.c, 534
 extern/libembroidery/src/formats/format_col.c, 535
 extern/libembroidery/src/formats/format_csd.c, 536
 extern/libembroidery/src/formats/format_csv.c, 537
 extern/libembroidery/src/formats/format_dat.c, 538
 extern/libembroidery/src/formats/format_dem.c, 539
 extern/libembroidery/src/formats/format_dsb.c, 539
 extern/libembroidery/src/formats/format_dst.c, 540
 extern/libembroidery/src/formats/format_dsz.c, 542
 extern/libembroidery/src/formats/format_dxf.c, 542
 extern/libembroidery/src/formats/format_edr.c, 543
 extern/libembroidery/src/formats/format_emd.c, 544
 extern/libembroidery/src/formats/format_exp.c, 544
 extern/libembroidery/src/formats/format_exy.c, 545
 extern/libembroidery/src/formats/format_eyc.c, 545
 extern/libembroidery/src/formats/format_fxy.c, 546
 extern/libembroidery/src/formats/format_gc.c, 546
 extern/libembroidery/src/formats/format_gnc.c, 547
 extern/libembroidery/src/formats/format_gt.c, 547
 extern/libembroidery/src/formats/format_hus.c, 548
 extern/libembroidery/src/formats/format_inb.c, 549
 extern/libembroidery/src/formats/format_inf.c, 549
 extern/libembroidery/src/formats/format_jef.c, 550
 extern/libembroidery/src/formats/format_ksm.c, 551
 extern/libembroidery/src/formats/format_max.c, 551
 extern/libembroidery/src/formats/format_mit.c, 552
 extern/libembroidery/src/formats/format_new.c, 553
 extern/libembroidery/src/formats/format_ofm.c, 553
 extern/libembroidery/src/formats/format_pcd.c, 554
 extern/libembroidery/src/formats/format_pcm.c, 555
 extern/libembroidery/src/formats/format_pcq.c, 555
 extern/libembroidery/src/formats/format_pcs.c, 556
 extern/libembroidery/src/formats/format_pec.c, 556
 extern/libembroidery/src/formats/format_pel.c, 558
 extern/libembroidery/src/formats/format_pem.c, 558
 extern/libembroidery/src/formats/format_pes.c, 559
 extern/libembroidery/src/formats/format_phb.c, 561
 extern/libembroidery/src/formats/format_phc.c, 562
 extern/libembroidery/src/formats/format_plt.c, 562
 extern/libembroidery/src/formats/format_rgb.c, 563
 extern/libembroidery/src/formats/format_sew.c, 563
 extern/libembroidery/src/formats/format_shv.c, 564
 extern/libembroidery/src/formats/format_sst.c, 564
 extern/libembroidery/src/formats/format_stx.c, 565
 extern/libembroidery/src/formats/format_svg.c, 565
 extern/libembroidery/src/formats/format_t01.c, 567
 extern/libembroidery/src/formats/format_t09.c, 567
 extern/libembroidery/src/formats/format_tap.c, 568
 extern/libembroidery/src/formats/format_thr.c, 568
 extern/libembroidery/src/formats/format_txt.c, 569
 extern/libembroidery/src/formats/format_u00.c, 569
 extern/libembroidery/src/formats/format_u01.c, 570
 extern/libembroidery/src/formats/format_vip.c, 570
 extern/libembroidery/src/formats/format_vp3.c, 572
 extern/libembroidery/src/formats/format_xxx.c, 573
 extern/libembroidery/src/formats/format_zsk.c, 574
 extern/libembroidery/src/geometry.c, 574
 extern/libembroidery/src/geometry/arc.c, 576
 extern/libembroidery/src/geometry/circle.c, 579
 extern/libembroidery/src/geometry/ellipse.c, 580
 extern/libembroidery/src/geometry/functions.c, 582

extern/libembroidery/src/geometry/line.c, 583
extern/libembroidery/src/geometry/path.c, 583
extern/libembroidery/src/geometry/polygon.c, 583
extern/libembroidery/src/geometry/polyline.c, 583
extern/libembroidery/src/geometry/rect.c, 584
extern/libembroidery/src/geometry/text.c, 584
extern/libembroidery/src/geometry/vector.c, 586
extern/libembroidery/src/image.c, 588
extern/libembroidery/src/main.c, 589
extern/libembroidery/src/pattern.c, 600
extern/libembroidery/src/thread-color.c, 605
exy, 16, 545
eys, 501, 546

F10Pressed
 CmdPrompt, 88
 CmdPromptInput, 101

F11Pressed
 CmdPrompt, 88
 CmdPromptInput, 101

F12Pressed
 CmdPrompt, 88
 CmdPromptInput, 101

F1Pressed
 CmdPrompt, 88
 CmdPromptInput, 101

F2Pressed
 CmdPrompt, 89
 CmdPromptInput, 102

F3Pressed
 CmdPrompt, 89
 CmdPromptInput, 102

F4Pressed
 CmdPrompt, 89
 CmdPromptInput, 102

F5Pressed
 CmdPrompt, 89
 CmdPromptInput, 102

F6Pressed
 CmdPrompt, 89
 CmdPromptInput, 102

F7Pressed
 CmdPrompt, 89
 CmdPromptInput, 102

F8Pressed
 CmdPrompt, 89
 CmdPromptInput, 102

F9Pressed
 CmdPrompt, 89
 CmdPromptInput, 102

factor
 UndoableScaleCommand, 339

fat
 _bcf_file, 51

fatEntries
 _bcf_file_fat, 52

fatEntryCount
 _bcf_file_fat, 53

fatSectorCount
 _bcf_file_fat, 52

fatSectorEntries
 _bcf_file_fat, 52

fieldEdited
 PropertyEditor, 264

fieldNewText
 PropertyEditor, 266

fieldNoText
 PropertyEditor, 266

fieldOffText
 PropertyEditor, 266

fieldOldText
 PropertyEditor, 266

fieldOnText
 PropertyEditor, 267

fieldVariesText
 PropertyEditor, 267

fieldYesText
 PropertyEditor, 267

file_toolbar
 embroidermodder.h, 368
 mainwindow-toolbars.cpp, 406

fileExtension
 MdiWindow, 229

fileMenu
 MainWindow, 207

filename
 EmbView_, 147

fileWasLoaded
 MdiWindow, 234

fill.c
 dragon_curve, 524
 embPattern_combine, 524
 embPattern_convertGeometry, 524
 embPattern_crossstitch, 524
 embPattern_horizontal_fill, 524
 embPattern_stitchArc, 524
 embPattern_stitchCircle, 524
 embPattern_stitchEllipse, 525
 embPattern_stitchPath, 525
 embPattern_stitchPolygon, 525
 embPattern_stitchPolyline, 525
 embPattern_stitchRect, 526
 embPattern_stitchText, 526
 embPolygon_reduceByDistance, 526
 embPolygon_reduceByNth, 526
 generate_dragon_curve, 526
 greedy_algorithm, 526
 hilbert_curve, 526
 hilbert_curve_l_system, 527
 join_short_stitches, 527
 lindenmayer_system, 527
 rules, 527
 save_points_to_pattern, 527
 threshold_method, 527

filled
 DimLeaderObject, 114

findIndex

PolygonObject, 248
 PolylineObject, 254
 findMdiWindow
 MainWindow, 181
 firstDifatSectorLocation
 _bcf_file_header, 54
 firstDirectorySectorLocation
 _bcf_file_header, 54
 firstMiniFATSectorLocation
 _bcf_file_header, 54
 firstRun
 UiObject, 330
 flag
 EmbGeometry, 131
 FLAG_CIRCLE
 main.c, 591
 FLAG_CIRCLE_SHORT
 main.c, 592
 FLAG_COMBINE
 main.c, 592
 FLAG_CROSS_STITCH
 main.c, 592
 FLAG_ELLIPSE
 main.c, 592
 FLAG_ELLIPSE_SHORT
 main.c, 592
 FLAG_FILL
 main.c, 592
 FLAG_FILL_SHORT
 main.c, 592
 FLAG_FORMATS
 main.c, 592
 FLAG_FORMATS_SHORT
 main.c, 592
 FLAG_FULL_TEST_SUITE
 main.c, 592
 FLAG_HELP
 main.c, 592
 FLAG_HELP_SHORT
 main.c, 592
 FLAG_HILBERT_CURVE
 main.c, 592
 FLAG_LINE
 main.c, 592
 FLAG_LINE_SHORT
 main.c, 592
 FLAG_POLYGON
 main.c, 592
 FLAG_POLYGON_SHORT
 main.c, 592
 FLAG_POLYLINE
 main.c, 592
 FLAG_POLYLINE_SHORT
 main.c, 593
 FLAG QUIET
 main.c, 593
 FLAG QUIET_SHORT
 main.c, 593
 FLAG_RENDER
 main.c, 593
 FLAG_RENDER_SHORT
 main.c, 593
 FLAG_SATIN
 main.c, 593
 FLAG_SATIN_SHORT
 main.c, 593
 FLAG_SIERPINSKI_TRIANGLE
 main.c, 593
 FLAG_SIMULATE
 main.c, 593
 FLAG_STITCH
 main.c, 593
 FLAG_STITCH_SHORT
 main.c, 593
 FLAG_TEST
 main.c, 593
 FLAG_TO
 main.c, 593
 FLAG_TO_SHORT
 main.c, 593
 FLAG_VERBOSE
 main.c, 593
 FLAG_VERBOSE_SHORT
 main.c, 593
 FLAG_VERSION
 main.c, 593
 FLAG_VERSION_SHORT
 main.c, 593
 flagList
 EmbPath, 137
 flags
 EmbStitch, 142
 Flared
 DimLeaderObject, 110
 Fletching
 DimLeaderObject, 110
 floatingChanged
 CmdPrompt, 89
 floatingChangedToolBar
 MainWindow, 182
 fname
 UiObject, 330
 focusWidget
 PropertyEditor, 267
 UndoEditor, 340
 forceRepaint
 MdiArea, 222
 SelectBox, 283
 format_100.c
 read100, 532
 write100, 532
 format_10o.c
 read10o, 533
 write10o, 533
 format_art.c
 readArt, 533

writeArt, 533
format_bmc.c
 readBmc, 534
 writeBmc, 534
format_bro.c
 readBro, 534
 writeBro, 534
format_cnd.c
 readCnd, 535
 writeCnd, 535
format_col.c
 readCol, 536
 writeCol, 536
format_csd.c
 _subMask, 537
 _xorMask, 537
 BuildDecryptionTable, 536
 csd_decryptArray, 537
 CsdSubMaskSize, 536
 CsdXorMaskSize, 536
 DecodeCsdByte, 537
 readCsd, 537
 writeCsd, 537
format_csv.c
 csvStitchFlagToStr, 538
 csvStrToStitchFlag, 538
 readCsv, 538
 writeCsv, 538
format_dat.c
 readDat, 538
 writeDat, 538
format_dem.c
 readDem, 539
 writeDem, 539
format_dsb.c
 readDsb, 540
 writeDsb, 540
format_dst.c
 cc1, 541
 decode_record_flags, 541
 encode_record, 542
 readDst, 542
 set_dst_variable, 542
 writeDst, 542
format_dsz.c
 readDsz, 542
 writeDsz, 542
format_dxf.c
 readDxf, 543
 readLine, 543
 writeDxf, 543
format_edr.c
 readEdr, 543
 writeEdr, 543
format_emd.c
 emdDecode, 544
 readEmd, 544
 writeEmd, 544
format_exp.c
 expDecode, 544
 readExp, 544
 writeExp, 545
format_exy.c
 decode_exy_flags, 545
 readExy, 545
 writeExy, 545
format_eyc.c
 readEys, 545
 writeEys, 546
format_fxy.c
 readFxy, 546
 writeFxy, 546
format_gc.c
 readGc, 546
 writeGc, 547
format_gnc.c
 readGnc, 547
 writeGnc, 547
format_gt.c
 readGt, 547
 writeGt, 548
format_hus.c
 husCompressData, 548
 husDecodeByte, 548
 husDecodeStitchType, 548
 husDecompressData, 548
 husEncodeByte, 548
 husEncodeStitchType, 548
 readHus, 549
 writeHus, 549
format_inb.c
 readInb, 549
 writeInb, 549
format_inf.c
 readInf, 549
 writeInf, 550
format_jef.c
 jefDecode, 550
 jefEncode, 550
 jefGetHoopSize, 550
 jefSetHoopFromId, 550
 read_hoop, 550
 readJef, 551
 writeJef, 551
format_ksm.c
 ksmEncode, 551
 readKsm, 551
 writeKsm, 551
format_max.c
 max_header, 552
 readMax, 552
 writeMax, 552
format_mit.c
 readMit, 552
 writeMit, 553
format_new.c

readNew, 553
writeNew, 553
format_ofm.c
 ofmDecode, 554
 ofmReadBlockHeader, 554
 ofmReadClass, 554
 ofmReadColorChange, 554
 ofmReadExpanded, 554
 ofmReadLibrary, 554
 ofmReadThreads, 554
 readOfm, 554
 writeOfm, 554
format_pcd.c
 readPcd, 555
 writePcd, 555
format_pcm.c
 readPcm, 555
 writePcm, 555
format_pcq.c
 readPcq, 556
 writePcq, 556
format_pcs.c
 readPcs, 556
 writePcs, 556
format_pec.c
 pecEncode, 557
 pecEncodeJump, 557
 pecEncodeStop, 557
 readPec, 557
 readPecStitches, 557
 writeImage, 557
 writePec, 557
 writePecStitches, 557
format_pel.c
 readPel, 558
 writePel, 558
format_pem.c
 readPem, 558
 writePem, 558
format_pes.c
 pes_version, 561
 pes_version_strings, 561
 pesWriteEmbOneSection, 559
 pesWriteSewSegSection, 559
 readDescriptions, 559
 readFeatherPatterns, 559
 readHoopName, 559
 readImageString, 560
 readMotifPatterns, 560
 readPes, 560
 readPESHeaderV10, 560
 readPESHeaderV5, 560
 readPESHeaderV6, 560
 readPESHeaderV7, 560
 readPESHeaderV8, 560
 readPESHeaderV9, 560
 readProgrammableFills, 560
 readThreads, 560
 writePes, 561
format_phb.c
 readPhb, 561
 writePhb, 561
format_phc.c
 readPhc, 562
 writePhc, 562
format_plt.c
 readPlt, 562
 writePlt, 562
format_rgb.c
 readRgb, 563
 writeRgb, 563
format_sew.c
 readSew, 563
 sewDecode, 563
 writeSew, 563
format_shv.c
 readShv, 564
 shvDecode, 564
 shvDecodeShort, 564
 writeShv, 564
format_sst.c
 readSst, 564
 writeSst, 565
format_stx.c
 readStx, 565
 stxReadThread, 565
 writeStx, 565
format_svg.c
 attributeList, 566
 current_element_id, 566
 currentAttribute, 566
 currentValue, 566
 n_attributes, 566
 readSvg, 566
 svgCreator, 566
 svgExpect, 566
 svgMultiValue, 566
 writeSvg, 566
format_t01.c
 readT01, 567
 writeT01, 567
format_t09.c
 readT09, 567
 writeT09, 567
format_tap.c
 decode_tap_record_flags, 568
 encode_tap_record, 568
 readTap, 568
 writeTap, 568
format_thr.c
 readThr, 568
 writeThr, 569
format_txt.c
 readTxt, 569
 writeTxt, 569
format_u00.c

readU00, 569
writeU00, 570
format_u01.c
readU01, 570
writeU01, 570
format_vip.c
readVip, 571
vipCompressData, 571
vipDecodeByte, 571
vipDecodeStitchType, 571
vipDecodingTable, 571
vipDecompressData, 571
vipEncodeByte, 571
vipEncodeStitchType, 571
writeVip, 571
format_vp3.c
readVp3, 572
vp3Decode, 572
vp3DecodeInt16, 572
vp3PatchByteCount, 572
vp3ReadHoopSection, 572
vp3ReadString, 573
vp3WriteString, 573
vp3WriteStringLen, 573
writeVp3, 573
format_xxx.c
readXxx, 573
writeXxx, 573
xxxDecodeByte, 573
xxxEncodeDesign, 574
xxxEncodeStitch, 574
xxxEncodeStop, 574
format_zsk.c
readZsk, 574
writeZsk, 574
formatFilterOpen
 MainWindow, 207
formatFilterSave
 MainWindow, 207
formats.c
binaryWriteInt, 529
binaryWriteIntBE, 529
binaryWriteShort, 529
binaryWriteUInt, 529
binaryWriteUIntBE, 529
binaryWriteUShort, 529
binaryWriteUShortBE, 529
emb_identify_format, 530
embFormat_getExtension, 530
embPattern_read, 530
embPattern_readAuto, 530
embPattern_write, 530
embPattern_writeAuto, 530
formatTable, 531
fpad, 531
fread_int16, 531
fread_int32_be, 531
fread_uint16, 531
imageWithFrame, 532
safe_free, 531
formatTable
 embroidery.h, 465
 formats.c, 531
formatType
 SaveObject, 282
Fortron, 501, 546
fpad
 embroidery_internal.h, 496
 formats.c, 531
fread_int16
 embroidery_internal.h, 496
 formats.c, 531
fread_int32_be
 embroidery_internal.h, 497
 formats.c, 531
fread_uint16
 embroidery_internal.h, 497
 formats.c, 531
fromCenter
 UndoableNavCommand, 336
fromTransform
 UndoableNavCommand, 336
Fufu_Polyester
 embroidery.h, 446
Fufu_Rayon
 embroidery.h, 446
full_test_matrix
 embroidery.h, 463
functions.c
 degrees, 582
 emb_round, 582
 radians, 582
fx, 16, 501, 546
g
 EmbColor_, 126
general_check_for_updates
 Settings_, 288
general_current_tip
 Settings_, 288
general_mdi_bg_color
 Settings_, 288
general_mdi_bg_logo
 Settings_, 288
general_mdi_bg_texture
 Settings_, 288
general_system_help_browser
 Settings_, 289
generate_dragon_curve
 fill.c, 526
geometry
 EmbArray_, 123
 EmbLayer_, 134
 EmbPattern_, 138
geometry.c
 embGeometry_boundingRect, 575
 embGeometry_free, 575

embGeometry_init, 575
embGeometry_move, 575
embGeometry_vulcanize, 575
get_action_index
 embroidermodder.h, 367
 mainwindow-toolbars.cpp, 406
get_trim_bounds
 main.c, 598
getAction
 MainWindow, 182
getApplication
 MainWindow, 182
getArcCenter
 arc.c, 579
 embroidery.h, 463
getArcDataFromBulge
 arc.c, 579
 embroidery.h, 463
getCircleCircleIntersections
 circle.c, 580
 embroidery.h, 464
getCircleTangentPoints
 circle.c, 580
 embroidery.h, 464
getCurrentColor
 MainWindow, 182
 MdiWindow, 229
getCurrentFile
 MdiWindow, 229
getCurrentLayer
 MainWindow, 182
 MdiWindow, 229
getCurrentLineType
 MainWindow, 183
 MdiWindow, 230
getCurrentLineWidth
 MainWindow, 183
 MdiWindow, 230
getCurrentText
 CmdPrompt, 89
GetFile
 embroidery_internal.h, 497
 main.c, 598
getFileSeparator
 MainWindow, 183
getHistory
 CmdPrompt, 89
 getInfo
 EmbDetailsDialog, 127
getMdiArea
 MainWindow, 183
getPrefix
 CmdPrompt, 90
getScene
 MdiWindow, 230
getShortCurrentFile
 MdiWindow, 230
getUndoStack

View, 347
getView
 MdiWindow, 230
gnc, 16, 501, 547
Gold Thread, 501, 548
Great Notions, 501, 547
greedy_algorithm
 fill.c, 526
GREEN_TERM_COLOR
 embroidery_internal.h, 486
grid_center
 Settings_, 289
grid_center_on_origin
 Settings_, 289
grid_color
 Settings_, 289
grid_color_match_crosshair
 Settings_, 289
grid_load_from_file
 Settings_, 289
grid_mode
 EmbView_, 147
grid_show_on_load
 Settings_, 289
grid_show_origin
 Settings_, 289
grid_size
 Settings_, 289
grid_size_radius
 Settings_, 289
grid_spacing
 Settings_, 289
grid_spacing_angle
 Settings_, 289
grid_spacing_radius
 Settings_, 289
grid_type
 EmbView_, 147
 Settings_, 289
gridColor
 View, 352
gridPath
 View, 352
gripBaseObj
 View, 352
gripColorCool
 View, 352
gripColorHot
 View, 352
gripEdit
 ArcObject, 66
 BaseObject, 75
 CircleObject, 82
 DimLeaderObject, 111
 EllipseObject, 118
 ImageObject, 155
 LineObject, 164
 PathObject, 239

PointObject, 244
PolygonObject, 248
PolylineObject, 254
RectObject, 273
TextSingleObject, 324
gripIndex
 PolygonObject, 250
 PolylineObject, 256
grippingActive
 View, 352
gripSize
 View, 352
groupBoxes
 property-editor.cpp, 417
groupBoxGeometryCircle
 property-editor.cpp, 417
groupBoxGeometryDimAligned
 property-editor.cpp, 417
groupBoxGeometryDimAngular
 property-editor.cpp, 417
groupBoxGeometryDimArcLength
 property-editor.cpp, 417
groupBoxGeometryDimDiameter
 property-editor.cpp, 417
groupBoxGeometryDimLeader
 property-editor.cpp, 417
groupBoxGeometryDimLinear
 property-editor.cpp, 417
groupBoxGeometryDimOrdinate
 property-editor.cpp, 417
groupBoxGeometryDimRadius
 property-editor.cpp, 417
groupBoxGeometryEllipse
 property-editor.cpp, 418
groupBoxGeometryImage
 property-editor.cpp, 418
groupBoxGeometryInfiniteLine
 property-editor.cpp, 418
groupBoxGeometryLine
 property-editor.cpp, 418
groupBoxGeometryPath
 property-editor.cpp, 418
groupBoxGeometryPoint
 property-editor.cpp, 418
groupBoxGeometryPolygon
 property-editor.cpp, 418
groupBoxGeometryPolyline
 property-editor.cpp, 418
groupBoxGeometryRay
 property-editor.cpp, 418
groupBoxGeometryRectangle
 property-editor.cpp, 418
groupBoxGeometryTextMulti
 property-editor.cpp, 418
groupBoxGeometryTextSingle
 property-editor.cpp, 418
groupBoxMisclImage
 property-editor.cpp, 418
groupBoxMiscPath
 property-editor.cpp, 418
groupBoxMiscPolyline
 property-editor.cpp, 418
groupBoxMiscTextSingle
 property-editor.cpp, 418
groupBoxTextTextSingle
 property-editor.cpp, 418
gscene
 MdiWindow, 234
 SaveObject, 282
 View, 352
gt, 16, 501, 548
gview
 MdiWindow, 234
 UndoableAddCommand, 332
 UndoableDeleteCommand, 332
 UndoableGripEditCommand, 333
 UndoableMirrorCommand, 334
 UndoableMoveCommand, 335
 UndoableNavCommand, 337
 UndoableRotateCommand, 338
 UndoableScaleCommand, 339
handleMoved
 CmdPromptHandle, 94
handlePressed
 CmdPromptHandle, 94
handleReleased
 CmdPromptHandle, 95
Happy, 568
hash
 Action____, 58
hashDeletedObjects
 View, 352
haveExtraDIFATSectors
 main.c, 598
header
 _bcf_file, 51
HEART_MODE_NUM_POINTS
 embroidermodder.h, 366
HEART_MODE_STYLE
 embroidermodder.h, 366
HEART_MODE_XSCALE
 embroidermodder.h, 366
HEART_MODE_YSCALE
 embroidermodder.h, 366
height
 _vp3Hoop, 56
 EmblImage____, 133
help
 MainWindow, 183
help_toolbar
 mainwindow-toolbars.cpp, 406
helpMenu
 MainWindow, 208
Hemingworth_Polyester
 embroidery.h, 446
hex_code

thread_color_, 327
 hideAllGroups
 PropertyEditor, 265
 hideUnimplemented
 MainWindow, 183
 hilbert_curve
 embroidery.h, 464
 fill.c, 526
 hilbert_curve_l_system
 fill.c, 527
 historyAppended
 CmdPrompt, 90
 CmdPromptHistory, 97
 home
 EmbPattern_, 138
 HOOP_110X110
 embroidery_internal.h, 486
 HOOP_126X110
 embroidery_internal.h, 486
 HOOP_140X200
 embroidery_internal.h, 487
 HOOP_230X200
 embroidery_internal.h, 487
 HOOP_50X50
 embroidery_internal.h, 487
 hoop_height
 EmbPattern_, 138
 hoop_padding, 150
 bottom, 150
 left, 150
 right, 150
 top, 151
 hoop_width
 EmbPattern_, 138
 hoopSize
 ThredHeader_, 328
 hoopX
 ThredExtension_, 328
 hoopY
 ThredExtension_, 328
 hour
 EmbTime_, 145
 Huffman, 151
 default_value, 151
 lengths, 151
 nlengths, 151
 ntable, 151
 table, 151
 table_width, 151
 huffman
 embroidery_internal.h, 489
 huffman_build_table
 compress.c, 432
 embroidery_internal.h, 497
 huffman_lookup
 compress.c, 432
 huffman_lookup_data
 compress.c, 433
 huffman_table_lookup
 embroidery_internal.h, 497
 hus, 16, 548
 hus_compress
 compress.c, 432
 embroidery_internal.h, 497
 hus_decompress
 compress.c, 433
 embroidery_internal.h, 497
 hus_thread
 embroidery.h, 446
 husCompressData
 format_hus.c, 548
 husDecodeByte
 format_hus.c, 548
 husDecodeStitchType
 format_hus.c, 548
 husDecompressData
 format_hus.c, 548
 husEncodeByte
 format_hus.c, 548
 husEncodeStitchType
 format_hus.c, 548
 Husqvarna Viking, 548, 564
 husThreads
 embroidery.h, 465
 thread-color.c, 606
 i_value
 Parameter_, 235
 icon
 Action_, 58
 icon_size
 Settings_, 289
 icon_theme
 Settings_, 289
 icon_toolbar
 mainwindow-toolbars.cpp, 406
 iconDir
 PropertyEditor, 267
 UndoEditor, 341
 iconResize
 MainWindow, 183
 iconSize
 PropertyEditor, 267
 UndoEditor, 341
 id
 UiObject_, 330
 UndoableNavCommand, 336
 image.c
 image_diff, 588
 writeImage, 589
 image_diff
 image.c, 588
 ImageObject, 152
 ~ImageObject, 154
 allGripPoints, 155
 gripEdit, 155
 ImageObject, 154

init, 155
mouseSnapPoint, 155
objectArea, 155
objectBottomLeft, 155
objectBottomRight, 155
objectHeight, 155
objectTopLeft, 156
objectTopRight, 156
objectWidth, 156
paint, 156
setObjectRect, 156
Type, 154
type, 156
updatePath, 156
updateRubber, 156
vulcanize, 156
ImageWidget, 157
~ImageWidget, 158
ImageWidget, 157
img, 159
load, 158
paintEvent, 158
save, 158
imageWithFrame
embroidery_internal.h, 513
formats.c, 532
img
ImageWidget, 159
imgWidget
PreviewDialog, 257
inb, 16, 502, 549
Inbro, 502, 549
Index
embroidermodder.h, 361
inf, 502, 550
init
ArcObject, 66
CircleObject, 82
DimLeaderObject, 111
EllipseObject, 118
ImageObject, 155
LineObject, 164
PathObject, 239
PointObject, 244
PolygonObject, 248
PolylineObject, 254
RectObject, 274
TextSingleObject, 324
input_data
Compress, 107
input_length
Compress, 107
Isacord_Polyester
embroidery.h, 446
Isafil_Rayon
embroidery.h, 446
isBlinking
CmdPromptInput, 104
isCommandActive
CmdPrompt, 90
MainWindow, 183
isLwtEnabled
View, 347
isRapidFireEnabled
CmdPrompt, 90
isRealEnabled
View, 347
isShiftPressed
MainWindow, 184
Janome, 550, 563
jef, 16, 550
jef_thread
embroidery.h, 446
jefDecode
format_jef.c, 550
jefEncode
format_jef.c, 550
jefGetHoopSize
format_jef.c, 550
jefSetHoopFromId
format_jef.c, 550
jefThreads
embroidery.h, 465
thread-color.c, 606
join_short_stitches
fill.c, 527
JUMP
embroidery.h, 446
ksm, 16, 551
ksmEncode
format_ksm.c, 551
L_system
embroidery.h, 450
labelTipOfDay
MainWindow, 208
language
Settings_, 289
lastCmd
CmdPromptInput, 104
lastCommand
CmdPrompt, 90
layer
EmbPattern_, 138
LayerManager, 159
~LayerManager, 160
addLayer, 160
LayerManager, 159
layerModel, 160
layerModelSorted, 160
treeView, 161
layerManager
MainWindow, 184
layerModel
LayerManager, 160

layerModelSorted
 LayerManager, 160
 layerPrevious
 MainWindow, 184
 layerSelector
 MainWindow, 208
 layerSelectorIndexChanged
 MainWindow, 184
 layoutState
 MainWindow, 208
 left
 _vp3Hoop, 56
 EmbRect_, 140
 hoop_padding, 150
 left2
 _vp3Hoop, 56
 leftBrush
 SelectBox, 284
 leftBrushColor
 SelectBox, 284
 leftPen
 SelectBox, 284
 leftPenColor
 SelectBox, 284
 leftSiblingId
 _bcf_directory_entry, 50
 length
 EmbArray_, 123
 EmbSatinOutline_, 141
 ThredHeader_, 328
 lengths
 Huffman, 151
 LIBEMBROIDERY_EMBEDDED_VERSION
 embroidery.h, 446
 lindenmayer_system
 embroidery.h, 464
 fill.c, 527
 line
 BaseObject, 75
 EmbGeometry_, 131
 line.c
 embLine_intersectionPoint, 583
 embLine_normalVector, 583
 embLine_toVector, 583
 lineEditArcArea
 property-editor.cpp, 418
 lineEditArcCenterY
 property-editor.cpp, 419
 lineEditArcChord
 property-editor.cpp, 419
 lineEditArcEndAngle
 property-editor.cpp, 419
 lineEditArcEndX
 property-editor.cpp, 419
 lineEditArcEndY
 property-editor.cpp, 419
 lineEditArcInclAngle
 property-editor.cpp, 419
 lineEditArcLength
 property-editor.cpp, 419
 lineEditArcRadius
 property-editor.cpp, 419
 lineEditArcStartAngle
 property-editor.cpp, 419
 lineEditArcStartX
 property-editor.cpp, 419
 lineEditArcStartY
 property-editor.cpp, 419
 lineEditBlockX
 property-editor.cpp, 419
 lineEditBlockY
 property-editor.cpp, 419
 lineEditCircleArea
 property-editor.cpp, 419
 lineEditCircleCenterX
 property-editor.cpp, 419
 lineEditCircleCenterY
 property-editor.cpp, 419
 lineEditCircleCircumference
 property-editor.cpp, 419
 lineEditCircleDiameter
 property-editor.cpp, 419
 lineEditCircleRadius
 property-editor.cpp, 420
 lineEditEllipseCenterX
 property-editor.cpp, 420
 lineEditEllipseCenterY
 property-editor.cpp, 420
 lineEditEllipseDiameterMajor
 property-editor.cpp, 420
 lineEditEllipseDiameterMinor
 property-editor.cpp, 420
 lineEditEllipseRadiusMajor
 property-editor.cpp, 420
 lineEditEllipseRadiusMinor
 property-editor.cpp, 420
 lineEditImageHeight
 property-editor.cpp, 420
 lineEditImageName
 property-editor.cpp, 420
 lineEditImagePath
 property-editor.cpp, 420
 lineEditImageWidth
 property-editor.cpp, 420
 lineEditImageX
 property-editor.cpp, 420
 lineEditImageY
 property-editor.cpp, 420
 lineEditInfiniteLineVectorX
 property-editor.cpp, 420
 lineEditInfiniteLineVectorY
 property-editor.cpp, 420
 lineEditInfiniteLineX2
 property-editor.cpp, 420
 lineEditInfiniteLineY1
 property-editor.cpp, 420

lineEditInfiniteLineY2
 property-editor.cpp, 420
lineEditLineAngle
 property-editor.cpp, 421
lineEditLineDeltaX
 property-editor.cpp, 421
lineEditLineDeltaY
 property-editor.cpp, 421
lineEditLineEndX
 property-editor.cpp, 421
lineEditLineEndY
 property-editor.cpp, 421
lineEditLineLength
 property-editor.cpp, 421
lineEditLineStartX
 property-editor.cpp, 421
lineEditLineStartY
 property-editor.cpp, 421
lineEditPathArea
 property-editor.cpp, 421
lineEditPathLength
 property-editor.cpp, 421
lineEditPathVertexX
 property-editor.cpp, 421
lineEditPathVertexY
 property-editor.cpp, 421
lineEditPointX
 property-editor.cpp, 421
lineEditPointY
 property-editor.cpp, 421
lineEditPolygonCenterX
 property-editor.cpp, 421
lineEditPolygonCenterY
 property-editor.cpp, 421
lineEditPolygonDiameterSide
 property-editor.cpp, 421
lineEditPolygonDiameterVertex
 property-editor.cpp, 421
lineEditPolygonInteriorAngle
 property-editor.cpp, 422
lineEditPolygonRadiusSide
 property-editor.cpp, 422
lineEditPolygonRadiusVertex
 property-editor.cpp, 422
lineEditPolylineArea
 property-editor.cpp, 422
lineEditPolylineLength
 property-editor.cpp, 422
lineEditPolylineVertexX
 property-editor.cpp, 422
lineEditPolylineVertexY
 property-editor.cpp, 422
lineEditRayVectorX
 property-editor.cpp, 422
lineEditRayVectorY
 property-editor.cpp, 422
lineEditRayX1
 property-editor.cpp, 422
lineEditRayX2
 property-editor.cpp, 422
lineEditRayY1
 property-editor.cpp, 422
lineEditRayY2
 property-editor.cpp, 422
lineEditRectangleArea
 property-editor.cpp, 422
lineEditRectangleCorner1X
 property-editor.cpp, 422
lineEditRectangleCorner1Y
 property-editor.cpp, 422
lineEditRectangleCorner2X
 property-editor.cpp, 422
lineEditRectangleCorner2Y
 property-editor.cpp, 422
lineEditRectangleCorner3X
 property-editor.cpp, 423
lineEditRectangleCorner3Y
 property-editor.cpp, 423
lineEditRectangleCorner4X
 property-editor.cpp, 423
lineEditRectangleCorner4Y
 property-editor.cpp, 423
lineEditRectangleHeight
 property-editor.cpp, 423
lineEditRectangleWidth
 property-editor.cpp, 423
lineEdits
 property-editor.cpp, 423
lineEditTextMultiX
 property-editor.cpp, 423
lineEditTextMultiY
 property-editor.cpp, 423
lineEditTextSingleContents
 property-editor.cpp, 423
lineEditTextSingleHeight
 property-editor.cpp, 423
lineEditTextSingleRotation
 property-editor.cpp, 423
lineEditTextSingleX
 property-editor.cpp, 423
lineEditTextSingleY
 property-editor.cpp, 423
LineObject, 161
 ~LineObject, 163
 allGripPoints, 164
 gripEdit, 164
 init, 164
 LineObject, 163
 mouseSnapPoint, 164
 objectAngle, 164
 objectDeltaX, 164
 objectDeltaY, 164
 objectEndPoint1, 164
 objectEndPoint2, 165
 objectLength, 165
 objectMidPoint, 165

objectSavePath, 165
 objectX1, 165
 objectX2, 165
 objectY1, 165
 objectY2, 165
 paint, 165
 setObjectEndPoint1, 165
 setObjectEndPoint2, 166
 setObjectX1, 166
 setObjectX2, 166
 setObjectY1, 166
 setObjectY2, 166
 Type, 163
 type, 166
 updateRubber, 166
 vulcanize, 166
 lineStyle
 DimLeaderObject, 110
 lineStyleAngle
 DimLeaderObject, 114
 lineStyleLength
 DimLeaderObject, 114
 lineStylePath
 DimLeaderObject, 115
 LINETO
 embroidery_internal.h, 487
 lineType
 EmbGeometry_, 131
 EmbLine_, 135
 EmbPath_, 137
 EmbPoint_, 139
 linetypeSelector
 MainWindow, 208
 linetypeSelectorIndexChanged
 MainWindow, 184
 lineWeightPen
 BaseObject, 75
 linewidthSelector
 MainWindow, 208
 lineweightSelectorIndexChanged
 MainWindow, 184
 listMdiWin
 MainWindow, 208
 listTipOfDay
 MainWindow, 208
 load
 ImageWidget, 158
 LoadCommand
 MainWindow, 184
 loadFatFromSector
 embroidery_internal.h, 498
 main.c, 598
 loadFile
 MdiWindow, 230
 loadFormats
 MainWindow, 184
 loadRulerSettings
 View, 347
 logPromptInput
 MainWindow, 184
 MdiWindow, 230
 LSYSTEM, 167
 alphabet, 167
 axiom, 167
 constants, 167
 rules, 167
 lwt_default_lwt
 Settings_, 289
 lwt_mode
 EmbView_, 148
 lwt_real_render
 Settings_, 290
 lwt_show_lwt
 Settings_, 290
 lwtPen
 BaseObject, 79
 Madeira_Polyester
 embroidery.h, 446
 Madeira_Rayon
 embroidery.h, 446
 magicCode
 VipHeader_, 355
 main
 embroidermodder.cpp, 357
 main.c
 bcf_difat_create, 594
 bcf_directory_free, 594
 bcf_file_free, 594
 bcfFile_read, 594
 bcfFileFat_create, 594
 bcfFileHeader_read, 594
 binaryReadString, 594
 binaryReadUnicodeString, 595
 black_thread, 600
 check_header_present, 595
 CompoundFileDirectory, 595
 CompoundFileDirectoryEntry, 595
 copy_trim, 595
 difatEntriesInHeader, 600
 emb_error, 600
 emb_optOut, 595
 emb_readline, 596
 emb_verbose, 600
 embArc_print, 596
 embColor_distance, 596
 embColor_read, 596
 embColor_write, 596
 embConstantPi, 600
 embSatinOutline_generateSatinOutline, 596
 embSatinOutline_renderStitches, 596
 embThread_findNearestColor, 597
 embThread_findNearestThread, 597
 embThread_getRandom, 597
 embTime_initNow, 597
 embTime_time, 597
 embVector_print, 597

entriesInDifatSector, 598
FLAG_CIRCLE, 591
FLAG_CIRCLE_SHORT, 592
FLAG_COMBINE, 592
FLAG_CROSS_STITCH, 592
FLAG_ELLIPSE, 592
FLAG_ELLIPSE_SHORT, 592
FLAG_FILL, 592
FLAG_FILL_SHORT, 592
FLAG_FORMATS, 592
FLAG_FORMATS_SHORT, 592
FLAG_FULL_TEST_SUITE, 592
FLAG_HELP, 592
FLAG_HELP_SHORT, 592
FLAG_HILBERT_CURVE, 592
FLAG_LINE, 592
FLAG_LINE_SHORT, 592
FLAG_POLYGON, 592
FLAG_POLYGON_SHORT, 592
FLAG_POLYLINE, 592
FLAG_POLYLINE_SHORT, 593
FLAG QUIET, 593
FLAG QUIET_SHORT, 593
FLAG_RENDER, 593
FLAG_RENDER_SHORT, 593
FLAG_SATIN, 593
FLAG_SATIN_SHORT, 593
FLAG_SIERPINSKI_TRIANGLE, 593
FLAG_SIMULATE, 593
FLAG_STITCH, 593
FLAG_STITCH_SHORT, 593
FLAG_TEST, 593
FLAG_TO, 593
FLAG_TO_SHORT, 593
FLAG_VERBOSE, 593
FLAG_VERBOSE_SHORT, 593
FLAG_VERSION, 593
FLAG_VERSION_SHORT, 593
get_trim_bounds, 598
GetFile, 598
haveExtraDIFATSectors, 598
loadFatFromSector, 598
NUM_FLAGS, 594
parseDIFATSectors, 598
parseDirectoryEntryName, 598
parseTime, 598
readFullSector, 599
readNextSector, 599
sectorSize, 599
seekToSector, 599
sizeOfChainingEntryAtEndOfDifatSector, 600
sizeOfDifatEntry, 600
sizeOfDirectoryEntry, 600
sizeOfFatEntry, 600
stringInArray, 599
WHITE SPACE, 600
write_24bit, 599
mainWidget
EmbDetailsDialog, 127
mainWin
embroidermodder.h, 367
MainWindow, 208
mainwindow.cpp, 409
MdiArea, 225
MdiWindow, 235
Settings_Dialog, 314
StatusBarButton, 319
View, 352
MainWindow, 167
~MainWindow, 177
about, 177
actionHash, 207
activeCommand, 177
activeMdiWindow, 177
activeScene, 177
activeUndoStack, 177
activeView, 177
actuator, 177
buttonTipOfTheDayClicked, 178
changelog, 178
checkBoxTipOfTheDay, 207
checkBoxTipOfTheDayStateChanged, 178
checkForUpdates, 178
closeEvent, 178
closeToolBar, 178
colorSelector, 207
colorSelectorIndexChanged, 178
copy, 179
create_icon, 179
create_toolbar, 179
createAllActions, 179
createAllMenus, 179
createAllToolbars, 179
createEditMenu, 179
createFileMenu, 179
createHelpMenu, 180
createHelpToolbar, 180
createIconToolbar, 180
createLayerToolbar, 180
createPanToolbar, 180
createPromptToolbar, 180
createPropertiesToolbar, 180
createSettingsMenu, 180
createTextToolbar, 180
createViewMenu, 180
createWindowMenu, 181
cut, 181
cutCopyObjectList, 207
dayVision, 181
deletePressed, 181
designDetails, 181
disableMoveRapidFire, 181
disablePromptRapidFire, 181
docIndex, 207
dockPropEdit, 207
dockUndoEdit, 207

doNothing, 181
editMenu, 207
enableMoveRapidFire, 181
enablePromptRapidFire, 181
escapePressed, 181
exit, 181
fileMenu, 207
findMdiWindow, 181
floatingChangedToolBar, 182
formatFilterOpen, 207
formatFilterSave, 207
getAction, 182
getApplication, 182
getCurrentColor, 182
getCurrentLayer, 182
getCurrentLineType, 183
getCurrentLineWeight, 183
getFileSeparator, 183
getMdiArea, 183
help, 183
helpMenu, 208
hideUnimplemented, 183
iconResize, 183
isCommandActive, 183
isShiftPressed, 184
labelTipOfDay, 208
layerManager, 184
layerPrevious, 184
layerSelector, 208
layerSelectorIndexChanged, 184
layoutState, 208
linetypeSelector, 208
linetypeSelectorIndexChanged, 184
lineweightSelector, 208
lineweightSelectorIndexChanged, 184
listMdiWin, 208
listTipOfDay, 208
LoadCommand, 184
loadFormats, 184
logPromptInput, 184
mainWin, 208
MainWindow, 177
makeLayerActive, 184
mdiArea, 208
menuHash, 208
myFileSeparator, 209
nativeAddArc, 185
nativeAddCircle, 185
nativeAddDimLeader, 185
nativeAddEllipse, 185
nativeAddHorizontalDimension, 185
nativeAddImage, 185
nativeAddInfiniteLine, 186
nativeAddLine, 186
nativeAddPath, 186
nativeAddPoint, 186
nativeAddPolygon, 186
nativeAddPolyline, 186
nativeAddRay, 187
nativeAddRectangle, 187
nativeAddRegularPolygon, 187
nativeAddRoundedRectangle, 187
nativeAddSlot, 187
nativeAddTextMulti, 188
nativeAddTextSingle, 188
nativeAddToSelection, 188
nativeAddTriangle, 188
nativeAddVerticalDimension, 188
nativeAlert, 189
nativeAllowRubber, 189
nativeAppendPromptHistory, 189
nativeBlinkPrompt, 189
nativeCalculateAngle, 189
nativeCalculateDistance, 189
nativeClearRubber, 189
nativeClearSelection, 189
nativeCopySelected, 189
nativeCutSelected, 190
nativeDeleteSelected, 190
nativeDisableMoveRapidFire, 190
nativeDisablePromptRapidFire, 190
nativeEnableMoveRapidFire, 190
nativeEnablePromptRapidFire, 190
nativeEndCommand, 190
nativeExit, 190
nativeInitCommand, 190
nativeMessageBox, 191
nativeMirrorSelected, 191
nativeMouseX, 191
nativeMouseY, 191
nativeMoveSelected, 191
nativeNewFile, 191
nativeNumSelected, 192
nativeOpenFile, 192
nativePasteSelected, 192
nativePerpendicularDistance, 192
nativePreviewOff, 192
nativePreviewOn, 192
nativePrintArea, 192
nativeQSnapX, 192
nativeQSnapY, 193
nativeRotateSelected, 193
nativeScaleSelected, 193
nativeSelectAll, 193
nativeSetBackgroundColor, 193
nativeSetCrossHairColor, 193
nativeSetCursorShape, 194
nativeSetGridColor, 194
nativeSetPromptPrefix, 194
nativeSetRubberMode, 194
nativeSetRubberPoint, 194
nativeSetRubberText, 194
nativeSpareRubber, 194
nativeTextAngle, 194
nativeTextBold, 194
nativeTextFont, 195

nativeTextItalic, 195
nativeTextOverline, 195
nativeTextSize, 195
nativeTextStrikeOut, 195
nativeTextUnderline, 195
nativeTipOfTheDay, 195
nativeVulcanize, 195
nativeWindowCascade, 195
nativeWindowClose, 195
nativeWindowCloseAll, 195
nativeWindowNext, 196
nativeWindowPrevious, 196
nativeWindowTile, 196
newFile, 196
nightVision, 196
numOfDocs, 209
onCloseMdiWin, 196
onCloseWindow, 196
onWindowActivated, 196
openFile, 197
openFilePath, 209
openFilesSelected, 197
openrecentfile, 197
panDown, 197
panLeft, 197
panMenu, 209
panpoint, 197
panrealtime, 198
panRight, 198
panUp, 198
paste, 198
pickAddModeToggled, 198
platformString, 198
print, 198
prompt, 209
promptHistoryAppended, 198
promptInputNext, 198
promptInputPrevious, 198
quit, 198
readSettings, 199
recentMenu, 209
recentMenuAboutToShow, 199
redo, 199
resizeEvent, 199
run_script, 199
run_script_file, 200
runCommand, 200
runCommandClick, 200
runCommandContext, 200
runCommandMain, 200
runCommandMove, 200
runCommandPrompt, 200
saveasfile, 201
savefile, 201
selectAll, 201
setShiftPressed, 201
setShiftReleased, 201
setTextAngle, 201
setTextBold, 201
setTextFont, 201
setTextItalic, 201
setTextOverline, 201
setTextSize, 202
setTextStrikeOut, 202
setTextUnderline, 202
settings_display_bg_color, 209
settings_display_crosshair_color, 209
settings_display_crosshair_percent, 209
settings_display_renderhint_aa, 209
settings_display_renderhint_high_aa, 209
settings_display_renderhint_noncosmetic, 210
settings_display_renderhint_smooth_pix, 210
settings_display_renderhint_text_aa, 210
settings_display_scrollbar_widget_num, 210
settings_display_selectbox_alpha, 210
settings_display_selectbox_left_color, 210
settings_display_selectbox_left_fill, 210
settings_display_selectbox_right_color, 210
settings_display_selectbox_right_fill, 210
settings_display_show_scrollbars, 210
settings_display_units, 210
settings_display_use_opengl, 211
settings_display_zoomscale_in, 211
settings_display_zoomscale_out, 211
settings_general_check_for_updates, 211
settings_general_current_tip, 211
settings_general_icon_size, 211
settings_general_icon_theme, 211
settings_general_language, 211
settings_general_mdi_bg_color, 211
settings_general_mdi_bg_logo, 211
settings_general_mdi_bg_texture, 211
settings_general_mdi_bg_use_color, 212
settings_general_mdi_bg_use_logo, 212
settings_general_mdi_bg_use_texture, 212
settings_general_system_help_browser, 212
settings_general_tip_of_the_day, 212
settings_grid_center_on_origin, 212
settings_grid_center_x, 212
settings_grid_center_y, 212
settings_grid_color, 212
settings_grid_color_match_crosshair, 212
settings_grid_load_from_file, 212
settings_grid_show_on_load, 213
settings_grid_show_origin, 213
settings_grid_size_radius, 213
settings_grid_size_x, 213
settings_grid_size_y, 213
settings_grid_spacing_angle, 213
settings_grid_spacing_radius, 213
settings_grid_spacing_x, 213
settings_grid_spacing_y, 213
settings_grid_type, 213
settings_lwt_default_lwt, 213
settings_lwt_real_render, 214
settings_lwt_show_lwt, 214

settings_opensave_custom_filter, 214
settings_opensave_open_format, 214
settings_opensave_open_thumbnail, 214
settings_opensave_recent_directory, 214
settings_opensave_recent_list_of_files, 214
settings_opensave_recent_max_files, 214
settings_opensave_save_format, 214
settings_opensave_save_thumbnail, 214
settings_opensave_trim_dst_num_jumps, 214
settings_printing_default_device, 215
settings_printing_disable_bg, 215
settings_printing_use_last_device, 215
settings_prompt_bg_color, 215
settings_prompt_font_family, 215
settings_prompt_font_size, 215
settings_prompt_font_style, 215
settings_prompt_save_history, 215
settings_prompt_save_history_as_html, 215
settings_prompt_save_history_filename, 215
settings_prompt_text_color, 215
settings_qsnap_aperture_size, 216
settings_qsnap_apparent, 216
settings_qsnap_center, 216
settings_qsnap_enabled, 216
settings_qsnap_endpoint, 216
settings_qsnap_extension, 216
settings_qsnap_insertion, 216
settings_qsnap_intersection, 216
settings_qsnap_locator_color, 216
settings_qsnap_locator_size, 216
settings_qsnap_midpoint, 216
settings_qsnap_nearest, 217
settings_qsnap_node, 217
settings_qsnap_parallel, 217
settings_qsnap_perpendicular, 217
settings_qsnap_quadrant, 217
settings_qsnap_tangent, 217
settings_ruler_color, 217
settings_ruler_metric, 217
settings_ruler_pixel_size, 217
settings_ruler_show_on_load, 217
settings_selection_coolgrip_color, 217
settings_selection_grip_size, 218
settings_selection_hotgrip_color, 218
settings_selection_mode_pickadd, 218
settings_selection_mode_pickdrag, 218
settings_selection_mode_pickfirst, 218
settings_selection_pickbox_size, 218
settings_text_angle, 218
settings_text_font, 218
settings_text_size, 218
settings_text_style_bold, 218
settings_text_style_italic, 218
settings_text_style_overline, 219
settings_text_style_strikeout, 219
settings_text_style_underline, 219
settingsDialog, 202
settingsMenu, 219
settingsPrompt, 202
setUndoCleanIcon, 202
shiftKeyPressedState, 219
statusbar, 219
stub_implement, 202
stub_testing, 202
textAngle, 202
textBold, 203
textFont, 203
textFontSelector, 219
textFontSelectorCurrentFontChanged, 203
textItalic, 203
textOverline, 203
textSize, 203
textSizeSelector, 219
textSizeSelectorIndexChanged, 203
textStrikeOut, 203
textUnderline, 203
tipOfDay, 203
toggleGrid, 203
toggleLwt, 204
toggleRuler, 204
toolbarEdit, 219
toolbarFile, 219
toolbarHash, 219
toolbarHelp, 220
toolbarIcon, 220
toolbarLayer, 220
toolbarPan, 220
toolbarPrompt, 220
toolbarProperties, 220
toolbarText, 220
toolbarView, 220
toolbarZoom, 220
undo, 204
updateAllViewBackgroundColors, 204
updateAllViewCrossHairColors, 204
updateAllViewGridColors, 204
updateAllViewRulerColors, 204
updateAllViewScrollBars, 204
updateAllViewSelectBoxColors, 204
updateMenuToolbarStatusbar, 204
updatePickAddMode, 205
validFormat, 205
viewMenu, 220
whatsThisContextHelp, 205
windowMenu, 220
windowMenuAboutToShow, 205
windowMenuActivated, 205
wizardTipOfDay, 221
writeSettings, 205
zoomAll, 206
zoomCenter, 206
zoomDynamic, 206
zoomExtents, 206
zoomIn, 206
zoomMenu, 221
zoomOut, 206

zoomPrevious, 206
zoomRealtime, 206
zoomScale, 206
zoomSelected, 206
zoomWindow, 206
mainwindow-settings.cpp
 read_configuration, 405
 SettingsDir, 405
 SettingsPath, 405
 to_string_vector, 405
 write_setting, 405
mainwindow-toolbars.cpp
 edit_toolbar, 406
 file_toolbar, 406
 get_action_index, 406
 help_toolbar, 406
 icon_toolbar, 406
 pan_toolbar, 407
 view_toolbar, 407
 zoom_toolbar, 407
mainwindow.cpp
 _mainWin, 409
 action_labels, 409
 action_table, 409
 CIRCLE_MODE_1P_DIA_, 409
 CIRCLE_MODE_1P_RAD_, 409
 CIRCLE_MODE_2P_, 409
 CIRCLE_MODE_3P_, 410
 CIRCLE_MODE_TTR_, 410
 convert_args_to_type, 408
 dialog, 410
 DOLPHIN_MODE_NUM_POINTS_, 410
 DOLPHIN_MODE_XSCALE_, 410
 DOLPHIN_MODE_YSCALE_, 410
 Error, 409
 mainWin, 409
 Parameter, 408
 preview, 410
 read_settings, 409
 read_string_setting, 409
 settings, 410
 SINGLE_LINE_TEXT_MODE_JUSTIFY_, 410
 SINGLE_LINE_TEXT_MODE_RAPID_, 410
 SINGLE_LINE_TEXT_MODE_SETFONT_, 410
 SINGLE_LINE_TEXT_MODE_SETGEOM_, 410
 STAR_MODE_CENTER_PT_, 410
 STAR_MODE_NUM_POINTS_, 410
 STAR_MODE_RAD_INNER_, 410
 STAR_MODE_RAD_OUTER_, 410
 Todo, 409
 validRGB, 409
major_tick_seperation
 Settings_, 290
majorVersion
 _bcf_file_header, 54
makeLayerActive
 MainWindow, 184
manufacturer_code
 thread_color_, 327
mapSignal
 PropertyEditor, 265
Marathon_Polyester
 embroidery.h, 446
Marathon_Rayon
 embroidery.h, 446
max, 552
max_header
 format_max.c, 552
MAX_STITCHES
 embroidery.h, 446
MAX_THREADS
 embroidery.h, 446
maxNumberOfDirectoryEntries
 _bcf_directory, 49
maxPoints
 UiObject_, 330
mdi_bg_use_color
 Settings_, 290
mdi_bg_use_logo
 Settings_, 290
mdi_bg_use_texture
 Settings_, 290
MdiArea, 221
 ~MdiArea, 222
 bgColor, 225
 bgLogo, 225
 bgTexture, 225
 cascade, 222
 forceRepaint, 222
 mainWin, 225
 MdiArea, 222
 mouseDoubleClickEvent, 223
 paintEvent, 223
 setBackgroundColor, 223
 setBackgroundLogo, 223
 setBackgroundTexture, 223
 tile, 224
 useBackgroundColor, 224
 useBackgroundLogo, 224
 useBackgroundTexture, 224
 useColor, 225
 useLogo, 225
 useTexture, 225
 zoomExtentsAllSubWindows, 224
mdiArea
 MainWindow, 208
 MdiWindow, 235
MdiWindow, 225
 ~MdiWindow, 227
 closeEvent, 228
 curColor, 234
 curFile, 234
 curLayer, 234
 curlineType, 234
 curlineWeight, 234
 currentColorChanged, 228

currentLayerChanged, 228
 currentLinetypeChanged, 228
 currentLineweightChanged, 228
 deletePressed, 229
 designDetails, 229
 escapePressed, 229
 fileExtension, 229
 fileWasLoaded, 234
 getCurrentColor, 229
 getCurrentFile, 229
 getCurrentLayer, 229
 getCurrentLineType, 230
 getCurrentLineWeight, 230
 getScene, 230
 getShortCurrentFile, 230
 getView, 230
 gscene, 234
 gview, 234
 loadFile, 230
 logPromptInput, 230
 mainWin, 235
 mdiArea, 235
 MdiWindow, 227
 myIndex, 235
 onWindowActivated, 231
 print, 231
 printer, 235
 promptHistory, 235
 promptHistoryAppended, 231
 promptInputList, 235
 promptInputNext, 231
 promptInputNum, 235
 promptInputPrevious, 231
 promptInputPrevNext, 231
 saveBMC, 231
 saveFile, 232
 sendCloseMdiWin, 232
 setCurrentColor, 232
 setCurrentFile, 232
 setCurrentLayer, 232
 setCurrentLineType, 233
 setCurrentLineWeight, 233
 setViewBackgroundColor, 233
 setViewCrossHairColor, 233
 setViewGridColor, 233
 setViewRulerColor, 233
 setViewSelectBoxColors, 233
 showViewScrollBars, 233
 sizeHint, 233
 updateColorLinetypeLineweight, 234
 Mega 2560 or another board with equal or, 23
 Melco, 535, 539, 544, 554
 menu_action
 Settings __, 290
 menu_name
 Action __, 58
 menu_position
 Action __, 58
 menuHash
 MainWindow, 208
 mergeWith
 UndoableNavCommand, 336
 metric
 EmbView __, 148
 Metro_Polyester
 embroidery.h, 446
 mid
 EmbArc __, 122
 miniSectorShift
 _bcf_file_header, 54
 miniStreamCutoffSize
 _bcf_file_header, 54
 minorVersion
 _bcf_file_header, 54
 minPoints
 UiObject __, 330
 minute
 EmbTime __, 145
 mirror
 UndoableMirrorCommand, 334
 mirrorLine
 UndoableMirrorCommand, 334
 mirrorSelected
 View, 347
 mit, 502, 553
 mitDecodeStitch
 embroidery_internal.h, 498
 encoding.c, 522
 mitEncodeStitch
 embroidery_internal.h, 498
 encoding.c, 522
 Mitsubishi, 502, 553
 mode
 UiObject __, 330
 modifiedTime
 _bcf_directory_entry, 50
 modifierName
 ThredExtension __, 328
 month
 EmbTime __, 145
 mouseDoubleClickEvent
 MdiArea, 223
 View, 347
 mouseMoveEvent
 CmdPromptHandle, 95
 View, 347
 mousePressEvent
 CmdPromptHandle, 95
 View, 347
 mouseReleaseEvent
 CmdPromptHandle, 95
 View, 347
 mouseSnapPoint
 ArcObject, 66
 BaseObject, 75
 CircleObject, 83

DimLeaderObject, 111
EllipseObject, 118
ImageObject, 155
LineObject, 164
PathObject, 239
PointObject, 244
PolygonObject, 249
PolylineObject, 254
RectObject, 274
TextSingleObject, 324
moveAction
 View, 347
movePoint
 View, 352
moveResizeHistory
 CmdPromptSplitter, 106
moveSelected
 View, 347
MOVETO
 embroidery_internal.h, 487
moveY
 CmdPromptHandle, 95
movingActive
 View, 352
myFileSeparator
 MainWindow, 209
myIndex
 MdiWindow, 235
n_attributes
 format_svg.c, 566
n_controlPoints
 UiObject_, 330
N_PES_VERSIONS
 embroidery_internal.h, 487
n_selected
 EmbView_, 148
name
 EmblImage_, 133
 EmbLayer_, 134
 SvgAttribute_, 321
 thread_color_, 327
nativeAddArc
 MainWindow, 185
nativeAddCircle
 MainWindow, 185
nativeAddDimLeader
 MainWindow, 185
nativeAddEllipse
 MainWindow, 185
nativeAddHorizontalDimension
 MainWindow, 185
nativeAddImage
 MainWindow, 185
nativeAddInfiniteLine
 MainWindow, 186
nativeAddLine
 MainWindow, 186
nativeAddPath
 MainWindow, 186
nativeAddPoint
 MainWindow, 186
nativeAddPolygon
 MainWindow, 186
nativeAddPolyline
 MainWindow, 186
nativeAddRay
 MainWindow, 187
nativeAddRectangle
 MainWindow, 187
nativeAddRegularPolygon
 MainWindow, 187
nativeAddRoundedRectangle
 MainWindow, 187
nativeAddSlot
 MainWindow, 187
nativeAddTextMulti
 MainWindow, 188
nativeAddTextSingle
 MainWindow, 188
nativeAddToSelection
 MainWindow, 188
nativeAddTriangle
 MainWindow, 188
nativeAddVerticalDimension
 MainWindow, 188
nativeAlert
 MainWindow, 189
nativeAllowRubber
 MainWindow, 189
nativeAppendPromptHistory
 MainWindow, 189
nativeBlinkPrompt
 MainWindow, 189
nativeCalculateAngle
 MainWindow, 189
nativeCalculateDistance
 MainWindow, 189
nativeClearRubber
 MainWindow, 189
nativeClearSelection
 MainWindow, 189
nativeCopySelected
 MainWindow, 189
nativeCutSelected
 MainWindow, 190
nativeDeleteSelected
 MainWindow, 190
nativeDisableMoveRapidFire
 MainWindow, 190
nativeDisablePromptRapidFire
 MainWindow, 190
nativeEnableMoveRapidFire
 MainWindow, 190
nativeEnablePromptRapidFire
 MainWindow, 190
nativeEndCommand

MainWindow, 190
nativeExit
 MainWindow, 190
nativeInitCommand
 MainWindow, 190
nativeMessageBox
 MainWindow, 191
nativeMirrorSelected
 MainWindow, 191
nativeMouseX
 MainWindow, 191
nativeMouseY
 MainWindow, 191
nativeMoveSelected
 MainWindow, 191
nativeNewFile
 MainWindow, 191
nativeNumSelected
 MainWindow, 192
nativeOpenFile
 MainWindow, 192
nativePasteSelected
 MainWindow, 192
nativePerpendicularDistance
 MainWindow, 192
nativePreviewOff
 MainWindow, 192
nativePreviewOn
 MainWindow, 192
nativePrintArea
 MainWindow, 192
nativeQSnapX
 MainWindow, 192
nativeQSnapY
 MainWindow, 193
nativeRotateSelected
 MainWindow, 193
nativeScaleSelected
 MainWindow, 193
nativeSelectAll
 MainWindow, 193
nativeSetBackgroundColor
 MainWindow, 193
nativeSetCrossHairColor
 MainWindow, 193
nativeSetCursorShape
 MainWindow, 194
nativeSetGridColor
 MainWindow, 194
nativeSetPromptPrefix
 MainWindow, 194
nativeSetRubberMode
 MainWindow, 194
nativeSetRubberPoint
 MainWindow, 194
nativeSetRubberText
 MainWindow, 194
nativeSpareRubber
 MainWindow, 194
nativeTextAngle
 MainWindow, 194
nativeTextBold
 MainWindow, 194
nativeTextFont
 MainWindow, 195
nativeTextItalic
 MainWindow, 195
nativeTextOverline
 MainWindow, 195
nativeTextSize
 MainWindow, 195
nativeTextStrikeOut
 MainWindow, 195
nativeTextUnderline
 MainWindow, 195
nativeTipOfDay
 MainWindow, 195
nativeVulcanize
 MainWindow, 195
nativeWindowCascade
 MainWindow, 195
nativeWindowClose
 MainWindow, 195
nativeWindowCloseAll
 MainWindow, 195
nativeWindowNext
 MainWindow, 196
nativeWindowPrevious
 MainWindow, 196
nativeWindowTile
 MainWindow, 196
navType
 UndoableNavCommand, 337
needle_speed
 Settings_, 290
negativeXHoopSize
 VipHeader_, 355
negativeYHoopSize
 VipHeader_, 355
new, 502, 553
newFile
 MainWindow, 196
next
 _bcf_directory_entry, 50
nightVision
 MainWindow, 196
nlenghts
 Huffman, 151
NoArrow
 DimLeaderObject, 110
NoLine
 DimLeaderObject, 110
NORMAL
 embroidery.h, 446
normalPath
 PathObject, 241

PolygonObject, 250
PolylineObject, 256
ntable
 Huffman, 151
NUM_FLAGS
 main.c, 594
numberOfBytesRemaining
 _vp3Hoop, 56
numberOfColors
 _vp3Hoop, 56
 VipHeader_, 355
numberOfDifatSectors
 _bcf_file_header, 54
numberOfDirectorySectors
 _bcf_file_header, 54
numberOfEntriesInDifatSector
 embroidery_internal.h, 498
numberOfEntriesInFatSector
 _bcf_file_fat, 53
numberOfFATSectors
 _bcf_file_header, 54
numberOfFormats
 embroidery.h, 446
numberOfMiniFatSectors
 _bcf_file_header, 55
numberOfStitches
 VipHeader_, 355
numOfDocs
 MainWindow, 209
numPoints
 UiObject_, 330
numSelected
 View, 348
numStiches
 ThredHeader_, 328

OBJ_COLOR
 embroidermodder.h, 362
OBJ_KEYS
 embroidermodder.h, 362
OBJ_LAYER
 embroidermodder.h, 362
OBJ_LTYPE
 embroidermodder.h, 362
OBJ_LTYPE_CENTER
 embroidermodder.h, 362
OBJ_LTYPE_CONT
 embroidermodder.h, 362
OBJ_LTYPE_DOT
 embroidermodder.h, 362
OBJ_LTYPE_FISHBONE
 embroidermodder.h, 363
OBJ_LTYPE_HIDDEN
 embroidermodder.h, 362
OBJ_LTYPE_PHANTOM
 embroidermodder.h, 362
OBJ_LTYPE_RUNNING
 embroidermodder.h, 363
OBJ_LTYPE_SATIN

 embroidermodder.h, 363
OBJ_LTYPE_VALUES
 embroidermodder.h, 362
OBJ_LTYPE_ZIGZAG
 embroidermodder.h, 363
OBJ_LWT
 embroidermodder.h, 362
OBJ_LWT_01
 embroidermodder.h, 363
OBJ_LWT_02
 embroidermodder.h, 363
OBJ_LWT_03
 embroidermodder.h, 363
OBJ_LWT_04
 embroidermodder.h, 363
OBJ_LWT_05
 embroidermodder.h, 363
OBJ_LWT_06
 embroidermodder.h, 363
OBJ_LWT_07
 embroidermodder.h, 363
OBJ_LWT_08
 embroidermodder.h, 363
OBJ_LWT_09
 embroidermodder.h, 363
OBJ_LWT_10
 embroidermodder.h, 363
OBJ_LWT_11
 embroidermodder.h, 363
OBJ_LWT_12
 embroidermodder.h, 363
OBJ_LWT_13
 embroidermodder.h, 363
OBJ_LWT_14
 embroidermodder.h, 363
OBJ_LWT_15
 embroidermodder.h, 363
OBJ_LWT_16
 embroidermodder.h, 363
OBJ_LWT_17
 embroidermodder.h, 363
OBJ_LWT_18
 embroidermodder.h, 363
OBJ_LWT_19
 embroidermodder.h, 363
OBJ_LWT_20
 embroidermodder.h, 363
OBJ_LWT_21
 embroidermodder.h, 363
OBJ_LWT_22
 embroidermodder.h, 363
OBJ_LWT_23
 embroidermodder.h, 363
OBJ_LWT_24
 embroidermodder.h, 363
OBJ_LWT_BYBLOCK
 embroidermodder.h, 363
OBJ_LWT_BYLAYER

embroidermodder.h, 363
OBJ_LWT_DEFAULT
 embroidermodder.h, 363
OBJ_LWT_VALUES
 embroidermodder.h, 363
OBJ_NAME
 embroidermodder.h, 362
OBJ_RUBBER
 embroidermodder.h, 362
OBJ_RUBBER_CIRCLE_1P_DIA
 embroidermodder.h, 364
OBJ_RUBBER_CIRCLE_1P_RAD
 embroidermodder.h, 363
OBJ_RUBBER_CIRCLE_2P
 embroidermodder.h, 364
OBJ_RUBBER_CIRCLE_3P
 embroidermodder.h, 364
OBJ_RUBBER_CIRCLE_TTR
 embroidermodder.h, 364
OBJ_RUBBER_CIRCLE_TTT
 embroidermodder.h, 364
OBJ_RUBBER_DIMLEADER_LINE
 embroidermodder.h, 364
OBJ_RUBBER_ELLIPSE_LINE
 embroidermodder.h, 364
OBJ_RUBBER_ELLIPSE_MAJOR_DIAMETER_MINOR_RADIUS_SNAP_VALUES
 embroidermodder.h, 364
OBJ_RUBBER_ELLIPSE_MAJOR_RADIUS_MINOR_RADIUS_SBJ_TYPE
 embroidermodder.h, 364
OBJ_RUBBER_ELLIPSE_ROTATION
 embroidermodder.h, 364
OBJ_RUBBER_GRIP
 embroidermodder.h, 364
OBJ_RUBBER_IMAGE
 embroidermodder.h, 364
OBJ_RUBBER_LINE
 embroidermodder.h, 364
OBJ_RUBBER_OFF
 embroidermodder.h, 363
OBJ_RUBBER_ON
 embroidermodder.h, 363
OBJ_RUBBER_POLYGON
 embroidermodder.h, 364
OBJ_RUBBER_POLYGON_CIRCUMSCRIBE
 embroidermodder.h, 364
OBJ_RUBBER_POLYGON_INSCRIBE
 embroidermodder.h, 364
OBJ_RUBBER_POLYLINE
 embroidermodder.h, 364
OBJ_RUBBER_RECTANGLE
 embroidermodder.h, 364
OBJ_RUBBER_TEXTSINGLE
 embroidermodder.h, 364
OBJ_RUBBER_VALUES
 embroidermodder.h, 363
OBJ_SNAP_APPINTERSECTION
 embroidermodder.h, 364
OBJ_SNAP_CENTER
 embroidermodder.h, 364
OBJ_SNAP_ENDPOINT
 embroidermodder.h, 364
OBJ_SNAP_EXTENSION
 embroidermodder.h, 364
OBJ_SNAP_INSERTION
 embroidermodder.h, 364
OBJ_SNAP_INTERSECTION
 embroidermodder.h, 364
OBJ_SNAP_MIDPOINT
 embroidermodder.h, 364
OBJ_SNAP_NEAREST
 embroidermodder.h, 364
OBJ_SNAP_NODE
 embroidermodder.h, 364
OBJ_SNAP_NULL
 embroidermodder.h, 364
OBJ_SNAP_PARALLEL
 embroidermodder.h, 364
OBJ_SNAP_PERPENDICULAR
 embroidermodder.h, 364
OBJ_SNAP_QUADRANT
 embroidermodder.h, 364
OBJ_SNAP_TANGENT
 embroidermodder.h, 364
OBJ_SNAP_VALUES
 embroidermodder.h, 364
OBJ_TYPE
 embroidermodder.h, 362
OBJ_TYPE_ARC
 embroidermodder.h, 365
OBJ_TYPE_BASE
 embroidermodder.h, 365
OBJ_TYPE_BLOCK
 embroidermodder.h, 365
OBJ_TYPE_CIRCLE
 embroidermodder.h, 365
OBJ_TYPE_DIMALIGNED
 embroidermodder.h, 365
OBJ_TYPE_DIMANGULAR
 embroidermodder.h, 365
OBJ_TYPE_DIMARCLENGTH
 embroidermodder.h, 365
OBJ_TYPE_DIMDIAMETER
 embroidermodder.h, 365
OBJ_TYPE_DIMLEADER
 embroidermodder.h, 365
OBJ_TYPE_DIMLINEAR
 embroidermodder.h, 365
OBJ_TYPE_DIMORDINATE
 embroidermodder.h, 365
OBJ_TYPE_DIMRADIUS
 embroidermodder.h, 365
OBJ_TYPE_ELLIPSE
 embroidermodder.h, 365
OBJ_TYPE_ELLIPSEARC
 embroidermodder.h, 365
OBJ_TYPE_GRID

embroidermodder.h, 365
OBJ_TYPE_HATCH
embroidermodder.h, 365
OBJ_TYPE_IMAGE
embroidermodder.h, 365
OBJ_TYPE_INFINITELINE
embroidermodder.h, 365
OBJ_TYPE_LINE
embroidermodder.h, 365
OBJ_TYPE_NULL
embroidermodder.h, 365
OBJ_TYPE_PATH
embroidermodder.h, 365
OBJ_TYPE_POINT
embroidermodder.h, 365
OBJ_TYPE_POLYGON
embroidermodder.h, 365
OBJ_TYPE_POLYLINE
embroidermodder.h, 365
OBJ_TYPE_RAY
embroidermodder.h, 365
OBJ_TYPE_RECTANGLE
embroidermodder.h, 365
OBJ_TYPE_RUBBER
embroidermodder.h, 365
OBJ_TYPE_SLOT
embroidermodder.h, 365
OBJ_TYPE_SPLINE
embroidermodder.h, 365
OBJ_TYPE_TEXTMULTI
embroidermodder.h, 365
OBJ_TYPE_TEXTSINGLE
embroidermodder.h, 365
OBJ_TYPE_VALUES
embroidermodder.h, 364
object
 EmbGeometry_, 131
 UndoableAddCommand, 332
 UndoableDeleteCommand, 332
 UndoableGripEditCommand, 333
 UndoableMirrorCommand, 334
 UndoableMoveCommand, 335
 UndoableRotateCommand, 338
 UndoableScaleCommand, 339
object-arc.cpp
 rotate_vector, 411
object_index
 UiObject_, 330
objectAngle
 DimLeaderObject, 111
 LineObject, 164
objectArcLength
 ArcObject, 67
objectArea
 ArcObject, 67
 CircleObject, 83
 ImageObject, 155
 RectObject, 274
objectBottomLeft
 ImageObject, 155
 RectObject, 274
objectBottomRight
 ImageObject, 155
 RectObject, 274
objectCenter
 BaseObject, 75
objectCenterX
 BaseObject, 75
objectCenterY
 BaseObject, 75
objectChord
 ArcObject, 67
objectCircumference
 CircleObject, 83
objectClockwise
 ArcObject, 67
objectColor
 BaseObject, 75
objectColorRGB
 BaseObject, 76
objectCopyPath
 PathObject, 239
 PolygonObject, 249
 PolylineObject, 254
objectDeltaX
 DimLeaderObject, 112
 LineObject, 164
objectDeltaY
 DimLeaderObject, 112
 LineObject, 164
objectDiameter
 CircleObject, 83
objectDiameterMajor
 EllipseObject, 118
objectDiameterMinor
 EllipseObject, 118
objectEndAngle
 ArcObject, 67
objectEndPoint
 ArcObject, 68
objectEndPoint1
 DimLeaderObject, 112
 LineObject, 164
objectEndPoint2
 DimLeaderObject, 112
 LineObject, 165
objectEndX
 ArcObject, 68
objectEndY
 ArcObject, 68
objectHeight
 EllipseObject, 119
 ImageObject, 155
 RectObject, 274
objectID
 BaseObject, 76

objectIncludedAngle
 ArcObject, 68
objectLength
 DimLeaderObject, 112
 LineObject, 165
objectLineType
 BaseObject, 76
objectLineWidth
 BaseObject, 76
objectMidPoint
 ArcObject, 68
 DimLeaderObject, 112
 LineObject, 165
objectMidX
 ArcObject, 69
objectMidY
 ArcObject, 69
objectPath
 BaseObject, 76
objectPen
 BaseObject, 76
objectPos
 PathObject, 239
 PointObject, 244
 PolygonObject, 249
 PolylineObject, 254
 RectObject, 274
 TextSingleObject, 324
objectQuadrant0
 CircleObject, 83
 EllipseObject, 119
objectQuadrant180
 CircleObject, 83
 EllipseObject, 119
objectQuadrant270
 CircleObject, 83
 EllipseObject, 119
objectQuadrant90
 CircleObject, 83
 EllipseObject, 119
objectRadius
 ArcObject, 69
 CircleObject, 83
objectRadiusMajor
 EllipseObject, 119
objectRadiusMinor
 EllipseObject, 119
objectRubberMode
 BaseObject, 76
objectRubberPoint
 BaseObject, 76
objectRubberText
 BaseObject, 76
objectSavePath
 CircleObject, 83
 EllipseObject, 119
 LineObject, 165
 PathObject, 239
 PointObject, 244
 PolygonObject, 249
 PolylineObject, 254
 RectObject, 274
 TextSingleObject, 324
objectSavePathList
 TextSingleObject, 324
objectStartAngle
 ArcObject, 69
objectStartPoint
 ArcObject, 69
objectStartX
 ArcObject, 69
objectStartY
 ArcObject, 70
objectTextJustifyList
 TextSingleObject, 324
objectTopLeft
 ImageObject, 156
 RectObject, 275
objectTopRight
 ImageObject, 156
 RectObject, 275
objectType
 _bcf_directory_entry, 50
ObjectTypeRootEntry
 embroidery_internal.h, 487
ObjectTypeStorage
 embroidery_internal.h, 487
ObjectTypeStream
 embroidery_internal.h, 487
ObjectTypeUnknown
 embroidery_internal.h, 487
objectWidth
 EllipseObject, 119
 ImageObject, 156
 RectObject, 275
objectX
 PathObject, 239
 PointObject, 244
 PolygonObject, 249
 PolylineObject, 255
 TextSingleObject, 324
objectX1
 DimLeaderObject, 112
 LineObject, 165
objectX2
 DimLeaderObject, 112
 LineObject, 165
objectY
 PathObject, 239
 PointObject, 244
 PolygonObject, 249
 PolylineObject, 255
 TextSingleObject, 324
objectY1
 DimLeaderObject, 112
 LineObject, 165
objectY2

DimLeaderObject, 112
LineObject, 165
objID
 BaseObject, 79
objLine
 BaseObject, 79
objPen
 BaseObject, 79
objRubberMode
 BaseObject, 79
objRubberPoints
 BaseObject, 79
objRubberTexts
 BaseObject, 79
objText
 TextSingleObject, 326
objTextBackward
 TextSingleObject, 326
objTextBold
 TextSingleObject, 326
objTextFont
 TextSingleObject, 326
objTextItalic
 TextSingleObject, 326
objTextJustify
 TextSingleObject, 326
objTextOverline
 TextSingleObject, 326
objTextPath
 TextSingleObject, 326
objTextSize
 TextSingleObject, 326
objTextStrikeOut
 TextSingleObject, 327
objTextUnderline
 TextSingleObject, 327
objTextUpsideDown
 TextSingleObject, 327
ofm, 554
ofmDecode
 format_ofm.c, 554
ofmReadBlockHeader
 format_ofm.c, 554
ofmReadClass
 format_ofm.c, 554
ofmReadColorChange
 format_ofm.c, 554
ofmReadExpanded
 format_ofm.c, 554
ofmReadLibrary
 format_ofm.c, 554
ofmReadThreads
 format_ofm.c, 554
onCloseMdiWin
 MainWindow, 196
onCloseWindow
 MainWindow, 196
onWindowActivated

MainWindow, 196
MdiWindow, 231
Open
 DimLeaderObject, 110
openFile
 MainWindow, 197
openFilePath
 MainWindow, 209
openFilesSelected
 MainWindow, 197
openrecentfile
 MainWindow, 197
opensave_custom_filter
 Settings_, 290
opensave_open_format
 Settings_, 290
opensave_open_thumbnail
 Settings_, 290
opensave_recent_directory
 Settings_, 290
opensave_recent_list_of_files
 Settings_, 290
opensave_recent_max_files
 Settings_, 290
opensave_save_format
 Settings_, 290
opensave_save_thumbnail
 Settings_, 290
opensave_trim_dst_num_jumps
 Settings_, 290
operator+
 embroidermodder.h, 367
operator-
 embroidermodder.h, 367
origin
 EmbView_, 148
originPath
 View, 352
ortho_mode
 EmbView_, 148
paint
 ArcObject, 70
 CircleObject, 84
 DimLeaderObject, 112
 EllipseObject, 119
 ImageObject, 156
 LineObject, 165
 PathObject, 239
 PointObject, 244
 PolygonObject, 249
 PolylineObject, 255
 RectObject, 275
 TextSingleObject, 324
paintEvent
 ImageWidget, 158
 MdiArea, 223
 SelectBox, 283
pan_toolbar

mainwindow-toolbars.cpp, 407
 panDistance
 View, 352
 panDown
 MainWindow, 197
 View, 348
 panLeft
 MainWindow, 197
 View, 348
 panMenu
 MainWindow, 209
 panningActive
 View, 353
 panningPointActive
 View, 353
 panningRealTimeActive
 View, 353
 panPoint
 View, 348
 panpoint
 MainWindow, 197
 panRealTime
 View, 348
 panrealtime
 MainWindow, 198
 panRight
 MainWindow, 198
 View, 348
 panStart
 View, 348
 panStartX
 View, 353
 panStartY
 View, 353
 Pantone
 embroidery.h, 447
 panUp
 MainWindow, 198
 View, 348
 Parameter
 mainwindow.cpp, 408
 Parameter_
 i_value, 235
 r_value, 236
 s_value, 236
 parseDIFATSectors
 main.c, 598
 parseDirectoryEntryName
 main.c, 598
 parseTime
 main.c, 598
 paste
 MainWindow, 198
 View, 348
 pasteClip
 CmdPromptInput, 102
 pasteDelta
 View, 353
 pasteObjectItemGroup
 View, 353
 pastePressed
 CmdPrompt, 90
 CmdPromptInput, 102
 pastingActive
 View, 353
 path
 EmbGeometry_, 131
 EmblImage_, 133
 path_desc
 UiObject_, 330
 PathObject, 236
 ~PathObject, 238
 allGripPoints, 238
 gripEdit, 239
 init, 239
 mouseSnapPoint, 239
 normalPath, 241
 objectCopyPath, 239
 objectPos, 239
 objectSavePath, 239
 objectX, 239
 objectY, 239
 paint, 239
 PathObject, 238
 setObjectPos, 240
 setObjectX, 240
 setObjectY, 240
 Type, 238
 type, 240
 updatePath, 240
 updateRubber, 240
 vulcanize, 240
 pattern
 EmbView_, 148
 pattern.c
 convert, 601
 embPattern_addCircleAbs, 601
 embPattern_addEllipseAbs, 601
 embPattern_addLineAbs, 602
 embPattern_addPathAbs, 602
 embPattern_addPointAbs, 602
 embPattern_addPolygonAbs, 602
 embPattern_addPolylineObjectAbs, 602
 embPattern_addRectAbs, 602
 embPattern_addStitchAbs, 602
 embPattern_addStitchRel, 602
 embPattern_addThread, 602
 embPattern_calcBoundingBox, 603
 embPattern_center, 603
 embPattern_changeColor, 603
 embPattern_color_count, 603
 embPattern_combineJumpStitches, 603
 embPattern_copyPolylinesToStitch_list, 603
 embPattern_copyStitch_listToPolylines, 603
 embPattern_correctForMaxStitchLength, 603
 embPattern_create, 603

embPattern_designDetails, 603
embPattern_end, 604
embPattern_fixColorCount, 604
embPattern_flip, 604
embPattern_flipHorizontal, 604
embPattern_flipVertical, 604
embPattern_free, 604
embPattern_hideStitchesOverLength, 604
embPattern_jumpStitches, 604
embPattern_lengthHistogram, 604
embPattern_loadExternalColorFile, 604
embPattern_maximumStitchLength, 604
embPattern_minimumStitchLength, 604
embPattern_movePolylinesToStitch_list, 605
embPattern_movestitch_listToPolylines, 605
embPattern_realStitches, 605
embPattern_scale, 605
embPattern_totalStitchLength, 605
embPattern_trimStitches, 605
pattern_index
 Settings_, 290
 UiObject_, 330
pcd, 16, 503, 555
pcm, 16, 503, 555
pcm_thread
 embroidery.h, 447
pcmThreads
 embroidery.h, 465
 thread-color.c, 606
pcq, 16, 503, 556
pcs, 16, 503, 556
pec, 16, 503, 557
pec_thread
 embroidery.h, 447
pecEncode
 format_pec.c, 557
pecEncodeJump
 format_pec.c, 557
pecEncodeStop
 format_pec.c, 557
pecThreadCount
 embroidery.h, 465
 thread-color.c, 606
pecThreads
 embroidery.h, 466
 thread-color.c, 607
pel, 16, 503, 558
pem, 16, 504, 558
pes, 16, 561
PES0001
 embroidery_internal.h, 487
PES0020
 embroidery_internal.h, 487
PES0022
 embroidery_internal.h, 487
PES0030
 embroidery_internal.h, 487
PES0040
 embroidery_internal.h, 487
 embroidery_internal.h, 487
PES0050
 embroidery_internal.h, 487
PES0055
 embroidery_internal.h, 487
PES0056
 embroidery_internal.h, 488
PES0060
 embroidery_internal.h, 488
PES0070
 embroidery_internal.h, 488
PES0080
 embroidery_internal.h, 488
PES0090
 embroidery_internal.h, 488
PES0100
 embroidery_internal.h, 488
pes_version
 format_pes.c, 561
pes_version_strings
 format_pes.c, 561
pesWriteEmbOneSection
 format_pes.c, 559
pesWriteSewSegSection
 format_pes.c, 559
Pfaff, 466, 503, 505, 551, 552, 555, 556, 567, 572, 573
pfaffDecode
 embroidery_internal.h, 498
 encoding.c, 522
pfaffEncode
 embroidery_internal.h, 498
 encoding.c, 523
phb, 16, 504, 561
phc, 16, 504, 562
pickAdd
 PropertyEditor, 267
pickAddModeToggled
 MainWindow, 198
 PropertyEditor, 265
pickBoxSize
 View, 353
pivotX
 UndoableRotateCommand, 338
pivotY
 UndoableRotateCommand, 338
platformString
 MainWindow, 198
plt, 505, 562
point
 EmbGeometry_, 131
pointList
 EmbPath_, 137
PointObject, 241
 ~PointObject, 243
 allGripPoints, 243
 gripEdit, 244
 init, 244
 mouseSnapPoint, 244

objectPos, 244
 objectSavePath, 244
 objectX, 244
 objectY, 244
 paint, 244
 PointObject, 243
 setObjectPos, 244, 245
 setObjectX, 245
 setObjectY, 245
 Type, 243
 type, 245
 updateRubber, 245
 vulcanize, 245
polar_mode
 EmbView_, 148
polygon
 EmbGeometry_, 131
PolygonObject, 245
 ~PolygonObject, 248
 allGripPoints, 248
 findIndex, 248
 gripEdit, 248
 gripIndex, 250
 init, 248
 mouseSnapPoint, 249
 normalPath, 250
 objectCopyPath, 249
 objectPos, 249
 objectSavePath, 249
 objectX, 249
 objectY, 249
 paint, 249
 PolygonObject, 248
 setObjectPos, 249
 setObjectX, 250
 setObjectY, 250
 Type, 248
 type, 250
 updatePath, 250
 updateRubber, 250
 vulcanize, 250
polyline
 EmbGeometry_, 132
PolylineObject, 251
 ~PolylineObject, 253
 allGripPoints, 254
 findIndex, 254
 gripEdit, 254
 gripIndex, 256
 init, 254
 mouseSnapPoint, 254
 normalPath, 256
 objectCopyPath, 254
 objectPos, 254
 objectSavePath, 254
 objectX, 255
 objectY, 255
 paint, 255
 PolylineObject, 253
 setObjectPos, 255
 setObjectX, 255
 setObjectY, 255
 Type, 253
 type, 255
 updatePath, 255
 updateRubber, 256
 vulcanize, 256
position
 EmbAlignedDim_, 121
 EmbAngularDim_, 121
 EmbArcLengthDim_, 122
 EmbBlock_, 124
 EmbDiameterDim_, 128
 EmbImage_, 133
 EmbInfiniteLine_, 134
 EmbLeaderDim_, 135
 EmbLinearDim_, 136
 EmbOrdinateDim_, 136
 EmbPoint_, 139
 EmbRadiusDim_, 139
 EmbRay_, 140
 EmbTextMulti_, 143
 EmbTextSingle_, 144
 UndoHistory_, 341
postitiveXHoopSize
 VipHeader_, 356
postitiveYHoopSize
 VipHeader_, 356
precisionAngle
 PropertyEditor, 267
precisionLength
 PropertyEditor, 267
prefix
 CmdPromptInput, 104
pressPoint
 View, 353
pressResizeHistory
 CmdPromptSplitter, 106
pressY
 CmdPromptHandle, 95
preview
 mainwindow.cpp, 410
PREVIEW_CLONE_NULL
 embroidermodder.h, 365
PREVIEW_CLONE_RUBBER
 embroidermodder.h, 365
PREVIEW_CLONE_SELECTED
 embroidermodder.h, 365
PREVIEW_CLONE_VALUES
 embroidermodder.h, 365
preview_display_bg_color
 Settings_Dialog, 314
preview_display_crosshair_color
 Settings_Dialog, 314
preview_display_selectbox_alpha
 Settings_Dialog, 314

preview_display_selectbox_left_color
 Settings_Dialog, 314
preview_display_selectbox_left_fill
 Settings_Dialog, 314
preview_display_selectbox_right_color
 Settings_Dialog, 314
preview_display_selectbox_right_fill
 Settings_Dialog, 314
preview_display_show_scrollbars
 Settings_Dialog, 314
preview_general_mdi_bg_color
 Settings_Dialog, 315
preview_general_mdi_bg_use_color
 Settings_Dialog, 315
preview_general_mdi_bg_use_logo
 Settings_Dialog, 315
preview_general_mdi_bg_use_texture
 Settings_Dialog, 315
preview_grid_color
 Settings_Dialog, 315
preview_lwt_real_render
 Settings_Dialog, 315
preview_lwt_show_lwt
 Settings_Dialog, 315
PREVIEW_MODE_MOVE
 embroidermodder.h, 365
PREVIEW_MODE_NULL
 embroidermodder.h, 365
PREVIEW_MODE_ROTATE
 embroidermodder.h, 365
PREVIEW_MODE_SCALE
 embroidermodder.h, 366
PREVIEW_MODE_VALUES
 embroidermodder.h, 365
preview_prompt_bg_color
 Settings_Dialog, 315
preview_prompt_font_family
 Settings_Dialog, 315
preview_prompt_font_size
 Settings_Dialog, 315
preview_prompt_font_style
 Settings_Dialog, 315
preview_prompt_text_color
 Settings_Dialog, 315
preview_ruler_color
 Settings_Dialog, 315
previewActive
 View, 353
previewData
 View, 353
PreviewDialog, 256
 ~PreviewDialog, 257
 imgWidget, 257
 PreviewDialog, 256
previewMode
 View, 353
previewObjectItemGroup
 View, 353
previewObjectList
 View, 353
previewOff
 View, 348
previewOn
 View, 348
previewPoint
 View, 353
print
 MainWindow, 198
 MdiWindow, 231
printArcResults
 embroidery_internal.h, 498
printer
 MdiWindow, 235
printing_default_device
 Settings_, 291
printing_disable_bg
 Settings_, 291
printing_use_last_device
 Settings_, 291
privacy_policy.md, 607
processInput
 CmdPrompt, 90
 CmdPromptInput, 102
prompt
 MainWindow, 209
promptDivider
 CmdPrompt, 93
promptHistory
 CmdPrompt, 93
 MdiWindow, 235
promptHistoryAppended
 MainWindow, 198
 MdiWindow, 231
promptInput
 CmdPrompt, 93
promptInputList
 MdiWindow, 235
promptInputNext
 MainWindow, 198
 MdiWindow, 231
promptInputNum
 MdiWindow, 235
promptInputPrevious
 MainWindow, 198
 MdiWindow, 231
promptInputPrevNext
 MdiWindow, 231
promptSplitter
 CmdPrompt, 93
promptVBoxLayout
 CmdPrompt, 93
property-editor.cpp
 comboBoxArcClockwise, 416
 comboBoxes, 416
 comboBoxGeneralLineType, 416
 comboBoxGeneralLineWidth, 416

comboBoxPathClosed, 417
comboBoxPathVertexNum, 417
comboBoxPolylineClosed, 417
comboBoxPolylineVertexNum, 417
comboBoxTextSingleBackward, 417
comboBoxTextSingleFont, 417
comboBoxTextSingleJustify, 417
comboBoxTextSingleUpsideDown, 417
groupBoxes, 417
groupBoxGeometryCircle, 417
groupBoxGeometryDimAligned, 417
groupBoxGeometryDimAngular, 417
groupBoxGeometryDimArcLength, 417
groupBoxGeometryDimDiameter, 417
groupBoxGeometryDimLeader, 417
groupBoxGeometryDimLinear, 417
groupBoxGeometryDimOrdinate, 417
groupBoxGeometryDimRadius, 417
groupBoxGeometryEllipse, 418
groupBoxGeometryImage, 418
groupBoxGeometryInfiniteLine, 418
groupBoxGeometryLine, 418
groupBoxGeometryPath, 418
groupBoxGeometryPoint, 418
groupBoxGeometryPolygon, 418
groupBoxGeometryPolyline, 418
groupBoxGeometryRay, 418
groupBoxGeometryRectangle, 418
groupBoxGeometryTextMulti, 418
groupBoxGeometryTextSingle, 418
groupBoxMisclImage, 418
groupBoxMiscPath, 418
groupBoxMiscPolyline, 418
groupBoxMiscTextSingle, 418
groupBoxTextTextSingle, 418
lineEditArcArea, 418
lineEditArcCenterY, 419
lineEditArcChord, 419
lineEditArcEndAngle, 419
lineEditArcEndX, 419
lineEditArcEndY, 419
lineEditArcInclAngle, 419
lineEditArcLength, 419
lineEditArcRadius, 419
lineEditArcStartAngle, 419
lineEditArcStartX, 419
lineEditArcStartY, 419
lineEditBlockX, 419
lineEditBlockY, 419
lineEditCircleArea, 419
lineEditCircleCenterX, 419
lineEditCircleCenterY, 419
lineEditCircleCircumference, 419
lineEditCircleDiameter, 419
lineEditCircleRadius, 420
lineEditEllipseCenterX, 420
lineEditEllipseCenterY, 420
lineEditEllipseDiameterMajor, 420
lineEditEllipseDiameterMinor, 420
lineEditEllipseRadiusMajor, 420
lineEditEllipseRadiusMinor, 420
lineEditImageHeight, 420
lineEditImageName, 420
lineEditImagePath, 420
lineEditImageWidth, 420
lineEditImageX, 420
lineEditImageY, 420
lineEditInfiniteLineVectorX, 420
lineEditInfiniteLineVectorY, 420
lineEditInfiniteLineX2, 420
lineEditInfiniteLineY1, 420
lineEditInfiniteLineY2, 420
lineEditLineAngle, 421
lineEditLineDeltaX, 421
lineEditLineDeltaY, 421
lineEditLineEndX, 421
lineEditLineEndY, 421
lineEditLineLength, 421
lineEditLineStartX, 421
lineEditLineStartY, 421
lineEditPathArea, 421
lineEditPathLength, 421
lineEditPathVertexX, 421
lineEditPathVertexY, 421
lineEditPointX, 421
lineEditPointY, 421
lineEditPolygonCenterX, 421
lineEditPolygonCenterY, 421
lineEditPolygonDiameterSide, 421
lineEditPolygonDiameterVertex, 421
lineEditPolygonInteriorAngle, 422
lineEditPolygonRadiusSide, 422
lineEditPolygonRadiusVertex, 422
lineEditPolylineArea, 422
lineEditPolylineLength, 422
lineEditPolylineVertexX, 422
lineEditPolylineVertexY, 422
lineEditRayVectorX, 422
lineEditRayVectorY, 422
lineEditRayX1, 422
lineEditRayX2, 422
lineEditRayY1, 422
lineEditRayY2, 422
lineEditRectangleArea, 422
lineEditRectangleCorner1X, 422
lineEditRectangleCorner1Y, 422
lineEditRectangleCorner2X, 422
lineEditRectangleCorner2Y, 422
lineEditRectangleCorner3X, 423
lineEditRectangleCorner3Y, 423
lineEditRectangleCorner4X, 423
lineEditRectangleCorner4Y, 423
lineEditRectangleHeight, 423
lineEditRectangleWidth, 423
lineEdits, 423
lineEditTextMultiX, 423

lineEditTextMultiY, 423
lineEditTextSingleContents, 423
lineEditTextSingleHeight, 423
lineEditTextSingleRotation, 423
lineEditTextSingleX, 423
lineEditTextSingleY, 423
toolButtonArcClockwise, 423
toolButtonBlockX, 423
toolButtonBlockY, 423
toolButtonCircleArea, 423
toolButtonCircleCenterX, 424
toolButtonCircleCenterY, 424
toolButtonCircleCircumference, 424
toolButtonCircleDiameter, 424
toolButtonCircleRadius, 424
toolButtonEllipseCenterX, 424
toolButtonEllipseCenterY, 424
toolButtonEllipseDiameterMajor, 424
toolButtonEllipseDiameterMinor, 424
toolButtonEllipseRadiusMajor, 424
toolButtonEllipseRadiusMinor, 424
toolButtonImageHeight, 424
toolButtonImageName, 424
toolButtonImagePath, 424
toolButtonImageWidth, 424
toolButtonImageX, 424
toolButtonImageY, 424
toolButtonInfiniteLineVectorX, 424
toolButtonInfiniteLineVectorY, 425
toolButtonInfiniteLineX2, 425
toolButtonInfiniteLineY1, 425
toolButtonInfiniteLineY2, 425
toolButtonLineAngle, 425
toolButtonLineDeltaX, 425
toolButtonLineDeltaY, 425
toolButtonLineEndX, 425
toolButtonLineEndY, 425
toolButtonLineLength, 425
toolButtonLineStartX, 425
toolButtonLineStartY, 425
toolButtonPathArea, 425
toolButtonPathClosed, 425
toolButtonPathLength, 425
toolButtonPathVertexNum, 425
toolButtonPathVertexX, 425
toolButtonPathVertexY, 425
toolButtonPointX, 426
toolButtonPointY, 426
toolButtonPolygonCenterX, 426
toolButtonPolygonCenterY, 426
toolButtonPolygonDiameterSide, 426
toolButtonPolygonDiameterVertex, 426
toolButtonPolygonInteriorAngle, 426
toolButtonPolygonRadiusSide, 426
toolButtonPolygonRadiusVertex, 426
toolButtonPolylineArea, 426
toolButtonPolylineClosed, 426
toolButtonPolylineLength, 426
toolButtonPolylineVertexNum, 426
toolButtonPolylineVertexX, 426
toolButtonPolylineVertexY, 426
toolButtonRayVectorX, 426
toolButtonRayVectorY, 426
toolButtonRayX1, 426
toolButtonRayX2, 427
toolButtonRayY1, 427
toolButtonRayY2, 427
toolButtonRectangleArea, 427
toolButtonRectangleCorner1X, 427
toolButtonRectangleCorner1Y, 427
toolButtonRectangleCorner2X, 427
toolButtonRectangleCorner2Y, 427
toolButtonRectangleCorner3X, 427
toolButtonRectangleCorner3Y, 427
toolButtonRectangleCorner4X, 427
toolButtonRectangleCorner4Y, 427
toolButtonRectangleHeight, 427
toolButtonRectangleWidth, 427
toolButtons, 427
toolButtonTextMultiX, 427
toolButtonTextMultiY, 427
toolButtonTextSingleBackward, 427
toolButtonTextSingleContents, 428
toolButtonTextSingleFont, 428
toolButtonTextSingleHeight, 428
toolButtonTextSingleJustify, 428
toolButtonTextSingleRotation, 428
toolButtonTextSingleUpsideDown, 428
toolButtonTextSingleX, 428
toolButtonTextSingleY, 428

PropertyEditor, 257
~PropertyEditor, 260
clearAllFields, 260
comboBoxSelected, 266
createComboBox, 260
createComboBoxSelected, 260
createFontComboBox, 260
createGroupBoxGeneral, 260
createGroupBoxGeometryArc, 260
createGroupBoxGeometryBlock, 261
createGroupBoxGeometryCircle, 261
createGroupBoxGeometryDimAligned, 261
createGroupBoxGeometryDimAngular, 261
createGroupBoxGeometryDimArcLength, 261
createGroupBoxGeometryDimDiameter, 261
createGroupBoxGeometryDimLeader, 261
createGroupBoxGeometryDimLinear, 261
createGroupBoxGeometryDimOrdinate, 262
createGroupBoxGeometryDimRadius, 262
createGroupBoxGeometryEllipse, 262
createGroupBoxGeometryImage, 262
createGroupBoxGeometryInfiniteLine, 262
createGroupBoxGeometryLine, 262
createGroupBoxGeometryPath, 262
createGroupBoxGeometryPoint, 262
createGroupBoxGeometryPolygon, 263

createGroupBoxGeometryPolyline, 263
createGroupBoxGeometryRay, 263
createGroupBoxGeometryRectangle, 263
createGroupBoxGeometryTextMulti, 263
createGroupBoxGeometryTextSingle, 263
createGroupBoxMiscArc, 263
createGroupBoxMiscImage, 263
createGroupBoxMiscPath, 263
createGroupBoxMiscPolyline, 264
createGroupBoxMiscTextSingle, 264
createGroupBoxTextTextSingle, 264
createLineEdit, 264
createToolButton, 264
createToolButtonPickAdd, 264
createToolButtonQSelect, 264
eventFilter, 264
fieldEdited, 264
fieldNewText, 266
fieldNoText, 266
fieldOffText, 266
fieldOldText, 266
fieldOnText, 267
fieldVariesText, 267
fieldYesText, 267
focusWidget, 267
hideAllGroups, 265
iconDir, 267
iconSize, 267
mapSignal, 265
pickAdd, 267
pickAddModeToggled, 265
precisionAngle, 267
precisionLength, 267
PropertyEditor, 259
propertyEditorButtonStyle, 267
selectedItemList, 267
setSelectedItems, 265
showGroups, 265
showOneType, 265
signalMapper, 268
tempArcObj, 268
tempBlockObj, 268
tempCircleObj, 268
tempDimAlignedObj, 268
tempDimAngularObj, 268
tempDimArcLenObj, 268
tempDimDiamObj, 268
tempDimLeaderObj, 268
tempDimLinearObj, 268
tempDimOrdObj, 268
tempDimRadiusObj, 269
tempEllipseArcObj, 269
tempEllipseObj, 269
tempHatchObj, 269
tempImageObj, 269
tempInflLineObj, 269
tempLineObj, 269
tempPathObj, 269
tempPointObj, 269
tempPolygonObj, 269
tempPolylineObj, 269
tempRayObj, 270
tempRectObj, 270
tempSplineObj, 270
tempTextMultiObj, 270
tempTextSingleObj, 270
togglePickAddMode, 265
toolButtonPickAdd, 270
toolButtonQSelect, 270
updateComboBoxBoolIfVaries, 265
updateComboBoxStrIfVaries, 265
updateFontComboBoxStrIfVaries, 266
updateLineEditNumIfVaries, 266
updateLineEditStrIfVaries, 266
updatePickAddModeButton, 266
propertyEditorButtonStyle
 PropertyEditor, 267

qsnap_aperture_size
 Settings_, 291
qsnap_apparent
 Settings_, 291
qsnap_center
 Settings_, 291
qsnap_enabled
 Settings_, 291
qsnap_endpoint
 Settings_, 291
qsnap_extension
 Settings_, 291
qsnap_insertion
 Settings_, 291
qsnap_intersection
 Settings_, 291
qsnap_locator_color
 Settings_, 291
qsnap_locator_size
 Settings_, 291
qsnap_midpoint
 Settings_, 291
qsnap_mode
 EmbView_, 148
qsnap_nearest
 Settings_, 291
qsnap_node
 Settings_, 291
qsnap_parallel
 Settings_, 291
qsnap_perpendicular
 Settings_, 291
qsnap_quadrant
 Settings_, 292
qsnap_tangent
 Settings_, 292
qSnapActive
 View, 353
qsnapApertureSize

View, 353
qsnapLocatorColor
 View, 354
qsnapLocatorSize
 View, 354
qSnapToggle
 View, 354
qtrack_mode
 EmbView_, 148
QUADTOCONTROL
 embroidery_internal.h, 488
QUADTOEND
 embroidery_internal.h, 488
quit
 MainWindow, 198

r
 EmbColor_, 126
r_value
 Parameter_, 236
radians
 embroidermodder.h, 367
 embroidery.h, 464
 functions.c, 582
radius
 EmbCircle_, 125
 EmbEllipse_, 129
 EmbRect_, 140
rapidFireEnabled
 CmdPromptInput, 104
rapidMoveActive
 View, 354
read100
 embroidery_internal.h, 499
 format_100.c, 532
read10o
 embroidery_internal.h, 499
 format_10o.c, 533
read_configuration
 mainwindow-settings.cpp, 405
read_hoop
 format_jef.c, 550
read_settings
 embroidermodder.h, 367
 mainwindow.cpp, 409
read_string_setting
 mainwindow.cpp, 409
readArt
 embroidery_internal.h, 499
 format_art.c, 533
readBmc
 embroidery_internal.h, 499
 format_bmc.c, 534
readBro
 embroidery_internal.h, 499
 format_bro.c, 534
readCnd
 embroidery_internal.h, 499
 format_cnd.c, 535
readCol
 embroidery_internal.h, 499
 format_col.c, 536
readCsd
 embroidery_internal.h, 499
 format_csd.c, 537
readCsv
 embroidery_internal.h, 499
 format_csv.c, 538
readDat
 embroidery_internal.h, 499
 format_dat.c, 538
readDem
 embroidery_internal.h, 499
 format_dem.c, 539
readDescriptions
 embroidery_internal.h, 500
 format_pes.c, 559
readDsb
 embroidery_internal.h, 500
 format_dsb.c, 540
readDst
 embroidery_internal.h, 500
 format_dst.c, 542
readDsz
 embroidery_internal.h, 500
 format_dsz.c, 542
readDxf
 embroidery_internal.h, 500
 format_dxf.c, 543
readEdr
 embroidery_internal.h, 500
 format_edr.c, 543
readEmd
 embroidery_internal.h, 500
 format_emd.c, 544
reader_state
 EmbFormatList_, 130
readExp
 embroidery_internal.h, 500
 format_exp.c, 544
readExy
 embroidery_internal.h, 500
 format_exy.c, 545
readEys
 embroidery_internal.h, 500
 format_eys.c, 545
readFeatherPatterns
 embroidery_internal.h, 501
 format_pes.c, 559
readFullSector
 embroidery_internal.h, 501
 main.c, 599
readFxy
 embroidery_internal.h, 501
 format_fxy.c, 546
readGc
 embroidery_internal.h, 501

format_gc.c, 546
readGnc
 embroidery_internal.h, 501
 format_gnc.c, 547
readGt
 embroidery_internal.h, 501
 format_gt.c, 547
readHoopName
 embroidery_internal.h, 501
 format_pes.c, 559
readHus
 embroidery_internal.h, 501
 format_hus.c, 549
readImageString
 embroidery_internal.h, 501
 format_pes.c, 560
readInb
 embroidery_internal.h, 502
 format_inb.c, 549
readInf
 embroidery_internal.h, 502
 format_inf.c, 549
readJef
 embroidery_internal.h, 502
 format_jef.c, 551
readKsm
 embroidery_internal.h, 502
 format_ksm.c, 551
readLine
 format_dxf.c, 543
readMax
 embroidery_internal.h, 502
 format_max.c, 552
readMit
 embroidery_internal.h, 502
 format_mit.c, 552
readMotifPatterns
 embroidery_internal.h, 502
 format_pes.c, 560
readNew
 embroidery_internal.h, 502
 format_new.c, 553
readNextSector
 embroidery_internal.h, 502
 main.c, 599
readOfm
 embroidery_internal.h, 502
 format_ofm.c, 554
readPcd
 embroidery_internal.h, 503
 format_pcd.c, 555
readPcm
 embroidery_internal.h, 503
 format_pcm.c, 555
readPcq
 embroidery_internal.h, 503
 format_pcq.c, 556
readPcs
 embroidery_internal.h, 503
 format_pcs.c, 556
readPec
 embroidery_internal.h, 503
 format_pec.c, 557
readPecStitches
 embroidery_internal.h, 503
 format_pec.c, 557
readPel
 embroidery_internal.h, 503
 format_pel.c, 558
readPem
 embroidery_internal.h, 503
 format_pem.c, 558
readPes
 embroidery_internal.h, 504
 format_pes.c, 560
readPESHeaderV10
 embroidery_internal.h, 504
 format_pes.c, 560
readPESHeaderV5
 embroidery_internal.h, 504
 format_pes.c, 560
readPESHeaderV6
 embroidery_internal.h, 504
 format_pes.c, 560
readPESHeaderV7
 embroidery_internal.h, 504
 format_pes.c, 560
readPESHeaderV8
 embroidery_internal.h, 504
 format_pes.c, 560
readPESHeaderV9
 embroidery_internal.h, 504
 format_pes.c, 560
readPhb
 embroidery_internal.h, 504
 format_phb.c, 561
readPhc
 embroidery_internal.h, 504
 format_phc.c, 562
readPlt
 embroidery_internal.h, 504
 format_plt.c, 562
readProgrammableFills
 embroidery_internal.h, 505
 format_pes.c, 560
readRgb
 embroidery_internal.h, 505
 format_rgb.c, 563
readSettings
 MainWindow, 199
readSew
 embroidery_internal.h, 505
 format_sew.c, 563
readShv
 embroidery_internal.h, 505
 format_shv.c, 564

readSst
 embroidery_internal.h, 505
 format_sst.c, 564
readStx
 embroidery_internal.h, 505
 format_stx.c, 565
readSvg
 embroidery_internal.h, 505
 format_svg.c, 566
readT01
 embroidery_internal.h, 505
 format_t01.c, 567
readT09
 embroidery_internal.h, 505
 format_t09.c, 567
readTap
 embroidery_internal.h, 505
 format_tap.c, 568
readThr
 embroidery_internal.h, 506
 format_thr.c, 568
readThreads
 embroidery_internal.h, 506
 format_pes.c, 560
readTxt
 embroidery_internal.h, 506
 format_txt.c, 569
readU00
 embroidery_internal.h, 506
 format_u00.c, 569
readU01
 embroidery_internal.h, 506
 format_u01.c, 570
readVip
 embroidery_internal.h, 506
 format_vip.c, 571
readVp3
 embroidery_internal.h, 506
 format_vp3.c, 572
readXxx
 embroidery_internal.h, 506
 format_xxx.c, 573
readZsk
 embroidery_internal.h, 506
 format_zsk.c, 574
real_render
 EmbView_, 148
realRender
 BaseObject, 76
recalculateLimits
 View, 348
recentMenu
 MainWindow, 209
recentMenuAboutToShow
 MainWindow, 199
rect
 BaseObject, 76
 EmbGeometry_, 132
rect.c
 embRect_area, 584
 embRect_init, 584
RectObject, 270
 ~RectObject, 273
 allGripPoints, 273
 gripEdit, 273
 init, 274
 mouseSnapPoint, 274
 objectArea, 274
 objectBottomLeft, 274
 objectBottomRight, 274
 objectHeight, 274
 objectPos, 274
 objectSavePath, 274
 objectTopLeft, 275
 objectTopRight, 275
 objectWidth, 275
 paint, 275
 RectObject, 273
 setObjectRect, 275
 Type, 273
 type, 275
 updatePath, 275
 updateRubber, 275
 vulcanize, 275
RED_TERM_COLOR
 embroidery_internal.h, 488
redo
 MainWindow, 199
 UndoableAddCommand, 331
 UndoableDeleteCommand, 332
 UndoableGripEditCommand, 333
 UndoableMirrorCommand, 334
 UndoableMoveCommand, 335
 UndoableNavCommand, 336
 UndoableRotateCommand, 337
 UndoableScaleCommand, 339
 UndoEditor, 340
redoPressed
 CmdPrompt, 90
 CmdPromptInput, 103
redoText
 UndoEditor, 340
rejectChanges
 Settings_Dialog, 306
releasePoint
 View, 354
releaseResizeHistory
 CmdPromptSplitter, 106
releaseY
 CmdPromptHandle, 95
repeatAction
 View, 348
report
 embroidery.h, 464
reserved
 ThredExtension_, 328

ThredHeader_, 328
 reserved1
 _bcf_file_header, 55
 reserved2
 _bcf_file_header, 55
 RESET_TERM_COLOR
 embroidery_internal.h, 488
 resizeEvent
 MainWindow, 199
 resizeHistory
 CmdPromptHistory, 97
 resizeTheHistory
 CmdPrompt, 90
 reverse_byte_order
 encoding.c, 523
 rgb, 16, 505, 563
 right
 _vp3Hoop, 56
 EmbRect_, 141
 hoop_padding, 150
 right2
 _vp3Hoop, 57
 rightBrush
 SelectBox, 284
 rightBrushColor
 SelectBox, 284
 rightPen
 SelectBox, 284
 rightPenColor
 SelectBox, 284
 rightSiblingId
 _bcf_directory_entry, 50
 RobisonAnton_Polyester
 embroidery.h, 447
 RobisonAnton_Rayon
 embroidery.h, 447
 rotate
 UndoableRotateCommand, 337
 ROTATE_MODE_NORMAL
 embroidermodder.h, 366
 ROTATE_MODE_REFERENCE
 embroidermodder.h, 366
 rotate_vector
 object-arc.cpp, 411
 rotateAction
 View, 348
 rotateSelected
 View, 348
 rotation
 EmbEllipse_, 129
 EmbRect_, 141
 UiObject_, 330
 roundToMultiple
 View, 349
 rubber_mode
 EmbView_, 148
 rubberRoomList
 View, 354
 ruler_color
 Settings_, 292
 ruler_metric
 Settings_, 292
 ruler_mode
 EmbView_, 149
 ruler_pixel_size
 Settings_, 292
 ruler_show_on_load
 Settings_, 292
 ruler_width
 Settings_, 292
 rulerColor
 View, 354
 rulerMetric
 View, 354
 rulerPixelSize
 View, 354
 rules
 fill.c, 527
 LSYSTEM, 167
 run_script
 MainWindow, 199
 run_script_file
 MainWindow, 200
 runCommand
 CmdPrompt, 90
 CmdPromptInput, 103
 MainWindow, 200
 runCommandClick
 MainWindow, 200
 runCommandContext
 MainWindow, 200
 runCommandMain
 MainWindow, 200
 runCommandMove
 MainWindow, 200
 runCommandPrompt
 MainWindow, 200
 running
 Settings_, 292
 s_value
 Parameter_, 236
 safe_free
 embroidery_internal.h, 506
 formats.c, 531
 save
 ImageWidget, 158
 SaveObject, 281
 save_points_to_pattern
 fill.c, 527
 saveasfile
 MainWindow, 201
 saveBMC
 MdiWindow, 231
 saveFile
 MdiWindow, 232
 savefile

MainWindow, 201
saveHistory
 CmdPrompt, 90
SaveObject, 276
 ~SaveObject, 277
 addArc, 277
 addBlock, 277
 addCircle, 277
 addDimAligned, 278
 addDimAngular, 278
 addDimArcLength, 278
 addDimDiameter, 279
 addDimLeader, 279
 addDimLinear, 279
 addDimOrdinate, 279
 addDimRadius, 279
 addEllipse, 279
 addEllipseArc, 279
 addGrid, 280
 addHatch, 280
 addImage, 280
 addInfiniteLine, 280
 addLine, 280
 addPath, 280
 addPoint, 280
 addPolygon, 280
 addPolyline, 281
 addRay, 281
 addRectangle, 281
 addSlot, 281
 addSpline, 281
 addTextMulti, 281
 addTextSingle, 281
 formatType, 282
 gscene, 282
 save, 281
 SaveObject, 277
 toPolyline, 282
scale
 EmbView_, 149
 UiObject_, 330
SCALE_MODE_NORMAL
 embroidermodder.h, 366
SCALE_MODE_REFERENCE
 embroidermodder.h, 366
scaleAction
 View, 349
scaleSelected
 View, 349
sceneGripPoint
 View, 354
sceneMousePoint
 View, 354
sceneMovePoint
 View, 354
scenePressPoint
 View, 354
sceneReleasePoint
 View, 354
script
 Action__, 58
second
 EmbTime_, 145
sectionName
 StxThread_, 320
sectorShift
 _bcf_file_header, 55
sectorSize
 _bcf_file_difat, 52
 main.c, 599
seekToSector
 main.c, 599
selectable
 UiObject_, 330
selectAll
 MainWindow, 201
 View, 349
selectAllPressed
 CmdPrompt, 91
 CmdPromptInput, 103
SelectBox, 282
 alpha, 284
 boxDir, 284
 dirBrush, 284
 dirPen, 284
 forceRepaint, 283
 leftBrush, 284
 leftBrushColor, 284
 leftPen, 284
 leftPenColor, 284
 paintEvent, 283
 rightBrush, 284
 rightBrushColor, 284
 rightPen, 284
 rightPenColor, 284
 SelectBox, 283
 setColors, 283
 setDirection, 284
selectBox
 View, 354
selected
 EmbView_, 149
selectedItemList
 PropertyEditor, 267
selectingActive
 View, 354
selection_coolgrip_color
 Settings_, 292
selection_grip_size
 Settings_, 292
selection_hotgrip_color
 Settings_, 292
selection_mode_pickadd
 Settings_, 292
selection_mode_pickdrag
 Settings_, 292

selection_mode_pickfirst
 Settings_, 292
 selection_pickbox_size
 Settings_, 292
 selectionChanged
 View, 349
 sendCloseMdiWin
 MdiWindow, 232
 SEQUIN
 embroidery.h, 447
 set_dst_variable
 format_dst.c, 542
 set_object_color
 arc.c, 579
 setBackgroundColor
 MdiArea, 223
 View, 349
 setBackgroundLogo
 MdiArea, 223
 setBackgroundTexture
 MdiArea, 223
 setColors
 SelectBox, 283
 setCornerButton
 View, 349
 setCrossHairColor
 View, 349
 setCrossHairSize
 View, 349
 setCurrentColor
 MdiWindow, 232
 setCurrentFile
 MdiWindow, 232
 setCurrentLayer
 MdiWindow, 232
 setCurrentLineType
 MdiWindow, 233
 setCurrentLineWeight
 MdiWindow, 233
 setCurrentText
 CmdPrompt, 91
 setDirection
 SelectBox, 284
 setGridColor
 View, 349
 setHistory
 CmdPrompt, 91
 setLine
 BaseObject, 77
 setMainWin
 Application, 60
 setMouseCoord
 StatusBar, 316
 setObjectArea
 CircleObject, 84
 setObjectCenter
 BaseObject, 77
 setObjectCenterX
 BaseObject, 77
 setObjectCenterY
 BaseObject, 77
 setObjectCircumference
 CircleObject, 84
 setObjectColor
 BaseObject, 77
 setObjectColorRGB
 BaseObject, 77
 setObjectDiameter
 CircleObject, 84
 setObjectDiameterMajor
 EllipseObject, 119
 setObjectDiameterMinor
 EllipseObject, 120
 setObjectEndAngle
 ArcObject, 70
 setObjectEndPoint
 ArcObject, 70, 71
 setObjectEndPoint1
 DimLeaderObject, 113
 LineObject, 165
 setObjectEndPoint2
 DimLeaderObject, 113
 LineObject, 166
 setObjectLineType
 BaseObject, 77
 setObjectLineWeight
 BaseObject, 77
 setObjectMidPoint
 ArcObject, 71
 setObjectPath
 BaseObject, 78
 setObjectPos
 PathObject, 240
 PointObject, 244, 245
 PolygonObject, 249
 PolylineObject, 255
 TextSingleObject, 324, 325
 setObjectRadius
 ArcObject, 71
 CircleObject, 84
 setObjectRadiusMajor
 EllipseObject, 120
 setObjectRadiusMinor
 EllipseObject, 120
 setObjectRect
 ImageObject, 156
 RectObject, 275
 setObjectRubberMode
 BaseObject, 78
 setObjectRubberPoint
 BaseObject, 78
 setObjectRubberText
 BaseObject, 78
 setObjectSize
 EllipseObject, 120
 setObjectStartAngle

ArcObject, 71
setObjectStartPoint
 ArcObject, 71
setObjectText
 TextSingleObject, 325
setObjectTextBackward
 TextSingleObject, 325
setObjectTextBold
 TextSingleObject, 325
setObjectTextFont
 TextSingleObject, 325
setObjectTextItalic
 TextSingleObject, 325
setObjectTextJustify
 TextSingleObject, 325
setObjectTextOverline
 TextSingleObject, 325
setObjectTextSize
 TextSingleObject, 325
setObjectTextStrikeOut
 TextSingleObject, 325
setObjectTextStyle
 TextSingleObject, 325
setObjectTextUnderline
 TextSingleObject, 325
setObjectTextUpsideDown
 TextSingleObject, 325
setObjectX
 PathObject, 240
 PointObject, 245
 PolygonObject, 250
 PolylineObject, 255
 TextSingleObject, 326
setObjectX1
 DimLeaderObject, 113
 LineObject, 166
setObjectX2
 DimLeaderObject, 113
 LineObject, 166
setObjectY
 PathObject, 240
 PointObject, 245
 PolygonObject, 250
 PolylineObject, 255
 TextSingleObject, 326
setObjectY1
 DimLeaderObject, 113
 LineObject, 166
setObjectY2
 DimLeaderObject, 113
 LineObject, 166
setPrefix
 CmdPrompt, 91
setPromptBackgroundColor
 CmdPrompt, 91
setPromptFontFamily
 CmdPrompt, 91
setPromptFontSize
 CmdPrompt, 91
setPromptFontStyle
 CmdPrompt, 91
setPromptTextColor
 CmdPrompt, 91
setRect
 BaseObject, 78
setRubberMode
 View, 349
setRubberPoint
 View, 349
setRubberText
 View, 349
setRulerColor
 View, 350
setSelectBoxColors
 View, 350
setSelectedItems
 PropertyEditor, 265
setShiftPressed
 MainWindow, 201
setShiftReleased
 MainWindow, 201
setTextAngle
 MainWindow, 201
setTextBold
 MainWindow, 201
setTextFont
 MainWindow, 201
setTextItalic
 MainWindow, 201
setTextOverline
 MainWindow, 201
setTextSize
 MainWindow, 202
setTextStrikeOut
 MainWindow, 202
setTextUnderline
 MainWindow, 202
Settings
 embroidermodder.h, 361
settings
 embroidermodder.h, 368
 mainwindow.cpp, 410
Settings_
 assets_dir, 287
 current_directory, 287
 debug_mode, 287
 display_bg_color, 287
 display_crosshair_color, 287
 display_crosshair_percent, 287
 display_renderhint_aa, 287
 display_renderhint_high_aa, 287
 display_renderhint_noncosmetic, 287
 display_renderhint_smooth_pix, 288
 display_renderhint_text_aa, 288
 display_scrollbar_widget_num, 288
 display_selectbox_alpha, 288

display_selectbox_left_color, 288
 display_selectbox_left_fill, 288
 display_selectbox_right_color, 288
 display_selectbox_right_fill, 288
 display_show_scrollbars, 288
 display_units, 288
 display_use_opengl, 288
 display_zoomscale_in, 288
 display_zoomscale_out, 288
 general_check_for_updates, 288
 general_current_tip, 288
 general_mdi_bg_color, 288
 general_mdi_bg_logo, 288
 general_mdi_bg_texture, 288
 general_system_help_browser, 289
 grid_center, 289
 grid_center_on_origin, 289
 grid_color, 289
 grid_color_match_crosshair, 289
 grid_load_from_file, 289
 grid_show_on_load, 289
 grid_show_origin, 289
 grid_size, 289
 grid_size_radius, 289
 grid_spacing, 289
 grid_spacing_angle, 289
 grid_spacing_radius, 289
 grid_type, 289
 icon_size, 289
 icon_theme, 289
 language, 289
 lwt_default_lwt, 289
 lwt_real_render, 290
 lwt_show_lwt, 290
 major_tick_seperation, 290
 mdi_bg_use_color, 290
 mdi_bg_use_logo, 290
 mdi_bg_use_texture, 290
 menu_action, 290
 needle_speed, 290
 opensave_custom_filter, 290
 opensave_open_format, 290
 opensave_open_thumbnail, 290
 opensave_recent_directory, 290
 opensave_recent_list_of_files, 290
 opensave_recent_max_files, 290
 opensave_save_format, 290
 opensave_save_thumbnail, 290
 opensave_trim_dst_num_jumps, 290
 pattern_index, 290
 printing_default_device, 291
 printing_disable_bg, 291
 printing_use_last_device, 291
 qsnap_aperture_size, 291
 qsnap_apparent, 291
 qsnap_center, 291
 qsnap_enabled, 291
 qsnap_endpoint, 291
 qsnap_extension, 291
 qsnap_insertion, 291
 qsnap_intersection, 291
 qsnap_locator_color, 291
 qsnap_locator_size, 291
 qsnap_midpoint, 291
 qsnap_nearest, 291
 qsnap_node, 291
 qsnap_parallel, 291
 qsnap_perpendicular, 291
 qsnap_quadrant, 292
 qsnap_tangent, 292
 ruler_color, 292
 ruler_metric, 292
 ruler_pixel_size, 292
 ruler_show_on_load, 292
 ruler_width, 292
 running, 292
 selection_coolgrip_color, 292
 selection_grip_size, 292
 selection_hotgrip_color, 292
 selection_mode_pickadd, 292
 selection_mode_pickdrag, 292
 selection_mode_pickfirst, 292
 selection_pickbox_size, 292
 shift_held, 292
 shine_color, 292
 show_about_dialog, 292
 show_details_dialog, 293
 show_editor, 293
 show_open_file_dialog, 293
 show_settings_editor, 293
 stitch_time, 293
 testing, 293
 text_angle, 293
 text_font, 293
 text_size, 293
 text_style_bold, 293
 text_style_italic, 293
 text_style_overline, 293
 text_style_strikeout, 293
 text_style_underline, 293
 texture_list, 293
 tick_depth, 293
 ticks_color, 293
 tip_of_the_day, 293
 to_open, 294
 use_translation, 294
 version, 294
 zoomInLimit, 294
 zoomOutLimit, 294
 Settings_Dialog, 294
 ~Settings_Dialog, 299
 accept_display_bg_color, 308
 accept_display_crosshair_color, 308
 accept_display_selectbox_left_color, 308
 accept_display_selectbox_left_fill, 308
 accept_display_selectbox_right_color, 308

accept_display_selectbox_right_fill, 308
accept_general_mdi_bg_color, 308
accept_general_mdi_bg_logo, 308
accept_general_mdi_bg_texture, 308
accept_grid_color, 308
accept_prompt_bg_color, 309
accept_prompt_text_color, 309
accept_ruler_color, 309
acceptChanges, 299
addColorsToComboBox, 299
buttonBox, 309
buttonCustomFilterClearAll, 299
buttonCustomFilterClearAllClicked, 299
buttonCustomFilterSelectAll, 299
buttonCustomFilterSelectAllClicked, 299
buttonQSnapClearAll, 300
buttonQSnapClearAllClicked, 300
buttonQSnapSelectAll, 300
buttonQSnapSelectAllClicked, 300
checkBoxCustomFilterStateChanged, 300
checkBoxDisableBGStateChanged, 300
checkBoxGeneralMdiBGUseColorStateChanged,
 300
checkBoxGeneralMdiBGUseLogoStateChanged,
 300
checkBoxGeneralMdiBGUseTextureStateChanged,
 300
checkBoxGridCenterOnOriginStateChanged, 300
checkBoxGridColumnMatchCrossHairStateChanged,
 300
checkBoxGridLoadFromFileStateChanged, 300
checkBoxGridShowOnLoadStateChanged, 300
checkBoxGridShowOriginStateChanged, 301
checkBoxLwtRealRenderStateChanged, 301
checkBoxLwtShowLwtStateChanged, 301
checkBoxPromptSaveHistoryAsHtmlStateChanged,
 301
checkBoxPromptSaveHistoryStateChanged, 301
checkBoxQSnapApparentStateChanged, 301
checkBoxQSnapCenterStateChanged, 301
checkBoxQSnapEndPointStateChanged, 301
checkBoxQSnapExtensionStateChanged, 301
checkBoxQSnapInsertionStateChanged, 301
checkBoxQSnapIntersectionStateChanged, 301
checkBoxQSnapMidPointStateChanged, 301
checkBoxQSnapNearestStateChanged, 301
checkBoxQSnapNodeStateChanged, 302
checkBoxQSnapParallelStateChanged, 302
checkBoxQSnapPerpendicularStateChanged, 302
checkBoxQSnapQuadrantStateChanged, 302
checkBoxQSnapTangentStateChanged, 302
checkBoxRenderHintAAStateChanged, 302
checkBoxRenderHintHighAAStateChanged, 302
checkBoxRenderHintNonCosmeticStateChanged,
 302
checkBoxRenderHintSmoothPixStateChanged,
 302
checkBoxRenderHintTextAAStateChanged, 302
checkBoxRulerShowOnLoadStateChanged, 302
checkBoxSelectionModePickAddStateChanged,
 302
checkBoxSelectionModePickDragStateChanged,
 302
checkBoxSelectionModePickFirstStateChanged,
 303
checkBoxShowScrollBarsStateChanged, 303
checkBoxTipOfTheDayStateChanged, 303
checkBoxUseOpenGLStateChanged, 303
chooseDisplayBackgroundColor, 303
chooseDisplayCrossHairColor, 303
chooseDisplaySelectBoxLeftColor, 303
chooseDisplaySelectBoxLeftFill, 303
chooseDisplaySelectBoxRightColor, 303
chooseDisplaySelectBoxRightFill, 303
chooseGeneralMdiBackgroundColor, 303
chooseGeneralMdiBackgroundLogo, 303
chooseGeneralMdiBackgroundTexture, 303
chooseGridColor, 304
choosePromptBackgroundColor, 304
choosePromptTextColor, 304
chooseRulerColor, 304
comboBoxGridTypeCurrentIndexChanged, 304
comboBoxIconSizeCurrentIndexChanged, 304
comboBoxIconThemeCurrentIndexChanged, 304
comboBoxLanguageCurrentIndexChanged, 304
comboBoxPromptFontFamilyCurrentIndexChanged,
 304
comboBoxPromptFontStyleCurrentIndexChanged,
 304
comboBoxQSnapLocatorColorCurrentIndex-
 Changed, 304
comboBoxRulerMetricCurrentIndexChanged, 304
comboBoxScrollBarWidgetCurrentIndexChanged,
 304
comboBoxSelectionCoolGripColorCurrentIndex-
 Changed, 305
comboBoxSelectionHotGripColorCurrentIndex-
 Changed, 305
createTabDisplay, 305
createTabFilePaths, 305
createTabGeneral, 305
createTabGridRuler, 305
createTabLineWeight, 305
createTabOpenSave, 305
createTabOrthoPolar, 305
createTabPrinting, 305
createTabPrompt, 305
createTabQuickSnap, 305
createTabQuickTrack, 305
createTabSelection, 305
createTabSnap, 305
currentDisplayBackgroundColorChanged, 305
currentDisplayCrossHairColorChanged, 306
currentDisplaySelectBoxLeftColorChanged, 306
currentDisplaySelectBoxLeftFillChanged, 306
currentDisplaySelectBoxRightColorChanged, 306

currentDisplaySelectBoxRightFillChanged, 306
currentGeneralMdiBackgroundColorChanged, 306
currentGridColorChanged, 306
currentPromptBackgroundColorChanged, 306
currentPromptTextColorChanged, 306
currentRulerColorChanged, 306
dialog_display_bg_color, 309
dialog_display_crosshair_color, 309
dialog_display_crosshair_percent, 309
dialog_display_renderhint_aa, 309
dialog_display_renderhint_high_aa, 309
dialog_display_renderhint_noncosmetic, 309
dialog_display_renderhint_smooth_pix, 309
dialog_display_renderhint_text_aa, 309
dialog_display_scrollbar_widget_num, 309
dialog_display_selectbox_alpha, 309
dialog_display_selectbox_left_color, 309
dialog_display_selectbox_left_fill, 309
dialog_display_selectbox_right_color, 309
dialog_display_selectbox_right_fill, 309
dialog_display_show_scrollbars, 310
dialog_display_units, 310
dialog_display_use_opengl, 310
dialog_display_zoomscale_in, 310
dialog_display_zoomscale_out, 310
dialog_general_icon_size, 310
dialog_general_icon_theme, 310
dialog_general_language, 310
dialog_general_mdi_bg_color, 310
dialog_general_mdi_bg_logo, 310
dialog_general_mdi_bg_texture, 310
dialog_general_mdi_bg_use_color, 310
dialog_general_mdi_bg_use_logo, 310
dialog_general_mdi_bg_use_texture, 310
dialog_general_system_help_browser, 310
dialog_general_tip_of_the_day, 310
dialog_grid_center_on_origin, 310
dialog_grid_center_x, 310
dialog_grid_center_y, 311
dialog_grid_color, 311
dialog_grid_color_match_crosshair, 311
dialog_grid_load_from_file, 311
dialog_grid_show_on_load, 311
dialog_grid_show_origin, 311
dialog_grid_size_radius, 311
dialog_grid_size_x, 311
dialog_grid_size_y, 311
dialog_grid_spacing_angle, 311
dialog_grid_spacing_radius, 311
dialog_grid_spacing_x, 311
dialog_grid_spacing_y, 311
dialog_grid_type, 311
dialog_lwt_default_lwt, 311
dialog_lwt_real_render, 311
dialog_lwt_show_lwt, 311
dialog_opensave_custom_filter, 311
dialog_opensave_open_format, 312
dialog_opensave_open_thumbnail, 312
dialog_opensave_recent_max_files, 312
dialog_opensave_save_format, 312
dialog_opensave_save_thumbnail, 312
dialog_opensave_trim_dst_num_jumps, 312
dialog_printing_default_device, 312
dialog_printing_disable_bg, 312
dialog_printing_use_last_device, 312
dialog_prompt_bg_color, 312
dialog_prompt_font_family, 312
dialog_prompt_font_size, 312
dialog_prompt_font_style, 312
dialog_prompt_save_history, 312
dialog_prompt_save_history_as_html, 312
dialog_prompt_save_history_filename, 312
dialog_prompt_text_color, 312
dialog_qsnap_aperture_size, 312
dialog_qsnap_apparent, 313
dialog_qsnap_center, 313
dialog_qsnap_enabled, 313
dialog_qsnap_endpoint, 313
dialog_qsnap_extension, 313
dialog_qsnap_insertion, 313
dialog_qsnap_intersection, 313
dialog_qsnap_locator_color, 313
dialog_qsnap_locator_size, 313
dialog_qsnap_midpoint, 313
dialog_qsnap_nearest, 313
dialog_qsnap_node, 313
dialog_qsnap_parallel, 313
dialog_qsnap_perpendicular, 313
dialog_qsnap_quadrant, 313
dialog_qsnap_tangent, 313
dialog_ruler_color, 313
dialog_ruler_metric, 313
dialog_ruler_pixel_size, 314
dialog_ruler_show_on_load, 314
dialog_selection_coolgrip_color, 314
dialog_selection_grip_size, 314
dialog_selection_hotgrip_color, 314
dialog_selection_mode_pickadd, 314
dialog_selection_mode_pickdrag, 314
dialog_selection_mode_pickfirst, 314
dialog_selection_pickbox_size, 314
mainWin, 314
preview_display_bg_color, 314
preview_display_crosshair_color, 314
preview_display_selectbox_alpha, 314
preview_display_selectbox_left_color, 314
preview_display_selectbox_left_fill, 314
preview_display_selectbox_right_color, 314
preview_display_selectbox_right_fill, 314
preview_display_show_scrollbars, 314
preview_general_mdi_bg_color, 315
preview_general_mdi_bg_use_color, 315
preview_general_mdi_bg_use_logo, 315
preview_general_mdi_bg_use_texture, 315
preview_grid_color, 315
preview_lwt_real_render, 315

preview_lwt_show_lwt, 315
preview_prompt_bg_color, 315
preview_prompt_font_family, 315
preview_prompt_font_size, 315
preview_prompt_font_style, 315
preview_prompt_text_color, 315
preview_ruler_color, 315
rejectChanges, 306
Settings_Dialog, 299
sliderQSnapApertureSizeValueChanged, 306
sliderQSnapLocatorSizeValueChanged, 306
sliderSelectionGripSizeValueChanged, 307
sliderSelectionPickBoxSizeValueChanged, 307
spinBoxDisplaySelectBoxAlphaValueChanged,
 307
spinBoxGridCenterXValueChanged, 307
spinBoxGridCenterYValueChanged, 307
spinBoxGridSizeRadiusValueChanged, 307
spinBoxGridSizeXValueChanged, 307
spinBoxGridSizeYValueChanged, 307
spinBoxGridSpacingAngleValueChanged, 307
spinBoxGridSpacingRadiusValueChanged, 307
spinBoxGridSpacingXValueChanged, 307
spinBoxGridSpacingYValueChanged, 307
spinBoxPromptFontSizeValueChanged, 307
spinBoxRecentMaxFilesValueChanged, 308
spinBoxRulerPixelSizeValueChanged, 308
spinBoxTrimDstNumJumpsValueChanged, 308
spinBoxZoomScaleInValueChanged, 308
spinBoxZoomScaleOutValueChanged, 308
tabWidget, 315
settings_display_bg_color
 MainWindow, 209
settings_display_crosshair_color
 MainWindow, 209
settings_display_crosshair_percent
 MainWindow, 209
settings_display_renderhint_aa
 MainWindow, 209
settings_display_renderhint_high_aa
 MainWindow, 209
settings_display_renderhint_noncosmetic
 MainWindow, 210
settings_display_renderhint_smooth_pix
 MainWindow, 210
settings_display_renderhint_text_aa
 MainWindow, 210
settings_display_scrollbar_widget_num
 MainWindow, 210
settings_display_selectbox_alpha
 MainWindow, 210
settings_display_selectbox_left_color
 MainWindow, 210
settings_display_selectbox_left_fill
 MainWindow, 210
settings_display_selectbox_right_color
 MainWindow, 210
settings_display_selectbox_right_fill
 MainWindow, 210
settings_display_show_scrollbars
 MainWindow, 210
settings_display_units
 MainWindow, 210
settings_display_use_opengl
 MainWindow, 211
settings_display_zoomscale_in
 MainWindow, 211
settings_display_zoomscale_out
 MainWindow, 211
settings_general_check_for_updates
 MainWindow, 211
settings_general_current_tip
 MainWindow, 211
settings_general_icon_size
 MainWindow, 211
settings_general_icon_theme
 MainWindow, 211
settings_general_language
 MainWindow, 211
settings_general_mdi_bg_color
 MainWindow, 211
settings_general_mdi_bg_logo
 MainWindow, 211
settings_general_mdi_bg_texture
 MainWindow, 211
settings_general_mdi_bg_use_color
 MainWindow, 212
settings_general_mdi_bg_use_logo
 MainWindow, 212
settings_general_mdi_bg_use_texture
 MainWindow, 212
settings_general_system_help_browser
 MainWindow, 212
settings_general_tip_of_the_day
 MainWindow, 212
settings_grid_center_on_origin
 MainWindow, 212
settings_grid_center_x
 MainWindow, 212
settings_grid_center_y
 MainWindow, 212
settings_grid_color
 MainWindow, 212
settings_grid_color_match_crosshair
 MainWindow, 212
settings_grid_load_from_file
 MainWindow, 212
settings_grid_show_on_load
 MainWindow, 213
settings_grid_show_origin
 MainWindow, 213
settings_grid_size_radius
 MainWindow, 213
settings_grid_size_x
 MainWindow, 213
settings_grid_size_y

MainWindow, 213
settings_grid_spacing_angle
 MainWindow, 213
settings_grid_spacing_radius
 MainWindow, 213
settings_grid_spacing_x
 MainWindow, 213
settings_grid_spacing_y
 MainWindow, 213
settings_grid_type
 MainWindow, 213
settings_lwt_default_lwt
 MainWindow, 213
settings_lwt_real_render
 MainWindow, 214
settings_lwt_show_lwt
 MainWindow, 214
settings_opensave_custom_filter
 MainWindow, 214
settings_opensave_open_format
 MainWindow, 214
settings_opensave_open_thumbnail
 MainWindow, 214
settings_opensave_recent_directory
 MainWindow, 214
settings_opensave_recent_list_of_files
 MainWindow, 214
settings_opensave_recent_max_files
 MainWindow, 214
settings_opensave_save_format
 MainWindow, 214
settings_opensave_save_thumbnail
 MainWindow, 214
settings_opensave_trim_dst_num_jumps
 MainWindow, 214
settings_printing_default_device
 MainWindow, 215
settings_printing_disable_bg
 MainWindow, 215
settings_printing_use_last_device
 MainWindow, 215
settings_prompt_bg_color
 MainWindow, 215
settings_prompt_font_family
 MainWindow, 215
settings_prompt_font_size
 MainWindow, 215
settings_prompt_font_style
 MainWindow, 215
settings_prompt_save_history
 MainWindow, 215
settings_prompt_save_history_as_html
 MainWindow, 215
settings_prompt_save_history_filename
 MainWindow, 215
settings_prompt_text_color
 MainWindow, 215
settings_qsnap_aperture_size
 MainWindow, 216
settings_qsnap_apparent
 MainWindow, 216
settings_qsnap_center
 MainWindow, 216
settings_qsnap_enabled
 MainWindow, 216
settings_qsnap_endpoint
 MainWindow, 216
settings_qsnap_extension
 MainWindow, 216
settings_qsnap_insertion
 MainWindow, 216
settings_qsnap_intersection
 MainWindow, 216
settings_qsnap_locator_color
 MainWindow, 216
settings_qsnap_locator_size
 MainWindow, 216
settings_qsnap_midpoint
 MainWindow, 216
settings_qsnap_nearest
 MainWindow, 217
settings_qsnap_node
 MainWindow, 217
settings_qsnap_parallel
 MainWindow, 217
settings_qsnap_perpendicular
 MainWindow, 217
settings_qsnap_quadrant
 MainWindow, 217
settings_qsnap_tangent
 MainWindow, 217
settings_ruler_color
 MainWindow, 217
settings_ruler_metric
 MainWindow, 217
settings_ruler_pixel_size
 MainWindow, 217
settings_ruler_show_on_load
 MainWindow, 217
settings_selection_coolgrip_color
 MainWindow, 217
settings_selection_grip_size
 MainWindow, 218
settings_selection_hotgrip_color
 MainWindow, 218
settings_selection_mode_pickadd
 MainWindow, 218
settings_selection_mode_pickdrag
 MainWindow, 218
settings_selection_mode_pickfirst
 MainWindow, 218
settings_selection_pickbox_size
 MainWindow, 218
settings_text_angle
 MainWindow, 218
settings_text_font

MainWindow, 218
settings_text_size
 MainWindow, 218
settings_text_style_bold
 MainWindow, 218
settings_text_style_italic
 MainWindow, 218
settings_text_style_overline
 MainWindow, 219
settings_text_style_strikeout
 MainWindow, 219
settings_text_style_underline
 MainWindow, 219
settingsDialog
 MainWindow, 202
SettingsDir
 mainwindow-settings.cpp, 405
settingsGrid
 StatusBarButton, 318
settingsLwt
 StatusBarButton, 318
settingsMenu
 MainWindow, 219
settingsOrtho
 StatusBarButton, 318
SettingsPath
 mainwindow-settings.cpp, 405
settingsPolar
 StatusBarButton, 318
settingsPrompt
 MainWindow, 202
settingsQSnap
 StatusBarButton, 318
settingsQTrack
 StatusBarButton, 318
settingsRuler
 StatusBarButton, 318
settingsSnap
 StatusBarButton, 318
setUndoCleanIcon
 MainWindow, 202
setViewBackgroundColor
 MdiWindow, 233
setViewCrossHairColor
 MdiWindow, 233
setViewGridColor
 MdiWindow, 233
setViewRulerColor
 MdiWindow, 233
setViewSelectBoxColors
 MdiWindow, 233
sew, 16, 563
sewDecode
 format_sew.c, 563
shape
 BaseObject, 78
shift_hold
 Settings_, 292
shiftKeyPressedState
 MainWindow, 219
shiftPressed
 CmdPrompt, 91
 CmdPromptInput, 103
shiftReleased
 CmdPrompt, 92
 CmdPromptInput, 103
shine_color
 Settings_, 292
shortcut
 Action__, 58
show_about_dialog
 Settings_, 292
show_details_dialog
 Settings_, 293
show_editor
 Settings_, 293
show_open_file_dialog
 Settings_, 293
show_settings_editor
 Settings_, 293
showGroups
 PropertyEditor, 265
showOneType
 PropertyEditor, 265
showScrollBars
 View, 350
showSettings
 CmdPrompt, 92
 CmdPromptInput, 103
showViewScrollBars
 MdiWindow, 233
shv, 16, 564
shv_thread
 embroidery.h, 447
shvDecode
 format_shv.c, 564
shvDecodeShort
 format_shv.c, 564
shvThreadCount
 embroidery.h, 466
 thread-color.c, 607
shvThreads
 embroidery.h, 466
 thread-color.c, 607
side1
 EmbSatinOutline_, 141
side2
 EmbSatinOutline_, 141
Sierra_Expanded, 501, 546
Sigma_Polyester
 embroidery.h, 447
signalMapper
 PropertyEditor, 268
signature
 _bcf_file_header, 55
sigVersion

ThredHeader_, 328
 simulate
 EmbView_, 149
 simulation_start
 EmbView_, 149
 Singer, 536, 574
 SINGLE_LINE_TEXT_MODE_JUSTIFY
 embroidermodder.h, 366
 SINGLE_LINE_TEXT_MODE_JUSTIFY_
 mainwindow.cpp, 410
 SINGLE_LINE_TEXT_MODE_RAPID
 embroidermodder.h, 366
 SINGLE_LINE_TEXT_MODE_RAPID_
 mainwindow.cpp, 410
 SINGLE_LINE_TEXT_MODE_SETFONT
 embroidermodder.h, 366
 SINGLE_LINE_TEXT_MODE_SETFONT_
 mainwindow.cpp, 410
 SINGLE_LINE_TEXT_MODE_SETGEOM
 embroidermodder.h, 366
 SINGLE_LINE_TEXT_MODE_SETGEOM_
 mainwindow.cpp, 410
 sizeHint
 MdiWindow, 233
 sizeOfChainingEntryAtEndOfDifatSector
 main.c, 600
 sizeOfDifatEntry
 main.c, 600
 sizeOfDirectoryEntry
 main.c, 600
 sizeOfFatEntry
 main.c, 600
 sliderQSnapApertureSizeValueChanged
 Settings_Dialog, 306
 sliderQSnapLocatorSizeValueChanged
 Settings_Dialog, 306
 sliderSelectionGripSizeValueChanged
 Settings_Dialog, 307
 sliderSelectionPickBoxSizeValueChanged
 Settings_Dialog, 307
 snap_mode
 EmbView_, 149
 SNOWFLAKE_MODE_NUM_POINTS
 embroidermodder.h, 366
 SNOWFLAKE_MODE_XSCALE
 embroidermodder.h, 366
 SNOWFLAKE_MODE_YSCALE
 embroidermodder.h, 366
 someInt
 SubDescriptor_, 320
 someNum
 SubDescriptor_, 320
 someOtherInt
 SubDescriptor_, 320
 SPARE_RUBBER_OFF
 embroidermodder.h, 366
 SPARE_RUBBER_PATH
 embroidermodder.h, 366
 SPARE_RUBBER_POLYGON
 embroidermodder.h, 366
 SPARE_RUBBER_POLYLINE
 embroidermodder.h, 366
 SPARE_RUBBER_VALUES
 embroidermodder.h, 366
 spareRubber
 View, 350
 spareRubberList
 View, 354
 spinBoxDisplaySelectBoxAlphaValueChanged
 Settings_Dialog, 307
 spinBoxGridCenterXValueChanged
 Settings_Dialog, 307
 spinBoxGridCenterYValueChanged
 Settings_Dialog, 307
 spinBoxGridSizeRadiusValueChanged
 Settings_Dialog, 307
 spinBoxGridSizeXValueChanged
 Settings_Dialog, 307
 spinBoxGridSizeYValueChanged
 Settings_Dialog, 307
 spinBoxGridSpacingAngleValueChanged
 Settings_Dialog, 307
 spinBoxGridSpacingRadiusValueChanged
 Settings_Dialog, 307
 spinBoxGridSpacingXValueChanged
 Settings_Dialog, 307
 spinBoxGridSpacingYValueChanged
 Settings_Dialog, 307
 spinBoxPromptFontSizeValueChanged
 Settings_Dialog, 307
 spinBoxRecentMaxFilesValueChanged
 Settings_Dialog, 308
 spinBoxRulerPixelSizeValueChanged
 Settings_Dialog, 308
 spinBoxTrimDstNumJumpsValueChanged
 Settings_Dialog, 308
 spinBoxZoomScaleInValueChanged
 Settings_Dialog, 308
 spinBoxZoomScaleOutValueChanged
 Settings_Dialog, 308
 spline
 EmbGeometry_, 132
 sst, 16, 505, 565
 STAR_MODE_CENTER_PT
 embroidermodder.h, 366
 STAR_MODE_CENTER_PT_
 mainwindow.cpp, 410
 STAR_MODE_NUM_POINTS
 embroidermodder.h, 366
 STAR_MODE_NUM_POINTS_
 mainwindow.cpp, 410
 STAR_MODE_RAD_INNER
 embroidermodder.h, 366
 STAR_MODE_RAD_INNER_
 mainwindow.cpp, 410
 STAR_MODE_RAD_OUTER

embroidermodder.h, 366
STAR_MODE_RAD_OUTER_mainwindow.cpp, 410
start
 EmbArc_, 122
 EmbBezier_, 124
 EmbLine_, 135
startBlinking
 CmdPrompt, 92
startCommand
 CmdPrompt, 92
 CmdPromptInput, 103
startGripping
 View, 350
startingSectorLocation
 _bcf_directory_entry, 50
startResizeHistory
 CmdPromptHistory, 97
startResizingTheHistory
 CmdPrompt, 92
stateBits
 _bcf_directory_entry, 50
StatusBar, 315
 setMouseCoord, 316
 StatusBar, 316
 statusBarGridButton, 316
 statusBarLwtButton, 316
 statusBarMouseCoord, 316
 statusBarOrthoButton, 316
 statusBarPolarButton, 316
 statusBarQSnapButton, 316
 statusBarQTrackButton, 316
 statusBarRulerButton, 316
 statusBarSnapButton, 317
statusbar
 MainWindow, 219
 StatusBarButton, 319
StatusBarButton, 317
 contextMenuEvent, 318
 disableLwt, 318
 disableReal, 318
 enableLwt, 318
 enableReal, 318
 mainWin, 319
 settingsGrid, 318
 settingsLwt, 318
 settingsOrtho, 318
 settingsPolar, 318
 settingsQSnap, 318
 settingsQTrack, 318
 settingsRuler, 318
 settingsSnap, 318
 statusbar, 319
 StatusBarButton, 317
 toggleGrid, 318
 toggleLwt, 319
 toggleOrtho, 319
 togglePolar, 319
toggleQSnap, 319
toggleQTrack, 319
toggleRuler, 319
toggleSnap, 319
statusBarGridButton
 StatusBar, 316
statusBarLwtButton
 StatusBar, 316
statusBarMouseCoord
 StatusBar, 316
statusBarOrthoButton
 StatusBar, 316
statusBarPolarButton
 StatusBar, 316
statusBarQSnapButton
 StatusBar, 316
statusBarQTrackButton
 StatusBar, 316
statusBarRulerButton
 StatusBar, 316
statusBarSnapButton
 StatusBar, 317
statustip
 Action__, 59
stitch
 EmbArray_, 123
 EmbGeometry_, 132
stitch_list
 EmbPattern_, 138
stitch_time
 Settings_, 293
stitchesJump
 EmbDetailsDialog, 128
stitchesReal
 EmbDetailsDialog, 128
stitchesTotal
 EmbDetailsDialog, 128
stitchesTrim
 EmbDetailsDialog, 128
stitchGranularity
 ThredExtension_, 328
STOP
 embroidery.h, 447
stopBlinking
 CmdPrompt, 92
 CmdPromptInput, 103
stopGripping
 View, 350
stopResizeHistory
 CmdPromptHistory, 97
stopResizingTheHistory
 CmdPrompt, 92
streamSize
 _bcf_directory_entry, 50
streamSizeHigh
 _bcf_directory_entry, 51
stringInArray
 embroidery_internal.h, 507

main.c, 599
 stringVal
 VipHeader_, 356
 stub_implement
 MainWindow, 202
 stub_testing
 MainWindow, 202
 stx, 565
 stxColor
 StxThread_, 320
 stxReadThread
 format_stx.c, 565
 StxThread
 embroidery_internal.h, 489
 StxThread_, 319
 colorCode, 320
 colorName, 320
 sectionName, 320
 stxColor, 320
 subDescriptors, 320
 styleHash
 CmdPrompt, 93
 SubDescriptor
 embroidery_internal.h, 490
 SubDescriptor_, 320
 colorCode, 320
 colorName, 320
 someInt, 320
 someNum, 320
 someOtherInt, 320
 subDescriptors
 StxThread_, 320
 subPathList
 TextSingleObject, 326
 Sulky_Rayon
 embroidery.h, 447
 Sunstar, 505, 565
 svg, 16, 566
 SVG_ATTRIBUTE
 embroidery_internal.h, 488
 SVG_CATCH_ALL
 embroidery_internal.h, 488
 SVG_Colors
 embroidery.h, 447
 SVG_CREATOR_EMBROIDERMODDER
 embroidery_internal.h, 488
 SVG_CREATOR_ILLUSTRATOR
 embroidery_internal.h, 488
 SVG_CREATOR_INKSCAPE
 embroidery_internal.h, 488
 SVG_CREATOR_NULL
 embroidery_internal.h, 488
 SVG_ELEMENT
 embroidery_internal.h, 488
 SVG_EXPECT_ATTRIBUTE
 embroidery_internal.h, 488
 SVG_EXPECT_ELEMENT
 embroidery_internal.h, 489
 SVG_EXPECT_NULL
 embroidery_internal.h, 489
 SVG_EXPECT_VALUE
 embroidery_internal.h, 489
 SVG_MEDIA_PROPERTY
 embroidery_internal.h, 489
 SVG_NULL
 embroidery_internal.h, 489
 SVG_PROPERTY
 embroidery_internal.h, 489
 SvgAttribute
 embroidery_internal.h, 490
 SvgAttribute_, 321
 name, 321
 value, 321
 svgCreator
 format_svg.c, 566
 svgExpect
 format_svg.c, 566
 svgMultiValue
 format_svg.c, 566
 t01, 505, 567
 t09, 505, 567
 table
 Huffman, 151
 table_width
 Huffman, 151
 tabPressed
 CmdPrompt, 92
 CmdPromptInput, 103
 tabWidget
 Settings_Dialog, 315
 Tajima, 540
 tap, 16, 568
 tempArcObj
 PropertyEditor, 268
 tempBaseObj
 View, 354
 tempBlockObj
 PropertyEditor, 268
 tempCircleObj
 PropertyEditor, 268
 tempDimAlignedObj
 PropertyEditor, 268
 tempDimAngularObj
 PropertyEditor, 268
 tempDimArcLenObj
 PropertyEditor, 268
 tempDimDiamObj
 PropertyEditor, 268
 tempDimLeaderObj
 PropertyEditor, 268
 tempDimLinearObj
 PropertyEditor, 268
 tempDimOrdObj
 PropertyEditor, 268
 tempDimRadiusObj
 PropertyEditor, 269

tempEllipseArcObj
 PropertyEditor, 269
tempEllipseObj
 PropertyEditor, 269
tempHatchObj
 PropertyEditor, 269
templImageObj
 PropertyEditor, 269
tempInfLineObj
 PropertyEditor, 269
tempLineObj
 PropertyEditor, 269
tempPathObj
 PropertyEditor, 269
tempPointObj
 PropertyEditor, 269
tempPolygonObj
 PropertyEditor, 269
tempPolylineObj
 PropertyEditor, 269
tempRayObj
 PropertyEditor, 270
tempRectObj
 PropertyEditor, 270
tempSplineObj
 PropertyEditor, 270
tempTextMultiObj
 PropertyEditor, 270
tempTextSingleObj
 PropertyEditor, 270
testEmbCircle
 embroidery_internal.h, 507
testEmbCircle_2
 embroidery_internal.h, 507
testEmbFormat
 embroidery_internal.h, 507
testGeomArc
 embroidery_internal.h, 507
testing
 Settings_, 293
testMain
 embroidery.h, 464
testTangentPoints
 embroidery_internal.h, 507
testThreadColor
 embroidery_internal.h, 507
text
 EmbTextMulti_, 143
 EmbTextSingle_, 144
 UiObject_, 331
text.c
 textSingle_gripEdit, 584
 textSingle_mouseSnapPoint, 584
 textSingle_paint, 585
 textSingle_setJustify, 585
 textSingle_setTextBackward, 585
 textSingle_setTextBold, 585
 textSingle_setTextFont, 585
 textSingle_setTextItalic, 585
 textSingle_setTextOverline, 585
 textSingle_setTextSize, 585
 textSingle_setTextStrikeOut, 585
 textSingle_setTextStyle, 585
 textSingle_setTextUnderline, 585
 textSingle_setTextUpsideDown, 585
 textSingle_updateRubber, 585
text_angle
 EmbView_, 149
 Settings_, 293
text_font
 EmbView_, 149
 Settings_, 293
text_size
 EmbView_, 149
 Settings_, 293
text_style_bold
 EmbView_, 149
 Settings_, 293
text_style_italic
 EmbView_, 149
 Settings_, 293
text_style_overline
 EmbView_, 150
 Settings_, 293
text_style_strikeout
 EmbView_, 150
 Settings_, 293
text_style_underline
 EmbView_, 150
 Settings_, 293
textAngle
 MainWindow, 202
textBold
 MainWindow, 203
textFont
 MainWindow, 203
 UiObject_, 331
textFontSelector
 MainWindow, 219
textFontSelectorCurrentFontChanged
 MainWindow, 203
textHeight
 UiObject_, 331
textItalic
 MainWindow, 203
textJustify
 UiObject_, 331
textOverline
 MainWindow, 203
textRotation
 UiObject_, 331
textSingle_gripEdit
 text.c, 584
textSingle_mouseSnapPoint
 text.c, 584
textSingle_paint

text.c, 585
 textSingle_setJustify
 text.c, 585
 textSingle_setTextBackward
 text.c, 585
 textSingle_setTextBold
 text.c, 585
 textSingle_setTextFont
 text.c, 585
 textSingle_setTextItalic
 text.c, 585
 textSingle_setTextOverline
 text.c, 585
 textSingle_setTextSize
 text.c, 585
 textSingle_setTextStrikeOut
 text.c, 585
 textSingle_setTextStyle
 text.c, 585
 textSingle_setTextUnderline
 text.c, 585
 textSingle_setTextUpsideDown
 text.c, 585
 textSingle_updateRubber
 text.c, 585
 TextSingleObject, 321
 ~TextSingleObject, 324
 allGripPoints, 324
 gripEdit, 324
 init, 324
 mouseSnapPoint, 324
 objectPos, 324
 objectSavePathList, 324
 objectTextJustifyList, 324
 objectX, 324
 objectY, 324
 objText, 326
 objTextBackward, 326
 objTextBold, 326
 objTextFont, 326
 objTextItalic, 326
 objTextJustify, 326
 objTextOverline, 326
 objTextPath, 326
 objTextSize, 326
 objTextStrikeOut, 327
 objTextUnderline, 327
 objTextUpsideDown, 327
 paint, 324
 setObjectPos, 324, 325
 setObjectText, 325
 setObjectTextBackward, 325
 setObjectTextBold, 325
 setObjectTextFont, 325
 setObjectTextItalic, 325
 setObjectTextJustify, 325
 setObjectTextOverline, 325
 setObjectTextSize, 325
 setObjectTextStrikeOut, 325
 setObjectTextStyle, 325
 setObjectTextUnderline, 325
 setObjectTextUpsideDown, 325
 setObjectX, 326
 setObjectY, 326
 subPathList, 326
 TextSingleObject, 323
 Type, 323
 type, 326
 updateRubber, 326
 vulcanize, 326
 textSize
 MainWindow, 203
 textSizeSelector
 MainWindow, 219
 textSizeSelectorIndexChanged
 MainWindow, 203
 textStrikeOut
 MainWindow, 203
 textUnderline
 MainWindow, 203
 texture_list
 Settings_, 293
 thr, 506, 569
 thread
 EmbArray_, 123
 EmbGeometry_, 132
 thread-color.c
 _dxsetColorTable, 606
 brand_codes, 606
 brand_codes_files, 606
 husThreads, 606
 jefThreads, 606
 pcmThreads, 606
 pecThreadCount, 606
 pecThreads, 607
 shvThreadCount, 607
 shvThreads, 607
 threadColor, 606
 threadColorName, 606
 threadColorNum, 606
 thread_color
 embroidery.h, 450
 thread_color_
 327
 hex_code, 327
 manufacturer_code, 327
 name, 327
 thread_list
 EmbPattern_, 138
 ThreadArt_Polyester
 embroidery.h, 447
 ThreadArt_Rayon
 embroidery.h, 447
 threadColor
 embroidery.h, 465
 thread-color.c, 606
 threadColorName

embroidery.h, 465
thread-color.c, 606
threadColorNum
 embroidery.h, 465
 thread-color.c, 606
ThreaDelight_Polyester
 embroidery.h, 447
threadLength
 _vp3Hoop, 57
ThreadWorks, 506, 569
ThredExtension
 embroidery_internal.h, 490
ThredExtension_-, 327
 auxFormat, 327
 creatorName, 328
 hoopX, 328
 hoopY, 328
 modifierName, 328
 reserved, 328
 stitchGranularity, 328
ThredHeader
 embroidery_internal.h, 490
ThredHeader_-, 328
 hoopSize, 328
 length, 328
 numStiches, 328
 reserved, 328
 sigVersion, 328
threshold_method
 fill.c, 527
Tick
 DimLeaderObject, 110
tick_depth
 Settings_-, 293
ticks_color
 Settings_-, 293
tile
 MdiArea, 224
tip_of_the_day
 Settings_-, 293
tipOfDay
 MainWindow, 203
tmpHeight
 CmdPromptHistory, 97
to_EmbVector
 embroidermodder.h, 367
to_open
 Settings_-, 294
to_QPointF
 embroidermodder.h, 367
to_string_vector
 mainwindow-settings.cpp, 405
toCenter
 UndoableNavCommand, 337
Todo
 mainwindow.cpp, 409
toggleGrid
 MainWindow, 203
 StatusBarButton, 318
 View, 350
toggleLwt
 MainWindow, 204
 StatusBarButton, 319
 View, 350
toggleOrtho
 StatusBarButton, 319
 View, 350
togglePickAddMode
 PropertyEditor, 265
togglePolar
 StatusBarButton, 319
 View, 350
toggleQSnap
 StatusBarButton, 319
 View, 350
toggleQTrack
 StatusBarButton, 319
 View, 350
toggleReal
 View, 350
toggleRuler
 MainWindow, 204
 StatusBarButton, 319
 View, 351
toggleSnap
 StatusBarButton, 319
 View, 351
toolbar_name
 Action_-, 59
toolbar_position
 Action_-, 59
toolbarEdit
 MainWindow, 219
toolbarFile
 MainWindow, 219
toolbarHash
 MainWindow, 219
toolbarHelp
 MainWindow, 220
toolbarIcon
 MainWindow, 220
toolbarLayer
 MainWindow, 220
toolbarPan
 MainWindow, 220
toolbarPrompt
 MainWindow, 220
toolbarProperties
 MainWindow, 220
toolbarText
 MainWindow, 220
toolbarView
 MainWindow, 220
toolbarZoom
 MainWindow, 220
toolButtonArcClockwise

property-editor.cpp, 423
toolButtonBlockX
 property-editor.cpp, 423
toolButtonBlockY
 property-editor.cpp, 423
toolButtonCircleArea
 property-editor.cpp, 423
toolButtonCircleCenterX
 property-editor.cpp, 424
toolButtonCircleCenterY
 property-editor.cpp, 424
toolButtonCircleCircumference
 property-editor.cpp, 424
toolButtonCircleDiameter
 property-editor.cpp, 424
toolButtonCircleRadius
 property-editor.cpp, 424
toolButtonEllipseCenterX
 property-editor.cpp, 424
toolButtonEllipseCenterY
 property-editor.cpp, 424
toolButtonEllipseDiameterMajor
 property-editor.cpp, 424
toolButtonEllipseDiameterMinor
 property-editor.cpp, 424
toolButtonEllipseRadiusMajor
 property-editor.cpp, 424
toolButtonEllipseRadiusMinor
 property-editor.cpp, 424
toolButtonImageHeight
 property-editor.cpp, 424
toolButtonImageName
 property-editor.cpp, 424
toolButtonImagePath
 property-editor.cpp, 424
toolButtonImageWidth
 property-editor.cpp, 424
toolButtonImageX
 property-editor.cpp, 424
toolButtonImageY
 property-editor.cpp, 424
toolButtonInfiniteLineVectorX
 property-editor.cpp, 424
toolButtonInfiniteLineVectorY
 property-editor.cpp, 425
toolButtonInfiniteLineX2
 property-editor.cpp, 425
toolButtonInfiniteLineY1
 property-editor.cpp, 425
toolButtonInfiniteLineY2
 property-editor.cpp, 425
toolButtonLineAngle
 property-editor.cpp, 425
toolButtonLineDeltaX
 property-editor.cpp, 425
toolButtonLineDeltaY
 property-editor.cpp, 425
toolButtonLineEndX
 property-editor.cpp, 425
toolButtonLineEndY
 property-editor.cpp, 425
toolButtonLineLength
 property-editor.cpp, 425
toolButtonLineStartX
 property-editor.cpp, 425
toolButtonLineStartY
 property-editor.cpp, 425
toolButtonPathArea
 property-editor.cpp, 425
toolButtonPathClosed
 property-editor.cpp, 425
toolButtonPathLength
 property-editor.cpp, 425
toolButtonPathVertexNum
 property-editor.cpp, 425
toolButtonPathVertexX
 property-editor.cpp, 425
toolButtonPathVertexY
 property-editor.cpp, 425
toolButtonPickAdd
 PropertyEditor, 270
toolButtonPointX
 property-editor.cpp, 426
toolButtonPointY
 property-editor.cpp, 426
toolButtonPolygonCenterX
 property-editor.cpp, 426
toolButtonPolygonCenterY
 property-editor.cpp, 426
toolButtonPolygonDiameterSide
 property-editor.cpp, 426
toolButtonPolygonDiameterVertex
 property-editor.cpp, 426
toolButtonPolygonInteriorAngle
 property-editor.cpp, 426
toolButtonPolygonRadiusSide
 property-editor.cpp, 426
toolButtonPolygonRadiusVertex
 property-editor.cpp, 426
toolButtonPolylineArea
 property-editor.cpp, 426
toolButtonPolylineClosed
 property-editor.cpp, 426
toolButtonPolylineLength
 property-editor.cpp, 426
toolButtonPolylineVertexNum
 property-editor.cpp, 426
toolButtonPolylineVertexX
 property-editor.cpp, 426
toolButtonPolylineVertexY
 property-editor.cpp, 426
toolButtonQSelect
 PropertyEditor, 270
toolButtonRayVectorX
 property-editor.cpp, 426
toolButtonRayVectorY
 property-editor.cpp, 426

property-editor.cpp, 426
toolButtonRayX1
 property-editor.cpp, 426
toolButtonRayX2
 property-editor.cpp, 427
toolButtonRayY1
 property-editor.cpp, 427
toolButtonRayY2
 property-editor.cpp, 427
toolButtonRectangleArea
 property-editor.cpp, 427
toolButtonRectangleCorner1X
 property-editor.cpp, 427
toolButtonRectangleCorner1Y
 property-editor.cpp, 427
toolButtonRectangleCorner2X
 property-editor.cpp, 427
toolButtonRectangleCorner2Y
 property-editor.cpp, 427
toolButtonRectangleCorner3X
 property-editor.cpp, 427
toolButtonRectangleCorner3Y
 property-editor.cpp, 427
toolButtonRectangleCorner4X
 property-editor.cpp, 427
toolButtonRectangleCorner4Y
 property-editor.cpp, 427
toolButtonRectangleHeight
 property-editor.cpp, 427
toolButtonRectangleWidth
 property-editor.cpp, 427
toolButtons
 property-editor.cpp, 427
toolButtonTextMultiX
 property-editor.cpp, 427
toolButtonTextMultiY
 property-editor.cpp, 427
toolButtonTextSingleBackward
 property-editor.cpp, 427
toolButtonTextSingleContents
 property-editor.cpp, 428
toolButtonTextSingleFont
 property-editor.cpp, 428
toolButtonTextSingleHeight
 property-editor.cpp, 428
toolButtonTextSingleJustify
 property-editor.cpp, 428
toolButtonTextSingleRotation
 property-editor.cpp, 428
toolButtonTextSingleUpsideDown
 property-editor.cpp, 428
toolButtonTextSingleX
 property-editor.cpp, 428
toolButtonTextSingleY
 property-editor.cpp, 428
tooltip
 Action__, 59
top
 _vp3Hoop, 57
 EmbRect_, 141
 hoop_padding, 151
top2
 _vp3Hoop, 57
toPolyline
 SaveObject, 282
toTransform
 UndoableNavCommand, 337
Toyota, 532
transactionSignatureNumber
 _bcf_file_header, 55
treeView
 LayerManager, 161
TRIM
 embroidery.h, 447
txt, 506, 569
Type
 ArcObject, 64
 BaseObject, 74
 CircleObject, 82
 DimLeaderObject, 110
 EllipseObject, 117
 ImageObject, 154
 LineObject, 163
 PathObject, 238
 PointObject, 243
 PolygonObject, 248
 PolylineObject, 253
 RectObject, 273
 TextSingleObject, 323
type
 ArcObject, 71
 BaseObject, 78
 CircleObject, 84
 DimLeaderObject, 113
 EllipseObject, 120
 EmbArray_, 123
 EmbFormatList_, 130
 EmbGeometry_, 132
 ImageObject, 156
 LineObject, 166
 PathObject, 240
 PointObject, 245
 PolygonObject, 250
 PolylineObject, 255
 RectObject, 275
 TextSingleObject, 326
 UiObject_, 331
u00, 506, 570
u01, 16, 506, 570
ui_mode
 EmbView_, 150
UiMode
 embroidermodder.h, 366
UiObject
 embroidermodder.h, 362
UiObject_, 329

center, 329
 color, 330
 command, 330
 controlPointLabels, 330
 controlPoints, 330
 firstRun, 330
 fname, 330
 id, 330
 maxPoints, 330
 minPoints, 330
 mode, 330
 n_controlPoints, 330
 numPoints, 330
 object_index, 330
 path_desc, 330
 pattern_index, 330
 rotation, 330
 scale, 330
 selectable, 330
 text, 331
 textFont, 331
 textHeight, 331
 textJustify, 331
 textRotation, 331
 type, 331
undo
 MainWindow, 204
 UndoableAddCommand, 331
 UndoableDeleteCommand, 332
 UndoableGripEditCommand, 333
 UndoableMirrorCommand, 334
 UndoableMoveCommand, 335
 UndoableNavCommand, 336
 UndoableRotateCommand, 338
 UndoableScaleCommand, 339
 UndoEditor, 340
undo_history
 EmbView_, 150
 UndoableAddCommand, 331
 gview, 332
 object, 332
 redo, 331
 undo, 331
 UndoableAddCommand, 331
 UndoableDeleteCommand, 332
 gview, 332
 object, 332
 redo, 332
 undo, 332
 UndoableDeleteCommand, 332
 UndoableGripEditCommand, 333
 after, 333
 before, 333
 gview, 333
 object, 333
 redo, 333
 undo, 333
 UndoableGripEditCommand, 333
 UndoableMirrorCommand, 334
 gview, 334
 mirror, 334
 mirrorLine, 334
 object, 334
 redo, 334
 undo, 334
 UndoableMoveCommand, 335
 dx, 335
 dy, 335
 gview, 335
 object, 335
 redo, 335
 undo, 335
 UndoableMoveCommand, 335
undoableNavCommand, 336
 done, 336
 fromCenter, 336
 fromTransform, 336
 gview, 337
 id, 336
 mergeWith, 336
 navType, 337
 redo, 336
 toCenter, 337
 toTransform, 337
 undo, 336
 UndoableNavCommand, 336
undoableRotateCommand, 337
 angle, 338
 gview, 338
 object, 338
 pivotX, 338
 pivotY, 338
 redo, 337
 rotate, 337
 undo, 338
 UndoableRotateCommand, 337
undoableScaleCommand, 338
 dx, 339
 dy, 339
 factor, 339
 gview, 339
 object, 339
 redo, 339
 undo, 339
 UndoableScaleCommand, 338
undoEditor, 339
 ~UndoEditor, 340
 addStack, 340
 canRedo, 340
 canUndo, 340
 focusWidget, 340
 iconDir, 341
 iconSize, 341
 redo, 340
 redoText, 340

undo, 340
UndoEditor, 340
undoGroup, 341
undoText, 340
undoView, 341
updateCleanIcon, 340
undoGroup
 UndoEditor, 341
UndoHistory
 embroidermodder.h, 362
UndoHistory_, 341
 data, 341
 position, 341
undoPressed
 CmdPrompt, 92
 CmdPromptInput, 103
undoStack
 View, 355
undoText
 UndoEditor, 340
undoView
 UndoEditor, 341
unknown
 VipHeader_, 356
unknown2
 _vp3Hoop, 57
unknown3
 _vp3Hoop, 57
unknown4
 _vp3Hoop, 57
updateAllViewBackgroundColors
 MainWindow, 204
updateAllViewCrossHairColors
 MainWindow, 204
updateAllViewGridColors
 MainWindow, 204
updateAllViewRulerColors
 MainWindow, 204
updateAllViewScrollBars
 MainWindow, 204
updateAllViewSelectBoxColors
 MainWindow, 204
updateArcRect
 ArcObject, 71
updateCleanIcon
 UndoEditor, 340
updateColorLinetypeLineweight
 MdiWindow, 234
updateComboBoxBoolIfVaries
 PropertyEditor, 265
updateComboBoxStrIfVaries
 PropertyEditor, 265
updateCurrentText
 CmdPromptInput, 103
updateFontComboBoxStrIfVaries
 PropertyEditor, 266
updateLeader
 DimLeaderObject, 114
updateLineEditNumIfVaries
 PropertyEditor, 266
updateLineEditStrIfVaries
 PropertyEditor, 266
updateMenuToolbarStatusbar
 MainWindow, 204
updateMouseCoords
 View, 351
updatePath
 ArcObject, 72
 CircleObject, 84
 EllipseObject, 120
 ImageObject, 156
 PathObject, 240
 PolygonObject, 250
 PolylineObject, 255
 RectObject, 275
updatePickAddMode
 MainWindow, 205
updatePickAddModeButton
 PropertyEditor, 266
updateRubber
 ArcObject, 72
 CircleObject, 84
 DimLeaderObject, 114
 EllipseObject, 120
 ImageObject, 156
 LineObject, 166
 PathObject, 240
 PointObject, 245
 PolygonObject, 250
 PolylineObject, 256
 RectObject, 275
 TextSingleObject, 326
updateStyle
 CmdPrompt, 92
upPressed
 CmdPrompt, 92
 CmdPromptInput, 104
usage
 embroidermodder.cpp, 357
use_translation
 Settings_, 294
useBackgroundColor
 MdiArea, 224
useBackgroundLogo
 MdiArea, 224
useBackgroundTexture
 MdiArea, 224
useColor
 MdiArea, 225
useLogo
 MdiArea, 225
useTexture
 MdiArea, 225
validFileFormat
 MainWindow, 205
validRGB

mainwindow.cpp, 409
value
 SvgAttribute_, 321
vector
 EmbGeometry_, 132
vector.c
 embVector_add, 586
 embVector_angle, 586
 embVector_average, 586
 embVector_cross, 586
 embVector_distance, 587
 embVector_dot, 587
 embVector_length, 587
 embVector_multiply, 587
 embVector_normalize, 587
 embVector_relativeX, 587
 embVector_relativeY, 588
 embVector_subtract, 588
 embVector_transpose_product, 588
 embVector_unit, 588
version
 embroidermodder.cpp, 357
 Settings_, 294
View, 341
 ~View, 345
 addObject, 345
 addToRubberRoom, 345
 alignScenePointWithViewPoint, 345
 allowRubber, 345
 allowZoomIn, 345
 allowZoomOut, 345
 center, 345
 centerAt, 345
 clearRubberRoom, 345
 clearSelection, 345
 contextMenuEvent, 345
 copy, 345
 copySelected, 345
 cornerButtonClicked, 346
 createGrid, 346
 createGridIso, 346
 createGridPolar, 346
 createGridRect, 346
 createObjectList, 346
 createOrigin, 346
 createRulerTextPath, 346
 crosshairColor, 352
 crosshairSize, 352
 cut, 346
 cutCopyMousePoint, 352
 deleteObject, 346
 deletePressed, 346
 deleteSelected, 346
 disableMoveRapidFire, 346
 drawBackground, 346
 drawForeground, 346
 enableMoveRapidFire, 347
 enterEvent, 347
 escapePressed, 347
 getUndoStack, 347
 gridColor, 352
 gridPath, 352
 gripBaseObj, 352
 gripColorCool, 352
 gripColorHot, 352
 grippingActive, 352
 gripSize, 352
 gscene, 352
 hashDeletedObjects, 352
 isLwtEnabled, 347
 isRealEnabled, 347
 loadRulerSettings, 347
 mainWin, 352
 mirrorSelected, 347
 mouseDoubleClickEvent, 347
 mouseMoveEvent, 347
 mousePressEvent, 347
 mouseReleaseEvent, 347
 moveAction, 347
 movePoint, 352
 moveSelected, 347
 movingActive, 352
 numSelected, 348
 originPath, 352
 panDistance, 352
 panDown, 348
 panLeft, 348
 panningActive, 353
 panningPointActive, 353
 panningRealTimeActive, 353
 panPoint, 348
 panRealTime, 348
 panRight, 348
 panStart, 348
 panStartX, 353
 panStartY, 353
 panUp, 348
 paste, 348
 pasteDelta, 353
 pasteObjectItemGroup, 353
 pastingActive, 353
 pickBoxSize, 353
 pressPoint, 353
 previewActive, 353
 previewData, 353
 previewMode, 353
 previewObjectItemGroup, 353
 previewObjectList, 353
 previewOff, 348
 previewOn, 348
 previewPoint, 353
 qSnapActive, 353
 qsnapApertureSize, 353
 qsnapLocatorColor, 354
 qsnapLocatorSize, 354
 qSnapToggle, 354

rapidMoveActive, 354
recalculateLimits, 348
releasePoint, 354
repeatAction, 348
rotateAction, 348
rotateSelected, 348
roundToMultiple, 349
rubberRoomList, 354
rulerColor, 354
rulerMetric, 354
rulerPixelSize, 354
scaleAction, 349
scaleSelected, 349
sceneGripPoint, 354
sceneMousePoint, 354
sceneMovePoint, 354
scenePressPoint, 354
sceneReleasePoint, 354
selectAll, 349
selectBox, 354
selectingActive, 354
selectionChanged, 349
setBackgroundColor, 349
setCornerButton, 349
setCrossHairColor, 349
setCrossHairSize, 349
setGridColor, 349
setRubberMode, 349
setRubberPoint, 349
setRubberText, 349
setRulerColor, 350
setSelectBoxColors, 350
showScrollBars, 350
spareRubber, 350
spareRubberList, 354
startGripping, 350
stopGripping, 350
tempBaseObj, 354
toggleGrid, 350
toggleLwt, 350
toggleOrtho, 350
togglePolar, 350
toggleQSnap, 350
toggleQTrack, 350
toggleReal, 350
toggleRuler, 351
toggleSnap, 351
undoStack, 355
updateMouseCoords, 351
View, 344
viewMousePoint, 355
vulcanizeObject, 351
vulcanizeRubberRoom, 351
wheelEvent, 351
willOverflowInt32, 351
willUnderflowInt32, 351
zoomExtents, 351
zoomIn, 351
zoomOut, 351
zoomSelected, 351
zoomToPoint, 351
zoomWindow, 351
zoomWindowActive, 355
view_toolbar
 embroidermodder.h, 368
 mainwindow-toolbars.cpp, 407
viewMenu
 MainWindow, 220
viewMousePoint
 View, 355
vip, 16, 466, 572
vipCompressData
 format_vip.c, 571
vipDecodeByte
 format_vip.c, 571
vipDecodeStitchType
 format_vip.c, 571
vipDecodingTable
 embroidery.h, 466
 format_vip.c, 571
vipDecompressData
 format_vip.c, 571
vipEncodeByte
 format_vip.c, 571
vipEncodeStitchType
 format_vip.c, 571
VipHeader
 embroidery_internal.h, 490
VipHeader_, 355
 attributeOffset, 355
 colorLength, 355
 magicCode, 355
 negativeXHoopSize, 355
 negativeYHoopSize, 355
 numberOfColors, 355
 numberOfStitches, 355
 postitiveXHoopSize, 356
 postitiveYHoopSize, 356
 stringVal, 356
 unknown, 356
 xOffset, 356
 yOffset, 356
vp3, 16, 573
vp3Decode
 format_vp3.c, 572
vp3DecodeInt16
 format_vp3.c, 572
vp3Hoop
 embroidery_internal.h, 490
vp3PatchByteCount
 format_vp3.c, 572
vp3ReadHoopSection
 format_vp3.c, 572
vp3ReadString
 format_vp3.c, 573
vp3WriteString

format_vp3.c, 573
 vp3WriteStringLen
 format_vp3.c, 573
 vulcanize
 ArcObject, 72
 BaseObject, 78
 CircleObject, 84
 DimLeaderObject, 114
 EllipseObject, 120
 ImageObject, 156
 LineObject, 166
 PathObject, 240
 PointObject, 245
 PolygonObject, 250
 PolylineObject, 256
 RectObject, 275
 TextSingleObject, 326
 vulcanizeObject
 View, 351
 vulcanizeRubberRoom
 View, 351

 whatsThisContextHelp
 MainWindow, 205
 wheelEvent
 View, 351
 WHITESPACE
 main.c, 600
 width
 _vp3Hoop, 57
 EmblImage_, 133
 willOverflowInt32
 View, 351
 willUnderflowInt32
 View, 351
 windowMenu
 MainWindow, 220
 windowMenuAboutToShow
 MainWindow, 205
 windowMenuActivated
 MainWindow, 205
 wizardTipOfDay
 MainWindow, 221
 write100
 embroidery_internal.h, 507
 format_100.c, 532
 write10o
 embroidery_internal.h, 507
 format_10o.c, 533
 write_24bit
 embroidery_internal.h, 507
 encoding.c, 523
 main.c, 599
 write_external_color_file
 EmbFormatList_, 130
 write_setting
 mainwindow-settings.cpp, 405
 write_settings
 embroidermodder.h, 367

 writeArt
 embroidery_internal.h, 508
 format_art.c, 533
 writeBmc
 embroidery_internal.h, 508
 format_bmc.c, 534
 writeBro
 embroidery_internal.h, 508
 format_bro.c, 534
 writeCnd
 embroidery_internal.h, 508
 format_cnd.c, 535
 writeCol
 embroidery_internal.h, 508
 format_col.c, 536
 writeCsd
 embroidery_internal.h, 508
 format_csd.c, 537
 writeCsv
 embroidery_internal.h, 508
 format_csv.c, 538
 writeDat
 embroidery_internal.h, 508
 format_dat.c, 538
 writeDem
 embroidery_internal.h, 508
 format_dem.c, 539
 writeDsb
 embroidery_internal.h, 508
 format_dsb.c, 540
 writeDst
 embroidery_internal.h, 508
 format_dst.c, 542
 writeDsz
 embroidery_internal.h, 509
 format_dsz.c, 542
 writeDxf
 embroidery_internal.h, 509
 format_dxf.c, 543
 writeEdr
 embroidery_internal.h, 509
 format_edr.c, 543
 writeEmd
 embroidery_internal.h, 509
 format_emd.c, 544
 writeExp
 embroidery_internal.h, 509
 format_exp.c, 545
 writeExy
 embroidery_internal.h, 509
 format_exy.c, 545
 writeEys
 embroidery_internal.h, 509
 format_eys.c, 546
 writeFxy
 embroidery_internal.h, 509
 format_fxy.c, 546
 writeGc

embroidery_internal.h, 509
format_gc.c, 547
writeGnc
 embroidery_internal.h, 509
 format_gnc.c, 547
writeGt
 embroidery_internal.h, 509
 format_gt.c, 548
writeHus
 embroidery_internal.h, 510
 format_hus.c, 549
writeImage
 format_pec.c, 557
 image.c, 589
writeInb
 embroidery_internal.h, 510
 format_inb.c, 549
writeInf
 embroidery_internal.h, 510
 format_inf.c, 550
writeJef
 embroidery_internal.h, 510
 format_jef.c, 551
writeKsm
 embroidery_internal.h, 510
 format_ksm.c, 551
writeMax
 embroidery_internal.h, 510
 format_max.c, 552
writeMit
 embroidery_internal.h, 510
 format_mit.c, 553
writeNew
 embroidery_internal.h, 510
 format_new.c, 553
writeOfm
 embroidery_internal.h, 510
 format_ofm.c, 554
writePcd
 embroidery_internal.h, 510
 format_pcd.c, 555
writePcm
 embroidery_internal.h, 510
 format_pcm.c, 555
writePcq
 embroidery_internal.h, 511
 format_pcq.c, 556
writePcs
 embroidery_internal.h, 511
 format_pcs.c, 556
writePec
 embroidery_internal.h, 511
 format_pec.c, 557
writePecStitches
 embroidery_internal.h, 511
 format_pec.c, 557
writePel
 embroidery_internal.h, 511
 format_pel.c, 558
writePem
 embroidery_internal.h, 511
 format_pem.c, 558
writePes
 embroidery_internal.h, 511
 format_pes.c, 561
writePhb
 embroidery_internal.h, 511
 format_phb.c, 561
writePhc
 embroidery_internal.h, 511
 format_phc.c, 562
writePlt
 embroidery_internal.h, 511
 format_plt.c, 562
writer_state
 EmbFormatList_, 130
writeRgb
 embroidery_internal.h, 512
 format_rgb.c, 563
writeSettings
 MainWindow, 205
writeSew
 embroidery_internal.h, 512
 format_sew.c, 563
writeShv
 embroidery_internal.h, 512
 format_shv.c, 564
writeSst
 embroidery_internal.h, 512
 format_sst.c, 565
writeStx
 embroidery_internal.h, 512
 format_stx.c, 565
writeSvg
 embroidery_internal.h, 512
 format_svg.c, 566
writeT01
 embroidery_internal.h, 512
 format_t01.c, 567
writeT09
 embroidery_internal.h, 512
 format_t09.c, 567
writeTap
 embroidery_internal.h, 512
 format.tap.c, 568
writeThr
 embroidery_internal.h, 512
 format_thr.c, 569
writeTxt
 embroidery_internal.h, 512
 format_txt.c, 569
writeU00
 embroidery_internal.h, 513
 format_u00.c, 570
writeU01
 embroidery_internal.h, 513

format_u01.c, 570
writeVip
embroidery_internal.h, 513
format_vip.c, 571
writeVp3
embroidery_internal.h, 513
format_vp3.c, 573
writeXxx
embroidery_internal.h, 513
format_xxx.c, 573
writeZsk
embroidery_internal.h, 513
format_zsk.c, 574

x
EmbStitch_, 142
EmbVector_, 146
xOffset
_vp3Hoop, 57
VipHeader_, 356
xxx, 16, 574
xxxDecodeByte
format_xxx.c, 573
xxxEncodeDesign
format_xxx.c, 574
xxxEncodeStitch
format_xxx.c, 574
xxxEncodeStop
format_xxx.c, 574

y
EmbStitch_, 143
EmbVector_, 146
year
EmbTime_, 145
YELLOW_TERM_COLOR
embroidery_internal.h, 489
yOffset
_vp3Hoop, 57
VipHeader_, 356

Z102_Isacord_Polyester
embroidery.h, 447
zoom_toolbar
embroidermodder.h, 368
mainwindow-toolbars.cpp, 407
zoomAll
MainWindow, 206
zoomCenter
MainWindow, 206
zoomDynamic
MainWindow, 206
zoomExtents
MainWindow, 206
View, 351
zoomExtentsAllSubWindows
MdiArea, 224
zoomIn
MainWindow, 206

View, 351
zoomInLimit
Settings_, 294
zoomMenu
MainWindow, 221
zoomOut
MainWindow, 206
View, 351
zoomOutLimit
Settings_, 294
zoomPrevious
MainWindow, 206
zoomRealtime
MainWindow, 206
zoomScale
MainWindow, 206
zoomSelected
MainWindow, 206
View, 351
zoomToPoint
View, 351
zoomWindow
MainWindow, 206
View, 351
zoomWindowActive
View, 355
zsk, 16, 574
ZSK USA, 500, 542, 574