

Embroidermodder

2.0.0 alpha

Generated by Doxygen 1.9.6

1 Overview	1
1.0.1 License	2
2 About	2
2.1 The Embroidermodder Project and Team	2
2.1.1 "Core Development Team"	3
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"	5
2.3 "About"	5
2.3.1 History	5
2.4 Contact us	6
3 Downloads	6
3.1 Alpha Build	6
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	7
4.3 Advanced Features	7
5 Post History	7
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	11
5.2.2 The New Settings System	11
5.2.3 Reducing Reliance on Qt5	11
5.2.4 Palettes	11
5.2.5 Conclusion	12
5.3 June 2022 Backer Update (2022-06-22)	12
5.3.1 Fill Algorithms	12
5.3.2 Working Render Algorithm	12
5.3.3 Timetable	12
5.4 New Website! (2013-09-09)	13
5.5 Crowdfunding Campaign Coming Soon! (2013-09-09)	13
5.6 Our Kickstarter Crowdfunding Campaign is LIVE! (2014-03-17)	13
5.7 Fast Forward (2014-02-13)	14
6 Changelog	14
7 Ideas	14

7.0.1 libembroidery C formats	15
7.0.2 FORMAT READ WRITE NOTES	15
8 Formats	15
8.1 Overview	15
8.1.1 Read/Write Support Levels	15
8.1.2 Table of Format Support Levels	16
8.1.3 Support	17
8.1.4 libembroidery C formats	18
9 Geometry and Algorithms	19
9.1 To Do	19
9.1.1 Development	19
9.1.2 Testing	20
9.1.3 Contributing	20
9.1.4 Embroidermodder Project Coding Standards	20
9.1.5 Code Style	21
9.1.6 Version Control	21
9.1.7 Donations	21
9.1.8 Project Coding Standards	22
9.1.9 Code Style	22
9.1.10 Ideas	23
9.1.11 development	25
9.1.12 Development	25
9.2 Embroiderbot and Libembroidery on Embedded Systems	25
9.2.1 Compatible Boards	25
9.2.2 Arduino Considerations	26
9.2.3 Space	26
9.2.4 Tables	26
9.2.5 Current Pattern Memory Management	26
9.2.6 Special Notes	26
9.2.7 The Assembly Split	26
9.3 The <tt>embroider</tt> Command Line Program	27
9.3.1 Embroider pipeline	27
9.3.2 embroider CLI	27
10 GNU Free Documentation License	27
11 Contributor Covenant Code of Conduct	32
11.1 Our Pledge	32
11.2 Our Standards	32
11.3 Enforcement Responsibilities	32
11.4 Scope	33
11.5 Enforcement	33

11.6 Enforcement Guidelines	33
11.6.1 1. Correction	33
11.6.2 2. Warning	33
11.6.3 3. Temporary Ban	33
11.6.4 4. Permanent Ban	33
11.7 Attribution	33
12 Privacy Policy for Embroidery Viewer	34
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	45
16.1 File List	45
17 Class Documentation	48
17.1 _bcf_directory Struct Reference	48
17.1.1 Detailed Description	49
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	50
17.3.1 Member Data Documentation	50
17.4 _bcf_file_difat Struct Reference	51
17.4.1 Member Data Documentation	51
17.5 _bcf_file_fat Struct Reference	51
17.5.1 Member Data Documentation	51
17.6 _bcf_file_header Struct Reference	52
17.6.1 Detailed Description	52
17.6.2 Member Data Documentation	52
17.7 _vp3Hoop Struct Reference	53
17.7.1 Member Data Documentation	54
17.8 Action__ Struct Reference	55
17.8.1 Detailed Description	55
17.8.2 Member Data Documentation	55
17.9 Application Class Reference	56
17.9.1 Detailed Description	56
17.9.2 Constructor & Destructor Documentation	56

17.9.3 Member Function Documentation	56
17.9.4 Member Data Documentation	57
17.10 ArcObject Class Reference	57
17.10.1 Member Enumeration Documentation	60
17.10.2 Constructor & Destructor Documentation	60
17.10.3 Member Function Documentation	61
17.10.4 Member Data Documentation	67
17.11 BaseObject Class Reference	67
17.11.1 Member Enumeration Documentation	68
17.11.2 Constructor & Destructor Documentation	68
17.11.3 Member Function Documentation	69
17.11.4 Member Data Documentation	72
17.12 CircleObject Class Reference	72
17.12.1 Member Enumeration Documentation	74
17.12.2 Constructor & Destructor Documentation	74
17.12.3 Member Function Documentation	74
17.13 CmdPrompt Class Reference	76
17.13.1 Detailed Description	78
17.13.2 Constructor & Destructor Documentation	78
17.13.3 Member Function Documentation	78
17.13.4 Member Data Documentation	82
17.14 CmdPromptHandle Class Reference	82
17.14.1 Detailed Description	83
17.14.2 Constructor & Destructor Documentation	83
17.14.3 Member Function Documentation	83
17.14.4 Member Data Documentation	83
17.15 CmdPromptHistory Class Reference	84
17.15.1 Detailed Description	84
17.15.2 Constructor & Destructor Documentation	84
17.15.3 Member Function Documentation	84
17.15.4 Member Data Documentation	85
17.16 CmdPromptInput Class Reference	85
17.16.1 Constructor & Destructor Documentation	87
17.16.2 Member Function Documentation	87
17.16.3 Member Data Documentation	90
17.17 CmdPromptSplitter Class Reference	90
17.17.1 Detailed Description	91
17.17.2 Constructor & Destructor Documentation	91
17.17.3 Member Function Documentation	91
17.18 Compress Struct Reference	91
17.18.1 Member Data Documentation	91
17.19 DimLeaderObject Class Reference	92

17.19.1 Member Enumeration Documentation	94
17.19.2 Constructor & Destructor Documentation	95
17.19.3 Member Function Documentation	95
17.19.4 Member Data Documentation	97
17.20 EllipseObject Class Reference	98
17.20.1 Member Enumeration Documentation	100
17.20.2 Constructor & Destructor Documentation	100
17.20.3 Member Function Documentation	100
17.21 EmbAlignedDim_ Struct Reference	102
17.21.1 Member Data Documentation	103
17.22 EmbAngularDim_ Struct Reference	103
17.22.1 Member Data Documentation	103
17.23 EmbArc_ Struct Reference	103
17.23.1 Detailed Description	103
17.23.2 Member Data Documentation	103
17.24 EmbArcLengthDim_ Struct Reference	104
17.24.1 Member Data Documentation	104
17.25 EmbArray_ Struct Reference	104
17.25.1 Member Data Documentation	104
17.26 EmbBezier_ Struct Reference	104
17.26.1 Member Data Documentation	105
17.27 EmbBlock_ Struct Reference	105
17.27.1 Member Data Documentation	105
17.28 EmbCircle_ Struct Reference	105
17.28.1 Member Data Documentation	105
17.29 EmbColor_ Struct Reference	106
17.29.1 Detailed Description	106
17.29.2 Member Data Documentation	106
17.30 EmbDetailsDialog Class Reference	106
17.30.1 Detailed Description	107
17.30.2 Constructor & Destructor Documentation	107
17.30.3 Member Function Documentation	107
17.30.4 Member Data Documentation	107
17.31 EmbDiameterDim_ Struct Reference	108
17.31.1 Member Data Documentation	108
17.32 EmbEllipse_ Struct Reference	108
17.32.1 Member Data Documentation	108
17.33 EmbFormatList_ Struct Reference	108
17.33.1 Member Data Documentation	109
17.34 EmbGeometry_ Struct Reference	109
17.34.1 Member Data Documentation	110
17.35 EmblImage_ Struct Reference	111

17.35.1 Member Data Documentation	111
17.36 EmbInfiniteLine_ Struct Reference	112
17.36.1 Member Data Documentation	112
17.37 EmbLayer_ Struct Reference	112
17.37.1 Member Data Documentation	112
17.38 EmbLeaderDim_ Struct Reference	112
17.38.1 Member Data Documentation	112
17.39 EmbLine_ Struct Reference	112
17.39.1 Member Data Documentation	113
17.40 EmbLinearDim_ Struct Reference	113
17.40.1 Member Data Documentation	113
17.41 EmbOrdinateDim_ Struct Reference	113
17.41.1 Member Data Documentation	113
17.42 EmbPath_ Struct Reference	114
17.42.1 Member Data Documentation	114
17.43 EmbPattern_ Struct Reference	114
17.43.1 Member Data Documentation	114
17.44 EmbPoint_ Struct Reference	115
17.44.1 Member Data Documentation	115
17.45 EmbRadiusDim_ Struct Reference	115
17.45.1 Member Data Documentation	115
17.46 EmbRay_ Struct Reference	116
17.46.1 Member Data Documentation	116
17.47 EmbRect_ Struct Reference	116
17.47.1 Member Data Documentation	116
17.48 EmbSatinOutline_ Struct Reference	117
17.48.1 Member Data Documentation	117
17.49 EmbSpline_ Struct Reference	117
17.49.1 Member Data Documentation	117
17.50 EmbStitch_ Struct Reference	117
17.50.1 Member Data Documentation	117
17.51 EmbTextMulti_ Struct Reference	118
17.51.1 Member Data Documentation	118
17.52 EmbTextSingle_ Struct Reference	118
17.52.1 Member Data Documentation	118
17.53 EmbThread_ Struct Reference	119
17.53.1 Member Data Documentation	119
17.54 EmbTime_ Struct Reference	119
17.54.1 Member Data Documentation	119
17.55 EmbVector_ Struct Reference	120
17.55.1 Detailed Description	120
17.55.2 Member Data Documentation	120

17.56 EmbView_ Struct Reference	120
17.56.1 Detailed Description	121
17.56.2 EmbViews	121
17.56.3 Member Data Documentation	121
17.57 hoop_padding Struct Reference	123
17.57.1 Member Data Documentation	123
17.58 Huffman Struct Reference	123
17.58.1 Member Data Documentation	123
17.59 ImageObject Class Reference	124
17.59.1 Member Enumeration Documentation	125
17.59.2 Constructor & Destructor Documentation	126
17.59.3 Member Function Documentation	126
17.60 ImageWidget Class Reference	127
17.60.1 Detailed Description	128
17.60.2 Constructor & Destructor Documentation	128
17.60.3 Member Function Documentation	128
17.60.4 Member Data Documentation	129
17.61 LayerManager Class Reference	129
17.61.1 Detailed Description	129
17.61.2 Constructor & Destructor Documentation	129
17.61.3 Member Function Documentation	130
17.61.4 Member Data Documentation	130
17.62 LineObject Class Reference	130
17.62.1 Member Enumeration Documentation	132
17.62.2 Constructor & Destructor Documentation	133
17.62.3 Member Function Documentation	133
17.63 LSYSTEM Struct Reference	135
17.63.1 Member Data Documentation	135
17.64 MainWindow Class Reference	136
17.64.1 Detailed Description	145
17.64.2 Constructor & Destructor Documentation	145
17.64.3 Member Function Documentation	145
17.64.4 Member Data Documentation	167
17.65 MdiArea Class Reference	176
17.65.1 Constructor & Destructor Documentation	177
17.65.2 Member Function Documentation	177
17.65.3 Member Data Documentation	179
17.66 MdiWindow Class Reference	179
17.66.1 Constructor & Destructor Documentation	181
17.66.2 Member Function Documentation	181
17.66.3 Member Data Documentation	186
17.67 Parameter_ Struct Reference	187

17.67.1 Member Data Documentation	187
17.68 PathObject Class Reference	187
17.68.1 Member Enumeration Documentation	189
17.68.2 Constructor & Destructor Documentation	189
17.68.3 Member Function Documentation	189
17.68.4 Member Data Documentation	191
17.69 PointObject Class Reference	191
17.69.1 Member Enumeration Documentation	193
17.69.2 Constructor & Destructor Documentation	193
17.69.3 Member Function Documentation	193
17.70 PolygonObject Class Reference	195
17.70.1 Member Enumeration Documentation	196
17.70.2 Constructor & Destructor Documentation	197
17.70.3 Member Function Documentation	197
17.70.4 Member Data Documentation	199
17.71 PolylineObject Class Reference	199
17.71.1 Member Enumeration Documentation	201
17.71.2 Constructor & Destructor Documentation	201
17.71.3 Member Function Documentation	201
17.71.4 Member Data Documentation	203
17.72 PreviewDialog Class Reference	203
17.72.1 Constructor & Destructor Documentation	203
17.72.2 Member Data Documentation	203
17.73 PropertyEditor Class Reference	204
17.73.1 Constructor & Destructor Documentation	206
17.73.2 Member Function Documentation	206
17.73.3 Member Data Documentation	210
17.74 RectObject Class Reference	213
17.74.1 Member Enumeration Documentation	215
17.74.2 Constructor & Destructor Documentation	215
17.74.3 Member Function Documentation	215
17.75 SaveObject Class Reference	217
17.75.1 Constructor & Destructor Documentation	218
17.75.2 Member Function Documentation	218
17.75.3 Member Data Documentation	222
17.76 SelectBox Class Reference	223
17.76.1 Constructor & Destructor Documentation	223
17.76.2 Member Function Documentation	223
17.76.3 Member Data Documentation	224
17.77 Settings_Struct Reference	225
17.77.1 Detailed Description	227
17.77.2 Member Data Documentation	227

17.78 Settings_Dialog Class Reference	234
17.78.1 Constructor & Destructor Documentation	239
17.78.2 Member Function Documentation	239
17.78.3 Member Data Documentation	248
17.79 StatusBar Class Reference	255
17.79.1 Constructor & Destructor Documentation	256
17.79.2 Member Function Documentation	256
17.79.3 Member Data Documentation	256
17.80 StatusBarButton Class Reference	257
17.80.1 Constructor & Destructor Documentation	257
17.80.2 Member Function Documentation	258
17.80.3 Member Data Documentation	259
17.81 StxThread_ Struct Reference	259
17.81.1 Member Data Documentation	260
17.82 SubDescriptor_ Struct Reference	260
17.82.1 Member Data Documentation	260
17.83 SvgAttribute_ Struct Reference	261
17.83.1 Member Data Documentation	261
17.84 TextSingleObject Class Reference	261
17.84.1 Member Enumeration Documentation	263
17.84.2 Constructor & Destructor Documentation	263
17.84.3 Member Function Documentation	264
17.84.4 Member Data Documentation	266
17.85 thread_color_ Struct Reference	267
17.85.1 Member Data Documentation	267
17.86 ThredExtension_ Struct Reference	267
17.86.1 Member Data Documentation	267
17.87 ThredHeader_ Struct Reference	268
17.87.1 Member Data Documentation	268
17.88 UiObject_ Struct Reference	269
17.88.1 Detailed Description	269
17.88.2 Member Data Documentation	269
17.89 UndoableAddCommand Class Reference	271
17.89.1 Constructor & Destructor Documentation	271
17.89.2 Member Function Documentation	271
17.89.3 Member Data Documentation	272
17.90 UndoableDeleteCommand Class Reference	272
17.90.1 Constructor & Destructor Documentation	272
17.90.2 Member Function Documentation	272
17.90.3 Member Data Documentation	272
17.91 UndoableGripEditCommand Class Reference	273
17.91.1 Constructor & Destructor Documentation	273

17.91.2 Member Function Documentation	273
17.91.3 Member Data Documentation	273
17.92 UndoableMirrorCommand Class Reference	274
17.92.1 Constructor & Destructor Documentation	274
17.92.2 Member Function Documentation	274
17.92.3 Member Data Documentation	274
17.93 UndoableMoveCommand Class Reference	275
17.93.1 Constructor & Destructor Documentation	275
17.93.2 Member Function Documentation	275
17.93.3 Member Data Documentation	275
17.94 UndoableNavCommand Class Reference	276
17.94.1 Constructor & Destructor Documentation	276
17.94.2 Member Function Documentation	276
17.94.3 Member Data Documentation	276
17.95 UndoableRotateCommand Class Reference	277
17.95.1 Constructor & Destructor Documentation	277
17.95.2 Member Function Documentation	277
17.95.3 Member Data Documentation	278
17.96 UndoableScaleCommand Class Reference	278
17.96.1 Constructor & Destructor Documentation	278
17.96.2 Member Function Documentation	279
17.96.3 Member Data Documentation	279
17.97 UndoEditor Class Reference	279
17.97.1 Constructor & Destructor Documentation	280
17.97.2 Member Function Documentation	280
17.97.3 Member Data Documentation	280
17.98 UndoHistory_ Struct Reference	281
17.98.1 Detailed Description	281
17.98.2 Member Data Documentation	281
17.99 View Class Reference	281
17.99.1 Constructor & Destructor Documentation	284
17.99.2 Member Function Documentation	285
17.99.3 Member Data Documentation	292
17.100 VipHeader_ Struct Reference	295
17.100.1 Member Data Documentation	295
18 File Documentation	296
18.1 CODE_OF_CONDUCT.md File Reference	296
18.2 embroidermodder2/cmdprompt.cpp File Reference	296
18.3 embroidermodder2/embdetails-dialog.cpp File Reference	296
18.4 embroidermodder2/embroidermodder.cpp File Reference	296
18.4.1 Function Documentation	297

18.4.2 Variable Documentation	297
18.5 embroidermodder2/embroidermodder.h File Reference	297
18.5.1 Detailed Description	301
18.5.2 Typedef Documentation	301
18.5.3 EmbViews	301
18.5.4 Enumeration Type Documentation	302
18.5.5 Function Documentation	307
18.5.6 Variable Documentation	307
18.6 embroidermodder.h	308
18.7 embroidermodder2/imagewidget.cpp File Reference	344
18.8 embroidermodder2/layer-manager.cpp File Reference	344
18.8.1 Detailed Description	344
18.9 embroidermodder2/mainwindow-commands.cpp File Reference	344
18.10 embroidermodder2/mainwindow-menus.cpp File Reference	344
18.11 embroidermodder2/mainwindow-settings.cpp File Reference	344
18.11.1 Function Documentation	345
18.12 embroidermodder2/mainwindow-toolbars.cpp File Reference	345
18.12.1 Function Documentation	345
18.12.2 Variable Documentation	345
18.13 embroidermodder2/mainwindow.cpp File Reference	347
18.13.1 Typedef Documentation	348
18.13.2 Function Documentation	348
18.13.3 Variable Documentation	349
18.14 embroidermodder2/mdiarea.cpp File Reference	351
18.15 embroidermodder2 mdiwindow.cpp File Reference	351
18.16 embroidermodder2/object-arc.cpp File Reference	351
18.16.1 Function Documentation	351
18.17 embroidermodder2/object-base.cpp File Reference	351
18.18 embroidermodder2/object-circle.cpp File Reference	351
18.19 embroidermodder2/object-dimleader.cpp File Reference	351
18.20 embroidermodder2/object-ellipse.cpp File Reference	351
18.21 embroidermodder2/object-image.cpp File Reference	351
18.22 embroidermodder2/object-line.cpp File Reference	352
18.23 embroidermodder2/object-path.cpp File Reference	352
18.24 embroidermodder2/object-point.cpp File Reference	352
18.25 embroidermodder2/object-polygon.cpp File Reference	352
18.26 embroidermodder2/object-polyline.cpp File Reference	352
18.27 embroidermodder2/object-rect.cpp File Reference	352
18.27.1 Detailed Description	352
18.28 embroidermodder2/object-save.cpp File Reference	352
18.29 embroidermodder2/object-textsingle.cpp File Reference	352
18.30 embroidermodder2/preview-dialog.cpp File Reference	352

18.31 embroidermodder2/property-editor.cpp File Reference	352
18.31.1 Variable Documentation	356
18.32 embroidermodder2/README.md File Reference	368
18.33 embroidermodder2/selectbox.cpp File Reference	368
18.34 embroidermodder2/settings-dialog.cpp File Reference	368
18.35 embroidermodder2/statusbar-button.cpp File Reference	368
18.36 embroidermodder2/statusbar.cpp File Reference	368
18.37 embroidermodder2/undo-commands.cpp File Reference	369
18.38 embroidermodder2/undo-editor.cpp File Reference	369
18.38.1 Detailed Description	369
18.39 embroidermodder2/view.cpp File Reference	369
18.39.1 Detailed Description	369
18.40 extern/libembroidery/src/array.c File Reference	369
18.40.1 Function Documentation	369
18.41 extern/libembroidery/src/compress.c File Reference	371
18.41.1 Detailed Description	372
18.41.2 Function Documentation	372
18.41.3 Variable Documentation	373
18.42 extern/libembroidery/src/embroidery.h File Reference	373
18.42.1 Macro Definition Documentation	380
18.42.2 Typedef Documentation	388
18.42.3 Function Documentation	390
18.42.4 Variable Documentation	405
18.43 embroidery.h	406
18.44 extern/libembroidery/src/embroidery_internal.h File Reference	413
18.44.1 Macro Definition Documentation	421
18.44.2 Typedef Documentation	429
18.44.3 Enumeration Type Documentation	430
18.44.4 Function Documentation	431
18.44.5 Variable Documentation	453
18.45 embroidery_internal.h	453
18.46 extern/libembroidery/src/encoding.c File Reference	460
18.46.1 Detailed Description	461
18.46.2 Function Documentation	461
18.47 extern/libembroidery/src/fill.c File Reference	463
18.47.1 Function Documentation	464
18.47.2 Variable Documentation	467
18.48 extern/libembroidery/src/formats.c File Reference	468
18.48.1 Function Documentation	469
18.48.2 Variable Documentation	471
18.49 extern/libembroidery/src/formats/format_100.c File Reference	472
18.49.1 Detailed Description	472

18.49.2 Function Documentation	472
18.50 extern/libembroidery/src/formats/format_10o.c File Reference	472
18.50.1 Detailed Description	472
18.50.2 Function Documentation	473
18.51 extern/libembroidery/src/formats/format_art.c File Reference	473
18.51.1 Detailed Description	473
18.51.2 Function Documentation	473
18.52 extern/libembroidery/src/formats/format_bmc.c File Reference	473
18.52.1 Detailed Description	474
18.52.2 Function Documentation	474
18.53 extern/libembroidery/src/formats/format_bro.c File Reference	474
18.53.1 Detailed Description	474
18.53.2 Function Documentation	474
18.54 extern/libembroidery/src/formats/format_cnd.c File Reference	474
18.54.1 Detailed Description	475
18.54.2 Function Documentation	475
18.55 extern/libembroidery/src/formats/format_col.c File Reference	475
18.55.1 Detailed Description	475
18.55.2 Function Documentation	476
18.56 extern/libembroidery/src/formats/format_csd.c File Reference	476
18.56.1 Detailed Description	476
18.56.2 Macro Definition Documentation	476
18.56.3 Function Documentation	476
18.56.4 Variable Documentation	477
18.57 extern/libembroidery/src/formats/format_csv.c File Reference	477
18.57.1 Detailed Description	478
18.57.2 Function Documentation	478
18.58 extern/libembroidery/src/formats/format_dat.c File Reference	478
18.58.1 Function Documentation	478
18.59 extern/libembroidery/src/formats/format_dem.c File Reference	479
18.59.1 Detailed Description	479
18.59.2 Function Documentation	479
18.60 extern/libembroidery/src/formats/format_dsb.c File Reference	479
18.60.1 Detailed Description	479
18.60.2 Function Documentation	480
18.61 extern/libembroidery/src/formats/format_dst.c File Reference	480
18.61.1 Detailed Description	480
18.61.2 Macro Definition Documentation	481
18.61.3 Function Documentation	481
18.62 extern/libembroidery/src/formats/format_dsz.c File Reference	482
18.62.1 Function Documentation	482
18.63 extern/libembroidery/src/formats/format_dxf.c File Reference	482

18.63.1 Function Documentation	483
18.64 extern/libembroidery/src/formats/format_edr.c File Reference	483
18.64.1 Function Documentation	483
18.65 extern/libembroidery/src/formats/format_emd.c File Reference	484
18.65.1 Detailed Description	484
18.65.2 Function Documentation	484
18.66 extern/libembroidery/src/formats/format_exp.c File Reference	484
18.66.1 Function Documentation	484
18.67 extern/libembroidery/src/formats/format_exy.c File Reference	485
18.67.1 Function Documentation	485
18.68 extern/libembroidery/src/formats/format_eyc.c File Reference	485
18.68.1 Function Documentation	485
18.69 extern/libembroidery/src/formats/format_fxy.c File Reference	486
18.69.1 Function Documentation	486
18.70 extern/libembroidery/src/formats/format_gc.c File Reference	486
18.70.1 Function Documentation	486
18.71 extern/libembroidery/src/formats/format_gnc.c File Reference	487
18.71.1 Function Documentation	487
18.72 extern/libembroidery/src/formats/format_gt.c File Reference	487
18.72.1 Function Documentation	487
18.73 extern/libembroidery/src/formats/format_hus.c File Reference	488
18.73.1 Function Documentation	488
18.74 extern/libembroidery/src/formats/format_inb.c File Reference	489
18.74.1 Function Documentation	489
18.75 extern/libembroidery/src/formats/format_inf.c File Reference	489
18.75.1 Function Documentation	489
18.76 extern/libembroidery/src/formats/format_jef.c File Reference	490
18.76.1 Function Documentation	490
18.77 extern/libembroidery/src/formats/format_ksm.c File Reference	491
18.77.1 Function Documentation	491
18.78 extern/libembroidery/src/formats/format_max.c File Reference	491
18.78.1 Function Documentation	492
18.78.2 Variable Documentation	492
18.79 extern/libembroidery/src/formats/format_mit.c File Reference	492
18.79.1 Function Documentation	492
18.80 extern/libembroidery/src/formats/format_new.c File Reference	493
18.80.1 Function Documentation	493
18.81 extern/libembroidery/src/formats/format_ofm.c File Reference	493
18.81.1 Function Documentation	494
18.82 extern/libembroidery/src/formats/format_pcd.c File Reference	494
18.82.1 Function Documentation	495
18.83 extern/libembroidery/src/formats/format_pcm.c File Reference	495

18.83.1 Function Documentation	495
18.84 extern/libembroidery/src/formats/format_pcq.c File Reference	495
18.84.1 Function Documentation	496
18.85 extern/libembroidery/src/formats/format_pcs.c File Reference	496
18.85.1 Function Documentation	496
18.86 extern/libembroidery/src/formats/format_pec.c File Reference	496
18.86.1 Function Documentation	497
18.87 extern/libembroidery/src/formats/format_pel.c File Reference	498
18.87.1 Function Documentation	498
18.88 extern/libembroidery/src/formats/format_pem.c File Reference	498
18.88.1 Function Documentation	498
18.89 extern/libembroidery/src/formats/format_pes.c File Reference	499
18.89.1 Function Documentation	499
18.89.2 Variable Documentation	501
18.90 extern/libembroidery/src/formats/format_phb.c File Reference	501
18.90.1 Function Documentation	501
18.91 extern/libembroidery/src/formats/format_phc.c File Reference	502
18.91.1 Function Documentation	502
18.92 extern/libembroidery/src/formats/format_plt.c File Reference	502
18.92.1 Function Documentation	502
18.93 extern/libembroidery/src/formats/format_rgb.c File Reference	503
18.93.1 Function Documentation	503
18.94 extern/libembroidery/src/formats/format_sew.c File Reference	503
18.94.1 Function Documentation	503
18.95 extern/libembroidery/src/formats/format_shv.c File Reference	504
18.95.1 Function Documentation	504
18.96 extern/libembroidery/src/formats/format_sst.c File Reference	504
18.96.1 Function Documentation	504
18.97 extern/libembroidery/src/formats/format_stx.c File Reference	505
18.97.1 Function Documentation	505
18.98 extern/libembroidery/src/formats/format_svg.c File Reference	505
18.98.1 Function Documentation	506
18.98.2 Variable Documentation	506
18.99 extern/libembroidery/src/formats/format_t01.c File Reference	507
18.99.1 Function Documentation	507
18.100 extern/libembroidery/src/formats/format_t09.c File Reference	507
18.100.1 Function Documentation	507
18.101 extern/libembroidery/src/formats/format_tap.c File Reference	508
18.101.1 Function Documentation	508
18.102 extern/libembroidery/src/formats/format_thr.c File Reference	508
18.102.1 Function Documentation	508
18.103 extern/libembroidery/src/formats/format_txt.c File Reference	509

18.103.1 Function Documentation	509
18.104 extern/libembroidery/src/formats/format_u00.c File Reference	509
18.104.1 Function Documentation	509
18.105 extern/libembroidery/src/formats/format_u01.c File Reference	510
18.105.1 Function Documentation	510
18.106 extern/libembroidery/src/formats/format_vip.c File Reference	510
18.106.1 Function Documentation	511
18.106.2 Variable Documentation	511
18.107 extern/libembroidery/src/formats/format_vp3.c File Reference	512
18.107.1 Function Documentation	512
18.108 extern/libembroidery/src/formats/format_xxx.c File Reference	513
18.108.1 Function Documentation	513
18.109 extern/libembroidery/src/formats/format_zsk.c File Reference	514
18.109.1 Detailed Description	514
18.109.2 Function Documentation	514
18.110 extern/libembroidery/src/geometry.c File Reference	514
18.110.1 Function Documentation	515
18.111 extern/libembroidery/src/geometry/arc.c File Reference	516
18.111.1 Function Documentation	516
18.112 extern/libembroidery/src/geometry/circle.c File Reference	519
18.112.1 Function Documentation	520
18.113 extern/libembroidery/src/geometry/ellipse.c File Reference	520
18.113.1 Function Documentation	521
18.114 extern/libembroidery/src/geometry/functions.c File Reference	522
18.114.1 Function Documentation	522
18.115 extern/libembroidery/src/geometry/line.c File Reference	523
18.115.1 Function Documentation	523
18.116 extern/libembroidery/src/geometry/path.c File Reference	523
18.117 extern/libembroidery/src/geometry/polygon.c File Reference	523
18.118 extern/libembroidery/src/geometry/polyline.c File Reference	523
18.119 extern/libembroidery/src/geometry/rect.c File Reference	524
18.119.1 Function Documentation	524
18.120 extern/libembroidery/src/geometry/text.c File Reference	524
18.120.1 Function Documentation	524
18.121 extern/libembroidery/src/geometry/vector.c File Reference	526
18.121.1 Function Documentation	526
18.122 extern/libembroidery/src/image.c File Reference	528
18.122.1 Detailed Description	528
18.122.2 Function Documentation	528
18.123 extern/libembroidery/src/main.c File Reference	529
18.123.1 Macro Definition Documentation	531
18.123.2 Function Documentation	534

18.123.3 Variable Documentation	540
18.124 extern/libembroidery/src/pattern.c File Reference	540
18.124.1 Detailed Description	541
18.124.2 Function Documentation	541
18.125 extern/libembroidery/src/thread-color.c File Reference	545
18.125.1 Function Documentation	546
18.125.2 Variable Documentation	546
18.126 privacy_policy.md File Reference	547
Bibliography	548
Index	549

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 seperate 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.

2.3 "About"

2.3.1 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.

2.4 Contact us

For general questions email: [embroidermodder at gmail.com](mailto:embroidermodder@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)! <div class="video"><iframe src="https://www.youtube.com/embed/KqiKfn4lxBk" frameborder="0" allowfullscreen></iframe></div> ← 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 @subpage crowdfunding-1 Crowdfunding Campaign Coming Soon! \author Jonathan https://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 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\]\(https://www.kickstarter.com/projects/redteam316/369640335?token=57f7685e\)](https://www.kickstarter.com/projects/redteam316/369640335?token=57f7685e)

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-arm>

... 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)! [video](https://www.youtube.com/embed/KqiKfn4lxBk) → Jonathan [Permanent link to this article](news0.html::fast-forward).

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. /* TODO: Josh, Review this file 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)

7.0.1 libembroidery C formats

7.0.2 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) 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)

Support for Singer FHE, CHE (Compucon) formats?

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

Format	Ratings	Score
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
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 Support

| FORMAT | READ | WRITE | NOTES | -----|-----|-----|-----| | 100 | 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) || 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) |

Support for Singer FHE, CHE (Compucon) formats?

Todo Josh, Review this file 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)

8.1.4 libembroidery C formats

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)
art			none
bro	YES		read (complete)(maybe figure out detail of header)
rnd			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?)

FORMAT	READ	WRITE	NOTES
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)

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. Should your idea pass this test:

- A new kind of GUI structure it goes in `src/ui.c`.
- If it's something the user can do, make a section of the `actuator` function (which lives in `src/actuator.c`) using the guide "The Actuator's Behaviour".
- Potentially variable data that is global goes in `src/data.c`.
- If the data will not vary declare it as a compiler definition using the "Compiler definitions" section and put it in `src/em2.h`.
- All other C code goes in `src/em2.c`.

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 more code without a payoff in speed of execution or clarity.

9.1.4.4 Conventions Name variables and functions intelligently to minimize the need for comments. It should be immediately obvious what information it represents. Short names such as `\texttt{x}` and `\texttt{y}` are fine when referring to coordinates. Short names such as `\texttt{i}` and `\texttt{j}` are fine when doing loops.

Variable names should be `camelCase`, starting with a lowercase word followed by uppercase word(s). C Functions that attempt to simulate namespacing, should be `\texttt{nameSpace_camelCase}`.

All files and directories shall be lowercase and contain no spaces.

9.1.5 Code Style

Tabs should not be used when indenting. Setup your IDE or text editor to use 4 spaces.

9.1.5.1 Braces For functions: please put each brace on a new line.

```
void function_definition(int argument)
{
    /* code block */
}
```

For control statements: please put the first brace on the same line.

```
if (condition) {
    /* code block */
}
```

Use exceptions sparingly.

Do not use ternary operator (`? :`) in place of if/else.

Do not repeat a variable name that already occurs in an outer scope.

9.1.6 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 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 }`).

9.1.7.1 Format Support Support for Singer FHE, CHE (Compucon) formats?

9.1.8 Project Coding Standards

A basic set of guidelines to use when submitting code.

9.1.8.1 Naming Conventions Name variables and functions intelligently to minimize the need for comments. It should be immediately obvious what information it represents. Short names such as x and y are fine when referring to coordinates. Short names such as i and j are fine when doing loops.

Variable names should be "camelCase", starting with a lowercase word followed by uppercase word(s). C++ Class Names should be "CamelCase", using all uppercase word(s). C Functions that attempt to simulate namespaces, should be "nameSpace_camelCase".

All files and directories shall be lowercase and contain no spaces.

9.1.9 Code Style

Tabs should not be used when indenting. Setup your IDE or text editor to use 4 spaces.

9.1.9.1 Braces For functions: please put each brace on a new line.

```
void function_definition(int argument)
{
}
```

For control statements: please put the first brace on the same line.

```
if (condition) {
}
```

Use exceptions sparingly.

Do not use ternary operator (?:) in place of if/else.

Do not repeat a variable name that already occurs in an outer scope.

9.1.9.2 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...

9.1.9.3 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.
 */
```

9.1.10 Ideas

9.1.10.1 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.

9.1.10.2 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.

9.1.10.3 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.

9.1.10.4 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.

9.1.10.5 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

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

9.1.10.7 Code Style Rather than maintain our own standard for style, please defer to the Python's PEP 7 [4] for C style. If it passes the linters for that we consider it well styled for a pull request.

As for other languages we have no house style other than whatever "major" styles exist, for example Java in Google style [3] would be acceptable. We'll elect specific standards if it becomes an issue.

9.1.10.8 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.

9.1.10.9 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 .

9.1.10.10 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 (`emb←Pattern_flip`).

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

9.1.10.11 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 later `.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

9.1.10.12 features

- Incorporate `#if Oed` 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.

9.1.10.13 Optimisations and Simplifications

- 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.

9.1.10.14 Documentation

- Create csv data files for thread tables.
- Convert tex to markdown, make tex an output of `build.bash`.
- Run `sloccount` on `extern/` and `src/` (and `tests/`) 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.

9.1.10.15 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.

9.1.11 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_\leftrightarrow 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.

9.1.12 Development

9.1.12.1 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.

9.1.12.2 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.

9.1.12.3 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 $\times 86$ 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 <tt>embroider</tt> 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.

10.0.0.0.3 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.

10.0.0.0.4 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.

10.0.0.10 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.

10.0.0.11 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.

10.0.0.12 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 \(char const *s\)](#)

description

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

Josh, Review this file 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 emblnt_read

Member fread_uint16 (FILE *f)

replace with emblnt_read

Member generate_dragon_curve (char *state, int iterations)

find citation for paper folding method

Page Geometry and Algorithms

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

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.

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) Fix emb-outline files

(Arduino) Fix thread-color files

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

(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

Bibliography style to plainnat.

US letter paper version of printed docs. /* TODO: Josh, Review this file 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. */

Serif font for printed docs.

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::createGroupBoxMiscImage ()`

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	50
_bcf_file_difat	51
_bcf_file_fat	51
_bcf_file_header	52
_vp3Hoop	53
Action__	55
Compress	91
EmbAlignedDim__	102
EmbAngularDim__	103
EmbArc__	103
EmbArcLengthDim__	104
EmbArray__	104
EmbBezier__	104
EmbBlock__	105
EmbCircle__	105
EmbColor__	106
EmbDiameterDim__	108
EmbEllipse__	108
EmbFormatList__	108
EmbGeometry__	109
EmblImage__	111
EmblInfiniteLine__	112
EmbLayer__	112
EmbLeaderDim__	112
EmbLine__	112
EmbLinearDim__	113
EmbOrdinateDim__	113
EmbPath__	114
EmbPattern__	114
EmbPoint__	115

EmbRadiusDim_	115
EmbRay_	116
EmbRect_	116
EmbSatinOutline_	117
EmbSpline_	117
EmbStitch_	117
EmbTextMulti_	118
EmbTextSingle_	118
EmbThread_	119
EmbTime_	119
EmbVector_	120
EmbView_	120
hoop_padding	123
Huffman	123
LSYSTEM	135
Parameter_ QApplication	187
Application QDialog	56
EmbDetailsDialog	106
LayerManager	129
Settings_Dialog QDockWidget	234
PropertyEditor	204
UndoEditor QFileDialog	279
PreviewDialog QGraphicsPathItem	203
BaseObject	67
ArcObject	57
CircleObject	72
DimLeaderObject	92
EllipseObject	98
ImageObject	124

LineObject	130
PathObject	187
PointObject	191
PolygonObject	195
PolylineObject	199
RectObject	213
TextSingleObject	261
QGraphicsView	
View	281
QLineEdit	
CmdPromptInput	85
QMainWindow	
MainWindow	136
QMdiArea	
MdiArea	176
QMdiSubWindow	
MdiWindow	179
QObject	
SaveObject	217
QRubberBand	
SelectBox	223
QSplitter	
CmdPromptSplitter	90
QSplitterHandle	
CmdPromptHandle	82
QStatusBar	
StatusBar	255
QTextBrowser	
CmdPromptHistory	84
QToolButton	
StatusBarButton	257
QUndoCommand	
UndoableAddCommand	271
UndoableDeleteCommand	272
UndoableGripEditCommand	273
UndoableMirrorCommand	274
UndoableMoveCommand	275
UndoableNavCommand	276

UndoableRotateCommand	277
UndoableScaleCommand	278
QWidget	
CmdPrompt	76
ImageWidget	127
Settings_	225
StxThread_	259
SubDescriptor_	260
SvgAttribute_	261
thread_color_	267
ThredExtension_	267
ThredHeader_	268
UiObject_	269
UndoHistory_	281
VipHeader_	295

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	50
_bcf_file_difat	51
_bcf_file_fat	51
_bcf_file_header	52
_vp3Hoop	53
Action_	55
Application	56
ArcObject	57
BaseObject	67
CircleObject	72
CmdPrompt	76

CmdPromptHandle	
82	
CmdPromptHistory	
The Command Prompt History class	84
CmdPromptInput	
	85
CmdPromptSplitter	
90	
Compress	
	91
DimLeaderObject	
	92
EllipseObject	
	98
EmbAlignedDim_	
	102
EmbAngularDim_	
	103
EmbArc_	
Absolute position (not relative)	103
EmbArcLengthDim_	
	104
EmbArray_	
	104
EmbBezier_	
	104
EmbBlock_	
	105
EmbCircle_	
	105
EmbColor_	
	106
EmbDetailsDialog	
106	
EmbDiameterDim_	
	108
EmbEllipse_	
	108
EmbFormatList_	
	108
EmbGeometry_	
	109
EmblImage_	
	111
EmblInfiniteLine_	
	112
EmbLayer_	
	112
EmbLeaderDim_	
	112
EmbLine_	
	112
EmbLinearDim_	
	113
EmbOrdinateDim_	
	113
EmbPath_	
	114

EmbPattern_	114
EmbPoint_	115
EmbRadiusDim_	115
EmbRay_	116
EmbRect_	116
EmbSatinOutline_	117
EmbSpline_	117
EmbStitch_	117
EmbTextMulti_	118
EmbTextSingle_	118
EmbThread_	119
EmbTime_	119
EmbVector_	120
EmbView_	120
hoop_padding	123
Huffman	123
ImageObject	124
ImageWidget	127
LayerManager	129
LineObject	130
LSYSTEM	135
MainWindow	136
The MainWindow class	136
MdiArea	176
MdiWindow	179
Parameter_	187
PathObject	187
PointObject	191
PolygonObject	195
PolylineObject	199
PreviewDialog	203

PropertyEditor	204
RectObject	213
SaveObject	217
SelectBox	223
Settings_	
Settings System	225
Settings_Dialog	234
StatusBar	255
StatusBarButton	257
StxThread_	259
SubDescriptor_	260
SvgAttribute_	261
TextSingleObject	261
thread_color_	267
ThredExtension_	267
ThredHeader_	268
UiObject_	
This covers the inbuilt designs: Dolphin, Snowflake and Heart. Covers Rotate, Scale and Point UI events	269
UndoableAddCommand	271
UndoableDeleteCommand	272
UndoableGripEditCommand	273
UndoableMirrorCommand	274
UndoableMoveCommand	275
UndoableNavCommand	276
UndoableRotateCommand	277
UndoableScaleCommand	278
UndoEditor	279
UndoHistory_	281
View	281
VipHeader_	295

16 File Index

16.1 File List

Here is a list of all files with brief descriptions:

embroidermodder2/cmdprompt.cpp	296
embroidermodder2/embdetails-dialog.cpp	296
embroidermodder2/embroidermodder.cpp	296
embroidermodder2/embroidermodder.h	297
embroidermodder2/imagewidget.cpp	344
embroidermodder2/layer-manager.cpp	344
embroidermodder2/mainwindow-commands.cpp	344
embroidermodder2/mainwindow-menus.cpp	344
embroidermodder2/mainwindow-settings.cpp	344
embroidermodder2/mainwindow-toolbars.cpp	345
embroidermodder2/mainwindow.cpp	347
embroidermodder2/mdiarea.cpp	351
embroidermodder2/mdiwindow.cpp	351
embroidermodder2/object-arc.cpp	351
embroidermodder2/object-base.cpp	351
embroidermodder2/object-circle.cpp	351
embroidermodder2/object-dimleader.cpp	351
embroidermodder2/object-ellipse.cpp	351
embroidermodder2/object-image.cpp	351
embroidermodder2/object-line.cpp	352
embroidermodder2/object-path.cpp	352
embroidermodder2/object-point.cpp	352
embroidermodder2/object-polygon.cpp	352
embroidermodder2/object-polyline.cpp	352
embroidermodder2/object-rect.cpp	352
embroidermodder2/object-save.cpp	352
embroidermodder2/object-textsingle.cpp	352
embroidermodder2/preview-dialog.cpp	352
embroidermodder2/property-editor.cpp	352

embroidermodder2/selectbox.cpp	368
embroidermodder2/settings-dialog.cpp	368
embroidermodder2/statusbar-button.cpp	368
embroidermodder2/statusbar.cpp	368
embroidermodder2/undo-commands.cpp	369
embroidermodder2/undo-editor.cpp	369
embroidermodder2/view.cpp	369
extern/libembroidery/src/array.c	369
extern/libembroidery/src/compress.c	371
extern/libembroidery/src/embroidery.h	373
extern/libembroidery/src/embroidery_internal.h	413
extern/libembroidery/src/encoding.c	460
extern/libembroidery/src/fill.c	463
extern/libembroidery/src/formats.c	468
extern/libembroidery/src/geometry.c	514
extern/libembroidery/src/image.c	528
extern/libembroidery/src/main.c	529
extern/libembroidery/src/pattern.c	540
extern/libembroidery/src/thread-color.c	545
extern/libembroidery/src/formats/format_100.c	472
extern/libembroidery/src/formats/format_10o.c	472
extern/libembroidery/src/formats/format_art.c	473
extern/libembroidery/src/formats/format_bmc.c	473
extern/libembroidery/src/formats/format_bro.c	474
extern/libembroidery/src/formats/format_cnd.c	474
extern/libembroidery/src/formats/format_col.c	475
extern/libembroidery/src/formats/format_csd.c	476
extern/libembroidery/src/formats/format_csv.c	477
extern/libembroidery/src/formats/format_dat.c	478
extern/libembroidery/src/formats/format_dem.c	479
extern/libembroidery/src/formats/format_dsb.c	479
extern/libembroidery/src/formats/format_dst.c	480

extern/libembroidery/src/formats/format_dsz.c	482
extern/libembroidery/src/formats/format_dxf.c	482
extern/libembroidery/src/formats/format_edr.c	483
extern/libembroidery/src/formats/format_emd.c	484
extern/libembroidery/src/formats/format_exp.c	484
extern/libembroidery/src/formats/format_exy.c	485
extern/libembroidery/src/formats/format_eyc.c	485
extern/libembroidery/src/formats/format_fxy.c	486
extern/libembroidery/src/formats/format_gc.c	486
extern/libembroidery/src/formats/format_gnc.c	487
extern/libembroidery/src/formats/format_gt.c	487
extern/libembroidery/src/formats/format_hus.c	488
extern/libembroidery/src/formats/format_inb.c	489
extern/libembroidery/src/formats/format_inf.c	489
extern/libembroidery/src/formats/format_jef.c	490
extern/libembroidery/src/formats/format_ksm.c	491
extern/libembroidery/src/formats/format_max.c	491
extern/libembroidery/src/formats/format_mit.c	492
extern/libembroidery/src/formats/format_new.c	493
extern/libembroidery/src/formats/format_ofm.c	493
extern/libembroidery/src/formats/format_pcd.c	494
extern/libembroidery/src/formats/format_pcm.c	495
extern/libembroidery/src/formats/format_pcq.c	495
extern/libembroidery/src/formats/format_pcs.c	496
extern/libembroidery/src/formats/format_pec.c	496
extern/libembroidery/src/formats/format_pel.c	498
extern/libembroidery/src/formats/format_pem.c	498
extern/libembroidery/src/formats/format_pes.c	499
extern/libembroidery/src/formats/format_phb.c	501
extern/libembroidery/src/formats/format_phc.c	502
extern/libembroidery/src/formats/format_plt.c	502
extern/libembroidery/src/formats/format_rgb.c	503

extern/libembroidery/src/formats/ format_sew.c	503
extern/libembroidery/src/formats/ format_shv.c	504
extern/libembroidery/src/formats/ format_sst.c	504
extern/libembroidery/src/formats/ format_stx.c	505
extern/libembroidery/src/formats/ format_svg.c	505
extern/libembroidery/src/formats/ format_t01.c	507
extern/libembroidery/src/formats/ format_t09.c	507
extern/libembroidery/src/formats/ format_tap.c	508
extern/libembroidery/src/formats/ format_thr.c	508
extern/libembroidery/src/formats/ format_txt.c	509
extern/libembroidery/src/formats/ format_u00.c	509
extern/libembroidery/src/formats/ format_u01.c	510
extern/libembroidery/src/formats/ format_vip.c	510
extern/libembroidery/src/formats/ format_vp3.c	512
extern/libembroidery/src/formats/ format_xxx.c	513
extern/libembroidery/src/formats/ format_zsk.c	514
extern/libembroidery/src/geometry/ arc.c	516
extern/libembroidery/src/geometry/ circle.c	519
extern/libembroidery/src/geometry/ ellipse.c	520
extern/libembroidery/src/geometry/ functions.c	522
extern/libembroidery/src/geometry/ line.c	523
extern/libembroidery/src/geometry/ path.c	523
extern/libembroidery/src/geometry/ polygon.c	523
extern/libembroidery/src/geometry/ polyline.c	523
extern/libembroidery/src/geometry/ rect.c	524
extern/libembroidery/src/geometry/ text.c	524
extern/libembroidery/src/geometry/ vector.c	526

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](#)

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 script std::vector<std::string> script

17.8.2.5 shortcut std::string shortcut

17.8.2.6 statustip std::string statustip

17.8.2.7 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>	
<i>argv</i>	

17.9.3 Member Function Documentation

17.9.3.1 event() bool event (QEvent * event) [protected], [virtual]
Application::event.

Parameters

event	
-------	--

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`
`ArcObject::objectRadius.`
- `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** 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

- 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

<code>arc</code>	<input type="button" value=""/>
<code>rgb</code>	<input type="button" value=""/>
<code>parent</code>	<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>	
<i>startY</i>	
<i>midX</i>	
<i>midY</i>	
<i>endX</i>	
<i>endY</i>	
<i>rgb</i>	
<i>parent</i>	

17.10.2.3 ArcObject() [3/3]

```

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

Parameters

<i>obj</i>	
<i>parent</i>	

17.10.2.4 ~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()

```

    EmbReal startX,
    EmbReal startY,
    EmbReal midX,
    EmbReal midY,
```

```
    EmbReal endX,  
    EmbReal endY )  
ArcObject::calculateArcData.
```

Parameters

<i>startX</i>	
<i>startY</i>	
<i>midX</i>	
<i>midY</i>	
<i>endX</i>	
<i>endY</i>	

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.
```

Returns

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

<i>painter</i>	
<i>option</i>	

17.10.3.24 setObjectEndAngle() void setObjectEndAngle (
 EmbReal angle)

17.10.3.25 setObjectEndPoint() [1/2] void setObjectEndPoint (
 const QPointF & point)
ArcObject::setObjectEndPoint.

Parameters

<i>point</i>	
--------------	--

17.10.3.26 setObjectEndPoint() [2/2] void setObjectEndPoint (
 EmbReal pointX,
 EmbReal pointY)
ArcObject::setObjectEndPoint.

Parameters

<i>pointX</i>	<input type="text"/>
<i>pointY</i>	<input type="text"/>

17.10.3.27 `setObjectMidPoint()` [1/2] void setObjectMidPoint (const QPointF & *point*)

17.10.3.28 `setObjectMidPoint()` [2/2] void setObjectMidPoint (EmbReal *pointX*, EmbReal *pointY*)

17.10.3.29 `setObjectRadius()` void setObjectRadius (EmbReal *radius*)

17.10.3.30 `setObjectStartAngle()` void setObjectStartAngle (EmbReal *angle*)

17.10.3.31 `setObjectStartPoint()` [1/2] void setObjectStartPoint (const QPointF & *point*)

17.10.3.32 `setObjectStartPoint()` [2/2] void setObjectStartPoint (EmbReal *pointX*, EmbReal *pointY*)

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

17.10.3.34 `updateArcRect()` void updateArcRect (EmbReal *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>	
----------------	--

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](#) [objectLineWeight](#) () 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 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`

17.12.1 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,
QSsplitter * 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:

- embroiderymodder2/[embroidermodder.h](#)
- embroiderymodder2/[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:

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

- 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	
------	--

17.19.1.2 ArrowStyle `enum ArrowStyle`

Enumerator

NoArrow	
Open	
Closed	

Enumerator

Dot	
Box	
Tick	

17.19.1.3 `lineStyle` enum `lineStyle`**Enumerator**

NoLine	
Flared	
Fletching	

17.19.2 Constructor & Destructor Documentation**17.19.2.1 `DimLeaderObject()` [1/2]** `DimLeaderObject` (

```
    EmbReal 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 &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](#)

17.20.1 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  
} object
```

- `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` (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`

17.59.1 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]**

```
EmbReal x,  
EmbReal y,  
EmbReal w,  
EmbReal h,  
QRgb rgb,  
QGraphicsItem * parent = 0 )
```

17.59.2.2 `ImageObject()` [2/2]

```
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

fileName	<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 &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`

17.62.1 Member Enumeration Documentation

17.62.1.1 anonymous enum anonymous enum

Enumerator

Type	
------	--

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 & endPt1)`

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 ()`
MainWindow::enablePromptRapidFire.
- void `disablePromptRapidFire ()`
MainWindow::disablePromptRapidFire.
- void `enableMoveRapidFire ()`
MainWindow::enableMoveRapidFire.
- void `disableMoveRapidFire ()`
MainWindow::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.
- 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="")
 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.
 - `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.
 - `void nativePrintArea (EmbReal x, EmbReal y, EmbReal w, EmbReal h)`
MainWindow::nativePrintArea.
 - `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)
- 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.
- void `nativeCopySelected (EmbReal x, EmbReal y)`
MainWindow::nativeCopySelected.
- void `nativePasteSelected (EmbReal x, EmbReal y)`
MainWindow::nativePasteSelected.
- void `nativeMoveSelected (EmbReal dx, EmbReal dy)`
MainWindow::nativeMoveSelected.
- void `nativeScaleSelected (EmbReal x, EmbReal y, EmbReal factor)`
MainWindow::nativeScaleSelected.
- void `nativeRotateSelected (EmbReal x, EmbReal y, EmbReal rot)`
MainWindow::nativeRotateSelected.
- void `nativeMirrorSelected (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2)`
MainWindow::nativeMirrorSelected.
- 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 * `wizardTipOfDay`
- QLabel * `labelTipOfDay`
- QCheckBox * `checkBoxTipOfDay`
- QStringList `listTipOfDay`
- 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 * `lineweightSelector`
- 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="text"/>
----------------	----------------------

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="text"/>
--------------	----------------------

17.64.3.13 closeToolBar void closeToolBar (QAction * *action*) [slot]
[MainWindow::closeToolBar](#).

Parameters

<i>action</i>	<input type="text"/>
---------------	----------------------

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>	
-----------------	--

Returns

17.64.3.46 floatingChangedToolBar void floatingChangedToolBar (bool *isFloating*) [slot]
[MainWindow::floatingChangedToolBar](#).

Parameters

<i>isFloating</i>	<input type="checkbox"/>
-------------------	--------------------------

17.64.3.47 QAction * getAction QAction * *getAction* (int *actionEnum*) [slot]
[MainWindow::getAction](#).

Parameters

<i>actionEnum</i>	<input type="checkbox"/>
-------------------	--------------------------

Returns

17.64.3.48 MainWindow * getApplication() MainWindow * *getApplication* ()
[MainWindow::getApplication](#).

Returns

17.64.3.49 QRgb getCurrentColor QRgb *getCurrentColor* () [slot]

17.64.3.50 QString getCurrentLayer QString *getCurrentLayer* () [slot]

17.64.3.51 QString getCurrentLineType QString *getCurrentLineType* () [slot]

17.64.3.52 QString getCurrentLineWeight QString *getCurrentLineWeight* () [slot]

17.64.3.53 QAction * getFileSeparator() QAction * *getFileSeparator* () [protected]
[MainWindow::getFileSeparator](#).

Returns

17.64.3.54 MdiArea * 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 loadFormats() void loadFormats () [protected]
[MainWindow::loadFormats](#).

17.64.3.66 logPromptInput void logPromptInput (const QString & txt) [slot]

17.64.3.67 makeLayerActive void makeLayerActive () [slot]

17.64.3.68 nativeAddArc() void nativeAddArc (EmbReal startX, EmbReal startY, EmbReal midX, EmbReal midY, EmbReal endX, EmbReal endY, int rubberMode)

```
17.64.3.69 nativeAddCircle() void nativeAddCircle (
    EmbReal centerX,
    EmbReal centerY,
    EmbReal radius,
    bool fill,
    int rubberMode )
```

```
17.64.3.70 nativeAddDimLeader() void nativeAddDimLeader (
    EmbReal x1,
    EmbReal y1,
    EmbReal x2,
    EmbReal y2,
    EmbReal rot,
    int rubberMode )
```

```
17.64.3.71 nativeAddEllipse() void nativeAddEllipse (
    EmbReal centerX,
    EmbReal centerY,
    EmbReal width,
    EmbReal height,
    EmbReal rot,
    bool fill,
    int rubberMode )
```

```
17.64.3.72 nativeAddHorizontalDimension() void nativeAddHorizontalDimension (
    EmbReal x1,
    EmbReal y1,
    EmbReal x2,
    EmbReal y2,
    EmbReal legHeight )
```

```
17.64.3.73 nativeAddImage() void nativeAddImage (
    const QString & img,
    EmbReal x,
    EmbReal y,
    EmbReal w,
    EmbReal h,
    EmbReal rot )
```

```
17.64.3.74 nativeAddInfiniteLine() void nativeAddInfiniteLine (
    EmbReal x1,
    EmbReal y1,
    EmbReal x2,
    EmbReal y2,
    EmbReal rot )
```

```
17.64.3.75 nativeAddLine() void nativeAddLine (
    EmbReal x1,
    EmbReal y1,
    EmbReal x2,
    EmbReal y2,
```

```
    EmbReal rot,  
    int rubberMode )
```

17.64.3.76 nativeAddPath() void nativeAddPath (

```
    EmbReal startX,  
    EmbReal startY,  
    const QPainterPath & p,  
    int rubberMode )
```

17.64.3.77 nativeAddPoint() void nativeAddPoint (

```
    EmbReal x,  
    EmbReal y )
```

17.64.3.78 nativeAddPolygon() void nativeAddPolygon (

```
    EmbReal startX,  
    EmbReal startY,  
    const QPainterPath & p,  
    int rubberMode )
```

17.64.3.79 nativeAddPolyline() void nativeAddPolyline (

```
    EmbReal startX,  
    EmbReal startY,  
    const QPainterPath & p,  
    int rubberMode )
```

17.64.3.80 nativeAddRay() void nativeAddRay (

```
    EmbReal x1,  
    EmbReal y1,  
    EmbReal x2,  
    EmbReal y2,  
    EmbReal rot )
```

17.64.3.81 nativeAddRectangle() void nativeAddRectangle (

```
    EmbReal x,  
    EmbReal y,  
    EmbReal w,  
    EmbReal h,  
    EmbReal rot,  
    bool fill,  
    int rubberMode )
```

17.64.3.82 nativeAddRegularPolygon() void nativeAddRegularPolygon (

```
    EmbReal centerX,  
    EmbReal centerY,  
    quint16 sides,  
    uint8_t mode,  
    EmbReal rad,  
    EmbReal rot,  
    bool fill )
```

```
17.64.3.83 nativeAddRoundedRectangle() void nativeAddRoundedRectangle (
    EmbReal x,
    EmbReal y,
    EmbReal w,
    EmbReal h,
    EmbReal rad,
    EmbReal rot,
    bool fill )
```

```
17.64.3.84 nativeAddSlot() void nativeAddSlot (
    EmbReal centerX,
    EmbReal centerY,
    EmbReal diameter,
    EmbReal length,
    EmbReal rot,
    bool fill,
    int rubberMode )
```

```
17.64.3.85 nativeAddTextMulti() void nativeAddTextMulti (
    const QString & str,
    EmbReal x,
    EmbReal y,
    EmbReal rot,
    bool fill,
    int rubberMode )
```

```
17.64.3.86 nativeAddTextSingle() void nativeAddTextSingle (
    const QString & str,
    EmbReal x,
    EmbReal y,
    EmbReal rot,
    bool fill,
    int rubberMode )
```

```
17.64.3.87 nativeAddToSelection() void nativeAddToSelection (
    const QPainterPath path,
    Qt::SelectionMode mode )
```

```
17.64.3.88 nativeAddTriangle() void nativeAddTriangle (
    EmbReal x1,
    EmbReal y1,
    EmbReal x2,
    EmbReal y2,
    EmbReal x3,
    EmbReal y3,
    EmbReal rot,
    bool fill )
```

```
17.64.3.89 nativeAddVerticalDimension() void nativeAddVerticalDimension (
    EmbReal x1,
    EmbReal y1,
```

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

17.64.3.90 nativeAlert() void nativeAlert (const QString & txt)

17.64.3.91 nativeAllowRubber() bool nativeAllowRubber ()

17.64.3.92 nativeAppendPromptHistory() void nativeAppendPromptHistory (const QString & txt)

17.64.3.93 nativeBlinkPrompt() void nativeBlinkPrompt ()

17.64.3.94 nativeCalculateAngle() EmbReal nativeCalculateAngle (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2)

17.64.3.95 nativeCalculateDistance() EmbReal nativeCalculateDistance (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2)

17.64.3.96 nativeClearRubber() void nativeClearRubber ()

17.64.3.97 nativeClearSelection() void nativeClearSelection ()

17.64.3.98 nativeCopySelected() void nativeCopySelected (EmbReal x, EmbReal y)

MainWindow::nativeCopySelected.

Parameters

x	
y	

17.64.3.99 nativeCutSelected() void nativeCutSelected (EmbReal x, EmbReal y)

MainWindow::nativeCutSelected.

Parameters

<i>x</i>	
<i>y</i>	

17.64.3.100 nativeDeleteSelected() void nativeDeleteSelected ()
[MainWindow::nativeDeleteSelected](#).

17.64.3.101 nativeDisableMoveRapidFire() void nativeDisableMoveRapidFire ()

17.64.3.102 nativeDisablePromptRapidFire() void nativeDisablePromptRapidFire ()

17.64.3.103 nativeEnableMoveRapidFire() void nativeEnableMoveRapidFire ()

17.64.3.104 nativeEnablePromptRapidFire() void nativeEnablePromptRapidFire ()

17.64.3.105 nativeEndCommand() void nativeEndCommand ()

17.64.3.106 nativeExit() void nativeExit ()

17.64.3.107 nativeInitCommand() void nativeInitCommand ()

17.64.3.108 nativeMessageBox() void nativeMessageBox (
 const QString & type,
 const QString & title,
 const QString & text)
[MainWindow::nativeMessageBox](#).

Parameters

<i>type</i>	
<i>title</i>	
<i>text</i>	

17.64.3.109 nativeMirrorSelected() void nativeMirrorSelected (
 EmbReal x1,
 EmbReal y1,
 EmbReal x2,
 EmbReal y2)
[MainWindow::nativeMirrorSelected](#).

Parameters

<i>x1</i>	

Parameters

<i>y1</i>	
<i>x2</i>	
<i>y2</i>	

17.64.3.110 nativeMouseX() `EmbReal nativeMouseX ()`
`MainWindow::nativeMouseX.`

Returns

17.64.3.111 nativeMouseY() `EmbReal nativeMouseY ()`
`MainWindow::nativeMouseY.`

Returns

17.64.3.112 nativeMoveSelected() `void nativeMoveSelected (`
 `EmbReal dx,`
 `EmbReal dy)`
`MainWindow::nativeMoveSelected.`

Parameters

<i>dx</i>	
<i>dy</i>	

17.64.3.113 nativeNewFile() `void nativeNewFile ()`

17.64.3.114 nativeNumSelected() `int nativeNumSelected ()`

17.64.3.115 nativeOpenFile() `void nativeOpenFile ()`

17.64.3.116 nativePasteSelected() `void nativePasteSelected (`
 `EmbReal x,`
 `EmbReal y)`
`MainWindow::nativePasteSelected.`

Parameters

<i>x</i>	
<i>y</i>	

17.64.3.117 nativePerpendicularDistance() `EmbReal nativePerpendicularDistance (`
`EmbReal px,`
`EmbReal py,`
`EmbReal x1,`
`EmbReal y1,`
`EmbReal x2,`
`EmbReal y2)`

17.64.3.118 nativePreviewOff() `void nativePreviewOff ()`

17.64.3.119 nativePreviewOn() `void nativePreviewOn (`
`int clone,`
`int mode,`
`EmbReal x,`
`EmbReal y,`
`EmbReal data)`

17.64.3.120 nativePrintArea() `void nativePrintArea (`
`EmbReal x,`
`EmbReal y,`
`EmbReal w,`
`EmbReal h)`

[MainWindow::nativePrintArea.](#)

Parameters

<i>x</i>	
<i>y</i>	
<i>w</i>	
<i>h</i>	

17.64.3.121 nativeQSnapX() `EmbReal nativeQSnapX ()`
[MainWindow::nativeQSnapX.](#)

Returns

17.64.3.122 nativeQSnapY() `EmbReal nativeQSnapY ()`
[MainWindow::nativeQSnapY.](#)

Returns

17.64.3.123 nativeRotateSelected() `void nativeRotateSelected (`
`EmbReal x,`
`EmbReal y,`
`EmbReal rot)`

[MainWindow::nativeRotateSelected.](#)

Parameters

<i>x</i>	
<i>y</i>	
<i>rot</i>	

17.64.3.124 nativeScaleSelected() void nativeScaleSelected (EmbReal *x*, EmbReal *y*, EmbReal *factor*)

[MainWindow::nativeScaleSelected](#).

Parameters

<i>x</i>	
<i>y</i>	
<i>factor</i>	

17.64.3.125 nativeSelectAll() void nativeSelectAll ()

17.64.3.126 nativeSetBackgroundColor() void nativeSetBackgroundColor (uint8_t *r*, uint8_t *g*, uint8_t *b*)

17.64.3.127 nativeSetCrossHairColor() void nativeSetCrossHairColor (uint8_t *r*, uint8_t *g*, uint8_t *b*)

17.64.3.128 nativeSetCursorShape() void nativeSetCursorShape (const QString & *str*)

17.64.3.129 nativeSetGridColor() void nativeSetGridColor (uint8_t *r*, uint8_t *g*, uint8_t *b*)

17.64.3.130 nativeSetPromptPrefix() void nativeSetPromptPrefix (const QString & *txt*)

17.64.3.131 nativeSetRubberMode() void nativeSetRubberMode (int *mode*)

17.64.3.132 nativeSetRubberPoint() void nativeSetRubberPoint (const QString & key, EmbReal x, EmbReal y)

17.64.3.133 nativeSetRubberText() void nativeSetRubberText (const QString & key, const QString & txt)

17.64.3.134 nativeSpareRubber() void nativeSpareRubber (qint64 id)

17.64.3.135 nativeTextAngle() EmbReal nativeTextAngle ()

17.64.3.136 nativeTextBold() bool nativeTextBold ()

17.64.3.137 nativeTextFont() QString nativeTextFont ()

17.64.3.138 nativeTextItalic() bool nativeTextItalic ()

17.64.3.139 nativeTextOverline() bool nativeTextOverline ()

17.64.3.140 nativeTextSize() EmbReal nativeTextSize ()

17.64.3.141 nativeTextStrikeOut() bool nativeTextStrikeOut ()

17.64.3.142 nativeTextUnderline() bool nativeTextUnderline ()

17.64.3.143 nativeTipOfDay() void nativeTipOfDay ()

17.64.3.144 nativeVulcanize() void nativeVulcanize ()

17.64.3.145 nativeWindowCascade() void nativeWindowCascade ()

17.64.3.146 nativeWindowClose() void nativeWindowClose ()

17.64.3.147 nativeWindowCloseAll() void nativeWindowCloseAll ()

17.64.3.148 nativeWindowNext() void nativeWindowNext ()

17.64.3.149 nativeWindowPrevious() void nativeWindowPrevious ()

17.64.3.150 nativeWindowTile() void nativeWindowTile ()

17.64.3.151 newFile void newFile () [slot]
[MainWindow::newFile](#).

17.64.3.152 nightVision void nightVision () [slot]
[MainWindow::nightVision](#).

17.64.3.153 onCloseMdiWin void onCloseMdiWin (
 [MdiWindow * theMdiWin](#)) [virtual], [slot]
[MainWindow::onCloseMdiWin](#).

Parameters

theMdiWin	<input type="button" value=""/>
---------------------------	---------------------------------

17.64.3.154 onCloseWindow void onCloseWindow () [slot]
[MainWindow::onCloseWindow](#).

17.64.3.155 onWindowActivated void onWindowActivated (
 [QMdiSubWindow * w](#)) [slot]
[MainWindow::onWindowActivated](#).

Parameters

w	<input type="button" value=""/>
-------------------	---------------------------------

17.64.3.156 openFile void openFile (
 bool recent = *false*,
 const [QString & recentFile](#) = *""*) [slot]
[MainWindow::openFile](#).

Parameters

recent	<input type="button" value=""/>
recentFile	<input type="button" value=""/>

17.64.3.157 openFilesSelected void openFilesSelected (
 const [QStringList & filesToOpen](#)) [slot]
[MainWindow::openFilesSelected](#).

Parameters

<i>filesToOpen</i>	<input type="text"/>
--------------------	----------------------

17.64.3.158 openrecentfile void openrecentfile () [slot]
[MainWindow::openrecentfile](#).

17.64.3.159 panDown void panDown () [slot]
[MainWindow::panDown](#).

17.64.3.160 panLeft void panLeft () [slot]

17.64.3.161 panpoint void panpoint () [slot]

17.64.3.162 panrealtime void panrealtime () [slot]

17.64.3.163 panRight void panRight () [slot]

17.64.3.164 panUp void panUp () [slot]

17.64.3.165 paste void paste () [slot]

17.64.3.166 pickAddModeToggled void pickAddModeToggled () [slot]

17.64.3.167 platformString() QString platformString ()

17.64.3.168 print void print () [slot]

17.64.3.169 promptHistoryAppended void promptHistoryAppended (const QString & txt) [slot]

17.64.3.170 promptInputNext void promptInputNext () [slot]

17.64.3.171 promptInputPrevious void promptInputPrevious () [slot]

17.64.3.172 quit void quit () [slot]
[MainWindow::quit](#).

17.64.3.173 `readSettings` void readSettings () [slot]
[MainWindow::readSettings](#).

17.64.3.174 `recentMenuAboutToShow` void recentMenuAboutToShow () [slot]
[MainWindow::recentMenuAboutToShow](#).

17.64.3.175 `redo` void redo () [slot]

17.64.3.176 `resizeEvent()` void resizeEvent (
 QResizeEvent * e) [protected], [virtual]
[MainWindow::resizeEvent](#).

Parameters

e	
---	--

17.64.3.177 `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  
-----
```

17.64.3.178 `run_script_file()` std::string run_script_file (
 std::string fname)
[MainWindow::run_script_file](#).

Parameters

fname	The path of the script to run.
-------	--------------------------------

17.64.3.179 `runCommand` void runCommand () [slot]

17.64.3.180 runCommandClick void runCommandClick (const QString & *cmd*, EmbReal *x*, EmbReal *y*) [slot]

17.64.3.181 runCommandContext void runCommandContext (const QString & *cmd*, const QString & *str*) [slot]

17.64.3.182 runCommandMain void runCommandMain (const QString & *cmd*) [slot]

17.64.3.183 runCommandMove void runCommandMove (const QString & *cmd*, EmbReal *x*, EmbReal *y*) [slot]

17.64.3.184 runCommandPrompt void runCommandPrompt (const QString & *cmd*, const QString & *str*) [slot]

17.64.3.185 saveasfile void saveasfile () [slot]
MainWindow::saveasfile.

17.64.3.186 savefile void savefile () [slot]
MainWindow::savefile.

17.64.3.187 selectAll void selectAll () [slot]

17.64.3.188 setShiftPressed void setShiftPressed () [slot]

17.64.3.189 setShiftReleased void setShiftReleased () [slot]

17.64.3.190 setTextAngle void setTextAngle (EmbReal *num*) [slot]

17.64.3.191 setTextBold void setTextBold (bool *val*) [slot]

17.64.3.192 setTextFont void setTextFont (const QString & *str*) [slot]

17.64.3.193 `setTextItalic` void setTextItalic (bool val) [slot]

17.64.3.194 `setTextOverline` void setTextOverline (bool val) [slot]

17.64.3.195 `setTextSize` void setTextSize (EmbReal num) [slot]

17.64.3.196 `setTextStrikeOut` void setTextStrikeOut (bool val) [slot]

17.64.3.197 `setTextUnderline` void setTextUnderline (bool val) [slot]

17.64.3.198 `settingsDialog` void settingsDialog (const QString & showTab = QString()) [slot]

17.64.3.199 `settingsPrompt` void settingsPrompt () [slot]

17.64.3.200 `setUndoCleanIcon()` void setUndoCleanIcon (bool opened)

17.64.3.201 `stub_implement` void stub_implement (QString txt) [slot]
[MainWindow::stub_implement](#).

Parameters

<code>txt</code>	<input type="text"/>
------------------	----------------------

17.64.3.202 `stub_testing` void stub_testing () [slot]
[MainWindow::stub_testing](#).

17.64.3.203 `textAngle` EmbReal textAngle () [slot]

17.64.3.204 `textBold` bool textBold () [slot]

17.64.3.205 `textFont` QString textFont () [slot]

17.64.3.206 `textFontSelectorCurrentFontChanged` void textFontSelectorCurrentFontChanged (const QFont & *font*) [slot]

17.64.3.207 `textItalic` bool textItalic () [slot]

17.64.3.208 `textOverline` bool textOverline () [slot]

17.64.3.209 `textSize` EmbReal textSize () [slot]

17.64.3.210 `textSizeSelectorIndexChanged` void textSizeSelectorIndexChanged (int *index*) [slot]

17.64.3.211 `textStrikeOut` bool textStrikeOut () [slot]

17.64.3.212 `textUnderline` bool textUnderline () [slot]

17.64.3.213 `tipOfDay` void tipOfDay () [slot]

17.64.3.214 `toggleGrid` void toggleGrid () [slot]

17.64.3.215 `toggleLwt` void toggleLwt () [slot]

17.64.3.216 `toggleRuler` void toggleRuler () [slot]

17.64.3.217 `undo` void undo () [slot]

17.64.3.218 `updateAllViewBackgroundColors` void updateAllViewBackgroundColors (QRgb *color*) [slot]

17.64.3.219 `updateAllViewCrossHairColors` void updateAllViewCrossHairColors (QRgb *color*) [slot]

17.64.3.220 `updateAllViewGridColors` void updateAllViewGridColors (QRgb *color*) [slot]

17.64.3.221 `updateAllViewRulerColors` void updateAllViewRulerColors (QRgb *color*) [slot]

17.64.3.222 updateAllViewScrollBars void updateAllViewScrollBars (bool val) [slot]

17.64.3.223 updateAllViewSelectBoxColors void updateAllViewSelectBoxColors (QRgb colorL, QRgb fillL, QRgb colorR, QRgb fillR, int alpha) [slot]

17.64.3.224 updateMenuToolbarStatusbar() void updateMenuToolbarStatusbar () [virtual]
[MainWindow::updateMenuToolbarStatusbar](#).

17.64.3.225 updatePickAddMode void updatePickAddMode (bool val) [slot]

17.64.3.226 validFileFormat bool validFileFormat (const QString & fileName) [static], [slot]
[MainWindow::validFileFormat](#).

Parameters

fileName	<input type="text"/>
----------	----------------------

Returns

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

17.64.3.227 whatsThisContextHelp void whatsThisContextHelp () [slot]

17.64.3.228 windowMenuAboutToShow void windowMenuAboutToShow () [slot]
[MainWindow::windowMenuAboutToShow](#).

17.64.3.229 windowMenuActivated void windowMenuActivated (bool checked) [slot]
[MainWindow::windowMenuActivated](#).

Parameters

checked	<input type="checkbox"/>
---------	--------------------------

17.64.3.230 writeSettings void writeSettings () [slot]
[MainWindow::writeSettings](#).

17.64.3.231 zoomAll void zoomAll () [slot]

17.64.3.232 zoomCenter void zoomCenter () [slot]

17.64.3.233 zoomDynamic void zoomDynamic () [slot]

17.64.3.234 zoomExtents void zoomExtents () [slot]

17.64.3.235 zoomIn void zoomIn () [slot]

17.64.3.236 zoomOut void zoomOut () [slot]

17.64.3.237 zoomPrevious void zoomPrevious () [slot]

17.64.3.238 zoomRealtime void zoomRealtime () [slot]

17.64.3.239 zoomScale void zoomScale () [slot]

17.64.3.240 zoomSelected void zoomSelected () [slot]

17.64.3.241 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 \(const QString &fileName\)](#)

- 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](#) (

<i>MainWindow</i> *	<i>mw</i> ,
<i>QWidget</i> *	<i>parent</i> = 0)

MdiArea::MdiArea.

Parameters

<i>mw</i>	
<i>parent</i>	

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 (

<i>QMouseEvent</i> *	<i>e</i>) [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

fileName	<input type="text"/>
----------	----------------------

17.65.2.8 tile void tile () [slot]
[MdiArea::tile](#).

17.65.2.9 useBackgroundColor() void useBackgroundColor (bool use)

Parameters

use	<input type="text"/>
-----	----------------------

17.65.2.10 useBackgroundLogo() void useBackgroundLogo (bool use)
[MdiArea::useBackgroundLogo](#).

Parameters

use	<input type="text"/>
-----	----------------------

```
17.65.2.11 useBackgroundTexture() void useBackgroundTexture ( bool use )  
MdiArea::useBackgroundTexture.
```

Parameters

use	
-----	--

```
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 `currentLinewidthChanged` (const QString &weight)
`MdiWindow::currentLinewidthChanged.`
- 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.`

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

<code>color</code>	<input type="text"/>
--------------------	----------------------

17.66.2.3 currentLayerChanged void currentLayerChanged (const QString & layer) [slot]
[MdiWindow::currentLayerChanged](#).

Parameters

layer	<input type="text"/>
-------	----------------------

17.66.2.4 currentLinetypeChanged void currentLinetypeChanged (const QString & type) [slot]
[MdiWindow::currentLinetypeChanged](#).

Parameters

type	<input type="text"/>
------	----------------------

17.66.2.5 currentLineweightChanged void currentLineweightChanged (const QString & weight) [slot]
[MdiWindow::currentLineweightChanged](#).

Parameters

weight	<input type="text"/>
--------	----------------------

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

fileName	<input type="text"/>
----------	----------------------

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>	<input type="text"/>
-----------------------	----------------------

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

prev	<input type="text"/>
------	----------------------

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

fileName	<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

<i>fileName</i>	<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 updateColorLinetypeLineweight void updateColorLinetypeLineweight () [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	<input type="button" value=""/>
------	---------------------------------

17.68.2 Constructor & Destructor Documentation**17.68.2.1 PathObject()** [1/2] [PathObject](#) (
 EmbrReal x,
 EmbrReal 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](#) (
 EmbrReal x,
 EmbrReal 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 \(QPainter *painter=0\)](#)
- 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](#) 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 [lwrtPen](#)
- 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](#) ([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`
- `int griplIndex`

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)`

17.70.1 Member Enumeration Documentation

17.70.1.1 anonymous enum anonymous enum

Enumerator

Type	
------	--

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 &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`
- 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< QGraphicsItem * > 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 * createToolBar` (`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 * createToolBarQSelect` ()
- `QToolButton * createToolBarPickAdd` ()
- `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 templImageObj `ImageObject*` `tempImageObj` [private]

17.73.3.33 templInfLineObj `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 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`

17.74.1 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

<i>pattern</i>	<input type="text"/>
<i>item</i>	<input type="text"/>

17.75.2.2 addBlock() void addBlock (

```
EmbPattern * pattern,
QGraphicsItem * item )
```

SaveObject::addBlock.

Parameters

<i>pattern</i>	<input type="text"/>
<i>item</i>	<input type="text"/>

17.75.2.3 addCircle() void addCircle (

```
EmbPattern * pattern,
QGraphicsItem * item )
```

SaveObject::addCircle.

Parameters

<i>pattern</i>	<input type="text"/>
<i>item</i>	<input type="text"/>

17.75.2.4 addDimAligned() void addDimAligned (

```
EmbPattern * pattern,
QGraphicsItem * item )
```

SaveObject::addDimAligned.

Parameters

<i>pattern</i>	<input type="text"/>
<i>item</i>	<input type="text"/>

17.75.2.5 addDimAngular() void addDimAngular (

```
EmbPattern * pattern,
QGraphicsItem * item )
```

SaveObject::addDimAngular.

Parameters

<i>pattern</i>	<input type="text"/>
<i>item</i>	<input type="text"/>

17.75.2.6 addDimArcLength() void addDimArcLength (

```
EmbPattern * pattern,
QGraphicsItem * item )
```

SaveObject::addDimArcLength.

Parameters

<i>pattern</i>	
<i>item</i>	

17.75.2.7 addDimDiameter() void addDimDiameter (

```
EmbPattern * pattern,
QGraphicsItem * item )
```

SaveObject::addDimDiameter.

Parameters

<i>pattern</i>	
<i>item</i>	

17.75.2.8 addDimLeader() void addDimLeader (

```
EmbPattern * pattern,
QGraphicsItem * item )
```

SaveObject::addDimLeader.

Parameters

<i>pattern</i>	
<i>item</i>	

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 checkBoxQSnapPerpendicularStateChanged (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 checkBoxRenderHintNonCosmeticStateChanged (int *checked*) [private], [slot]

17.78.2.41 `checkBoxRenderHintSmoothPixStateChanged` void checkBoxRenderHintSmoothPixStateChanged (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 checkBoxSelectionModePickAddStateChanged (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	
------	--

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 (
 EmbReal x1,
 EmbReal y1,
 EmbReal x2,
 EmbReal 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
00585
00586
00587
00588
00589
00590
00591 enum SPARE_RUBBER_VALUES {
00592     SPARE_RUBBER_OFF = 0, //NOTE: Allow this enum to evaluate false
00593     SPARE_RUBBER_PATH,
00594     SPARE_RUBBER_POLYGON,
00595     SPARE_RUBBER_POLYLINE
00596 };
00597
00598 enum PREVIEW_CLONE_VALUES {
00599     PREVIEW_CLONE_NULL = 0, //NOTE: Allow this enum to evaluate false
00600     PREVIEW_CLONE_SELECTED,
00601     PREVIEW_CLONE_RUBBER
00602 };
00603
00604 enum PREVIEW_MODE_VALUES {
00605     PREVIEW_MODE_NULL = 0, //NOTE: Allow this enum to evaluate false
00606     PREVIEW_MODE_MOVE,
00607     PREVIEW_MODE_ROTATE,
00608     PREVIEW_MODE_SCALE
00609 };
00610
00611 int read_settings(const char *settings_file);
00612 void write_settings(const char *fname);
00613
00614 static const EmbReal emb_constant_pi = 3.14159265358979323846;
00615
00616 /*
00617     * \brief Convert \a a to a QPointF.
00618 */
00619 inline QPointF
00620 to_QPointF(EmbVector a)
00621 {
00622     QPointF result(a.x(), a.y());
00623     return result;
00624 }
00625
00626 /*
00627     * \brief Convert \a a to an EmbVector.
00628 */
00629 inline EmbVector
00630 to_EmbVector(QPointF a)
00631 {
00632     EmbVector v;
00633     v.x = a.x();
00634     v.y = a.y();
00635     return v;
00636 }
00637
00638 /*
00639     * \brief Wrapper for embVector_add to use the syntax \a a + \a b.
00640 */
00641 inline EmbVector
00642 operator+(EmbVector a, EmbVector b)

```

```

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
00735     void drawRubberLine(const QLineF& rubLine, QPainter* painter = 0, const char* colorFromScene = 0);
00736
00737     virtual void vulcanize() = 0;
00738     virtual QPointF mouseSnapPoint(const QPointF& mousePoint) = 0;
00739     virtual QList<QPointF> allGripPoints() = 0;
00740     virtual void gripEdit(const QPointF& before, const QPointF& after) = 0;
00741 protected:
00742     QPen lineWeightPen() const { return lwtPen; }
00743     void realRender(QPainter* painter, const QPainterPath& renderPath);
00744 };
00745
00746
00747 class ArcObject : public BaseObject
00748 {
00749 public:
00750     QPointF arcStartPoint;
00751     QPointF arcMidPoint;
00752     QPointF arcEndPoint;
00753
00754     ArcObject(EmbArc arc, QRgb rgb, QGraphicsItem* parent = 0);
00755     ArcObject(EmbReal startX, EmbReal startY, EmbReal midX, EmbReal midY, EmbReal endX, EmbReal endY,
00756     QRgb rgb, QGraphicsItem* parent = 0);
00757     ArcObject(ArcObject* obj, QGraphicsItem* parent = 0);
00758     ~ArcObject();
00759
00760     enum { Type = OBJ_TYPE_ARC };
00761     virtual int type() const { return Type; }
00762
00763     void init(EmbReal startX, EmbReal startY, EmbReal midX, EmbReal midY, EmbReal endX, EmbReal endY,
00764     QRgb rgb, Qt::PenStyle lineType);
00765     void updatePath();
00766
00767     void calculateArcData(EmbReal startX, EmbReal startY, EmbReal midX, EmbReal midY, EmbReal endX,
00768     EmbReal endY);
00769     void updateArcRect(EmbReal radius);
00770
00771     EmbReal objectRadius() const { return rect().width()/2.0*scale(); }
00772     EmbReal objectStartAngle() const;
00773     EmbReal objectEndAngle() const;
00774     QPointF objectStartPoint() const;
00775     EmbReal objectStartX() const;
00776     EmbReal objectStartY() const;
00777     QPointF objectMidPoint() const;
00778     EmbReal objectMidX() const;
00779     EmbReal objectMidY() const;
00780     QPointF objectEndPoint() const;
00781     EmbReal objectEndX() const;
00782     EmbReal objectEndY() const;
00783     EmbReal objectArea() const;
00784     EmbReal objectArcLength() const;
00785     EmbReal objectChord() const;
00786     EmbReal objectIncludedAngle() const;
00787     bool objectClockwise() const;
00788
00789     void setObjectRadius(EmbReal radius);
00790     void setObjectStartAngle(EmbReal angle);
00791     void setObjectEndAngle(EmbReal angle);
00792     void setObjectStartPoint(const QPointF& point);
00793     void setObjectStartPoint(EmbReal pointX, EmbReal pointY);
00794     void setObjectMidPoint(const QPointF& point);
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);
00814     CircleObject(CircleObject* obj, QGraphicsItem* parent = 0);
00815     ~CircleObject();
00816
00817     void init(EmbReal centerX, EmbReal centerY, EmbReal radius, QRgb rgb, Qt::PenStyle lineType);
00818     void updatePath();
00819
00820     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

```

```

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     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
01846     QAction* actionHash[200];
01847     QHash<QString, QToolBar*> toolbarHash;
01848     QHash<QString, QMenu*> menuHash;
01849
01850     QString formatFilterOpen;
01851     QString formatFilterSave;
01852
01853     bool isCommandActive() { return prompt->isCommandActive(); }
01854     QString activeCommand() { return prompt->activeCommand(); }
01855     QIcon create_icon(QString stub);
01856     void create_toolbar(QToolBar* toolbar, std::string label, std::vector<std::string> entries);
01857
01858     QString platformString();
01859
01860 public slots:
01861     void enablePromptRapidFire();
01862     void disablePromptRapidFire();
01863
01864     void enableMoveRapidFire();
01865     void disableMoveRapidFire();
01866
01867     void onCloseWindow();
01868     virtual void onCloseMdiWin(MdiWindow*);
01869
01870     void recentMenuAboutToShow();
01871
01872     void onWindowActivated(QMdiSubWindow* w);
01873     void windowMenuAboutToShow();
01874     void windowMenuActivated( bool checked/*int id*/ );
01875     QAction*           getAction(int actionEnum);
01876
01877     void updateAllViewScrollBars(bool val);
01878     void updateAllViewCrossHairColors(QRgb color);
01879     void updateAllViewBackgroundColors(QRgb color);
01880     void updateAllViewSelectBoxColors(QRgb colorL, QRgb fillL, QRgb colorR, QRgb fillR, int alpha);
01881     void updateAllViewGridColors(QRgb color);
01882     void updateAllViewRulerColors(QRgb color);
01883
01884     void updatePickAddMode(bool val);
01885     void pickAddModeToggled();
01886
01887     void settingsPrompt();
01888
01889     void settingsDialog(const QString& showTab = QString());
01890     void readSettings();
01891     void writeSettings();
01892
01893     static bool validFileFormat(const QString &fileName);
01894
01895 protected:
01896     virtual void resizeEvent(QResizeEvent* );
01897     void closeEvent(QCloseEvent *event);
01898     QAction* getFileSeparator();
01899     void loadFormats();
01900
01901     bool shiftKeyPressedState;
01902
01903     QByteArray layoutState;
01904
01905     int numOfDocs;
01906     int docIndex;
01907
01908     QList<MdiWindow*> listMdiWin;
01909     QMdiSubWindow* findMdiWindow(const QString &fileName);
01910     QString openFilesPath;
01911
01912     QAction* myFileSeparator;
01913
01914     QWizard* wizardTipOfTheDay;
01915     QLabel* labelTipOfTheDay;
01916     QCheckBox* checkBoxTipOfTheDay;
01917     QStringList listTipOfTheDay;
01918
01919     void createAllActions();
01920
01921 //Toolbars
01922

```

```
01923 //=====
01924 void createAllToolbars();
01925 void createPanToolbar();
01926 void createIconToolbar();
01927 void createHelpToolbar();
01928 void createLayerToolbar();
01929 void createPropertiesToolbar();
01930 void createTextToolbar();
01931 void createPromptToolbar();
01932
01933 QToolBar* toolbarFile;
01934 QToolBar* toolbarEdit;
01935 QToolBar* toolbarView;
01936 QToolBar* toolbarZoom;
01937 QToolBar* toolbarPan;
01938 QToolBar* toolbarIcon;
01939 QToolBar* toolbarHelp;
01940 QToolBar* toolbarLayer;
01941 QToolBar* toolbarText;
01942 QToolBar* toolbarProperties;
01943 QToolBar* toolbarPrompt;
01944
01945 //Selectors
01946 //=====
01947 QComboBox* layerSelector;
01948 QComboBox* colorSelector;
01949 QComboBox* linetypeSelector;
01950 QComboBox* linewidthSelector;
01951 QFontComboBox* textFontSelector;
01952 QComboBox* textSizeSelector;
01953
01954 //Menus
01955 //=====
01956 void createAllMenus();
01957 void createFileMenu();
01958 void createEditMenu();
01959 void createViewMenu();
01960 void createSettingsMenu();
01961 void createWindowMenu();
01962 void createHelpMenu();
01963
01964 QMenu* fileMenu;
01965 QMenu* editMenu;
01966 QMenu* viewMenu;
01967 QMenu* settingsMenu;
01968 QMenu* windowMenu;
01969 QMenu* helpMenu;
01970
01971 //SubMenus
01972 //=====
01973 QMenu* recentMenu;
01974 QMenu* zoomMenu;
01975 QMenu* panMenu;
01976
01977 private slots:
01978 void hideUnimplemented();
01979
01980 public slots:
01981
01982 void stub_implement(QString txt);
01983 void stub_testing();
01984
01985 void promptHistoryAppended(const QString& txt);
01986 void logPromptInput(const QString& txt);
01987 void promptInputPrevious();
01988 void promptInputNext();
01989
01990 void runCommand();
01991 void runCommandMain(const QString& cmd);
01992 void runCommandClick(const QString& cmd, EmbReal x, EmbReal y);
01993 void runCommandMove(const QString& cmd, EmbReal x, EmbReal y);
01994 void runCommandContext(const QString& cmd, const QString& str);
01995 void runCommandPrompt(const QString& cmd, const QString& str);
01996
01997 void newFile();
01998 void openFile(bool recent = false, const QString& recentFile = "");
01999 void openFilesSelected(const QStringList&);
02000 void openrecentfile();
02001 void savefile();
02002 void saveasfile();
02003 void print();
02004 void designDetails();
02005 void exit();
02006 void quit();
02007 void checkForUpdates();
02008 // Help Menu
02009 void tipOfDay();
```

```

02010 void buttonTipOfDayClicked(int);
02011 void checkBoxTipOfDayStateChanged(int);
02012 void help();
02013 void changelog();
02014 void about();
02015 void whatsThisContextHelp();
02016
02017 void cut();
02018 void copy();
02019 void paste();
02020 void selectAll();
02021
02022 void closeToolBar(QAction* );
02023 void floatingChangedToolBar(bool);
02024
02025 void toggleGrid();
02026 void toggleRuler();
02027 void toggleLwt();
02028
02029 // Icons
02030 void iconResize(int iconSize);
02031
02032 //Selectors
02033 void layerSelectorIndexChanged(int index);
02034 void colorSelectorIndexChanged(int index);
02035 void linetypeSelectorIndexChanged(int index);
02036 void linewidthSelectorIndexChanged(int index);
02037 void textFontSelectorCurrentFontChanged(const QFont& font);
02038 void textSizeSelectorIndexChanged(int index);
02039
02040 QString textFont();
02041 EmbReal textSize();
02042 EmbReal textAngle();
02043 bool textBold();
02044 bool textItalic();
02045 bool textUnderline();
02046 bool textStrikeOut();
02047 bool textOverline();
02048
02049 void setTextFont(const QString& str);
02050 void setTextSize(EmbReal num);
02051 void setTextAngle(EmbReal num);
02052 void setTextBold(bool val);
02053 void setTextItalic(bool val);
02054 void setTextUnderline(bool val);
02055 void setTextStrikeOut(bool val);
02056 void setTextOverline(bool val);
02057
02058 QString getCurrentLayer();
02059 QRgb getCurrentColor();
02060 QString getCurrentLineType();
02061 QString getCurrentLineWeight();
02062
02063 // Standard Slots
02064 void undo();
02065 void redo();
02066
02067 bool isShiftPressed();
02068 void setShiftPressed();
02069 void setShiftReleased();
02070
02071 void deletePressed();
02072 void escapePressed();
02073
02074 // Layer Toolbar
02075 void makeLayerActive();
02076 void layerManager();
02077 void layerPrevious();
02078 // Zoom Toolbar
02079 void zoomRealtime();
02080 void zoomPrevious();
02081 void zoomWindow();
02082 void zoomDynamic();
02083 void zoomScale();
02084 void zoomCenter();
02085 void zoomIn();
02086 void zoomOut();
02087 void zoomSelected();
02088 void zoomAll();
02089 void zoomExtents();
02090 // Pan SubMenu
02091 void panrealtime();
02092 void panpoint();
02093 void panLeft();
02094 void panRight();
02095 void panUp();
02096 void panDown();

```

```

02097
02098     void dayVision();
02099     void nightVision();
02100
02101     void doNothing();
02102
02103 public:
02104     //Natives
02105     void nativeAlert(const QString& txt);
02106     void nativeBlinkPrompt();
02107     void nativeSetPromptPrefix(const QString& txt);
02108     void nativeAppendPromptHistory(const QString& txt);
02109     void nativeEnablePromptRapidFire();
02110     void nativeDisablePromptRapidFire();
02111     void nativeInitCommand();
02112     void nativeEndCommand();
02113
02114     void nativeEnableMoveRapidFire();
02115     void nativeDisableMoveRapidFire();
02116
02117     void nativeNewFile();
02118     void nativeOpenFile();
02119
02120     void nativeExit();
02121     void nativeTipOfTheDay();
02122     void nativeWindowCascade();
02123     void nativeWindowTile();
02124     void nativeWindowClose();
02125     void nativeWindowCloseAll();
02126     void nativeWindowNext();
02127     void nativeWindowPrevious();
02128
02129     void nativeMessageBox(const QString& type, const QString& title, const QString& text);
02130
02131     void nativePrintArea(EmbReal x, EmbReal y, EmbReal w, EmbReal h);
02132
02133     void nativeSetBackgroundColor(uint8_t r, uint8_t g, uint8_t b);
02134     void nativeSetCrossHairColor(uint8_t r, uint8_t g, uint8_t b);
02135     void nativeSetGridColor(uint8_t r, uint8_t g, uint8_t b);
02136
02137     QString nativeTextFont();
02138     EmbReal nativeTextSize();
02139     EmbReal nativeTextAngle();
02140     bool nativeTextBold();
02141     bool nativeTextItalic();
02142     bool nativeTextUnderline();
02143     bool nativeTextStrikeOut();
02144     bool nativeTextOverline();
02145
02146     void nativePreviewOn(int clone, int mode, EmbReal x, EmbReal y, EmbReal data);
02147     void nativePreviewOff();
02148
02149     void nativeVulcanize();
02150     void nativeClearRubber();
02151     bool nativeAllowRubber();
02152     void nativeSpareRubber(qint64 id);
02153     //TODO: void nativeSetRubberFilter(qint64 id); //TODO: This is so more than 1 rubber object can
02154     //exist at one time without updating all rubber objects at once
02155     void nativeSetRubberMode(int mode);
02156     void nativeSetRubberPoint(const QString& key, EmbReal x, EmbReal y);
02157     void nativeSetRubberText(const QString& key, const QString& txt);
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);
02182     void nativeAddPolygon(EmbReal startX, EmbReal startY, const QPainterPath& p, int rubberMode);

```

```

02174     void nativeAddPolyline(EmbReal startX, EmbReal startY, const QPainterPath& p, int rubberMode);
02175     void nativeAddPath(EmbReal startX, EmbReal startY, const QPainterPath& p, int rubberMode);
02176     void nativeAddHorizontalDimension(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal
02177         legHeight);
02178     void nativeAddVerticalDimension(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal
02179         legHeight);
02180     void nativeAddImage(const QString& img, EmbReal x, EmbReal y, EmbReal w, EmbReal h, EmbReal rot);
02181
02182     void nativeAddDimLeader(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal rot, int
02183         rubberMode);
02184
02185     void nativeSetCursorShape(const QString& str);
02186     EmbReal nativeCalculateAngle(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2);
02187     EmbReal nativeCalculateDistance(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2);
02188     EmbReal nativePerpendicularDistance(EmbReal px, EmbReal py, EmbReal x1, EmbReal y1, EmbReal x2,
02189         EmbReal y2);
02190
02191     int nativeNumSelected();
02192     void nativeSelectAll();
02193     void nativeAddToSelection(const QPainterPath path, Qt::ItemSelectionMode mode);
02194     void nativeClearSelection();
02195     void nativeDeleteSelected();
02196     void nativeCutSelected(EmbReal x, EmbReal y);
02197     void nativeCopySelected(EmbReal x, EmbReal y);
02198     void nativePasteSelected(EmbReal x, EmbReal y);
02199     void nativeMoveSelected(EmbReal dx, EmbReal dy);
02200     void nativeScaleSelected(EmbReal x, EmbReal y, EmbReal factor);
02201     void nativeRotateSelected(EmbReal x, EmbReal y, EmbReal rot);
02202     void nativeMirrorSelected(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2);
02203
02204 };
02205
02206 MainWindow* mainWin();
02207
02208
02209 class MdiWindow: public QMdiSubWindow
02210 {
02211     Q_OBJECT
02212
02213 public:
02214     MdiWindow(const int theIndex, MainWindow* mw, QMdiArea* parent, Qt::WindowFlags wflags);
02215     ~MdiWindow();
02216
02217     virtual QSize           sizeHint() const;
02218     QString    getCurrentFile() { return curFile; }
02219     QString    getShortCurrentFile();
02220     View*      getView() { return gview; }
02221     QGraphicsScene*   getScene() { return gscene; }
02222     QString    getCurrentLayer() { return curLayer; }
02223     QRgb     getCurrentColor() { return curColor; }
02224     QString    getCurrentLineType() { return curLineType; }
02225     QString    getCurrentLineWidth() { return curLineWidth; }
02226     void      setCurrentLayer(const QString& layer) { curLayer = layer; }
02227     void      setCurrentColor(const QRgb& color) { curColor = color; }
02228     void      setCurrentLineType(const QString& lineType) { curLineType = lineType; }
02229     void      setCurrentLineWidth(const QString& lineWidth) { curLineWidth = lineWidth; }
02230     void      designDetails();
02231     bool     loadFile(const QString &fileName);
02232     bool     saveFile(const QString &fileName);
02233 signals:
02234     void sendCloseMdiWin(MdiWindow* );
02235
02236 public slots:
02237     void closeEvent(QCloseEvent* e);
02238     void onWindowActivated();
02239
02240     void currentLayerChanged(const QString& layer);
02241     void currentColorChanged(const QRgb& color);
02242     void currentLinetypeChanged(const QString& type);
02243     void currentLinewidthChanged(const QString& weight);
02244
02245     void updateColorLinetypeLinewidth();
02246     void deletePressed();
02247     void escapePressed();
02248
02249     void showViewScrollBars(bool val);
02250     void setViewCrossHairColor(QRgb color);
02251     void setViewBackgroundColor(QRgb color);
02252     void setViewSelectBoxColors(QRgb colorL, QRgb fillL, QRgb colorR, QRgb fillR, int alpha);
02253     void setViewGridColor(QRgb color);
02254     void setViewRulerColor(QRgb color);
02255
02256     void print();

```

```
02257     void saveBMC();
02258
02259     void promptHistoryAppended(const QString& txt);
02260     void logPromptInput(const QString& txt);
02261     void promptInputPrevious();
02262     void promptInputNext();
02263
02264 protected:
02265
02266 private:
02267     MainWindow*           mainWin;
02268     QMdiArea*             mdiArea;
02269     QGraphicsScene*        gscene;
02270     View*                 gview;
02271
02272     bool fileWasLoaded;
02273
02274     QString promptHistory;
02275     QList<QString> promptInputList;
02276     int promptInputNum;
02277
02278     QPrinter               printer;
02279
02280     QString curFile;
02281     void setCurrentFile(const QString& fileName);
02282     QString fileExtension(const QString& fileName);
02283
02284     int myIndex;
02285
02286     QString curLayer;
02287     QRgb curColor;
02288     QString curLineType;
02289     QString curLineWeight;
02290
02291     void promptInputPrevNext(bool prev);
02292 };
02293
02294 class MdiArea : public QMdiArea
02295 {
02296     Q_OBJECT
02297
02298 public:
02299     MainWindow* mainWin;
02300
02301     bool useLogo;
02302     bool useTexture;
02303     bool useColor;
02304
02305     QPixmap bgLogo;
02306     QPixmap bgTexture;
02307     QColor   bgColor;
02308
02309     void zoomExtentsAllSubWindows();
02310     void forceRepaint();
02311
02312     MdiArea(MainWindow* mw, QWidget* parent = 0);
02313     ~MdiArea();
02314
02315     void useBackgroundLogo(bool use);
02316     void useBackgroundTexture(bool use);
02317     void useBackgroundColor(bool use);
02318
02319     void setBackgroundLogo(const QString& fileName);
02320     void setBackgroundTexture(const QString& fileName);
02321     void setBackgroundColor(const QColor& color);
02322
02323     public slots:
02324         void cascade();
02325         void tile();
02326     protected:
02327         virtual void mouseDoubleClickEvent(QMouseEvent* e);
02328         virtual void paintEvent(QPaintEvent* e);
02329 };
02330
02331 class PreviewDialog : public QFileDialog
02332 {
02333     Q_OBJECT
02334
02335 public:
02336     PreviewDialog(QWidget* parent = 0,
02337                   const QString& caption = QString(),
02338                   const QString& directory = QString(),
02339                   const QString& filter = QString());
02340     ~PreviewDialog();
02341
02342     ImageWidget* imgWidget;
02343 }
```

```

02350
02351
02352 class PropertyEditor : public QDockWidget
02353 {
02354     Q_OBJECT
02355
02356 public:
02357     PropertyEditor(const QString& iconDirectory = QString(), bool pickAddMode = true, QWidget*
02358     widgetToFocus = 0, QWidget* parent = 0); //, Qt::WindowFlags flags = 0);
02359     ~PropertyEditor();
02359
02360 protected:
02361     bool eventFilter(QObject *obj, QEvent *event);
02362
02363 signals:
02364     void pickAddModeToggled();
02365
02366 public slots:
02367     void setSelectedItems(QList<QGraphicsItem*> itemList);
02368     void updatePickAddModeButton(bool pickAddMode);
02369
02370 private slots:
02371     void fieldEdited(QObject* fieldObj);
02372     void showGroups(int objType);
02373     void showOneType(int index);
02374     void hideAllGroups();
02375     void clearAllFields();
02376     void togglePickAddMode();
02377
02378 private:
02379     QWidget* focusWidget;
02380
02381     QString iconDir;
02382     int iconSize;
02383     Qt::ToolButtonStyle propertyEditorButtonStyle;
02384
02385     bool pickAdd;
02386
02387     QList<QGraphicsItem*> selectedItemList;
02388
02389     ArcObject* tempArcObj;
02390     BlockObject* tempBlockObj;
02391     CircleObject* tempCircleObj;
02392     DimAlignedObject* tempDimAlignedObj;
02393     DimAngularObject* tempDimAngularObj;
02394     DimArcLengthObject* tempDimArcLenObj;
02395     DimDiameterObject* tempDimDiamObj;
02396     DimLeaderObject* tempDimLeaderObj;
02397     DimLinearObject* tempDimLinearObj;
02398     DimOrdinateObject* tempDimOrdObj;
02399     DimRadiusObject* tempDimRadiusObj;
02400     EllipseObject* tempEllipseObj;
02401     EllipseArcObject* tempEllipseArcObj;
02402     HatchObject* tempHatchObj;
02403     ImageObject* tempImageObj;
02404     InfiniteLineObject* tempInflLineObj;
02405     LineObject* tempLineObj;
02406     PathObject* tempPathObj;
02407     PointObject* tempPointObj;
02408     PolygonObject* tempPolygonObj;
02409     PolylineObject* tempPolylineObj;
02410     RayObject* tempRayObj;
02411     RectObject* tempRectObj;
02412     SplineObject* tempSplineObj;
02413     TextMultiObject* tempTextMultiObj;
02414     TextSingleObject* tempTextSingleObj;
02415
02416     //Helper functions
02417     QToolButton* createToolButton(const QString& iconName, const QString& txt);
02418     QLineEdit* createLineEdit(const QString& validatorType = QString(), bool readOnly = false);
02419     QComboBox* createComboBox(bool disable = false);
02420     QFontComboBox* createFontComboBox(bool disable = false);
02421
02422     int precisionAngle;
02423     int precisionLength;
02424
02425     //Used when checking if fields vary
02426     QString fieldOldText;
02427     QString fieldNewText;
02428     QString fieldVariesText;
02429     QString fieldYesText;
02430     QString fieldNoText;
02431     QString fieldOnText;
02432     QString fieldOffText;
02433
02434     void updateLineEditStrIfVaries(LineEdit* lineEdit, const QString& str);
02435     void updateLineEditNumIfVaries(LineEdit* lineEdit, EmbReal num, bool useAnglePrecision);

```

```

02436     void updateFontComboBoxStrIfVaries(QFontComboBox* fontComboBox, const QString& str);
02437     void updateComboBoxStrIfVaries(QComboBox* comboBox, const QString& str, const QStringList&
strList);
02438     void updateComboBoxBoolIfVaries(QComboBox* comboBox, bool val, bool yesOrNoText);
02439
02440     QSignalMapper* signalMapper;
02441     void mapSignal(QObject* fieldObj, const QString& name, QVariant value);
02442
02443 //=====
02444 //Selection
02445 //=====
02446     QComboBox* createComboBoxSelected();
02447     QToolButton* createToolButtonQSelect();
02448     QToolButton* createToolButtonPickAdd();
02449
02450     QComboBox* comboBoxSelected;
02451     QToolButton* toolButtonQSelect;
02452     QToolButton* toolButtonPickAdd;
02453
02454 //TODO: Alphabetic/Categorized TabWidget
02455
02456     QGroupBox* createGroupBoxGeneral();
02457
02458     QGroupBox* createGroupBoxGeometryArc();
02459     QGroupBox* createGroupBoxMiscArc();
02460     QGroupBox* createGroupBoxGeometryBlock();
02461     QGroupBox* createGroupBoxGeometryCircle();
02462     QGroupBox* createGroupBoxGeometryDimAligned();
02463     QGroupBox* createGroupBoxGeometryDimAngular();
02464     QGroupBox* createGroupBoxGeometryDimArcLength();
02465     QGroupBox* createGroupBoxGeometryDimDiameter();
02466     QGroupBox* createGroupBoxGeometryDimLeader();
02467     QGroupBox* createGroupBoxGeometryDimLinear();
02468     QGroupBox* createGroupBoxGeometryDimOrdinate();
02469     QGroupBox* createGroupBoxGeometryDimRadius();
02470     QGroupBox* createGroupBoxGeometryEllipse();
02471     QGroupBox* createGroupBoxGeometryImage();
02472     QGroupBox* createGroupBoxMiscImage();
02473     QGroupBox* createGroupBoxGeometryInfiniteLine();
02474     QGroupBox* createGroupBoxGeometryLine();
02475     QGroupBox* createGroupBoxGeometryPath();
02476     QGroupBox* createGroupBoxMiscPath();
02477     QGroupBox* createGroupBoxGeometryPoint();
02478     QGroupBox* createGroupBoxGeometryPolygon();
02479     QGroupBox* createGroupBoxGeometryPolyline();
02480     QGroupBox* createGroupBoxMiscPolyline();
02481     QGroupBox* createGroupBoxGeometryRay();
02482     QGroupBox* createGroupBoxGeometryRectangle();
02483     QGroupBox* createGroupBoxGeometryTextMulti();
02484     QGroupBox* createGroupBoxTextTextSingle();
02485     QGroupBox* createGroupBoxGeometryTextSingle();
02486     QGroupBox* createGroupBoxMiscTextSingle();
02487 };
02488
02489
02490 class SelectBox : public QRubberBand
02491 {
02492     Q_OBJECT
02493
02494 public:
02495     SelectBox(Shape s, QWidget* parent = 0);
02496
02497     QColor leftBrushColor;
02498     QColor rightBrushColor;
02499     QColor leftPenColor;
02500     QColor rightPenColor;
02501     quint8 alpha;
02502
02503     QBrush dirBrush;
02504     QBrush leftBrush;
02505     QBrush rightBrush;
02506
02507     QPen dirPen;
02508     QPen leftPen;
02509     QPen rightPen;
02510
02511     bool boxDir;
02512
02513     void forceRepaint();
02514
02515 public slots:
02516     void setDirection(int dir);
02517     void setColors(const QColor& colorL, const QColor& fillL, const QColor& colorR, const QColor&
fillR, int newAlpha);
02518
02519 protected:
02520     void paintEvent(QPaintEvent* );

```

```

02521 };
02522
02523 class Settings_Dialog : public QDialog
02524 {
02525     Q_OBJECT
02526
02527 public:
02528     Settings_Dialog(MainWindow* mw, const QString& showTab = QString(), QWidget *parent = 0);
02529     ~Settings_Dialog();
02530
02531     MainWindow* mainWin;
02532
02533     QTabWidget* tabWidget;
02534
02535     QWidget* createTabGeneral();
02536     QWidget* createTabFilePaths();
02537     QWidget* createTabDisplay();
02538     QWidget* createTabPrompt();
02539     QWidget* createTabOpenSave();
02540     QWidget* createTabPrinting();
02541     QWidget* createTabSnap();
02542     QWidget* createTabGridRuler();
02543     QWidget* createTabOrthoPolar();
02544     QWidget* createTabQuickSnap();
02545     QWidget* createTabQuickTrack();
02546     QWidget* createTabLineWeight();
02547     QWidget* createTabSelection();
02548
02549     QDialogButtonBox* buttonBox;
02550
02551     void addColorsToComboBox(QComboBox* comboBox);
02552
02553
02554 //Temporary for instant preview
02555     bool preview_general_mdi_bg_use_logo;
02556     bool preview_general_mdi_bg_use_texture;
02557     bool preview_general_mdi_bg_use_color;
02558
02559     QString accept_general_mdi_bg_logo;
02560     QString accept_general_mdi_bg_texture;
02561     QRgb preview_general_mdi_bg_color;
02562     QRgb accept_general_mdi_bg_color;
02563
02564     bool preview_display_show_scrollbars;
02565
02566     QRgb preview_display_crosshair_color;
02567     QRgb accept_display_crosshair_color;
02568     QRgb preview_display_bg_color;
02569     QRgb accept_display_bg_color;
02570
02571     QRgb preview_display_selectbox_left_color;
02572     QRgb accept_display_selectbox_left_color;
02573     QRgb preview_display_selectbox_left_fill;
02574     QRgb accept_display_selectbox_left_fill;
02575     QRgb preview_display_selectbox_right_color;
02576     QRgb accept_display_selectbox_right_color;
02577     QRgb preview_display_selectbox_right_fill;
02578     QRgb accept_display_selectbox_right_fill;
02579     QRgb preview_display_selectbox_right_fill;
02580     QRgb accept_display_selectbox_right_fill;
02581     quint8 preview_display_selectbox_alpha;
02582
02583     QRgb preview_prompt_text_color;
02584     QRgb accept_prompt_text_color;
02585
02586     QRgb preview_prompt_bg_color;
02587     QRgb accept_prompt_bg_color;
02588
02589     QString preview_prompt_font_family;
02590     QString preview_prompt_font_style;
02591     quint8 preview_prompt_font_size;
02592
02593     QRgb preview_grid_color;
02594     QRgb accept_grid_color;
02595
02596     QRgb preview_ruler_color;
02597     QRgb accept_ruler_color;
02598
02599     bool preview_lwt_show_lwt;
02600     bool preview_lwt_real_render;
02601
02602 //Temporary until changes are accepted
02603     QString dialog_general_language;
02604     QString dialog_general_icon_theme;
02605     int dialog_general_icon_size;
02606     bool dialog_general_mdi_bg_use_logo;
02607     bool dialog_general_mdi_bg_use_texture;
02608     bool dialog_general_mdi_bg_use_color;
02609     QString dialog_general_mdi_bg_logo;
02610     QString dialog_general_mdi_bg_texture;

```

```
02611 QRgb dialog_general_mdi_bg_color;
02612 bool dialog_general_tip_of_the_day;
02613 bool dialog_general_system_help_browser;
02614 bool dialog_display_use_opengl;
02615 bool dialog_display_renderhint_aa;
02616 bool dialog_display_renderhint_text_aa;
02617 bool dialog_display_renderhint_smooth_pix;
02618 bool dialog_display_renderhint_high_aa;
02619 bool dialog_display_renderhint_noncosmetic;
02620 bool dialog_display_show_scrollbars;
02621 int dialog_display_scrollbar_widget_num;
02622 QRgb dialog_display_crosshair_color;
02623 QRgb dialog_display_bg_color;
02624 QRgb dialog_display_selectbox_left_color;
02625 QRgb dialog_display_selectbox_left_fill;
02626 QRgb dialog_display_selectbox_right_color;
02627 QRgb dialog_display_selectbox_right_fill;
02628 quint8 dialog_display_selectbox_alpha;
02629 EmbReal dialog_display_zoomscale_in;
02630 EmbReal dialog_display_zoomscale_out;
02631 quint8 dialog_display_crosshair_percent;
02632 QString dialog_display_units;
02633 QRgb dialog_prompt_text_color;
02634 QRgb dialog_prompt_bg_color;
02635 QString dialog_prompt_font_family;
02636 QString dialog_prompt_font_style;
02637 quint8 dialog_prompt_font_size;
02638 bool dialog_prompt_save_history;
02639 bool dialog_prompt_save_history_as_html;
02640 QString dialog_prompt_save_history_filename;
02641 QString dialog_opensave_custom_filter;
02642 QString dialog_opensave_open_format;
02643 bool dialog_opensave_open_thumbnail;
02644 QString dialog_opensave_save_format;
02645 bool dialog_opensave_save_thumbnail;
02646 quint8 dialog_opensave_recent_max_files;
02647 quint8 dialog_opensave_trim_dst_num_jumps;
02648 QString dialog_printing_default_device;
02649 bool dialog_printing_use_last_device;
02650 bool dialog_printing_disable_bg;
02651 bool dialog_grid_show_on_load;
02652 bool dialog_grid_show_origin;
02653 bool dialog_grid_color_match_crosshair;
02654 QRgb dialog_grid_color;
02655 bool dialog_grid_load_from_file;
02656 QString dialog_grid_type;
02657 bool dialog_grid_center_on_origin;
02658 EmbReal dialog_grid_center_x;
02659 EmbReal dialog_grid_center_y;
02660 EmbReal dialog_grid_size_x;
02661 EmbReal dialog_grid_size_y;
02662 EmbReal dialog_grid_spacing_x;
02663 EmbReal dialog_grid_spacing_y;
02664 EmbReal dialog_grid_size_radius;
02665 EmbReal dialog_grid_spacing_radius;
02666 EmbReal dialog_grid_spacing_angle;
02667 bool dialog_ruler_show_on_load;
02668 bool dialog_ruler_metric;
02669 QRgb dialog_ruler_color;
02670 quint8 dialog_ruler_pixel_size;
02671 bool dialog_qsnap_enabled;
02672 QRgb dialog_qsnap_locator_color;
02673 quint8 dialog_qsnap_locator_size;
02674 quint8 dialog_qsnap_aperture_size;
02675 bool dialog_qsnap_endpoint;
02676 bool dialog_qsnap_midpoint;
02677 bool dialog_qsnap_center;
02678 bool dialog_qsnap_node;
02679 bool dialog_qsnap_quadrant;
02680 bool dialog_qsnap_intersection;
02681 bool dialog_qsnap_extension;
02682 bool dialog_qsnap_insertion;
02683 bool dialog_qsnap_perpendicular;
02684 bool dialog_qsnap_tangent;
02685 bool dialog_qsnap_nearest;
02686 bool dialog_qsnap_apparent;
02687 bool dialog_qsnap_parallel;
02688 bool dialog_lwt_show_lwt;
02689 bool dialog_lwt_real_render;
02690 EmbReal dialog_lwt_default_lwt;
02691 bool dialog_selection_mode_pickfirst;
02692 bool dialog_selection_mode_pickadd;
02693 bool dialog_selection_mode_pickedrag;
02694 QRgb dialog_selection_coolgrip_color;
02695 QRgb dialog_selection_hotgrip_color;
02696 quint8 dialog_selection_grip_size;
02697 quint8 dialog_selection_pickbox_size;
```

```

02698 private slots:
02699 void comboBoxLanguageCurrentIndexChanged(const QString&);
02700 void comboBoxIconThemeCurrentIndexChanged(const QString&);
02701 void comboBoxIconSizeCurrentIndexChanged(int);
02702 void checkBoxGeneralMdiBGUseLogoStateChanged(int);
02703 void chooseGeneralMdiBackgroundLogo();
02704 void checkBoxGeneralMdiBGUseTextureStateChanged(int);
02705 void chooseGeneralMdiBackgroundTexture();
02706 void checkBoxGeneralMdiBGUseColorStateChanged(int);
02707 void chooseGeneralMdiBackgroundColor();
02708 void currentGeneralMdiBackgroundColorChanged(const QColor&);
02709 void checkBoxTipOfTheDayStateChanged(int);
02710 void checkBoxUseOpenGLStateChanged(int);
02711 void checkBoxRenderHintAAStateChanged(int);
02712 void checkBoxRenderHintTextAAStateChanged(int);
02713 void checkBoxRenderHintSmoothPixStateChanged(int);
02714 void checkBoxRenderHintHighAAStateChanged(int);
02715 void checkBoxRenderHintNonCosmeticStateChanged(int);
02716 void checkBoxShowScrollBarsStateChanged(int);
02717 void comboBoxScrollBarWidgetCurrentIndexChanged(int);
02718 void spinBoxZoomScaleInValueChanged(double);
02719 void spinBoxZoomScaleOutValueChanged(double);
02720 void checkBoxDisableBGStateChanged(int);
02721 void chooseDisplayCrossHairColor();
02722 void currentDisplayCrossHairColorChanged(const QColor&);
02723 void chooseDisplayBackgroundColor();
02724 void currentDisplayBackgroundColorChanged(const QColor&);
02725 void chooseDisplaySelectBoxLeftColor();
02726 void currentDisplaySelectBoxLeftColorChanged(const QColor&);
02727 void chooseDisplaySelectBoxLeftFill();
02728 void currentDisplaySelectBoxLeftFillChanged(const QColor&);
02729 void chooseDisplaySelectBoxRightColor();
02730 void currentDisplaySelectBoxRightColorChanged(const QColor&);
02731 void chooseDisplaySelectBoxRightFill();
02732 void currentDisplaySelectBoxRightFillChanged(const QColor&);
02733 void spinBoxDisplaySelectBoxAlphaValueChanged(int);
02734 void choosePromptTextColor();
02735 void currentPromptTextColorChanged(const QColor&);
02736 void choosePromptBackgroundColor();
02737 void currentPromptBackgroundColorChanged(const QColor&);
02738 void comboBoxPromptFontFamilyCurrentIndexChanged(const QString&);
02739 void comboBoxPromptFontStyleCurrentIndexChanged(const QString&);
02740 void spinBoxPromptFontSizeValueChanged(int);
02741 void checkBoxPromptSaveHistoryStateChanged(int);
02742 void checkBoxPromptSaveHistoryAsHtmlStateChanged(int);
02743 void checkBoxCustomFilterSelectAllClicked();
02744 void buttonCustomFilterClearAllClicked();
02745 void spinBoxRecentMaxFilesValueChanged(int);
02746 void spinBoxTrimDstNumJumpsValueChanged(int);
02747 void checkBoxGridShowOnLoadStateChanged(int);
02748 void checkBoxGridShowOriginStateChanged(int);
02749 void checkBoxGridColorMatchCrossHairStateChanged(int);
02750 void chooseGridColor();
02751 void currentGridColorChanged(const QColor&);
02752 void checkBoxGridLoadFromFileStateChanged(int);
02753 void comboBoxGridTypeCurrentIndexChanged(const QString&);
02754 void checkBoxGridCenterOnOriginStateChanged(int);
02755 void spinBoxGridCenterXValueChanged(double);
02756 void spinBoxGridCenterYValueChanged(double);
02757 void spinBoxGridSizeXValueChanged(double);
02758 void spinBoxGridSizeYValueChanged(double);
02759 void spinBoxGridSpacingXValueChanged(double);
02760 void spinBoxGridSpacingYValueChanged(double);
02761 void spinBoxGridRadiusValueChanged(double);
02762 void spinBoxGridAngleValueChanged(double);
02763 void checkBoxRulerShowOnLoadStateChanged(int);
02764 void comboBoxRulerMetricCurrentIndexChanged(int);
02765 void chooseRulerColor();
02766 void currentRulerColorChanged(const QColor&);
02767 void spinBoxRulerPixelSizeValueChanged(double);
02768 void checkBoxQSnapEndPointStateChanged(int);
02769 void checkBoxQSnapMidPointStateChanged(int);
02770 void checkBoxQSnapCenterStateChanged(int);
02771 void checkBoxQSnapNodeStateChanged(int);
02772 void checkBoxQSnapQuadrantStateChanged(int);
02773 void checkBoxQSnapIntersectionStateChanged(int);
02774 void checkBoxQSnapExtensionStateChanged(int);
02775 void checkBoxQSnapInsertionStateChanged(int);
02776 void checkBoxQSnapPerpendicularStateChanged(int);
02777 void checkBoxQSnapTangentStateChanged(int);
02778 void checkBoxQSnapNearestStateChanged(int);
02779 void checkBoxQSnapApparentStateChanged(int);
02780 void checkBoxQSnapParallelStateChanged(int);
02781 void checkBoxQSnapSelectAllClicked();

```

```
02785     void buttonQSnapClearAllClicked();
02786     void comboBoxQSnapLocatorColorCurrentIndexChanged(int);
02787     void sliderQSnapLocatorSizeValueChanged(int);
02788     void sliderQSnapApertureSizeValueChanged(int);
02789     void checkBoxLwtShowLwtStateChanged(int);
02790     void checkBoxLwtRealRenderStateStateChanged(int);
02791     void checkBoxSelectionModePickFirstStateChanged(int);
02792     void checkBoxSelectionModePickAddStateChanged(int);
02793     void checkBoxSelectionModePickDragStateChanged(int);
02794     void comboBoxSelectionCoolGripColorCurrentIndexChanged(int);
02795     void comboBoxSelectionHotGripColorCurrentIndexChanged(int);
02796     void sliderSelectionGripSizeValueChanged(int);
02797     void sliderSelectionPickBoxSizeValueChanged(int);
02798
02799     void acceptChanges();
02800     void rejectChanges();
02801
02802 signals:
02803     void buttonCustomFilterSelectAll(bool);
02804     void buttonCustomFilterClearAll(bool);
02805     void buttonQSnapSelectAll(bool);
02806     void buttonQSnapClearAll(bool);
02807 };
02808
02809
02810 class StatusBar : public QStatusBar
02811 {
02812     Q_OBJECT
02813
02814 public:
02815     StatusBar(MainWindow* mw, QWidget* parent = 0);
02816
02817     StatusBarButton* statusBarSnapButton;
02818     StatusBarButton* statusBarGridButton;
02819     StatusBarButton* statusBarRulerButton;
02820     StatusBarButton* statusBarOrthoButton;
02821     StatusBarButton* statusBarPolarButton;
02822     StatusBarButton* statusBarQSnapButton;
02823     StatusBarButton* statusBarQTrackButton;
02824     StatusBarButton* statusBarLwtButton;
02825     QLabel* statusBarMouseCoord;
02826
02827     void setMouseCoord(EmbReal x, EmbReal y);
02828 };
02829
02830
02831 class StatusBarButton : public QToolButton
02832 {
02833     Q_OBJECT
02834
02835     Q_OBJECT
02836
02837 public:
02838     StatusBarButton(QString buttonText, MainWindow* mw, StatusBar* statbar, QWidget *parent = 0);
02839
02840     MainWindow* mainWin;
02841     StatusBar* statusbar;
02842
02843 protected:
02844     void contextMenuEvent(QContextMenuEvent *event = 0);
02845
02846 private slots:
02847     void settingsSnap();
02848     void settingsGrid();
02849     void settingsRuler();
02850     void settingsOrtho();
02851     void settingsPolar();
02852     void settingsQSnap();
02853     void settingsQTrack();
02854     void settingsLwt();
02855     void toggleSnap(bool on);
02856     void toggleGrid(bool on);
02857     void toggleRuler(bool on);
02858     void toggleOrtho(bool on);
02859     void togglePolar(bool on);
02860     void toggleQSnap(bool on);
02861     void toggleQTrack(bool on);
02862     void toggleLwt(bool on);
02863 public slots:
02864     void enableLwt();
02865     void disableLwt();
02866     void enableReal();
02867     void disableReal();
02868 };
02869
02870
02871 class UndoEditor : public QDockWidget
02872 {
02873     Q_OBJECT
02874
02875     Q_OBJECT
02876
02877 public:
```

```

02878     UndoEditor(const QString& iconDirectory = QString(), QWidget* widgetToFocus = 0, QWidget* parent =
02879     0); //, Qt::WindowFlags flags = 0);
02880     ~UndoEditor();
02881     void addStack(QUndoStack* stack);
02882     bool canUndo() const;
02883     bool canRedo() const;
02884     QWidget* focusWidget;
02885     QString iconDir;
02886     int iconSize;
02887     QUndoGroup* undoGroup;
02888     QUndoView* undoView;
02889     QString undoText() const;
02890     QString redoText() const;
02891     protected:
02892     public slots:
02893     void undo();
02894     void redo();
02895     void updateCleanIcon(bool opened);
02896 };
02897
02898 class UndoableAddCommand : public QUndoCommand
02899 {
02900     public:
02901         UndoableAddCommand(const QString& text, BaseObject* obj, View* v, QUndoCommand* parent = 0);
02902         void undo();
02903         void redo();
02904         BaseObject* object;
02905         View* gview;
02906     };
02907
02908 class UndoableDeleteCommand : public QUndoCommand
02909 {
02910     public:
02911         UndoableDeleteCommand(const QString& text, BaseObject* obj, View* v, QUndoCommand* parent = 0);
02912         void undo();
02913         void redo();
02914         BaseObject* object;
02915         View* gview;
02916     };
02917
02918 class UndoableMoveCommand : public QUndoCommand
02919 {
02920     public:
02921         UndoableMoveCommand(EmbReal deltaX, EmbReal deltaY, const QString& text, BaseObject* obj, View* v,
02922         QUndoCommand* parent = 0);
02923         void undo();
02924         void redo();
02925         BaseObject* object;
02926         View* gview;
02927     };
02928
02929 class UndoableRotateCommand : public QUndoCommand
02930 {
02931     public:
02932         UndoableRotateCommand(EmbReal pivotPointX, EmbReal pivotPointY, EmbReal rotAngle, const QString&
02933         text, BaseObject* obj, View* v, QUndoCommand* parent = 0);
02934         void undo();
02935         void redo();
02936         void rotate(EmbReal x, EmbReal y, EmbReal rot);
02937         BaseObject* object;
02938         View* gview;
02939         EmbReal pivotX;
02940         EmbReal pivotY;
02941         EmbReal angle;
02942     };
02943
02944 class UndableScaleCommand : public QUndoCommand
02945 {

```

```
02977 public:
02978     UndoableScaleCommand(EmbReal x, EmbReal y, EmbReal scaleFactor, const QString& text, BaseObject*
02979     obj, View* v, QUndoCommand* parent = 0);
02980     void undo();
02981     void redo();
02982
02983     BaseObject* object;
02984     View* gview;
02985     EmbReal dx;
02986     EmbReal dy;
02987     EmbReal factor;
02988 };
02989
02993 class UndoableNavCommand : public QUndoCommand
02994 {
02995 public:
02996     UndoableNavCommand(const QString& type, View* v, QUndoCommand* parent = 0);
02997
02998     int id() const { return 1234; }
02999     bool mergeWith(const QUndoCommand* command);
03000     void undo();
03001     void redo();
03002
03003     QString navType;
03004     QTransform fromTransform;
03005     QTransform toTransform;
03006     QPointF fromCenter;
03007     QPointF toCenter;
03008     bool done;
03009     View* gview;
03010 };
03011
03015 class UndoableGripEditCommand : public QUndoCommand
03016 {
03017 public:
03018     UndoableGripEditCommand(const QPointF beforePoint, const QPointF afterPoint, const QString& text,
03019     BaseObject* obj, View* v, QUndoCommand* parent = 0);
03020     void undo();
03021     void redo();
03022
03023     BaseObject* object;
03024     View* gview;
03025     QPointF before;
03026     QPointF after;
03027 };
03028
03032 class UndoableMirrorCommand : public QUndoCommand
03033 {
03034 public:
03035     UndoableMirrorCommand(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, const QString& text,
03036     BaseObject* obj, View* v, QUndoCommand* parent = 0);
03037     void undo();
03038     void redo();
03039     void mirror();
03040
03041     BaseObject* object;
03042     View* gview;
03043     QLineF mirrorLine;
03044
03045 };
03046
03050 class View : public QGraphicsView
03051 {
03052     Q_OBJECT
03053
03054 public:
03055     View(MainWindow* mw, QGraphicsScene* theScene, QWidget* parent);
03056     ~View();
03057
03058     bool allowZoomIn();
03059     bool allowZoomOut();
03060
03061     void recalculateLimits();
03062     void zoomToPoint(const QPoint& mousePoint, int zoomDir);
03063     void centerAt(const QPointF& centerPoint);
03064     QPointF center() { return mapToScene(rect().center()); }
03065
03066     QUndoStack* getUndoStack() { return undoStack; }
03067     void addObject(BaseObject* obj);
03068     void deleteObject(BaseObject* obj);
03069     void vulcanizeObject(BaseObject* obj);
03070
03071 public slots:
03072     void zoomIn();
```

```

03073     void zoomOut();
03074     void zoomWindow();
03075     void zoomSelected();
03076     void zoomExtents();
03077     void panRealTime();
03078     void panPoint();
03079     void panLeft();
03080     void panRight();
03081     void panUp();
03082     void panDown();
03083     void selectAll();
03084     void selectionChanged();
03085     void clearSelection();
03086     void deleteSelected();
03087     void moveSelected(EmbReal dx, EmbReal dy);
03088     void cut();
03089     void copy();
03090     void paste();
03091     void repeatAction();
03092     void moveAction();
03093     void scaleAction();
03094     void scaleSelected(EmbReal x, EmbReal y, EmbReal factor);
03095     void rotateAction();
03096     void rotateSelected(EmbReal x, EmbReal y, EmbReal rot);
03097     void mirrorSelected(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2);
03098     int numSelected();
03099
03100    void deletePressed();
03101    void escapePressed();
03102
03103    void cornerButtonClicked();
03104
03105    void showScrollBars(bool val);
03106    void setCornerButton();
03107    void setCrossHairColor(QRgb color);
03108    void setCrossHairSize(quint8 percent);
03109    void setBackgroundColor(QRgb color);
03110    void setSelectBoxColors(QRgb colorL, QRgb fillL, QRgb colorR, QRgb fillR, int alpha);
03111    void toggleSnap(bool on);
03112    void toggleGrid(bool on);
03113    void toggleRuler(bool on);
03114    void toggleOrtho(bool on);
03115    void togglePolar(bool on);
03116    void toggleQSnap(bool on);
03117    void toggleQTrack(bool on);
03118    void toggleLwt(bool on);
03119    void toggleReal(bool on);
03120    bool isLwtEnabled();
03121    bool isRealEnabled();
03122
03123    void setGridColor(QRgb color);
03124    void createGrid(const QString& gridType);
03125    void setRulerColor(QRgb color);
03126
03127    void previewOn(int clone, int mode, EmbReal x, EmbReal y, EmbReal data);
03128    void previewOff();
03129
03130    void enableMoveRapidFire();
03131    void disableMoveRapidFire();
03132
03133    bool allowRubber();
03134    void addToRubberRoom(QGraphicsItem* item);
03135    void vulcanizeRubberRoom();
03136    void clearRubberRoom();
03137    void spareRubber(qint64 id);
03138    void setRubberMode(int mode);
03139    void setRubberPoint(const QString& key, const QPointF& point);
03140    void setRubberText(const QString& key, const QString& txt);
03141
03142 protected:
03143    void mouseDoubleClickEvent(QMouseEvent* event);
03144    void mousePressEvent(QMouseEvent* event);
03145    void mouseMoveEvent(QMouseEvent* event);
03146    void mouseReleaseEvent(QMouseEvent* event);
03147    void wheelEvent(QWheelEvent* event);
03148    void contextMenuEvent(QContextMenuEvent* event);
03149    void drawBackground(QPainter* painter, const QRectF& rect);
03150    void drawForeground(QPainter* painter, const QRectF& rect);
03151    void enterEvent(QEvent* event);
03152
03153 private:
03154    QHash<qint64, QGraphicsItem*> hashDeletedObjects;
03155
03156    QList<qint64> spareRubberList;
03157
03158    QColor gridColumn;
03159    QPainterPath gridPath;

```

```
03160     void createGridRect();
03161     void createGridPolar();
03162     void createGridIso();
03163     QPainterPath originPath;
03164     void createOrigin();
03165
03166     bool rulerMetric;
03167     QColor rulerColor;
03168     quint8 rulerPixelSize;
03169     void loadRulerSettings();
03170
03171     bool willUnderflowInt32(qint64 a, qint64 b);
03172     bool willOverflowInt32(qint64 a, qint64 b);
03173     int roundToMultiple(bool roundUp, int numToRound, int multiple);
03174     QPainterPath createRulerTextPath(float x, EmbReal y, QString str, EmbReal height);
03175
03176     QList<QGraphicsItem*> previewObjectList;
03177     QGraphicsItemGroup* previewObjectItemGroup;
03178     QPointF previewPoint;
03179     EmbReal previewData;
03180     int previewMode;
03181
03182     QList<QGraphicsItem*> createObjectList(QList<QGraphicsItem*> list);
03183     QPointF cutCopyMousePoint;
03184     QGraphicsItemGroup* pasteObjectItemGroup;
03185     QPointF pasteDelta;
03186
03187     QList<QGraphicsItem*> rubberRoomList;
03188
03189     void copySelected();
03190
03191     bool grippingActive;
03192     bool rapidMoveActive;
03193     bool previewActive;
03194     bool pastingActive;
03195     bool movingActive;
03196     bool selectingActive;
03197     bool zoomWindowActive;
03198     bool panningRealTimeActive;
03199     bool panningPointActive;
03200     bool panningActive;
03201     bool qSnapActive;
03202     bool qSnapToggle;
03203
03204     void startGripping(BaseObject* obj);
03205     void stopGripping(bool accept = false);
03206
03207     BaseObject* gripBaseObj;
03208     BaseObject* tempBaseObj;
03209
03210     MainWindow* mainWin;
03211     QGraphicsScene* gscene;
03212     QUndoStack* undoStack;
03213
03214     SelectBox* selectBox;
03215     QPointF scenePressPoint;
03216     QPoint pressPoint;
03217     QPointF sceneMovePoint;
03218     QPoint movePoint;
03219     QPointF sceneReleasePoint;
03220     QPoint releasePoint;
03221     QPointF sceneGripPoint;
03222
03223     void updateMouseCoords(int x, int y);
03224     QPoint viewMousePoint;
03225     QPoint sceneMousePoint;
03226     QRgb qsnapLocatorColor;
03227     quint8 qsnapLocatorSize;
03228     quint8 qsnapApertureSize;
03229     QRgb gripColorCool;
03230     QRgb gripColorHot;
03231     quint8 gripSize;
03232     quint8 pickBoxSize;
03233     QRgb crosshairColor;
03234     quint32 crosshairSize;
03235
03236     void panStart(const QPoint& point);
03237     int panDistance;
03238     int panStartX;
03239     int panStartY;
03240
03241     void alignScenePointWithViewPoint(const QPointF& scenePoint, const QPoint& viewPoint);
03242 };
03243
03244 typedef struct Action_ {
03245     int hash;
03246     /*< Index in the actionHash array. */
```

```
03250     std::string icon;
03251     /*< The stub used for the icon and the basic command. */
03252     std::string tooltip;
03253     /*< The label in the menus and the message that appears when
03254      you hover over an icon. */
03255     std::string statustip;
03256     /*< The message that appears at the bottom of the . */
03257     std::string shortcut;
03258     /*< The keyboard shortcut for this action. */
03259     std::vector<std::string> aliases;
03260     /*< A list of all alternative commands, if empty only
03261      the icon string will be . */
03262     std::vector<std::string> script;
03263     /*< If this is a compound action this will be a
03264      list of commands or it can allow for command line
03265      style command aliases. For example: icon16 would become
03266      the string list {"iconResize 16"}.
03267 } Action;
03268
03269 int get_action_index(std::string cmd);
03270
03271 /* */
03272 extern Settings settings;
03273 extern Settings dialog;
03274 extern std::vector<Action> action_table;
03275 extern std::vector<std::string> file_toolbar;
03276 extern std::vector<std::string> edit_toolbar;
03277 extern std::vector<std::string> view_toolbar;
03278 extern std::vector<std::string> zoom_toolbar;
03279
03280 #endif
```

18.7 embroidermodder2/imagewidget.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"
```

Functions

- `QString SettingsDir ()`
- `QString SettingsPath ()`
- `std::vector< std::string > to_string_vector (QStringList list)`
- `void read_configuration ()`

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)

18.12 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

- void [c_split](#) (char input[200], int *argc, char argv[10][200])
- void [simplify_path](#) (char *path)

Simplifies a path by removing the .. and . symbols in place.
- 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`
- `Index * menu_layout`
- `Index * toolbar_layout`
- `EmbView views [50]`
- `int n_views = 0`
- `Settings dialog`
- `Settings preview`
- `Dictionary * translation_table`
- `EmbView * active_view = NULL`

18.13.1 Typedef Documentation

18.13.1.1 Parameter `typedef struct Parameter_ Parameter`

18.13.2 Function Documentation

```
18.13.2.1 c_split() void c_split (
    char input[200],
    int * argc,
    char argv[10][200] )
```

```
18.13.2.2 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.

```
18.13.2.3 Error() std::string Error (
    Parameter args[10] )
"debug": qDebug("%s", qPrintable(result[0].s_value));
```

```
18.13.2.4 mainWin() MainWindow * mainWin ( )
mainWin
```

Returns

```
18.13.2.5 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.

```
18.13.2.6 simplify_path() void simplify_path (
    char * path )
```

Simplifies a path by removing the .. and . symbols in place.

path The character array to operate on.

```
18.13.2.7 Todo() std::string Todo (
    Parameter result[10] )
```

```
18.13.2.8 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 active_view EmbView* active_view = NULL
```

The view focussed (that is the last view to have a click or keypress sent): this has to be manually set whenever it changes including being set to NULL when all views are closed.

```
18.13.3.5 CIRCLE_MODE_1P_DIA_ const int CIRCLE_MODE_1P_DIA_ = 1 [static]
```

```
18.13.3.6 CIRCLE_MODE_1P_RAD_ const int CIRCLE_MODE_1P_RAD_ = 0 [static]
```

```
18.13.3.7 CIRCLE_MODE_2P_ const int CIRCLE_MODE_2P_ = 2 [static]
```

```
18.13.3.8 CIRCLE_MODE_3P_ const int CIRCLE_MODE_3P_ = 3 [static]
```

```
18.13.3.9 CIRCLE_MODE_TTR_ const int CIRCLE_MODE_TTR_ = 4 [static]
```

```
18.13.3.10 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.

18.13.3.11 DOLPHIN_MODE_NUM_POINTS_ const int DOLPHIN_MODE_NUM_POINTS_ = 0 [static]

18.13.3.12 DOLPHIN_MODE_XSCALE_ const int DOLPHIN_MODE_XSCALE_ = 1 [static]

18.13.3.13 DOLPHIN_MODE_YSCALE_ const int DOLPHIN_MODE_YSCALE_ = 2 [static]

18.13.3.14 menu_layout [Index*](#) menu_layout

18.13.3.15 n_views int n_views = 0

18.13.3.16 preview [Settings](#) preview

18.13.3.17 settings [Settings](#) settings

The actuator changes the program state via these global variables.

18.13.3.18 SINGLE_LINE_TEXT_MODE_JUSTIFY_ const int SINGLE_LINE_TEXT_MODE_JUSTIFY_ = 0 [static]

18.13.3.19 SINGLE_LINE_TEXT_MODE_RAPID_ const int SINGLE_LINE_TEXT_MODE_RAPID_ = 3 [static]

18.13.3.20 SINGLE_LINE_TEXT_MODE_SETFONT_ const int SINGLE_LINE_TEXT_MODE_SETFONT_ = 1 [static]

18.13.3.21 SINGLE_LINE_TEXT_MODE_SETGEOM_ const int SINGLE_LINE_TEXT_MODE_SETGEOM_ = 2 [static]

18.13.3.22 STAR_MODE_CENTER_PT_ const int STAR_MODE_CENTER_PT_ = 1 [static]

18.13.3.23 STAR_MODE_NUM_POINTS_ const int STAR_MODE_NUM_POINTS_ = 0 [static]

18.13.3.24 STAR_MODE_RAD_INNER_ const int STAR_MODE_RAD_INNER_ = 3 [static]

18.13.3.25 STAR_MODE_RAD_OUTER_ const int STAR_MODE_RAD_OUTER_ = 2 [static]

18.13.3.26 toolbar_layout [Index*](#) toolbar_layout

18.13.3.27 translation_table [Dictionary*](#) translation_table

18.13.3.28 `views` `EmbView` `views[50]`

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 * `comboBoxGeneralLineWidth`
- 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 [EmblInfiniteLine_](#)
- 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 threshhold)
- EMB_PUBLIC void embPattern_horizontal_fill (EmbPattern *pattern, EmblImage *, int threshhold)
- 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)
 - pattern fileName format*
- 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

18.42.1.79 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)`

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)`

18.42.3.67 embPattern_copyPolylineToStitchList() `EMB_PUBLIC void embPattern_copyPolylineToStitchList (EmbPattern * pattern)`

18.42.3.68 embPattern_copyStitchListToPolylines() `EMB_PUBLIC void embPattern_copyStitchListToPolylines (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 threshhold 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 threshold )
```

pattern image threshold

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 $(-30, 30) \times (-30, 30)$.

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_movePolylinesToStitchList (EmbPattern * pattern)`

18.42.3.87 embPattern_moveStitchListToPolylines() `EMB_PUBLIC void embPattern_moveStitchListToPolylines (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 $(1 \ 0) \cdot (a) = (x_a \ y_a) \cdot (0 \ 1) \cdot (b) = (y_b)$

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
00303 typedef struct EmbInfiniteLine_ {
00304     EmbVector position;
00305 } EmbInfiniteLine;
00306
00311 typedef struct EmbRay_ {
00312     EmbVector position;
00313 } EmbRay;
00314
00319 typedef struct EmbTextMulti_ {
00320     EmbVector position;
00321     char text[200];
00322 } EmbTextMulti;
00323
00328 typedef struct EmbTextSingle_ {
00329     EmbVector position;
00330     char text[200];
00331 } EmbTextSingle;
00332
00337 typedef struct EmbTime_
00338 {
00339     unsigned int year;
00340     unsigned int month;
00341     unsigned int day;
00342     unsigned int hour;
00343     unsigned int minute;
00344     unsigned int second;
00345 } EmbTime;
00346
00351 typedef struct EmbPoint_
00352 {
00353     EmbVector position;
00354     int lineType;
00355     EmbColor color;
00356 } EmbPoint;
00357
00362 typedef struct EmbLine_
00363 {
00364     EmbVector start;
00365     EmbVector end;
00366     int lineType;
00367     EmbColor color;
00368 } EmbLine;
00369
00374 typedef struct EmbPath_
00375 {
00376     EmbArray* pointList;
00377     EmbArray* flagList;
00378     int lineType;
00379     EmbColor color;
00380 } EmbPath;
00381
00386 typedef struct EmbStitch_
00387 {
00388     int flags;
00389     EmbReal x;
00390     EmbReal y;
00391     int color;
00393 } EmbStitch;
00394
00399 typedef struct EmbThread_
00400 {
00401     EmbColor color;
00402     char description[50];
00403     char catalogNumber[30];
00404 } EmbThread;
00405
00410 typedef struct thread_color_ {
00411     char name[22];
00412     unsigned int hex_code;
00413     int manufacturer_code;
00414 } thread_color;
00415
```

```

00420 typedef struct EmbArc_
00421 {
00422     EmbVector start;
00423     EmbVector mid;
00424     EmbVector end;
00425 } EmbArc;
00426
00427 typedef struct EmbRect_
00428 {
00429     EmbReal top;
00430     EmbReal left;
00431     EmbReal bottom;
00432     EmbReal right;
00433     EmbReal rotation;
00434     EmbReal radius;
00435 } EmbRect;
00436
00437 typedef struct EmbCircle_
00438 {
00439     EmbVector center;
00440     EmbReal radius;
00441 } EmbCircle;
00442
00443 typedef EmbPath EmbPolygon;
00444
00445 typedef EmbPath EmbPolyline;
00446
00447 typedef int EmbFlag;
00448
00449 typedef struct EmbSatinOutline_
00450 {
00451     int length;
00452     EmbArray* sidel;
00453     EmbArray* side2;
00454 } EmbSatinOutline;
00455
00456 typedef struct EmbEllipse_
00457 {
00458     EmbVector center;
00459     EmbVector radius;
00460     EmbReal rotation;
00461 } EmbEllipse;
00462
00463 typedef struct EmbBezier_ {
00464     EmbVector start;
00465     EmbVector control1;
00466     EmbVector control2;
00467     EmbVector end;
00468 } EmbBezier;
00469
00470 typedef struct EmbSpline_ {
00471     EmbArray *beziers;
00472 } EmbSpline;
00473
00474 typedef struct LSYSTEM {
00475     char axiom;
00476     char *alphabet;
00477     char *constants;
00478     char **rules;
00479 } L_system;
00480
00481 typedef struct EmbGeometry_ {
00482     union {
00483         EmbArc arc;
00484         EmbCircle circle;
00485         EmbColor color;
00486         EmbEllipse ellipse;
00487         EmbLine line;
00488         EmbPath path;
00489         EmbPoint point;
00490         EmbPolygon polygon;
00491         EmbPolyline polyline;
00492         EmbRect rect;
00493         EmbSpline spline;
00494         EmbVector vector;
00495     } object;
00496     EmbStitch stitch;
00497     EmbThread thread;
00498     int flag;
00499     int type;
00500     int lineType;
00501     EmbColor color;
00502 } EmbGeometry;
00503
00504 struct EmbArray_ {
00505     EmbGeometry *geometry;
00506     EmbStitch *stitch;
00507 }
```

```

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,              EmbReal* chordMidX,
00670             EmbReal* chordMidY,          EmbReal* sagitta,
00671             EmbReal* apothem,            EmbReal* incAngleInDegrees,
00672             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 EmbRect embGeometry_boundingRect(EmbGeometry *obj);
00712
00713 EMB_PUBLIC EmbPattern* embPattern_create(void);
00714 EMB_PUBLIC void embPattern_hideStitchesOverLength(EmbPattern* p, int length);
00715 EMB_PUBLIC void embPattern_fixColorCount(EmbPattern* p);
00716 EMB_PUBLIC int embPattern_addThread(EmbPattern* p, EmbThread thread);
00717 EMB_PUBLIC void embPattern_addStitchAbs(EmbPattern* p, EmbReal x, EmbReal y, int flags, int
00718             isAutoColorIndex);
00719 EMB_PUBLIC void embPattern_addStitchRel(EmbPattern* p, EmbReal dx, EmbReal dy, int flags, int
00720             isAutoColorIndex);
00721 EMB_PUBLIC void embPattern_changeColor(EmbPattern* p, int index);
00722 EMB_PUBLIC void embPattern_free(EmbPattern* p);
00723 EMB_PUBLIC void embPattern_scale(EmbPattern* p, EmbReal scale);
00724 EMB_PUBLIC void embPattern_totalStitchLength(EmbPattern *pattern);
00725 EMB_PUBLIC void embPattern_minimumStitchLength(EmbPattern *pattern);
00726 EMB_PUBLIC void embPattern_maximumStitchLength(EmbPattern *pattern);
00727 EMB_PUBLIC void embPattern_lengthHistogram(EmbPattern *pattern, int *bin, int NUMBINS);
00728 EMB_PUBLIC void embPattern_realStitches(EmbPattern *pattern);
00729 EMB_PUBLIC void embPattern_jumpStitches(EmbPattern *pattern);
00730 EMB_PUBLIC void embPattern_trimStitches(EmbPattern *pattern);
00731 EMB_PUBLIC void embPattern_calcBoundingBox(EmbPattern* p);
00732 EMB_PUBLIC void embPattern_flipHorizontal(EmbPattern* p);
00733 EMB_PUBLIC void embPattern_flipVertical(EmbPattern* p);
00734 EMB_PUBLIC void embPattern_combineJumpStitches(EmbPattern* p);
00735 EMB_PUBLIC void embPattern_correctForMaxStitchLength(EmbPattern* p, EmbReal maxStitchLength, EmbReal
00736             maxJumpLength);
00737 EMB_PUBLIC void embPattern_center(EmbPattern* p);
00738 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 threshhold);
00743 EMB_PUBLIC void embPattern_horizontal_fill(EmbPattern *pattern, EmbImage *, int threshhold);
00744 EMB_PUBLIC int embPattern_render(EmbPattern *pattern, char *fname);
00745 EMB_PUBLIC int 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[numberFormats];
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 #endif /* LIBEMBROIDERY_HEADER */

```

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_ANIMATEEMOTION 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 * CompoundFileDirectoryEntry (FILE *file)`
file
- `bcf_directory * CompoundFileDirectory (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 `writelnb` (`EmbPattern` *pattern, `FILE` *file)
- char `readInf` (`EmbPattern` *pattern, `FILE` *file)
- char `writelnf` (`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 0xFFFFFFFFE

18.44.1.5 CompoundFileSector_FAT_Sector #define CompoundFileSector_FAT_Sector 0xFFFFFFFFD

18.44.1.6 CompoundFileSector_FreeSector #define CompoundFileSector_FreeSector 0xFFFFFFFFF

18.44.1.7 CompoundFileSector_MaxRegSector #define CompoundFileSector_MaxRegSector 0xFFFFFFFFA
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

<code>CSV_EXPECT_NULL</code>	
<code>CSV_EXPECT_QUOTE1</code>	
<code>CSV_EXPECT_QUOTE2</code>	
<code>CSV_EXPECT_COMMAS</code>	

18.44.3.2 CSV_MODE `enum CSV_MODE`

Enumerator

<code>CSV_MODE_NULL</code>	
<code>CSV_MODE_COMMENT</code>	
<code>CSV_MODE_VARIABLE</code>	
<code>CSV_MODE_THREAD</code>	

Enumerator

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 )
```

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 )
```

f

Returns

short

```
18.44.4.49 fread_int32_be() int fread_int32_be (
    FILE * f )
```

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 **Embroid 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 CSV_EXPECT_
00377 {
00378     CSV_EXPECT_NULL,
00379     CSV_EXPECT_QUOTE1,
00380     CSV_EXPECT_QUOTE2,
00381     CSV_EXPECT_COMMA
00382 } CSV_EXPECT;
00383
00384 typedef enum CSV_MODE_
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 *****/
00421
```

```

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_SIERNINSKI_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_SIERNINSKI_TRIANGLE #define FLAG_SIERNINSKI_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_movePolylinesToStitch_list() void embPattern_movePolylinesToStitch_list (EmbPattern * p)

Moves all of the EmbPolylineObjectList data to Embstitch_list data for pattern (p).

18.124.2.36 embPattern_movingStitch_listToPolylines() void embPattern_movingStitch_listToPolylines (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. [480](#)
- [2] KDE Community. Projects/liberty/file formats/tajima ternary - kde community wiki. [480](#)
- [3] Google et al. Google java style guide. [23](#)
- [4] G. van Rossum and B. Warsaw. Python pep 7. [23](#)

Index

appName
 embroidermodder.cpp, 297
appVer
 embroidermodder.cpp, 297
_bcf_directory, 48
 dirEntries, 48
 maxNumberOfDirectoryEntries, 48
_bcf_directory_entry, 48
 childId, 48
 CLSID, 49
 colorFlag, 49
 creationTime, 49
 directoryEntryName, 49
 directoryEntryNameLength, 49
 leftSiblingId, 49
 modifiedTime, 49
 next, 49
 objectType, 49
 rightSiblingId, 49
 startingSectorLocation, 49
 stateBits, 49
 streamSize, 49
 streamSizeHigh, 49
_bcf_file, 49
 difat, 50
 directory, 50
 fat, 50
 header, 50
_bcf_file_difat, 50
 fatSectorCount, 50
 fatSectorEntries, 50
 sectorSize, 50
_bcf_file_fat, 50
 fatEntries, 51
 fatEntryCount, 51
 numberOfEntriesInFatSector, 51
_bcf_file_header, 51
 byteOrder, 51
 CLSID, 51
 firstDifatSectorLocation, 51
 firstDirectorySectorLocation, 51
 firstMiniFATSectorLocation, 52
 majorVersion, 52
 miniSectorShift, 52
 miniStreamCutoffSize, 52
 minorVersion, 52
 numberOfDifatSectors, 52
 numberOfDirectorySectors, 52
 numberOfFATSectors, 52
 numberOfMiniFatSectors, 52
 reserved1, 52
 reserved2, 52
 sectorShift, 52
 signature, 52
 transactionSignatureNumber, 52
_dxintColorTable
 embroidery.h, 405
 thread-color.c, 546
_mainWin
 Application, 56
 mainwindow.cpp, 349
_subMask
 format_csd.c, 477
_vp3Hoop, 52
 bottom, 53
 bottom2, 53
 byte1, 53
 byte2, 53
 byte3, 53
 height, 53
 left, 53
 left2, 53
 numberOfBytesRemaining, 53
 numberOfColors, 53
 right, 54
 right2, 54
 threadLength, 54
 top, 54
 top2, 54
 unknown2, 54
 unknown3, 54
 unknown4, 54
 width, 54
 xOffset, 54
 yOffset, 54
_xorMask
 format_csd.c, 477
~ArcObject
 ArcObject, 60
~BaseObject
 BaseObject, 68
~CircleObject
 CircleObject, 74
~CmdPrompt
 CmdPrompt, 77
~CmdPromptHandle
 CmdPromptHandle, 82
~CmdPromptHistory
 CmdPromptHistory, 84
~CmdPromptInput
 CmdPromptInput, 86
~CmdPromptSplitter
 CmdPromptSplitter, 90
~DimLeaderObject
 DimLeaderObject, 95
~EllipseObject
 EllipseObject, 100
~EmbDetailsDialog
 EmbDetailsDialog, 107
~ImageObject

ImageObject, 126
 ~ImageWidget
 ImageWidget, 128
 ~LayerManager
 LayerManager, 130
 ~LineObject
 LineObject, 133
 ~MainWindow
 MainWindow, 145
 ~MdiArea
 MdiArea, 177
 ~MdiWindow
 MdiWindow, 181
 ~PathObject
 PathObject, 189
 ~PointObject
 PointObject, 193
 ~PolygonObject
 PolygonObject, 197
 ~PolylineObject
 PolylineObject, 201
 ~PreviewDialog
 PreviewDialog, 203
 ~PropertyEditor
 PropertyEditor, 206
 ~RectObject
 RectObject, 215
 ~SaveObject
 SaveObject, 218
 ~Settings_Dialog
 Settings_Dialog, 239
 ~TextSingleObject
 TextSingleObject, 264
 ~UndoEditor
 UndoEditor, 280
 ~View
 View, 285
 10o, 17, 472
 100, 17, 472

 about
 MainWindow, 145
 accept_display_bg_color
 Settings_Dialog, 248
 accept_display_crosshair_color
 Settings_Dialog, 248
 accept_display_selectbox_left_color
 Settings_Dialog, 248
 accept_display_selectbox_left_fill
 Settings_Dialog, 248
 accept_display_selectbox_right_color
 Settings_Dialog, 248
 accept_display_selectbox_right_fill
 Settings_Dialog, 248
 accept_general_mdi_bg_color
 Settings_Dialog, 248
 accept_general_mdi_bg_logo
 Settings_Dialog, 248

 accept_general_mdi_bg_texture
 Settings_Dialog, 248
 accept_grid_color
 Settings_Dialog, 248
 accept_prompt_bg_color
 Settings_Dialog, 249
 accept_prompt_text_color
 Settings_Dialog, 249
 accept_ruler_color
 Settings_Dialog, 249
 acceptChanges
 Settings_Dialog, 239
 Action
 embroidermodder.h, 301
 Action___, 54
 aliases, 54
 hash, 55
 icon, 55
 script, 55
 shortcut, 55
 statustip, 55
 tooltip, 55
 action_labels
 mainwindow.cpp, 349
 action_table
 embroidermodder.h, 307
 mainwindow.cpp, 349
 actionHash
 MainWindow, 167
 active_view
 mainwindow.cpp, 349
 activeCommand
 CmdPrompt, 77
 MainWindow, 145
 activeMdiWindow
 MainWindow, 145
 activeScene
 MainWindow, 145
 activeUndoStack
 MainWindow, 145
 activeView
 MainWindow, 145
 actuator
 MainWindow, 145
 addArc
 SaveObject, 218
 addBlock
 SaveObject, 219
 addCircle
 SaveObject, 219
 addColorsToComboBox
 Settings_Dialog, 239
 addCommand
 CmdPrompt, 77
 CmdPromptInput, 86
 addDimAligned
 SaveObject, 219
 addDimAngular

SaveObject, 219
addDimArcLength
 SaveObject, 219
addDimDiameter
 SaveObject, 220
addDimLeader
 SaveObject, 220
addDimLinear
 SaveObject, 220
addDimOrdinate
 SaveObject, 220
addDimRadius
 SaveObject, 220
addEllipse
 SaveObject, 220
addEllipseArc
 SaveObject, 220
addGrid
 SaveObject, 221
addHatch
 SaveObject, 221
addImage
 SaveObject, 221
addInfiniteLine
 SaveObject, 221
addLayer
 LayerManager, 130
addLine
 SaveObject, 221
addObject
 View, 285
addPath
 SaveObject, 221
addPoint
 SaveObject, 221
addPolygon
 SaveObject, 221
addPolyline
 SaveObject, 221
addRay
 SaveObject, 221
addRectangle
 SaveObject, 221
addSlot
 SaveObject, 222
addSpline
 SaveObject, 222
addStack
 UndoEditor, 280
addTextMulti
 SaveObject, 222
addTextSingle
 SaveObject, 222
addToRubberRoom
 View, 285
after
 UndoableGripEditCommand, 273
alert
 CmdPrompt, 77
aliases
 Action__, 54
aliasHash
 CmdPromptInput, 89
alignScenePointWithViewPoint
 View, 285
allGripPoints
 ArcObject, 60
 BaseObject, 68
 CircleObject, 74
 DimLeaderObject, 95
 EllipseObject, 101
 ImageObject, 126
 LineObject, 133
 PathObject, 189
 PointObject, 193
 PolygonObject, 197
 PolylineObject, 201
 RectObject, 215
 TextSingleObject, 264
allowRubber
 View, 285
allowZoomIn
 View, 285
allowZoomOut
 View, 285
alpha
 SelectBox, 224
alphabet
 LSYSTEM, 135
Ameco, 442, 493
angle
 UndoableRotateCommand, 278
appendHistory
 CmdPrompt, 77
 CmdPromptHistory, 84
 CmdPromptInput, 86
appendTheHistory
 CmdPrompt, 78
Application, 55
 _mainWin, 56
 Application, 55
 event, 56
 setMainWin, 56
applyFormatting
 CmdPromptHistory, 84
 CmdPromptInput, 86
arc
 EmbGeometry_, 110
arc.c
 Arc_clockwise, 516
 Base_objectRubberPoint, 516
 Base_objectRubberText, 517
 Base_setLineType, 517
 Base_setLineWeight, 517
 clockwise, 517
 embArc_arcLength, 517

embArc_area, 517
 embArc_chord, 517
 embArc_clockwise, 517
 embArc_endAngle, 517
 embArc_gripEdit, 517
 embArc_includedAngle, 517
 embArc_init, 517
 embArc_mouseSnapPoint, 518
 embArc_paint, 518
 embArc_setCenter, 518
 embArc_setEndAngle, 518
 embArc_setRadius, 518
 embArc_setStartAngle, 518
 embArc_startAngle, 518
 embArc_updatePath, 518
 embArc_updateRubber, 518
 embBaseSetColorRGB, 518
 embCircle_prompt, 518
 embCircle_setArea, 518
 embCircle_setCircumference, 519
 embEllipse_click, 519
 embEllipse_main, 519
 embRect_bottomLeft, 519
 embRect_bottomRight, 519
 getArcCenter, 519
 getArcDataFromBulge, 519
 set_object_color, 519
Arc_clockwise
 arc.c, 516
Arc_Polyester
 embroidery.h, 380
Arc_Rayon
 embroidery.h, 381
arcEndPoint
 ArcObject, 66
arcMidPoint
 ArcObject, 66
ArcObject, 56
 ~ArcObject, 60
 allGripPoints, 60
 arcEndPoint, 66
 arcMidPoint, 66
 ArcObject, 59, 60
 arcEndPoint, 66
 calculateArcData, 61
 gripEdit, 61
 init, 61
 mouseSnapPoint, 62
 objectArcLength, 62
 objectArea, 62
 objectChord, 62
 objectClockwise, 62
 objectEndAngle, 62
 objectEndPoint, 63
 objectEndX, 63
 objectEndY, 63
 objectIncludedAngle, 63
 objectMidPoint, 63
 objectMidX, 63
 objectMidY, 63
 objectRadius, 64
 objectStartAngle, 64
 objectStartPoint, 64
 objectStartX, 64
 objectStartY, 64
 paint, 64
 setObjectEndAngle, 64
 setObjectEndPoint, 65
 setObjectMidPoint, 65
 setObjectRadius, 65
 setObjectStartAngle, 65
 setObjectStartPoint, 65
 Type, 59
 type, 65
 updateArcRect, 65
 updatePath, 66
 updateRubber, 66
 vulcanize, 66
arcEndPoint
 ArcObject, 66
array.c
 embArray_addArc, 369
 embArray_addCircle, 370
 embArray_addEllipse, 370
 embArray_addFlag, 370
 embArray_addLine, 370
 embArray_addPath, 370
 embArray_addPoint, 370
 embArray_addPolygon, 370
 embArray_addPolyline, 370
 embArray_addRect, 370
 embArray_addStitch, 370
 embArray_addVector, 370
 embArray_copy, 371
 embArray_create, 371
 embArray_free, 371
 embArray_resize, 371
ArrowStyle
 DimLeaderObject, 95
arrowStyleAngle
 DimLeaderObject, 98
arrowStyleLength
 DimLeaderObject, 98
arrowStylePath
 DimLeaderObject, 98
art, 17, 473
assets_dir
 Settings_, 227
attributeList
 format_svg.c, 506
attributeOffset
 VipHeader_, 295
AutoCAD, 445, 483, 502
AutoDesk, 483
auxFormat
 ThredExtension_, 267

axiom
 LSYSTEM, 135

b
 EmbColor_, 106

Barudan, 446, 479, 510

Base_objectRubberPoint
 arc.c, 516

Base_objectRubberText
 arc.c, 517

Base_setLineType
 arc.c, 517

Base_setLineWeight
 arc.c, 517

BaseObject, 66
 ~BaseObject, 68

allGripPoints, 68

BaseObject, 68

boundingRect, 68

drawRubberLine, 68

gripEdit, 68

line, 68

lineWeightPen, 68

lwtPen, 71

mouseShapPoint, 68

objectCenter, 68

objectCenterX, 69

objectCenterY, 69

objectColor, 69

objectColorRGB, 69

objectID, 69

objectLineType, 69

objectLineWeight, 69

objectPath, 69

objectPen, 69

objectRubberMode, 69

objectRubberPoint, 69

objectRubberText, 69

objID, 71

objLine, 71

objPen, 71

objRubberMode, 71

objRubberPoints, 71

objRubberTexts, 71

realRender, 69

rect, 69

setLine, 69

setObjectCenter, 70

setObjectCenterX, 70

setObjectCenterY, 70

setObjectColor, 70

setObjectColorRGB, 70

setObjectLineType, 70

setObjectLineWeight, 70

setObjectPath, 70

setObjectRubberMode, 70

setObjectRubberPoint, 70

setObjectRubberText, 70

setRect, 70

shape, 71

Type, 68

type, 71

vulcanize, 71

bcf_difat_create
 embroidery_internal.h, 431

 main.c, 534

bcf_directory
 embroidery_internal.h, 429

bcf_directory_entry
 embroidery_internal.h, 429

bcf_directory_free
 embroidery_internal.h, 431

 main.c, 534

bcf_file
 embroidery_internal.h, 429

bcf_file_difat
 embroidery_internal.h, 429

bcf_file_difat_free
 embroidery_internal.h, 431

bcf_file_fat
 embroidery_internal.h, 429

bcf_file_fat_free
 embroidery_internal.h, 431

bcf_file_free
 embroidery_internal.h, 431

 main.c, 534

bcf_file_header
 embroidery_internal.h, 429

bcfFile_read
 embroidery_internal.h, 431

 main.c, 534

bcfFileFat_create
 embroidery_internal.h, 431

 main.c, 534

bcfFileHeader_isValid
 embroidery_internal.h, 431

bcfFileHeader_read
 embroidery_internal.h, 432

 main.c, 534

before
 UndoableGripEditCommand, 273

Bernina, 473

beziers
 EmbSpline_, 117

bgColor
 MdiArea, 179

bgLogo
 MdiArea, 179

bgTexture
 MdiArea, 179

binaryReadString
 embroidery_internal.h, 432

 main.c, 534

binaryReadUnicodeString
 embroidery_internal.h, 432

 main.c, 535

binaryWriteInt

embroidery_internal.h, 432
 formats.c, 469
 binaryWriteIntBE
 embroidery_internal.h, 432
 formats.c, 469
 binaryWriteShort
 embroidery_internal.h, 432
 formats.c, 469
 binaryWriteUInt
 embroidery_internal.h, 432
 formats.c, 469
 binaryWriteUIntBE
 embroidery_internal.h, 432
 formats.c, 469
 binaryWriteUShort
 embroidery_internal.h, 433
 formats.c, 469
 binaryWriteUShortBE
 embroidery_internal.h, 433
 formats.c, 469
 bit_position
 Compress, 91
 Bitmap Cache, 474
 Bits and Volts, 474
 bits_total
 Compress, 91
 black_thread
 embroidery.h, 405
 main.c, 540
 blink
 CmdPrompt, 78
 blinkState
 CmdPrompt, 81
 blinkTimer
 CmdPrompt, 81
 block_elements
 Compress, 91
 bmc, 474
 bottom
 _vp3Hoop, 53
 EmbRect_, 116
 hoop_padding, 123
 bottom2
 _vp3Hoop, 53
 boundingRect
 BaseObject, 68
 EmbDetailsDialog, 107
 Box
 DimLeaderObject, 95
 boxDir
 SelectBox, 224
 brand_codes
 thread-color.c, 546
 brand_codes_files
 thread-color.c, 546
 bro, 17, 474
 Brother, 443, 444, 497, 498, 501, 502
 BuildDecryptionTable
 format_csd.c, 476
 BULGETOCONTROL
 embroidery_internal.h, 421
 BULGETOEND
 embroidery_internal.h, 422
 buttonBox
 EmbDetailsDialog, 107
 Settings_Dialog, 249
 buttonCustomFilterClearAll
 Settings_Dialog, 239
 buttonCustomFilterClearAllClicked
 Settings_Dialog, 239
 buttonCustomFilterSelectAll
 Settings_Dialog, 239
 buttonCustomFilterSelectAllClicked
 Settings_Dialog, 239
 buttonQSnapClearAll
 Settings_Dialog, 240
 buttonQSnapClearAllClicked
 Settings_Dialog, 240
 buttonQSnapSelectAll
 Settings_Dialog, 240
 buttonQSnapSelectAllClicked
 Settings_Dialog, 240
 buttonTipOfTheDayClicked
 MainWindow, 146
 byte1
 _vp3Hoop, 53
 byte2
 _vp3Hoop, 53
 byte3
 _vp3Hoop, 53
 byteOrder
 _bcf_file_header, 51
 c_split
 mainwindow.cpp, 348
 calculateArcData
 ArcObject, 61
 canRedo
 UndoEditor, 280
 canUndo
 UndoEditor, 280
 cascade
 MdiArea, 177
 catalogNumber
 EmbThread_, 119
 cci
 format_dst.c, 481
 center
 EmbCircle_, 106
 EmbEllipse_, 108
 UiObject_, 269
 View, 285
 centerAt
 View, 285
 changeFormatting
 CmdPromptInput, 86
 changelog

MainWindow, 146
character_huffman
 Compress, 91
character_length_huffman
 Compress, 91
check_for_color_file
 EmbFormatList_, 109
check_header_present
 embroidery_internal.h, 433
 main.c, 535
checkBoxCustomFilterStateChanged
 Settings_Dialog, 240
checkBoxDisableBGStateChanged
 Settings_Dialog, 240
checkBoxGeneralMdiBGUseColorStateChanged
 Settings_Dialog, 240
checkBoxGeneralMdiBGUseLogoStateChanged
 Settings_Dialog, 240
checkBoxGeneralMdiBGUseTextureStateChanged
 Settings_Dialog, 240
checkBoxGridCenterOnOriginStateChanged
 Settings_Dialog, 240
checkBoxGridColumnMatchCrossHairStateChanged
 Settings_Dialog, 240
checkBoxGridLoadFromFileStateChanged
 Settings_Dialog, 240
checkBoxGridShowOnLoadStateChanged
 Settings_Dialog, 240
checkBoxGridShowOriginStateChanged
 Settings_Dialog, 241
checkBoxLwtRealRenderStateChanged
 Settings_Dialog, 241
checkBoxLwtShowLwtStateChanged
 Settings_Dialog, 241
checkBoxPromptSaveHistoryAsHtmlStateChanged
 Settings_Dialog, 241
checkBoxPromptSaveHistoryStateChanged
 Settings_Dialog, 241
checkBoxQSnapApparentStateChanged
 Settings_Dialog, 241
checkBoxQSnapCenterStateChanged
 Settings_Dialog, 241
checkBoxQSnapEndPointStateChanged
 Settings_Dialog, 241
checkBoxQSnapExtensionStateChanged
 Settings_Dialog, 241
checkBoxQSnapInsertionStateChanged
 Settings_Dialog, 241
checkBoxQSnapIntersectionStateChanged
 Settings_Dialog, 241
checkBoxQSnapMidPointStateChanged
 Settings_Dialog, 241
checkBoxQSnapNearestStateChanged
 Settings_Dialog, 241
checkBoxQSnapNodeStateChanged
 Settings_Dialog, 242
checkBoxQSnapParallelStateChanged
 Settings_Dialog, 242
checkBoxQSnapPerpendicularStateChanged
 Settings_Dialog, 242
checkBoxQSnapQuadrantStateChanged
 Settings_Dialog, 242
checkBoxQSnapTangentStateChanged
 Settings_Dialog, 242
checkBoxRenderHintAAStateChanged
 Settings_Dialog, 242
checkBoxRenderHintHighAAStateChanged
 Settings_Dialog, 242
checkBoxRenderHintNonCosmeticStateChanged
 Settings_Dialog, 242
checkBoxRenderHintSmoothPixStateChanged
 Settings_Dialog, 242
checkBoxRenderHintTextAAStateChanged
 Settings_Dialog, 242
checkBoxRulerShowOnLoadStateChanged
 Settings_Dialog, 242
checkBoxSelectionModePickAddStateChanged
 Settings_Dialog, 242
checkBoxSelectionModePickDragStateChanged
 Settings_Dialog, 242
checkBoxSelectionModePickFirstStateChanged
 Settings_Dialog, 243
checkBoxShowScrollBarsStateChanged
 Settings_Dialog, 243
checkBoxTipOfTheDay
 MainWindow, 167
checkBoxTipOfTheDayStateChanged
 MainWindow, 146
 Settings_Dialog, 243
checkBoxUseOpenGLStateChanged
 Settings_Dialog, 243
checkChangedText
 CmdPromptInput, 87
checkCursorPosition
 CmdPromptInput, 87
checkEditedText
 CmdPromptInput, 87
checkForUpdates
 MainWindow, 146
checkSelection
 CmdPromptInput, 87
childId
 _bcf_directory_entry, 48
chooseDisplayBackgroundColor
 Settings_Dialog, 243
chooseDisplayCrossHairColor
 Settings_Dialog, 243
chooseDisplaySelectBoxLeftColor
 Settings_Dialog, 243
chooseDisplaySelectBoxLeftFill
 Settings_Dialog, 243
chooseDisplaySelectBoxRightColor
 Settings_Dialog, 243
chooseDisplaySelectBoxRightFill
 Settings_Dialog, 243
chooseGeneralMdiBackgroundColor

Settings_Dialog, 243
 chooseGeneralMdiBackgroundLogo
 Settings_Dialog, 243
 chooseGeneralMdiBackgroundTexture
 Settings_Dialog, 243
 chooseGridColor
 Settings_Dialog, 244
 choosePromptBackgroundColor
 Settings_Dialog, 244
 choosePromptTextColor
 Settings_Dialog, 244
 chooseRulerColor
 Settings_Dialog, 244
 CHUNK_SIZE
 embroidery.h, 381
 circle
 EmbGeometry_, 110
 circle.c
 embCircle_area, 520
 embCircle_circumference, 520
 embCircle_init, 520
 getCircleCircleIntersections, 520
 getCircleTangentPoints, 520
 CIRCLE_MODE_1P_DIA
 brodermodder.h, 306
 CIRCLE_MODE_1P_DIA_
 mainwindow.cpp, 349
 CIRCLE_MODE_1P_RAD
 brodermodder.h, 306
 CIRCLE_MODE_1P_RAD_
 mainwindow.cpp, 349
 CIRCLE_MODE_2P
 brodermodder.h, 306
 CIRCLE_MODE_2P_
 mainwindow.cpp, 349
 CIRCLE_MODE_3P
 brodermodder.h, 306
 CIRCLE_MODE_3P_
 mainwindow.cpp, 349
 CIRCLE_MODE_TTR
 brodermodder.h, 306
 CIRCLE_MODE_TTR_
 mainwindow.cpp, 349
 CircleObject, 71
 ~CircleObject, 74
 allGripPoints, 74
 CircleObject, 73, 74
 gripEdit, 74
 init, 74
 mouseSnapPoint, 74
 objectArea, 74
 objectCircumference, 74
 objectDiameter, 74
 objectQuadrant0, 74
 objectQuadrant180, 74
 objectQuadrant270, 74
 objectQuadrant90, 74
 objectRadius, 75
 objectSavePath, 75
 paint, 75
 setObjectArea, 75
 setObjectCircumference, 75
 setObjectDiameter, 75
 setObjectRadius, 75
 Type, 73
 type, 75
 updatePath, 75
 updateRubber, 75
 vulcanize, 75
 clearAllFields
 PropertyEditor, 206
 clearFormatting
 CmdPromptInput, 87
 clearRubberRoom
 View, 285
 clearSelection
 View, 285
 clockwise
 arc.c, 517
 Closed
 DimLeaderObject, 95
 closeEvent
 MainWindow, 146
 MdiWindow, 181
 closeToolBar
 MainWindow, 146
 CLSID
 _bcf_directory_entry, 49
 _bcf_file_header, 51
 cmdActive
 CmdPromptInput, 89
 CmdPrompt, 75
 ~CmdPrompt, 77
 activeCommand, 77
 addCommand, 77
 alert, 77
 appendHistory, 77
 appendTheHistory, 78
 blink, 78
 blinkState, 81
 blinkTimer, 81
 CmdPrompt, 77
 copyPressed, 78
 cutPressed, 78
 deletePressed, 78
 disableRapidFire, 78
 downPressed, 78
 enableRapidFire, 78
 endCommand, 78
 escapePressed, 78
 F10Pressed, 78
 F11Pressed, 78
 F12Pressed, 78
 F1Pressed, 78
 F2Pressed, 78
 F3Pressed, 78

F4Pressed, 78
F5Pressed, 79
F6Pressed, 79
F7Pressed, 79
F8Pressed, 79
F9Pressed, 79
floatingChanged, 79
getCurrentText, 79
getHistory, 79
getPrefix, 79
historyAppended, 79
isCommandActive, 79
isRapidFireEnabled, 79
lastCommand, 79
pastePressed, 79
processInput, 79
promptDivider, 81
promptHistory, 81
promptInput, 81
promptSplitter, 81
promptVBoxLayout, 81
redoPressed, 79
resizeTheHistory, 79
runCommand, 80
saveHistory, 80
selectAllPressed, 80
setCurrentText, 80
setHistory, 80
setPrefix, 80
setPromptBackgroundColor, 80
setPromptFontFamily, 80
setPromptFontSize, 80
setPromptFontStyle, 80
setPromptTextColor, 80
shiftPressed, 80
shiftReleased, 80
showSettings, 80
startBlinking, 81
startCommand, 81
startResizingTheHistory, 81
stopBlinking, 81
stopResizingTheHistory, 81
styleHash, 82
tabPressed, 81
undoPressed, 81
updateStyle, 81
upPressed, 81
CmdPromptHandle, 82
~CmdPromptHandle, 82
CmdPromptHandle, 82
handleMoved, 82
handlePressed, 82
handleReleased, 83
mouseMoveEvent, 83
mousePressEvent, 83
mouseReleaseEvent, 83
moveY, 83
pressY, 83
releaseY, 83
CmdPromptHistory, 83
~CmdPromptHistory, 84
appendHistory, 84
applyFormatting, 84
CmdPromptHistory, 84
contextMenuEvent, 84
historyAppended, 84
resizeHistory, 84
startResizeHistory, 84
stopResizeHistory, 84
tmpHeight, 84
CmdPromptInput, 85
~CmdPromptInput, 86
addCommand, 86
aliasHash, 89
appendHistory, 86
applyFormatting, 86
changeFormatting, 86
checkChangedText, 87
checkCursorPosition, 87
checkEditedText, 87
checkSelection, 87
clearFormatting, 87
cmdActive, 89
CmdPromptInput, 86
contextMenuEvent, 87
copyClip, 87
copyPressed, 87
curCmd, 89
curText, 89
cutPressed, 87
defaultPrefix, 89
deletePressed, 87
downPressed, 87
endCommand, 87
escapePressed, 87
eventFilter, 87
F10Pressed, 87
F11Pressed, 87
F12Pressed, 88
F1Pressed, 88
F2Pressed, 88
F3Pressed, 88
F4Pressed, 88
F5Pressed, 88
F6Pressed, 88
F7Pressed, 88
F8Pressed, 88
F9Pressed, 88
isBlinking, 89
lastCmd, 89
pasteClip, 88
pastePressed, 88
prefix, 89
processInput, 88
rapidFireEnabled, 89
redoPressed, 88

runCommand, 88
selectAllPressed, 88
shiftPressed, 88
shiftReleased, 89
showSettings, 89
startCommand, 89
stopBlinking, 89
tabPressed, 89
undoPressed, 89
updateCurrentText, 89
upPressed, 89
CmdPromptSplitter, 90
~CmdPromptSplitter, 90
CmdPromptSplitter, 90
createHandle, 90
moveResizeHistory, 90
pressResizeHistory, 90
releaseResizeHistory, 90
cnd, 17, 475
CoatsAndClark_Rayon
embroidery.h, 381
CODE_OF_CONDUCT.md, 296
col, 17, 475
color
 EmbGeometry_, 110
 EmbLine_, 113
 EmbPath_, 114
 EmbPoint_, 115
 EmbStitch_, 118
 EmbThread_, 119
 UiObject_, 270
color_only
 EmbFormatList_, 109
colorChanges
 EmbDetailsDialog, 107
colorCode
 StxThread_, 260
 SubDescriptor_, 260
colorFlag
 _bcf_directory_entry, 49
colorLength
 VipHeader_, 295
colorName
 StxThread_, 260
 SubDescriptor_, 260
colorSelector
 MainWindow, 167
colorSelectorIndexChanged
 MainWindow, 146
colorTotal
 EmbDetailsDialog, 107
comboBoxArcClockwise
 property-editor.cpp, 356
comboBoxes
 property-editor.cpp, 356
comboBoxGeneralLineType
 property-editor.cpp, 357
comboBoxGeneralLineWidth
 property-editor.cpp, 357
comboBoxLanguageCurrentIndexChanged
 Settings_Dialog, 244
comboBoxPathClosed
 property-editor.cpp, 357
comboBoxPathVertexNum
 property-editor.cpp, 357
comboBoxPolylineClosed
 property-editor.cpp, 357
comboBoxPolylineVertexNum
 property-editor.cpp, 357
comboBoxPromptFontFamilyCurrentIndexChanged
 Settings_Dialog, 244
comboBoxPromptFontStyleCurrentIndexChanged
 Settings_Dialog, 244
comboBoxQSnapLocatorColorCurrentIndexChanged
 Settings_Dialog, 244
comboBoxRulerMetricCurrentIndexChanged
 Settings_Dialog, 244
comboBoxScrollBarWidgetCurrentIndexChanged
 Settings_Dialog, 244
comboBoxSelected
 PropertyEditor, 210
comboBoxSelectionCoolGripColorCurrentIndexChanged
 Settings_Dialog, 245
comboBoxSelectionHotGripColorCurrentIndexChanged
 Settings_Dialog, 245
comboBoxTextSingleBackward
 property-editor.cpp, 357
comboBoxTextSingleFont
 property-editor.cpp, 357
comboBoxTextSingleJustify
 property-editor.cpp, 357
comboBoxTextSingleUpsideDown
 property-editor.cpp, 357
command
 UiObject_, 270
CompoundFileDirectory
 embroidery_internal.h, 433
 main.c, 535
CompoundFileDirectoryEntry
 embroidery_internal.h, 433
 main.c, 535
CompoundFileSector_DIFAT_Sector
 embroidery_internal.h, 422
CompoundFileSector_EndOfChain
 embroidery_internal.h, 422
CompoundFileSector_FAT_Sector
 embroidery_internal.h, 422
CompoundFileSector_FreeSector
 embroidery_internal.h, 422
CompoundFileSector_MaxRegSector

embroidery_internal.h, 422
CompoundFileStreamId_MaxRegularStreamId
 embroidery_internal.h, 422
CompoundFileStreamId_NoStream
 embroidery_internal.h, 422
Compress, 91
 bit_position, 91
 bits_total, 91
 block_elements, 91
 character_huffman, 91
 character_length_huffman, 91
 distance_huffman, 91
 input_data, 91
 input_length, 91
compress
 embroidery_internal.h, 430
compress.c
 compress_get_bits, 372
 compress_get_position, 372
 compress_get_token, 372
 compress_init, 372
 compress_load_block, 372
 compress_load_character_huffman, 372
 compress_load_character_length_huffman, 372
 compress_load_distance_huffman, 372
 compress_peek, 372
 compress_pop, 372
 compress_read_variable_length, 372
 huffman_build_table, 373
 huffman_lookup, 373
 huffman_lookup_data, 373
 hus_compress, 373
 hus_decompress, 373
compress_get_bits
 compress.c, 372
 embroidery_internal.h, 433
compress_get_position
 compress.c, 372
 embroidery_internal.h, 433
compress_get_token
 compress.c, 372
 embroidery_internal.h, 434
compress_init
 compress.c, 372
compress_load_block
 compress.c, 372
 embroidery_internal.h, 434
compress_load_character_huffman
 compress.c, 372
 embroidery_internal.h, 434
compress_load_character_length_huffman
 compress.c, 372
 embroidery_internal.h, 434
compress_load_distance_huffman
 compress.c, 372
 embroidery_internal.h, 434
compress_peek
 compress.c, 372
compress_pop
 compress.c, 372
compress_read_variable_length
 compress.c, 372
constants
 LSYSTEM, 135
contextMenuEvent
 CmdPromptHistory, 84
 CmdPromptInput, 87
 StatusBarButton, 258
 View, 285
control1
 EmbBezier_, 105
control2
 EmbBezier_, 105
controlPointLabels
 UiObject_, 270
controlPoints
 UiObject_, 270
convert
 embroidery.h, 390
 pattern.c, 541
convert_args_to_type
 mainwindow.cpp, 348
copy
 MainWindow, 146
 View, 285
copy_trim
 embroidery_internal.h, 434
 main.c, 535
copyClip
 CmdPromptInput, 87
copyPressed
 CmdPrompt, 78
 CmdPromptInput, 87
copySelected
 View, 285
cornerButtonClicked
 View, 286
count
 EmbArray_, 104
create_icon
 MainWindow, 146
create_test_file_1
 embroidery_internal.h, 434
create_test_file_2
 embroidery_internal.h, 434
create_test_file_3
 embroidery_internal.h, 434
create_toolbar
 MainWindow, 146
createAllActions
 MainWindow, 146
createAllMenus
 MainWindow, 147
createAllToolbars

MainWindow, 147
createComboBox
 PropertyEditor, 206
createComboBoxSelected
 PropertyEditor, 206
createEditMenu
 MainWindow, 147
createFileMenu
 MainWindow, 147
createFontComboBox
 PropertyEditor, 206
createGrid
 View, 286
createGridIso
 View, 286
createGridPolar
 View, 286
createGridRect
 View, 286
createGroupBoxGeneral
 PropertyEditor, 206
createGroupBoxGeometryArc
 PropertyEditor, 207
createGroupBoxGeometryBlock
 PropertyEditor, 207
createGroupBoxGeometryCircle
 PropertyEditor, 207
createGroupBoxGeometryDimAligned
 PropertyEditor, 207
createGroupBoxGeometryDimAngular
 PropertyEditor, 207
createGroupBoxGeometryDimArcLength
 PropertyEditor, 207
createGroupBoxGeometryDimDiameter
 PropertyEditor, 207
createGroupBoxGeometryDimLeader
 PropertyEditor, 207
createGroupBoxGeometryDimLinear
 PropertyEditor, 207
createGroupBoxGeometryDimOrdinate
 PropertyEditor, 207
createGroupBoxGeometryDimRadius
 PropertyEditor, 207
createGroupBoxGeometryEllipse
 PropertyEditor, 207
createGroupBoxGeometryImage
 PropertyEditor, 208
createGroupBoxGeometryInfiniteLine
 PropertyEditor, 208
createGroupBoxGeometryLine
 PropertyEditor, 208
createGroupBoxGeometryPath
 PropertyEditor, 208
createGroupBoxGeometryPoint
 PropertyEditor, 208
createGroupBoxGeometryPolygon
 PropertyEditor, 208
createGroupBoxGeometryPolyline
 PropertyEditor, 208
createGroupBoxGeometryRay
 PropertyEditor, 208
createGroupBoxGeometryRectangle
 PropertyEditor, 208
createGroupBoxGeometryTextMulti
 PropertyEditor, 208
createGroupBoxGeometryTextSingle
 PropertyEditor, 208
createGroupBoxMiscArc
 PropertyEditor, 208
createGroupBoxMiscImage
 PropertyEditor, 208
createGroupBoxMiscPath
 PropertyEditor, 209
createGroupBoxMiscPolyline
 PropertyEditor, 209
createGroupBoxMiscTextSingle
 PropertyEditor, 209
createGroupBoxTextTextSingle
 PropertyEditor, 209
createHandle
 CmdPromptSplitter, 90
createHelpMenu
 MainWindow, 147
createHelpToolbar
 MainWindow, 147
createHistogram
 EmbDetailsDialog, 107
createIconToolbar
 MainWindow, 147
createLayerToolbar
 MainWindow, 147
createLineEdit
 PropertyEditor, 209
createMainWidget
 EmbDetailsDialog, 107
createObjectList
 View, 286
createOrigin
 View, 286
createPanToolbar
 MainWindow, 147
createPromptToolbar
 MainWindow, 147
createPropertiesToolbar
 MainWindow, 147
createRulerTextPath
 View, 286
createSettingsMenu
 MainWindow, 147
createTabDisplay
 Settings_Dialog, 245
createTabFilesPaths
 Settings_Dialog, 245
createTabGeneral
 Settings_Dialog, 245
createTabGridRuler

Settings_Dialog, 245
createTabLineWeight
 Settings_Dialog, 245
createTabOpenSave
 Settings_Dialog, 245
createTabOrthoPolar
 Settings_Dialog, 245
createTabPrinting
 Settings_Dialog, 245
createTabPrompt
 Settings_Dialog, 245
createTabQuickSnap
 Settings_Dialog, 245
createTabQuickTrack
 Settings_Dialog, 245
createTabSelection
 Settings_Dialog, 245
createTabSnap
 Settings_Dialog, 245
createTextToolbar
 MainWindow, 147
createToolButton
 PropertyEditor, 209
createToolButtonPickAdd
 PropertyEditor, 209
createToolButtonQSelect
 PropertyEditor, 209
createViewMenu
 MainWindow, 147
createWindowMenu
 MainWindow, 148
creationTime
 _bcf_directory_entry, 49
creatorName
 ThredExtension_, 268
crosshairColor
 View, 292
crosshairSize
 View, 292
csd, 17, 476
csd_decryptArray
 format_csd.c, 477
CsdSubMaskSize
 format_csd.c, 476
CsdXorMaskSize
 format_csd.c, 476
csv, 478
CSV_EXPECT
 embroidery_internal.h, 430
CSV_EXPECT_COMMA
 embroidery_internal.h, 430
CSV_EXPECT_NULL
 embroidery_internal.h, 430
CSV_EXPECT_QUOTE1
 embroidery_internal.h, 430
CSV_EXPECT_QUOTE2
 embroidery_internal.h, 430
CSV_MODE
 embroidery_internal.h, 430
CSV_MODE_COMMENT
 embroidery_internal.h, 430
CSV_MODE_NULL
 embroidery_internal.h, 430
CSV_MODE_STITCH
 embroidery_internal.h, 431
CSV_MODE_THREAD
 embroidery_internal.h, 430
CSV_MODE_VARIABLE
 embroidery_internal.h, 430
csvStitchFlagToStr
 format_csv.c, 478
csvStrToStitchFlag
 format_csv.c, 478
CUBICTOCONTROL1
 embroidery_internal.h, 422
CUBICTOCONTROL2
 embroidery_internal.h, 422
CUBICTOEND
 embroidery_internal.h, 422
curCmd
 CmdPromptInput, 89
curColor
 MdiWindow, 186
curFile
 MdiWindow, 186
curLayer
 MdiWindow, 186
curLineType
 MdiWindow, 186
curLineWeight
 MdiWindow, 186
current_directory
 Settings_, 227
current_element_id
 format_svg.c, 506
currentAttribute
 format_svg.c, 506
currentColorChanged
 MdiWindow, 181
currentColorIndex
 EmbPattern_, 114
currentDisplayBackgroundColorChanged
 Settings_Dialog, 245
currentDisplayCrossHairColorChanged
 Settings_Dialog, 246
currentDisplaySelectBoxLeftColorChanged
 Settings_Dialog, 246
currentDisplaySelectBoxLeftFillChanged
 Settings_Dialog, 246
currentDisplaySelectBoxRightColorChanged
 Settings_Dialog, 246
currentDisplaySelectBoxRightFillChanged
 Settings_Dialog, 246
currentGeneralMdiBackgroundColorChanged
 Settings_Dialog, 246
currentGridColorChanged

Settings_Dialog, 246
 currentLayerChanged
 MdiWindow, 182
 currentLinetypeChanged
 MdiWindow, 182
 currentLineweightChanged
 MdiWindow, 182
 currentPromptBackgroundColorChanged
 Settings_Dialog, 246
 currentPromptTextColorChanged
 Settings_Dialog, 246
 currentRulerColorChanged
 Settings_Dialog, 246
 currentValue
 format_svg.c, 506
 curText
 CmdPromptInput, 89
 curved
 DimLeaderObject, 98
 cut
 MainWindow, 148
 View, 286
 cutCopyMousePoint
 View, 292
 cutCopyObjectList
 MainWindow, 167
 cutPressed
 CmdPrompt, 78
 CmdPromptInput, 87

 dat, 17
 data
 EmblImage_, 111
 UndoHistory_, 281
 day
 EmbTime_, 119
 dayVision
 MainWindow, 148
 debug_mode
 Settings_, 227
 decode_exy_flags
 format_exy.c, 485
 decode_record_flags
 format_dst.c, 481
 decode_t01_record
 embroidery_internal.h, 434
 encoding.c, 461
 decode_tajima_ternary
 embroidery_internal.h, 435
 encoding.c, 461
 decode_tap_record_flags
 format_tap.c, 508
 DecodeCsdByte
 format_csd.c, 477
 decodeNewStitch
 embroidery_internal.h, 435
 encoding.c, 461
 DEFAULT_MODE
 embroidermodder.h, 306

 default_value
 Huffman, 123
 defaultPrefix
 CmdPromptInput, 89
 degrees
 embroidermodder.h, 307
 embroidery.h, 390
 functions.c, 522
 deleteObject
 View, 286
 deletePressed
 CmdPrompt, 78
 CmdPromptInput, 87
 MainWindow, 148
 MdiWindow, 182
 View, 286
 deleteSelected
 View, 286
 dem, 17, 479
 description
 EmbFormatList_, 109
 EmbThread_, 119
 designDetails
 MainWindow, 148
 MdiWindow, 182
 dialog
 embroidermodder.h, 307
 mainwindow.cpp, 349
 dialog_display_bg_color
 Settings_Dialog, 249
 dialog_display_crosshair_color
 Settings_Dialog, 249
 dialog_display_crosshair_percent
 Settings_Dialog, 249
 dialog_display_renderhint_aa
 Settings_Dialog, 249
 dialog_display_renderhint_high_aa
 Settings_Dialog, 249
 dialog_display_renderhint_noncosmetic
 Settings_Dialog, 249
 dialog_display_renderhint_smooth_pix
 Settings_Dialog, 249
 dialog_display_renderhint_text_aa
 Settings_Dialog, 249
 dialog_display_scrollbar_widget_num
 Settings_Dialog, 249
 dialog_display_selectbox_alpha
 Settings_Dialog, 249
 dialog_display_selectbox_left_color
 Settings_Dialog, 249
 dialog_display_selectbox_left_fill
 Settings_Dialog, 249
 dialog_display_selectbox_right_color
 Settings_Dialog, 249
 dialog_display_selectbox_right_fill
 Settings_Dialog, 249
 dialog_display_show_scrollbars
 Settings_Dialog, 250

dialog_display_units
 Settings_Dialog, 250
dialog_display_use_opengl
 Settings_Dialog, 250
dialog_display_zoomscale_in
 Settings_Dialog, 250
dialog_display_zoomscale_out
 Settings_Dialog, 250
dialog_general_icon_size
 Settings_Dialog, 250
dialog_general_icon_theme
 Settings_Dialog, 250
dialog_general_language
 Settings_Dialog, 250
dialog_general_mdi_bg_color
 Settings_Dialog, 250
dialog_general_mdi_bg_logo
 Settings_Dialog, 250
dialog_general_mdi_bg_texture
 Settings_Dialog, 250
dialog_general_mdi_bg_use_color
 Settings_Dialog, 250
dialog_general_mdi_bg_use_logo
 Settings_Dialog, 250
dialog_general_mdi_bg_use_texture
 Settings_Dialog, 250
dialog_general_system_help_browser
 Settings_Dialog, 250
dialog_general_tip_of_the_day
 Settings_Dialog, 250
dialog_grid_center_on_origin
 Settings_Dialog, 250
dialog_grid_center_x
 Settings_Dialog, 250
dialog_grid_center_y
 Settings_Dialog, 251
dialog_grid_color
 Settings_Dialog, 251
dialog_grid_color_match_crosshair
 Settings_Dialog, 251
dialog_grid_load_from_file
 Settings_Dialog, 251
dialog_grid_show_on_load
 Settings_Dialog, 251
dialog_grid_show_origin
 Settings_Dialog, 251
dialog_grid_size_radius
 Settings_Dialog, 251
dialog_grid_size_x
 Settings_Dialog, 251
dialog_grid_size_y
 Settings_Dialog, 251
dialog_grid_spacing_angle
 Settings_Dialog, 251
dialog_grid_spacing_radius
 Settings_Dialog, 251
dialog_grid_spacing_x
 Settings_Dialog, 251
dialog_grid_spacing_y
 Settings_Dialog, 251
dialog_grid_type
 Settings_Dialog, 251
dialog_lwt_default_lwt
 Settings_Dialog, 251
dialog_lwt_real_render
 Settings_Dialog, 251
dialog_lwt_show_lwt
 Settings_Dialog, 251
dialog_opensave_custom_filter
 Settings_Dialog, 251
dialog_opensave_open_format
 Settings_Dialog, 252
dialog_opensave_open_thumbnail
 Settings_Dialog, 252
dialog_opensave_recent_max_files
 Settings_Dialog, 252
dialog_opensave_save_format
 Settings_Dialog, 252
dialog_opensave_save_thumbnail
 Settings_Dialog, 252
dialog_opensave_trim_dst_num_jumps
 Settings_Dialog, 252
dialog_printing_default_device
 Settings_Dialog, 252
dialog_printing_disable_bg
 Settings_Dialog, 252
dialog_printing_use_last_device
 Settings_Dialog, 252
dialog_prompt_bg_color
 Settings_Dialog, 252
dialog_prompt_font_family
 Settings_Dialog, 252
dialog_prompt_font_size
 Settings_Dialog, 252
dialog_prompt_font_style
 Settings_Dialog, 252
dialog_prompt_save_history
 Settings_Dialog, 252
dialog_prompt_save_history_as_html
 Settings_Dialog, 252
dialog_prompt_save_history_filename
 Settings_Dialog, 252
dialog_prompt_text_color
 Settings_Dialog, 252
dialog_qsnap_aperture_size
 Settings_Dialog, 252
dialog_qsnap_apparent
 Settings_Dialog, 253
dialog_qsnap_center
 Settings_Dialog, 253
dialog_qsnap_enabled
 Settings_Dialog, 253
dialog_qsnap_endpoint
 Settings_Dialog, 253
dialog_qsnap_extension
 Settings_Dialog, 253

dialog_qsnap_insertion
 Settings_Dialog, 253
 dialog_qsnap_intersection
 Settings_Dialog, 253
 dialog_qsnap_locator_color
 Settings_Dialog, 253
 dialog_qsnap_locator_size
 Settings_Dialog, 253
 dialog_qsnap_midpoint
 Settings_Dialog, 253
 dialog_qsnap_nearest
 Settings_Dialog, 253
 dialog_qsnap_node
 Settings_Dialog, 253
 dialog_qsnap_parallel
 Settings_Dialog, 253
 dialog_qsnap_perpendicular
 Settings_Dialog, 253
 dialog_qsnap_quadrant
 Settings_Dialog, 253
 dialog_qsnap_tangent
 Settings_Dialog, 253
 dialog_ruler_color
 Settings_Dialog, 253
 dialog_ruler_metric
 Settings_Dialog, 253
 dialog_ruler_pixel_size
 Settings_Dialog, 254
 dialog_ruler_show_on_load
 Settings_Dialog, 254
 dialog_selection_coolgrip_color
 Settings_Dialog, 254
 dialog_selection_grip_size
 Settings_Dialog, 254
 dialog_selection_hotgrip_color
 Settings_Dialog, 254
 dialog_selection_mode_pickadd
 Settings_Dialog, 254
 dialog_selection_mode_pickdrag
 Settings_Dialog, 254
 dialog_selection_mode_pickfirst
 Settings_Dialog, 254
 dialog_selection_pickbox_size
 Settings_Dialog, 254
 Dictionary
 embroidermodder.h, 301
 difat
 _bcf_file, 50
 difatEntriesInHeader
 main.c, 540
 dimensions
 EmblImage_, 111
 DimLeaderObject, 91
 ~DimLeaderObject, 95
 allGripPoints, 95
 ArrowStyle, 95
 arrowStyleAngle, 98
 arrowStyleLength, 98
 arrowStylePath, 98
 Box, 95
 Closed, 95
 curved, 98
 DimLeaderObject, 95
 Dot, 95
 filled, 98
 Flared, 95
 Fletching, 95
 gripEdit, 95
 init, 96
 lineStyle, 95
 lineStyleAngle, 98
 lineStyleLength, 98
 lineStylePath, 98
 mouseSnapPoint, 96
 NoArrow, 95
 NoLine, 95
 objectAngle, 96
 objectDeltaX, 96
 objectDeltaY, 96
 objectEndPoint1, 96
 objectEndPoint2, 96
 objectLength, 96
 objectMidPoint, 96
 objectX1, 96
 objectX2, 96
 objectY1, 96
 objectY2, 96
 Open, 95
 paint, 96
 setObjectEndPoint1, 97
 setObjectEndPoint2, 97
 setObjectX1, 97
 setObjectX2, 97
 setObjectY1, 97
 setObjectY2, 97
 Tick, 95
 Type, 95
 type, 97
 updateLeader, 97
 updateRubber, 97
 vulcanize, 97
 dirBrush
 SelectBox, 224
 directory
 _bcf_file, 50
 directoryEntryName
 _bcf_directory_entry, 49
 directoryEntryNameLength
 _bcf_directory_entry, 49
 dirEntries
 _bcf_directory, 48
 dirPen
 SelectBox, 224
 disableLwt
 StatusBarButton, 258
 disableMoveRapidFire

MainWindow, 148
View, 286
disablePromptRapidFire
 MainWindow, 148
disableRapidFire
 CmdPrompt, 78
disableReal
 StatusBarButton, 258
display_bg_color
 Settings_, 227
display_crosshair_color
 Settings_, 227
display_crosshair_percent
 Settings_, 227
display_renderhint_aa
 Settings_, 227
display_renderhint_high_aa
 Settings_, 228
display_renderhint_noncosmetic
 Settings_, 228
display_renderhint_smooth_pix
 Settings_, 228
display_renderhint_text_aa
 Settings_, 228
display_scrollbar_widget_num
 Settings_, 228
display_selectbox_alpha
 Settings_, 228
display_selectbox_left_color
 Settings_, 228
display_selectbox_left_fill
 Settings_, 228
display_selectbox_right_color
 Settings_, 228
display_selectbox_right_fill
 Settings_, 228
display_show_scrollbars
 Settings_, 228
display_units
 Settings_, 228
display_use_opengl
 Settings_, 228
display_zoomscale_in
 Settings_, 228
display_zoomscale_out
 Settings_, 228
distance_huffman
 Compress, 91
docIndex
 MainWindow, 167
dockPropEdit
 MainWindow, 167
dockUndoEdit
 MainWindow, 167
DOLPHIN_MODE_NUM_POINTS
 embroidermodder.h, 306
DOLPHIN_MODE_NUM_POINTS_
 mainwindow.cpp, 349
DOLPHIN_MODE_XSCALE
 embroidermodder.h, 306
DOLPHIN_MODE_XSCALE_
 mainwindow.cpp, 350
DOLPHIN_MODE_YSCALE
 embroidermodder.h, 306
DOLPHIN_MODE_YSCALE_
 mainwindow.cpp, 350
done
 UndoableNavCommand, 276
doNothing
 MainWindow, 148
Dot
 DimLeaderObject, 95
downPressed
 CmdPrompt, 78
 CmdPromptInput, 87
dragon_curve
 fill.c, 464
drawBackground
 View, 286
drawForeground
 View, 286
drawRubberLine
 BaseObject, 68
dsb, 17, 479
dst, 17, 480
dstJumpsPerTrim
 EmbPattern_, 114
dsz, 17, 440, 482
dx
 UndoableMoveCommand, 275
 UndoableScaleCommand, 279
dxf, 17, 483
dxf_color
 embroidery.h, 381
DXF_VERSION_2000
 embroidery_internal.h, 422
DXF_VERSION_2002
 embroidery_internal.h, 422
DXF_VERSION_2004
 embroidery_internal.h, 422
DXF_VERSION_2006
 embroidery_internal.h, 422
DXF_VERSION_2007
 embroidery_internal.h, 422
DXF_VERSION_2009
 embroidery_internal.h, 422
DXF_VERSION_2010
 embroidery_internal.h, 423
DXF_VERSION_2013
 embroidery_internal.h, 423
DXF_VERSION_R10
 embroidery_internal.h, 423
DXF_VERSION_R11
 embroidery_internal.h, 423
DXF_VERSION_R12
 embroidery_internal.h, 423

DXF_VERSION_R13
embroidery_internal.h, 423

DXF_VERSION_R14
embroidery_internal.h, 423

DXF_VERSION_R15
embroidery_internal.h, 423

DXF_VERSION_R18
embroidery_internal.h, 423

DXF_VERSION_R21
embroidery_internal.h, 423

DXF_VERSION_R24
embroidery_internal.h, 423

DXF_VERSION_R27
embroidery_internal.h, 423

dy
UndoableMoveCommand, 275
UndoableScaleCommand, 279

edit_toolbar
brodermodder.h, 308
mainwindow-toolbars.cpp, 345

editMenu
MainWindow, 168

edr, 17, 440, 483

ELEMENT_A
embroidery_internal.h, 423

ELEMENT_ANIMATE
embroidery_internal.h, 423

ELEMENT_ANIMATECOLOR
embroidery_internal.h, 423

ELEMENT_ANIMATEMOTION
embroidery_internal.h, 423

ELEMENT_ANIMATETRANSFORM
embroidery_internal.h, 423

ELEMENT_ANIMATION
embroidery_internal.h, 423

ELEMENT_AUDIO
embroidery_internal.h, 424

ELEMENT_CIRCLE
embroidery_internal.h, 424

ELEMENT_DEFS
embroidery_internal.h, 424

ELEMENT_DESC
embroidery_internal.h, 424

ELEMENT_DISCARD
embroidery_internal.h, 424

ELEMENT_ELLIPSE
embroidery_internal.h, 424

ELEMENT_FONT
embroidery_internal.h, 424

ELEMENT_FONT_FACE
embroidery_internal.h, 424

ELEMENT_FONT_FACE_SRC
embroidery_internal.h, 424

ELEMENT_FONT_FACE_URI
embroidery_internal.h, 424

ELEMENT_FOREIGN_OBJECT
embroidery_internal.h, 424

ELEMENT_G
embroidery_internal.h, 424

ELEMENT_GLYPH
embroidery_internal.h, 424

ELEMENT_HANDLER
embroidery_internal.h, 424

ELEMENT_HKERN
embroidery_internal.h, 424

ELEMENT_IMAGE
embroidery_internal.h, 424

ELEMENT_LINE
embroidery_internal.h, 424

ELEMENT_LINEAR_GRADIENT
embroidery_internal.h, 424

ELEMENT_LISTENER
embroidery_internal.h, 425

ELEMENT_METADATA
embroidery_internal.h, 425

ELEMENT_MISSING_GLYPH
embroidery_internal.h, 425

ELEMENT_MPATH
embroidery_internal.h, 425

ELEMENT_PATH
embroidery_internal.h, 425

ELEMENT_POLYGON
embroidery_internal.h, 425

ELEMENT_POLYLINE
embroidery_internal.h, 425

ELEMENT_PREFETCH
embroidery_internal.h, 425

ELEMENT_RADIAL_GRADIENT
embroidery_internal.h, 425

ELEMENT_RECT
embroidery_internal.h, 425

ELEMENT_SCRIPT
embroidery_internal.h, 425

ELEMENT_SET
embroidery_internal.h, 425

ELEMENT_SOLID_COLOR
embroidery_internal.h, 425

ELEMENT_STOP
embroidery_internal.h, 425

ELEMENT_SVG
embroidery_internal.h, 425

ELEMENT_SWITCH
embroidery_internal.h, 425

ELEMENT_TBREAK
embroidery_internal.h, 425

ELEMENT_TEXT
embroidery_internal.h, 425

ELEMENT_TEXT_AREA
embroidery_internal.h, 426

ELEMENT_TITLE
embroidery_internal.h, 426

ELEMENT_TSPAN
embroidery_internal.h, 426

ELEMENT_USE
embroidery_internal.h, 426

ELEMENT_VIDEO

embroidery_internal.h, 426
ELEMENT_XML
 embroidery_internal.h, 426
ellipse
 EmbGeometry_, 110
ellipse.c
 ellipse_objectQuadrant0, 521
 ellipse_objectQuadrant180, 521
 ellipse_objectQuadrant270, 521
 ellipse_objectQuadrant90, 521
 embEllipse_area, 521
 embEllipse_diameterX, 521
 embEllipse_diameterY, 521
 embEllipse_height, 521
 embEllipse_init, 521
 embEllipse_perimeter, 521
 embEllipse_setDiameterMajor, 521
 embEllipse_setDiameterMinor, 522
 embEllipse_setRadiusMajor, 522
 embEllipse_setRadiusMinor, 522
 embEllipse_setSize, 522
 embEllipse_updatePath, 522
 embEllipse_width, 522
ELLIPSE_MODE_ELLIPSE_ROTATION
 embroidermodder.h, 306
ELLIPSE_MODE_MAJORDIAMETER_MINORRADIUS
 embroidermodder.h, 306
ELLIPSE_MODE_MAJORRADIUS_MINORRADIUS
 embroidermodder.h, 306
ellipse_objectQuadrant0
 ellipse.c, 521
ellipse_objectQuadrant180
 ellipse.c, 521
ellipse_objectQuadrant270
 ellipse.c, 521
ellipse_objectQuadrant90
 ellipse.c, 521
EllipseObject, 98
 ~EllipseObject, 100
 allGripPoints, 101
 EllipseObject, 100
 gripEdit, 101
 init, 101
 mouseSnapPoint, 101
 objectDiameterMajor, 101
 objectDiameterMinor, 101
 objectHeight, 101
 objectQuadrant0, 101
 objectQuadrant180, 101
 objectQuadrant270, 101
 objectQuadrant90, 101
 objectRadiusMajor, 102
 objectRadiusMinor, 102
 objectSavePath, 102
 objectWidth, 102
 paint, 102
 setObjectDiameterMajor, 102
 setObjectDiameterMinor, 102
 setObjectRadiusMajor, 102
 setObjectRadiusMinor, 102
 setObjectSize, 102
 Type, 100
 type, 102
 updatePath, 102
 updateRubber, 102
 vulcanize, 102
ELLIPSETOEND
 embroidery_internal.h, 426
ELLIPSETORAD
 embroidery_internal.h, 426
Elna, 484
Eltac, 485
EMB_ARC
 embroidery.h, 381
EMB_ARRAY
 embroidery.h, 381
EMB_BIG_ENDIAN
 embroidery_internal.h, 426
EMB_CIRCLE
 embroidery.h, 381
emb_constant_pi
 embroidermodder.h, 308
EMB_DIM_DIAMETER
 embroidery.h, 381
EMB_DIM_LEADER
 embroidery.h, 381
EMB_ELLIPSE
 embroidery.h, 381
emb_error
 embroidery.h, 405
 main.c, 540
EMB_FLAG
 embroidery.h, 381
EMB_FORMAT_100
 embroidery.h, 381
EMB_FORMAT_10O
 embroidery.h, 381
EMB_FORMAT_ART
 embroidery.h, 381
EMB_FORMAT_BMC
 embroidery.h, 381
EMB_FORMAT_BRO
 embroidery.h, 381
EMB_FORMAT_CND
 embroidery.h, 381
EMB_FORMAT_COL
 embroidery.h, 381
EMB_FORMAT_CSD
 embroidery.h, 382
EMB_FORMAT_CSV
 embroidery.h, 382
EMB_FORMAT_DAT
 embroidery.h, 382
EMB_FORMAT DEM
 embroidery.h, 382
EMB_FORMAT_DSB

embroidery.h, 382
EMB_FORMAT_DST
 embroidery.h, 382
EMB_FORMAT_DSZ
 embroidery.h, 382
EMB_FORMAT_DXF
 embroidery.h, 382
EMB_FORMAT_EDR
 embroidery.h, 382
EMB_FORMAT_EMD
 embroidery.h, 382
EMB_FORMAT_EXP
 embroidery.h, 382
EMB_FORMAT_EXY
 embroidery.h, 382
EMB_FORMAT_EYS
 embroidery.h, 382
EMB_FORMAT_FXY
 embroidery.h, 382
EMB_FORMAT_GC
 embroidery.h, 382
EMB_FORMAT_GNC
 embroidery.h, 382
EMB_FORMAT_GT
 embroidery.h, 382
EMB_FORMAT_HUS
 embroidery.h, 382
EMB_FORMAT_INB
 embroidery.h, 383
EMB_FORMAT_INF
 embroidery.h, 383
EMB_FORMAT_JEF
 embroidery.h, 383
EMB_FORMAT_KSM
 embroidery.h, 383
EMB_FORMAT_MAX
 embroidery.h, 383
EMB_FORMAT_MIT
 embroidery.h, 383
EMB_FORMAT_NEW
 embroidery.h, 383
EMB_FORMAT_OFM
 embroidery.h, 383
EMB_FORMAT_PCD
 embroidery.h, 383
EMB_FORMAT_PCM
 embroidery.h, 383
EMB_FORMAT_PCQ
 embroidery.h, 383
EMB_FORMAT_PCS
 embroidery.h, 383
EMB_FORMAT_PEC
 embroidery.h, 383
EMB_FORMAT_PEL
 embroidery.h, 383
EMB_FORMAT_PEM
 embroidery.h, 383
EMB_FORMAT_PES
 embroidery.h, 383
EMB_FORMAT_PHB
 embroidery.h, 383
EMB_FORMAT_PHC
 embroidery.h, 383
EMB_FORMAT_PLT
 embroidery.h, 384
EMB_FORMAT_RGB
 embroidery.h, 384
EMB_FORMAT_SEW
 embroidery.h, 384
EMB_FORMAT_SHV
 embroidery.h, 384
EMB_FORMAT_SST
 embroidery.h, 384
EMB_FORMAT_STX
 embroidery.h, 384
EMB_FORMAT_SVG
 embroidery.h, 384
EMB_FORMAT_T01
 embroidery.h, 384
EMB_FORMAT_T09
 embroidery.h, 384
EMB_FORMAT_TAP
 embroidery.h, 384
EMB_FORMAT_THR
 embroidery.h, 384
EMB_FORMAT_TXT
 embroidery.h, 384
EMB_FORMAT_U00
 embroidery.h, 384
EMB_FORMAT_U01
 embroidery.h, 384
EMB_FORMAT_VIP
 embroidery.h, 384
EMB_FORMAT_VP3
 embroidery.h, 384
EMB_FORMAT_XXX
 embroidery.h, 384
EMB_FORMAT_ZSK
 embroidery.h, 384
emb_identify_format
 embroidery.h, 390
 formats.c, 470
EMB_IMAGE
 embroidery.h, 385
EMB_INT16_BIG
 embroidery_internal.h, 426
EMB_INT16_LITTLE
 embroidery_internal.h, 426
EMB_INT32_BIG
 embroidery_internal.h, 426
EMB_INT32_LITTLE
 embroidery_internal.h, 426
EMB_LINE
 embroidery.h, 385
EMB_LITTLE_ENDIAN
 embroidery_internal.h, 426

EMB_MAX
embroidery_internal.h, 426

EMB_MAX_LAYERS
embroidery.h, 385

EMB_MIN
embroidery_internal.h, 426

emb_optOut
embroidery_internal.h, 435
main.c, 535

EMB_PATH
embroidery.h, 385

EMB_POINT
embroidery.h, 385

EMB_POLYGON
embroidery.h, 385

EMB_POLYLINE
embroidery.h, 385

EMB_PUBLIC
embroidery.h, 385

emb_readline
embroidery_internal.h, 435
main.c, 536

EMB_RECT
embroidery.h, 385

emb_round
embroidery.h, 390
functions.c, 522

EMB_SPLINE
embroidery.h, 385

EMB_STITCH
embroidery.h, 385

EMB_TEXT_MULTI
embroidery.h, 385

EMB_TEXT_SINGLE
embroidery.h, 385

EMB_THREAD
embroidery.h, 385

EMB_VECTOR
embroidery.h, 385

emb_verbose
embroidery.h, 405
main.c, 540

EmbAlignedDim
embroidery.h, 388

EmbAlignedDim_, 103
position, 103

EmbAngularDim
embroidery.h, 388

EmbAngularDim_, 103
position, 103

EmbArc
embroidery.h, 388

EmbArc_, 103
end, 103
mid, 104
start, 104

embArc_arcLength
arc.c, 517

embArc_area
arc.c, 517

embArc_chord
arc.c, 517

embArc_clockwise
arc.c, 517
embroidery.h, 390

embArc_endAngle
arc.c, 517

embArc_gripEdit
arc.c, 517

embArc_includedAngle
arc.c, 517

embArc_init
arc.c, 517
embroidery.h, 390

embArc_mouseSnapPoint
arc.c, 518

embArc_paint
arc.c, 518

embArc_print
main.c, 536

embArc_setCenter
arc.c, 518

embArc_setEndAngle
arc.c, 518

embArc_setRadius
arc.c, 518

embArc_setStartAngle
arc.c, 518

embArc_startAngle
arc.c, 518

embArc_updatePath
arc.c, 518

embArc_updateRubber
arc.c, 518

EmbArcLengthDim
embroidery.h, 388

EmbArcLengthDim_, 104
position, 104

EmbArray
embroidery.h, 388

EmbArray_, 104
count, 104
geometry, 104
length, 104
stitch, 104
thread, 104
type, 105

embArray_addArc
array.c, 369
embroidery.h, 391

embArray_addCircle
array.c, 370
embroidery.h, 391

embArray_addEllipse
array.c, 370
embroidery.h, 391

embArray_addFlag
 array.c, 370
 embroidery.h, 391

embArray_addLine
 array.c, 370
 embroidery.h, 391

embArray_addPath
 array.c, 370
 embroidery.h, 391

embArray_addPoint
 array.c, 370
 embroidery.h, 391

embArray_addPolygon
 array.c, 370
 embroidery.h, 391

embArray_addPolyline
 array.c, 370
 embroidery.h, 391

embArray_addRect
 array.c, 370
 embroidery.h, 391

embArray_addStitch
 array.c, 370
 embroidery.h, 391

embArray_addThread
 embroidery.h, 392

embArray_addVector
 array.c, 370
 embroidery.h, 392

embArray_copy
 array.c, 371
 embroidery.h, 392

embArray_create
 array.c, 371
 embroidery.h, 392

embArray_free
 array.c, 371
 embroidery.h, 392

embArray_resize
 array.c, 371
 embroidery.h, 392

embBaseSetColorRGB
 arc.c, 518

EmbBezier
 embroidery.h, 388

EmbBezier_, 105
 control1, 105
 control2, 105
 end, 105
 start, 105

EmbBlock
 embroidery.h, 388

EmbBlock_, 105
 position, 105

EmbCircle
 embroidery.h, 388

EmbCircle_, 105
 center, 106

radius, 106

embCircle_area
 circle.c, 520

embCircle_circumference
 circle.c, 520

embCircle_init
 circle.c, 520
 embroidery.h, 392

embCircle_prompt
 arc.c, 518

embCircle_setArea
 arc.c, 518

embCircle_setCircumference
 arc.c, 519

EmbColor
 embroidery.h, 388

EmbColor_, 106
 b, 106
 g, 106
 r, 106

embColor_create
 embroidery.h, 392

embColor_distance
 embroidery.h, 392
 main.c, 536

embColor_fromHexStr
 embroidery.h, 392
 encoding.c, 461

embColor_make
 embroidery.h, 393

embColor_read
 embroidery_internal.h, 435
 main.c, 536

embColor_write
 embroidery_internal.h, 435
 main.c, 536

embConstantPi
 embroidery.h, 405
 main.c, 540

EmbDetailsDialog, 106
 ~EmbDetailsDialog, 107
 boundingRect, 107
 buttonBox, 107
 colorChanges, 107
 colorTotal, 107
 createHistogram, 107
 createMainWidget, 107
 EmbDetailsDialog, 107
 getInfo, 107
 mainWidget, 107
 stitchesJump, 107
 stitchesReal, 108
 stitchesTotal, 108
 stitchesTrim, 108

EmbDiameterDim
 embroidery.h, 388

EmbDiameterDim_, 108
 position, 108

EmbEllipse
embroidery.h, 388
EmbEllipse_, 108
center, 108
radius, 108
rotation, 108
embEllipse_area
ellipse.c, 521
embroidery.h, 393
embEllipse_click
arc.c, 519
embEllipse_diameterX
ellipse.c, 521
embroidery.h, 393
embEllipse_diameterY
ellipse.c, 521
embroidery.h, 393
embEllipse_height
ellipse.c, 521
embroidery.h, 393
embEllipse_init
ellipse.c, 521
embroidery.h, 393
embEllipse_main
arc.c, 519
embEllipse_make
embroidery.h, 393
embEllipse_perimeter
ellipse.c, 521
embroidery.h, 393
embEllipse_setDiameterMajor
ellipse.c, 521
embEllipse_setDiameterMinor
ellipse.c, 522
embEllipse_setRadiusMajor
ellipse.c, 522
embEllipse_setRadiusMinor
ellipse.c, 522
embEllipse_setSize
ellipse.c, 522
embEllipse_updatePath
ellipse.c, 522
embEllipse_width
ellipse.c, 522
embroidery.h, 393
EmbFlag
embroidery.h, 388
embFormat_getExtension
formats.c, 470
EMBFORMAT_MAXDESC
embroidery.h, 385
EMBFORMAT_MAXEXT
embroidery.h, 385
EMBFORMAT_OBJECTONLY
embroidery.h, 385
EMBFORMAT_STCHANDOBJ
embroidery.h, 386
EMBFORMAT_STITCHONLY
embroidery.h, 386
EMBFORMAT_UNSUPPORTED
embroidery.h, 386
EmbFormatList
embroidery.h, 388
EmbFormatList_, 109
check_for_color_file, 109
color_only, 109
description, 109
extension, 109
reader_state, 109
type, 109
write_external_color_file, 109
writer_state, 109
EmbGeometry
embroidery.h, 388
EmbGeometry_, 109
arc, 110
circle, 110
color, 110
ellipse, 110
flag, 110
line, 110
lineType, 110
object, 110
path, 110
point, 110
polygon, 110
polyline, 110
rect, 110
spline, 111
stitch, 111
thread, 111
type, 111
vector, 111
embGeometry_boundingRect
embroidery.h, 393
geometry.c, 515
embGeometry_free
embroidery.h, 393
geometry.c, 515
embGeometry_init
embroidery.h, 394
geometry.c, 515
embGeometry_move
embroidery.h, 394
geometry.c, 515
embGeometry_vulcanize
embroidery.h, 394
geometry.c, 515
EmblImage
embroidery.h, 389
EmblImage_, 111
data, 111
dimensions, 111
height, 111
name, 111
path, 111

position, 111
width, 111
emblImage_create
 embroidery.h, 394
emblImage_free
 embroidery.h, 394
emblImage_read
 embroidery.h, 394
emblImage_write
 embroidery.h, 394
EmbInfiniteLine
 embroidery.h, 389
EmbInfiniteLine_, 112
 position, 112
embInt_read
 embroidery_internal.h, 436
 encoding.c, 462
embInt_write
 embroidery_internal.h, 436
 encoding.c, 462
Embird, 440, 478, 483
EmbLayer
 embroidery.h, 389
EmbLayer_, 112
 geometry, 112
 name, 112
EmbLeaderDim
 embroidery.h, 389
EmbLeaderDim_, 112
 position, 112
EmbLine
 embroidery.h, 389
EmbLine_, 113
 color, 113
 end, 113
 lineType, 113
 start, 113
embLine_intersectionPoint
 embroidery.h, 394
 line.c, 523
embLine_make
 embroidery.h, 394
embLine_normalVector
 embroidery.h, 395
 line.c, 523
embLine_toVector
 line.c, 523
EmbLinearDim
 embroidery.h, 389
EmbLinearDim_, 113
 position, 113
EmbOrdinateDim
 embroidery.h, 389
EmbOrdinateDim_, 113
 position, 113
EmbPath
 embroidery.h, 389
EmbPath_, 114
 color, 114
 flagList, 114
 lineType, 114
 pointList, 114
EmbPattern
 embroidery.h, 389
EmbPattern_, 114
 currentColorIndex, 114
 dstJumpsPerTrim, 114
 geometry, 115
 home, 115
 hoop_height, 115
 hoop_width, 115
 layer, 115
 stitch_list, 115
 thread_list, 115
embPattern_addCircleAbs
 embroidery.h, 395
 pattern.c, 541
embPattern_addEllipseAbs
 embroidery.h, 395
 pattern.c, 541
embPattern_addLineAbs
 embroidery.h, 395
 pattern.c, 542
embPattern_addPathAbs
 embroidery.h, 395
 pattern.c, 542
embPattern_addPointAbs
 embroidery.h, 395
 pattern.c, 542
embPattern_addPolygonAbs
 embroidery.h, 395
 pattern.c, 542
embPattern_addPolylineAbs
 embroidery.h, 395
 pattern.c, 542
embPattern_addPolylineObjectAbs
 pattern.c, 542
embPattern_addRectAbs
 embroidery.h, 395
 pattern.c, 542
embPattern_addStitchAbs
 embroidery.h, 395
 pattern.c, 542
embPattern_addStitchRel
 embroidery.h, 396
 pattern.c, 542
embPattern_addThread
 embroidery.h, 396
 pattern.c, 542
embPattern_calcBoundingBox
 embroidery.h, 396
 pattern.c, 543
embPattern_center
 embroidery.h, 396
 pattern.c, 543
embPattern_changeColor
 embroidery.h, 396

pattern.c, 543
embPattern_color_count
embroidery.h, 396
pattern.c, 543
embPattern_combine
embroidery.h, 396
fill.c, 464
embPattern_combineJumpStitches
embroidery.h, 396
pattern.c, 543
embPattern_convertGeometry
embroidery.h, 396
fill.c, 464
embPattern_copyPolylinesToStitch_list
pattern.c, 543
embPattern_copyPolylinesToStitchList
embroidery.h, 397
embPattern_copystitch_listToPolylines
pattern.c, 543
embPattern_copyStitchListToPolylines
embroidery.h, 397
embPattern_correctForMaxStitchLength
embroidery.h, 397
pattern.c, 543
embPattern_create
embroidery.h, 397
pattern.c, 543
embPattern_crossstitch
embroidery.h, 397
fill.c, 464
embPattern_designDetails
embroidery.h, 397
pattern.c, 543
embPattern_end
embroidery.h, 397
pattern.c, 544
embPattern_fixColorCount
embroidery.h, 397
pattern.c, 544
embPattern_flip
embroidery.h, 397
pattern.c, 544
embPattern_flipHorizontal
embroidery.h, 398
pattern.c, 544
embPattern_flipVertical
embroidery.h, 398
pattern.c, 544
embPattern_free
embroidery.h, 398
pattern.c, 544
embPattern_hideStitchesOverLength
embroidery.h, 398
pattern.c, 544
embPattern_horizontal_fill
embroidery.h, 398
fill.c, 464
embPattern_jumpStitches
embroidery.h, 398
pattern.c, 544
embPattern_lengthHistogram
embroidery.h, 398
pattern.c, 544
embPattern_loadExternalColorFile
embroidery.h, 398
pattern.c, 544
embPattern_maximumStitchLength
embroidery.h, 398
pattern.c, 544
embPattern_minimumStitchLength
embroidery.h, 398
pattern.c, 544
embPattern_movePolylinesToStitch_list
pattern.c, 545
embPattern_movePolylinesToStitchList
embroidery.h, 399
embPattern_movestitch_listToPolylines
pattern.c, 545
embPattern_moveStitchListToPolylines
embroidery.h, 399
embPattern_read
embroidery.h, 399
formats.c, 470
embPattern_readAuto
embroidery.h, 399
formats.c, 470
embPattern_realStitches
embroidery.h, 399
pattern.c, 545
embPattern_render
embroidery.h, 399
embPattern_scale
embroidery.h, 399
pattern.c, 545
embPattern_simulate
embroidery.h, 399
embPattern_stitchArc
fill.c, 464
embPattern_stitchCircle
fill.c, 464
embPattern_stitchEllipse
fill.c, 465
embPattern_stitchPath
fill.c, 465
embPattern_stitchPolygon
fill.c, 465
embPattern_stitchPolyline
fill.c, 465
embPattern_stitchRect
fill.c, 466
embPattern_stitchText
fill.c, 466
embPattern_totalStitchLength
embroidery.h, 399
pattern.c, 545
embPattern_trimStitches

embroidery.h, 400
 pattern.c, 545
 embPattern_write
 embroidery.h, 400
 formats.c, 470
 embPattern_writeAuto
 embroidery.h, 400
 formats.c, 470
 EmbPoint
 embroidery.h, 389
 EmbPoint_, 115
 color, 115
 lineType, 115
 position, 115
 EmbPolygon
 embroidery.h, 389
 embPolygon_reduceByDistance
 fill.c, 466
 embPolygon_reduceByNth
 fill.c, 466
 EmbPolyline
 embroidery.h, 389
 EmbRadiusDim
 embroidery.h, 389
 EmbRadiusDim_, 115
 position, 116
 EmbRay
 embroidery.h, 389
 EmbRay_, 116
 position, 116
 EmbReal
 embroidery.h, 389
 EmbRect
 embroidery.h, 389
 EmbRect_, 116
 bottom, 116
 left, 116
 radius, 116
 right, 116
 rotation, 116
 top, 117
 embRect_area
 embroidery.h, 400
 rect.c, 524
 embRect_bottomLeft
 arc.c, 519
 embRect_bottomRight
 arc.c, 519
 embRect_init
 embroidery.h, 400
 rect.c, 524
 embroidermodder.cpp
 appName, 297
 appVer, 297
 exitApp, 297
 main, 297
 usage, 297
 version, 297
 embroidermodder.h
 Action, 301
 action_table, 307
 CIRCLE_MODE_1P_DIA, 306
 CIRCLE_MODE_1P_RAD, 306
 CIRCLE_MODE_2P, 306
 CIRCLE_MODE_3P, 306
 CIRCLE_MODE_TTR, 306
 DEFAULT_MODE, 306
 degrees, 307
 dialog, 307
 Dictionary, 301
 DOLPHIN_MODE_NUM_POINTS, 306
 DOLPHIN_MODE_XSCALE, 306
 DOLPHIN_MODE_YSCALE, 306
 edit_toolbar, 308
 ELLIPSE_MODE_ELLIPSE_ROTATION, 306
 ELLIPSE_MODE_MAJORDIAMETER_MINORRADIUS, 306
 ELLIPSE_MODE_MAJORRADIUS_MINORRADIUS, 306
 emb_constant_pi, 308
 EmbView, 301
 file_toolbar, 308
 get_action_index, 307
 HEART_MODE_NUM_POINTS, 306
 HEART_MODE_STYLE, 306
 HEART_MODE_XSCALE, 306
 HEART_MODE_YSCALE, 306
 Index, 301
 mainWin, 307
 OBJ_COLOR, 302
 OBJ_KEYS, 302
 OBJ_LAYER, 302
 OBJ_LTYPE, 302
 OBJ_LTYPE_CENTER, 302
 OBJ_LTYPE_CONT, 302
 OBJ_LTYPE_DOT, 302
 OBJ_LTYPE_FISHBONE, 303
 OBJ_LTYPE_HIDDEN, 302
 OBJ_LTYPE_PHANTOM, 302
 OBJ_LTYPE_RUNNING, 303
 OBJ_LTYPE_SATIN, 303
 OBJ_LTYPE_VALUES, 302
 OBJ_LTYPE_ZIGZAG, 303
 OBJ_LWT, 302
 OBJ_LWT_01, 303
 OBJ_LWT_02, 303
 OBJ_LWT_03, 303
 OBJ_LWT_04, 303
 OBJ_LWT_05, 303
 OBJ_LWT_06, 303
 OBJ_LWT_07, 303
 OBJ_LWT_08, 303
 OBJ_LWT_09, 303
 OBJ_LWT_10, 303
 OBJ_LWT_11, 303
 OBJ_LWT_12, 303

OBJ_LWT_13, 303
OBJ_LWT_14, 303
OBJ_LWT_15, 303
OBJ_LWT_16, 303
OBJ_LWT_17, 303
OBJ_LWT_18, 303
OBJ_LWT_19, 303
OBJ_LWT_20, 303
OBJ_LWT_21, 303
OBJ_LWT_22, 303
OBJ_LWT_23, 303
OBJ_LWT_24, 303
OBJ_LWT_BYBLOCK, 303
OBJ_LWT_BYLAYER, 303
OBJ_LWT_DEFAULT, 303
OBJ_LWT_VALUES, 303
OBJ_NAME, 302
OBJ_RUBBER, 302
OBJ_RUBBER_CIRCLE_1P_DIA, 304
OBJ_RUBBER_CIRCLE_1P_RAD, 303
OBJ_RUBBER_CIRCLE_2P, 304
OBJ_RUBBER_CIRCLE_3P, 304
OBJ_RUBBER_CIRCLE_TTR, 304
OBJ_RUBBER_CIRCLE_TTT, 304
OBJ_RUBBER_DIMLEADER_LINE, 304
OBJ_RUBBER_ELLIPSE_LINE, 304
OBJ_RUBBER_ELLIPSE_MAJOR_DIAMETER_MINOR_RADIUS, 304
OBJ_RUBBER_ELLIPSE_MAJOR_RADIUS_MINOR_RADIUS, 304
OBJ_RUBBER_ELLIPSE_ROTATION, 304
OBJ_RUBBER_GRIP, 304
OBJ_RUBBER_IMAGE, 304
OBJ_RUBBER_LINE, 304
OBJ_RUBBER_OFF, 303
OBJ_RUBBER_ON, 303
OBJ_RUBBER_POLYGON, 304
OBJ_RUBBER_POLYGON_CIRCUMSCRIBE, 304
OBJ_RUBBER_POLYGON_INSCRIBE, 304
OBJ_RUBBER_POLYLINE, 304
OBJ_RUBBER_RECTANGLE, 304
OBJ_RUBBER_TEXTSINGLE, 304
OBJ_RUBBER_VALUES, 303
OBJ_SNAP_APPINTERSECTION, 304
OBJ_SNAP_CENTER, 304
OBJ_SNAP_ENDPOINT, 304
OBJ_SNAP_EXTENSION, 304
OBJ_SNAP_INSERTION, 304
OBJ_SNAP_INTERSECTION, 304
OBJ_SNAP_MIDPOINT, 304
OBJ_SNAP_NEAREST, 304
OBJ_SNAP_NODE, 304
OBJ_SNAP_NULL, 304
OBJ_SNAP_PARALLEL, 304
OBJ_SNAP_PERPENDICULAR, 304
OBJ_SNAP_QUADRANT, 304
OBJ_SNAP_TANGENT, 304
OBJ_SNAP_VALUES, 304
OBJ_TYPE, 302
OBJ_TYPE_ARC, 305
OBJ_TYPE_BASE, 305
OBJ_TYPE_BLOCK, 305
OBJ_TYPE_CIRCLE, 305
OBJ_TYPE_DIMALIGNED, 305
OBJ_TYPE_DIMANGULAR, 305
OBJ_TYPE_DIMARCLENGTH, 305
OBJ_TYPE_DIMDIAMETER, 305
OBJ_TYPE_DIMLEADER, 305
OBJ_TYPE_DIMLINEAR, 305
OBJ_TYPE_DIMORDINATE, 305
OBJ_TYPE_DIMRADIUS, 305
OBJ_TYPE_ELLIPSE, 305
OBJ_TYPE_ELLIPSEARC, 305
OBJ_TYPE_GRID, 305
OBJ_TYPE_HATCH, 305
OBJ_TYPE_IMAGE, 305
OBJ_TYPE_INFINITELINE, 305
OBJ_TYPE_LINE, 305
OBJ_TYPE_NULL, 305
OBJ_TYPE_PATH, 305
OBJ_TYPE_POINT, 305
OBJ_TYPE_POLYGON, 305
OBJ_TYPE_POLYLINE, 305
OBJ_TYPE_RAY, 305
OBJ_TYPE_RECTANGLE, 305
OBJ_TYPE_RUBBER, 305
OBJ_TYPE_SLOT, 305
OBJ_TYPE_SPLINE, 305
OBJ_TYPE_TEXTMULTI, 305
OBJ_TYPE_TEXTSINGLE, 305
OBJ_TYPE_VALUES, 304
operator+, 307
operator-, 307
PREVIEW_CLONE_NULL, 305
PREVIEW_CLONE_RUBBER, 305
PREVIEW_CLONE_SELECTED, 305
PREVIEW_CLONE_VALUES, 305
PREVIEW_MODE_MOVE, 305
PREVIEW_MODE_NULL, 305
PREVIEW_MODE_ROTATE, 305
PREVIEW_MODE_SCALE, 306
PREVIEW_MODE_VALUES, 305
radians, 307
read_settings, 307
ROTATE_MODE_NORMAL, 306
ROTATE_MODE_REFERENCE, 306
SCALE_MODE_NORMAL, 306
SCALE_MODE_REFERENCE, 306
Settings, 301
settings, 308
SINGLE_LINE_TEXT_MODE_JUSTIFY, 306
SINGLE_LINE_TEXT_MODE_RAPID, 306
SINGLE_LINE_TEXT_MODE_SETFONT, 306
SINGLE_LINE_TEXT_MODE_SETGEOM, 306
SNOWFLAKE_MODE_NUM_POINTS, 306
SNOWFLAKE_MODE_XSCALE, 306

SNOWFLAKE_MODE_YSCALE, 306
 SPARE_RUBBER_OFF, 306
 SPARE_RUBBER_PATH, 306
 SPARE_RUBBER_POLYGON, 306
 SPARE_RUBBER_POLYLINE, 306
 SPARE_RUBBER_VALUES, 306
 STAR_MODE_CENTER_PT, 306
 STAR_MODE_NUM_POINTS, 306
 STAR_MODE_RAD_INNER, 306
 STAR_MODE_RAD_OUTER, 306
 to_EmbVector, 307
 to_QPointF, 307
 UiMode, 306
 UiObject, 302
 UndoHistory, 302
 view_toolbar, 308
 write_settings, 307
 zoom_toolbar, 308
 embroidermodder2/cmdprompt.cpp, 296
 embroidermodder2/embdetails-dialog.cpp, 296
 embroidermodder2/embroidermodder.cpp, 296
 embroidermodder2/embroidermodder.h, 297, 308
 embroidermodder2/imagewidget.cpp, 344
 embroidermodder2/layer-manager.cpp, 344
 embroidermodder2/mainwindow-commands.cpp, 344
 embroidermodder2/mainwindow-menus.cpp, 344
 embroidermodder2/mainwindow-settings.cpp, 344
 embroidermodder2/mainwindow-toolbars.cpp, 345
 embroidermodder2/mainwindow.cpp, 347
 embroidermodder2/mdiarea.cpp, 351
 embroidermodder2/mdiwindow.cpp, 351
 embroidermodder2/object-arc.cpp, 351
 embroidermodder2/object-base.cpp, 351
 embroidermodder2/object-circle.cpp, 351
 embroidermodder2/object-dimleader.cpp, 351
 embroidermodder2/object-ellipse.cpp, 351
 embroidermodder2/object-image.cpp, 351
 embroidermodder2/object-line.cpp, 352
 embroidermodder2/object-path.cpp, 352
 embroidermodder2/object-point.cpp, 352
 embroidermodder2/object-polygon.cpp, 352
 embroidermodder2/object-polyline.cpp, 352
 embroidermodder2/object-rect.cpp, 352
 embroidermodder2/object-save.cpp, 352
 embroidermodder2/object-textsingle.cpp, 352
 embroidermodder2/preview-dialog.cpp, 352
 embroidermodder2/property-editor.cpp, 352
 embroidermodder2/README.md, 368
 embroidermodder2/selectbox.cpp, 368
 embroidermodder2/settings-dialog.cpp, 368
 embroidermodder2/statusbar-button.cpp, 368
 embroidermodder2/statusbar.cpp, 368
 embroidermodder2/undo-commands.cpp, 369
 embroidermodder2/undo-editor.cpp, 369
 embroidermodder2/view.cpp, 369
 embroidery.h
 _dxsetColorTable, 405
 Arc_Polyester, 380
 Arc_Rayon, 381
 black_thread, 405
 CHUNK_SIZE, 381
 CoatsAndClark_Rayon, 381
 convert, 390
 degrees, 390
 dxf_color, 381
 EMB_ARC, 381
 EMB_ARRAY, 381
 EMB_CIRCLE, 381
 EMB_DIM_DIAMETER, 381
 EMB_DIM_LEADER, 381
 EMB_ELLIPSE, 381
 emb_error, 405
 EMB_FLAG, 381
 EMB_FORMAT_100, 381
 EMB_FORMAT_10O, 381
 EMB_FORMAT_ART, 381
 EMB_FORMAT_BMC, 381
 EMB_FORMAT_BRO, 381
 EMB_FORMAT_CND, 381
 EMB_FORMAT_COL, 381
 EMB_FORMAT_CSD, 382
 EMB_FORMAT_CSV, 382
 EMB_FORMAT_DAT, 382
 EMB_FORMAT_DEM, 382
 EMB_FORMAT_DSB, 382
 EMB_FORMAT_DST, 382
 EMB_FORMAT_DSZ, 382
 EMB_FORMAT_DXF, 382
 EMB_FORMAT_EDR, 382
 EMB_FORMAT_EMD, 382
 EMB_FORMAT_EXP, 382
 EMB_FORMAT_EXY, 382
 EMB_FORMAT_EYS, 382
 EMB_FORMAT_FXY, 382
 EMB_FORMAT_GC, 382
 EMB_FORMAT_GNC, 382
 EMB_FORMAT_GT, 382
 EMB_FORMAT_HUS, 382
 EMB_FORMAT_INB, 383
 EMB_FORMAT_INF, 383
 EMB_FORMAT_JEF, 383
 EMB_FORMAT_KSM, 383
 EMB_FORMAT_MAX, 383
 EMB_FORMAT_MIT, 383
 EMB_FORMAT_NEW, 383
 EMB_FORMAT_OFM, 383
 EMB_FORMAT_PCD, 383
 EMB_FORMAT_PCM, 383
 EMB_FORMAT_PCQ, 383
 EMB_FORMAT_PCS, 383
 EMB_FORMAT_PEC, 383
 EMB_FORMAT_PEL, 383
 EMB_FORMAT_PEM, 383
 EMB_FORMAT_PES, 383
 EMB_FORMAT_PHB, 383
 EMB_FORMAT_PHC, 383

EMB_FORMAT_PLT, 384
EMB_FORMAT_RGB, 384
EMB_FORMAT_SEW, 384
EMB_FORMAT_SHV, 384
EMB_FORMAT_SST, 384
EMB_FORMAT_STX, 384
EMB_FORMAT_SVG, 384
EMB_FORMAT_T01, 384
EMB_FORMAT_T09, 384
EMB_FORMAT_TAP, 384
EMB_FORMAT_THR, 384
EMB_FORMAT_TXT, 384
EMB_FORMAT_U00, 384
EMB_FORMAT_U01, 384
EMB_FORMAT_VIP, 384
EMB_FORMAT_VP3, 384
EMB_FORMAT_XXX, 384
EMB_FORMAT_ZSK, 384
emb_identify_format, 390
EMB_IMAGE, 385
EMB_LINE, 385
EMB_MAX_LAYERS, 385
EMB_PATH, 385
EMB_POINT, 385
EMB_POLYGON, 385
EMB_POLYLINE, 385
EMB_PUBLIC, 385
EMB_RECT, 385
emb_round, 390
EMB_SPLINE, 385
EMB_STITCH, 385
EMB_TEXT_MULTI, 385
EMB_TEXT_SINGLE, 385
EMB_THREAD, 385
EMB_VECTOR, 385
emb_verbose, 405
EmbAlignedDim, 388
EmbAngularDim, 388
EmbArc, 388
embArc_clockwise, 390
embArc_init, 390
EmbArcLengthDim, 388
EmbArray, 388
embArray_addArc, 391
embArray_addCircle, 391
embArray_addEllipse, 391
embArray_addFlag, 391
embArray_addLine, 391
embArray_addPath, 391
embArray_addPoint, 391
embArray_addPolygon, 391
embArray_addPolyline, 391
embArray_addRect, 391
embArray_addStitch, 391
embArray_addThread, 392
embArray_addVector, 392
embArray_copy, 392
embArray_create, 392
embArray_free, 392
embArray_resize, 392
EmbBezier, 388
EmbBlock, 388
EmbCircle, 388
embCircle_init, 392
EmbColor, 388
embColor_create, 392
embColor_distance, 392
embColor_fromHexStr, 392
embColor_make, 393
embConstantPi, 405
EmbDiameterDim, 388
EmbEllipse, 388
embEllipse_area, 393
embEllipse_diameterX, 393
embEllipse_diameterY, 393
embEllipse_height, 393
embEllipse_init, 393
embEllipse_make, 393
embEllipse_perimeter, 393
embEllipse_width, 393
EmbFlag, 388
EMBFORMAT_MAXDESC, 385
EMBFORMAT_MAXEXT, 385
EMBFORMAT_OBJECTONLY, 385
EMBFORMAT_STCHANDOBJ, 386
EMBFORMAT_STITCHONLY, 386
EMBFORMAT_UNSUPPORTED, 386
EmbFormatList, 388
EmbGeometry, 388
embGeometry_boundingRect, 393
embGeometry_free, 393
embGeometry_init, 394
embGeometry_move, 394
embGeometry_vulcanize, 394
EmblImage, 389
emblImage_create, 394
emblImage_free, 394
emblImage_read, 394
emblImage_write, 394
EmblInfiniteLine, 389
EmbLayer, 389
EmbLeaderDim, 389
EmbLine, 389
embLine_intersectionPoint, 394
embLine_make, 394
embLine_normalVector, 395
EmbLinearDim, 389
EmbOrdinateDim, 389
EmbPath, 389
EmbPattern, 389
embPattern_addCircleAbs, 395
embPattern_addEllipseAbs, 395
embPattern_addLineAbs, 395
embPattern_addPathAbs, 395
embPattern_addPointAbs, 395
embPattern_addPolygonAbs, 395

embPattern_addPolylineAbs, 395
 embPattern_addRectAbs, 395
 embPattern_addStitchAbs, 395
 embPattern_addStitchRel, 396
 embPattern_addThread, 396
 embPattern_calcBoundingBox, 396
 embPattern_center, 396
 embPattern_changeColor, 396
 embPattern_color_count, 396
 embPattern_combine, 396
 embPattern_combineJumpStitches, 396
 embPattern_convertGeometry, 396
 embPattern_copyPolylinesToStitchList, 397
 embPattern_copyStitchListToPolylines, 397
 embPattern_correctForMaxStitchLength, 397
 embPattern_create, 397
 embPattern_crossstitch, 397
 embPattern_designDetails, 397
 embPattern_end, 397
 embPattern_fixColorCount, 397
 embPattern_flip, 397
 embPattern_flipHorizontal, 398
 embPattern_flipVertical, 398
 embPattern_free, 398
 embPattern_hideStitchesOverLength, 398
 embPattern_horizontal_fill, 398
 embPattern_jumpStitches, 398
 embPattern_lengthHistogram, 398
 embPattern_loadExternalColorFile, 398
 embPattern_maximumStitchLength, 398
 embPattern_minimumStitchLength, 398
 embPattern_movePolylinesToStitchList, 399
 embPattern_moveStitchListToPolylines, 399
 embPattern_read, 399
 embPattern_readAuto, 399
 embPattern_realStitches, 399
 embPattern_render, 399
 embPattern_scale, 399
 embPattern_simulate, 399
 embPattern_totalStitchLength, 399
 embPattern_trimStitches, 400
 embPattern_write, 400
 embPattern_writeAuto, 400
 EmbPoint, 389
 EmbPolygon, 389
 EmbPolyline, 389
 EmbRadiusDim, 389
 EmbRay, 389
 EmbReal, 389
 EmbRect, 389
 embRect_area, 400
 embRect_init, 400
 EmbSatinOutline, 389
 embSatinOutline_generateSatinOutline, 400
 embSatinOutline_renderStitches, 400
 EmbSpline, 389
 EmbStitch, 390
 EmbTextMulti, 390
 EmbTextSingle, 390
 EmbThread, 390
 embThread_findNearestColor, 400
 embThread_findNearestThread, 401
 embThread_getRandom, 401
 EmbTime, 390
 embTime_initNow, 401
 embTime_time, 401
 EmbVector, 390
 embVector_add, 401
 embVector_angle, 401
 embVector_average, 402
 embVector_cross, 402
 embVector_distance, 402
 embVector_dot, 402
 embVector_length, 402
 embVector_multiply, 402
 embVector_normalize, 403
 embVector_relativeX, 403
 embVector_relativeY, 403
 embVector_subtract, 403
 embVector_transpose_product, 403
 embVector_unit, 403
 END, 386
 Exquisite_Polyester, 386
 formatTable, 405
 Fufu_Polyester, 386
 Fufu_Rayon, 386
 full_test_matrix, 403
 getArcCenter, 403
 getArcDataFromBulge, 404
 getCircleCircleIntersections, 404
 getCircleTangentPoints, 404
 Hemingworth_Polyester, 386
 hilbert_curve, 404
 hus_thread, 386
 husThreads, 405
 Isacord_Polyester, 386
 Isafil_Rayon, 386
 jef_thread, 386
 jefThreads, 405
 JUMP, 386
 L_system, 390
 LIBEMBROIDERY_EMBEDDED_VERSION, 386
 lindenmayer_system, 404
 Madeira_Polyester, 386
 Madeira_Rayon, 386
 Marathon_Polyester, 386
 Marathon_Rayon, 386
 MAX_STITCHES, 387
 MAX_THREADS, 387
 Metro_Polyester, 387
 NORMAL, 387
 numberFormats, 387
 Pantone, 387
 pcm_thread, 387
 pcmThreads, 406
 pec_thread, 387

pecThreadCount, 406
pecThreads, 406
radians, 404
report, 404
RobisonAnton_Polyester, 387
RobisonAnton_Rayon, 387
SEQUIN, 387
shv_thread, 387
shvThreadCount, 406
shvThreads, 406
Sigma_Polyester, 387
STOP, 387
Sulky_Rayon, 387
SVG_Colors, 387
testMain, 405
thread_color, 390
ThreadArt_Polyester, 387
ThreadArt_Rayon, 387
threadColor, 405
threadColorName, 405
threadColorNum, 405
ThreaDelight_Polyester, 388
TRIM, 388
vipDecodingTable, 406
Z102_Isacord_Polyester, 388
embroidery_internal.h
 bcf_difat_create, 431
 bcf_directory, 429
 bcf_directory_entry, 429
 bcf_directory_free, 431
 bcf_file, 429
 bcf_file_difat, 429
 bcf_file_difat_free, 431
 bcf_file_fat, 429
 bcf_file_fat_free, 431
 bcf_file_free, 431
 bcf_file_header, 429
 bcfFile_read, 431
 bcfFileFat_create, 431
 bcfFileHeader_isValid, 431
 bcfFileHeader_read, 432
 binaryReadString, 432
 binaryReadUnicodeString, 432
 binaryWriteInt, 432
 binaryWriteIntBE, 432
 binaryWriteShort, 432
 binaryWriteUInt, 432
 binaryWriteUIntBE, 432
 binaryWriteUShort, 433
 binaryWriteUShortBE, 433
 BULGETOCONTROL, 421
 BULGETOEND, 422
 check_header_present, 433
 CompoundFileDirectory, 433
 CompoundFileDirectoryEntry, 433
 CompoundFileSector_DIFAT_Sector, 422
 CompoundFileSector_EndOfChain, 422
 CompoundFileSector_FAT_Sector, 422
 CompoundFileSector_FreeSector, 422
 CompoundFileSector_MaxRegSector, 422
 CompoundFileStreamId_MaxRegularStreamId, 422
 CompoundFileStreamId_NoStream, 422
 compress, 430
 compress_get_bits, 433
 compress_get_position, 433
 compress_get_token, 434
 compress_load_block, 434
 compress_load_character_huffman, 434
 compress_load_character_length_huffman, 434
 compress_load_distance_huffman, 434
 compress_pop, 434
 compress_read_variable_length, 434
 copy_trim, 434
 create_test_file_1, 434
 create_test_file_2, 434
 create_test_file_3, 434
 CSV_EXPECT, 430
 CSV_EXPECT_COMMA, 430
 CSV_EXPECT_NULL, 430
 CSV_EXPECT_QUOTE1, 430
 CSV_EXPECT_QUOTE2, 430
 CSV_MODE, 430
 CSV_MODE_COMMENT, 430
 CSV_MODE_NULL, 430
 CSV_MODE_STITCH, 431
 CSV_MODE_THREAD, 430
 CSV_MODE_VARIABLE, 430
 CUBICTOCONTROL1, 422
 CUBICTOCONTROL2, 422
 CUBICTOEND, 422
 decode_t01_record, 434
 decode_tajima_ternary, 435
 decodeNewStitch, 435
 DXF_VERSION_2000, 422
 DXF_VERSION_2002, 422
 DXF_VERSION_2004, 422
 DXF_VERSION_2006, 422
 DXF_VERSION_2007, 422
 DXF_VERSION_2009, 422
 DXF_VERSION_2010, 423
 DXF_VERSION_2013, 423
 DXF_VERSION_R10, 423
 DXF_VERSION_R11, 423
 DXF_VERSION_R12, 423
 DXF_VERSION_R13, 423
 DXF_VERSION_R14, 423
 DXF_VERSION_R15, 423
 DXF_VERSION_R18, 423
 DXF_VERSION_R21, 423
 DXF_VERSION_R24, 423
 DXF_VERSION_R27, 423
 ELEMENT_A, 423
 ELEMENT_ANIMATE, 423
 ELEMENT_ANIMATECOLOR, 423
 ELEMENT_ANIMATEMOTION, 423

ELEMENT_ANIMATETRANSFORM, 423
 ELEMENT_ANIMATION, 423
 ELEMENT_AUDIO, 424
 ELEMENT_CIRCLE, 424
 ELEMENT_DEFS, 424
 ELEMENT_DESC, 424
 ELEMENT_DISCARD, 424
 ELEMENT_ELLIPSE, 424
 ELEMENT_FONT, 424
 ELEMENT_FONT_FACE, 424
 ELEMENT_FONT_FACE_SRC, 424
 ELEMENT_FONT_FACE_URI, 424
 ELEMENT_FOREIGN_OBJECT, 424
 ELEMENT_G, 424
 ELEMENT_GLYPH, 424
 ELEMENT_HANDLER, 424
 ELEMENT_HKERN, 424
 ELEMENT_IMAGE, 424
 ELEMENT_LINE, 424
 ELEMENT_LINEAR_GRADIENT, 424
 ELEMENT_LISTENER, 425
 ELEMENT_METADATA, 425
 ELEMENT_MISSING_GLYPH, 425
 ELEMENT_MPATH, 425
 ELEMENT_PATH, 425
 ELEMENT_POLYGON, 425
 ELEMENT_POLYLINE, 425
 ELEMENT_PREFETCH, 425
 ELEMENT_RADIAL_GRADIENT, 425
 ELEMENT_RECT, 425
 ELEMENT_SCRIPT, 425
 ELEMENT_SET, 425
 ELEMENT_SOLID_COLOR, 425
 ELEMENT_STOP, 425
 ELEMENT_SVG, 425
 ELEMENT_SWITCH, 425
 ELEMENT_TBREAK, 425
 ELEMENT_TEXT, 425
 ELEMENT_TEXT_AREA, 426
 ELEMENT_TITLE, 426
 ELEMENT_TSPAN, 426
 ELEMENT_USE, 426
 ELEMENT_VIDEO, 426
 ELEMENT_XML, 426
 ELLIPSETOEND, 426
 ELLIPSETORAD, 426
 EMB_BIG_ENDIAN, 426
 EMB_INT16_BIG, 426
 EMB_INT16_LITTLE, 426
 EMB_INT32_BIG, 426
 EMB_INT32_LITTLE, 426
 EMB_LITTLE_ENDIAN, 426
 EMB_MAX, 426
 EMB_MIN, 426
 emb_optOut, 435
 emb_readline, 435
 embColor_read, 435
 embColor_write, 435
 emblInt_read, 436
 emblInt_write, 436
 encode_t01_record, 436
 encode_tajima_ternary, 436
 ENDIAN_HOST, 426
 entriesInDifatSector, 436
 fpad, 436
 fread_int16, 436
 fread_int32_be, 437
 fread_uint16, 437
 GetFile, 437
 GREEN_TERM_COLOR, 427
 HOOP_110X110, 427
 HOOP_126X110, 427
 HOOP_140X200, 427
 HOOP_230X200, 427
 HOOP_50X50, 427
 huffman, 430
 huffman_build_table, 437
 huffman_table_lookup, 437
 hus_compress, 437
 hus_decompress, 437
 imageWithFrame, 453
 LINETO, 427
 loadFatFromSector, 438
 mitDecodeStitch, 438
 mitEncodeStitch, 438
 MOVETO, 427
 N_PES VERSIONS, 427
 numberEntriesInDifatSector, 438
 ObjectTypeRootEntry, 427
 ObjectTypeStorage, 427
 ObjectTypeStream, 427
 ObjectTypeUnknown, 427
 PES0001, 427
 PES0020, 427
 PES0022, 427
 PES0030, 427
 PES0040, 428
 PES0050, 428
 PES0055, 428
 PES0056, 428
 PES0060, 428
 PES0070, 428
 PES0080, 428
 PES0090, 428
 PES0100, 428
 pfaffDecode, 438
 pfaffEncode, 438
 printArcResults, 438
 QUADTOCONTROL, 428
 QUADTOEND, 428
 read100, 439
 read10o, 439
 readArt, 439
 readBmc, 439
 readBro, 439
 readCnd, 439

readCol, 439
readCsd, 439
readCsv, 439
readDat, 439
readDem, 439
readDescriptions, 440
readDsb, 440
readDst, 440
readDsz, 440
readDxf, 440
readEdr, 440
readEmd, 440
readExp, 440
readExy, 440
readEys, 440
readFeatherPatterns, 441
readFullSector, 441
readFxy, 441
readGc, 441
readGnc, 441
readGt, 441
readHoopName, 441
readHus, 441
readImageString, 441
readInb, 442
readInf, 442
readJef, 442
readKsm, 442
readMax, 442
readMit, 442
readMotifPatterns, 442
readNew, 442
readNextSector, 442
readOfm, 442
readPcd, 443
readPcm, 443
readPcq, 443
readPcs, 443
readPec, 443
readPecStitches, 443
readPel, 443
readPem, 443
readPes, 444
readPESHeaderV10, 444
readPESHeaderV5, 444
readPESHeaderV6, 444
readPESHeaderV7, 444
readPESHeaderV8, 444
readPESHeaderV9, 444
readPhb, 444
readPhc, 444
readPlt, 444
readProgrammableFills, 445
readRgb, 445
readSew, 445
readShv, 445
readSst, 445
readStx, 445
readSvg, 445
readT01, 445
readT09, 445
readTap, 445
readThr, 446
readThreads, 446
readTxt, 446
readU00, 446
readU01, 446
readVip, 446
readVp3, 446
readXxx, 446
readZsk, 446
RED_TERM_COLOR, 428
RESET_TERM_COLOR, 428
safe_free, 446
stringInArray, 447
StxThread, 430
SubDescriptor, 430
SVG_ATTRIBUTE, 428
SVG_CATCH_ALL, 428
SVG_CREATOR_EMBROIDERMODDER, 428
SVG_CREATOR_ILLUSTRATOR, 428
SVG_CREATOR_INKSCAPE, 428
SVG_CREATOR_NULL, 429
SVG_ELEMENT, 429
SVG_EXPECT_ATTRIBUTE, 429
SVG_EXPECT_ELEMENT, 429
SVG_EXPECT_NULL, 429
SVG_EXPECT_VALUE, 429
SVG_MEDIA_PROPERTY, 429
SVG_NULL, 429
SVG_PROPERTY, 429
SvgAttribute, 430
testEmbCircle, 447
testEmbCircle_2, 447
testEmbFormat, 447
testGeomArc, 447
testTangentPoints, 447
testThreadColor, 447
ThredExtension, 430
ThredHeader, 430
VipHeader, 430
vp3Hoop, 430
write100, 447
write10o, 447
write_24bit, 447
writeArt, 448
writeBmc, 448
writeBro, 448
writeCnd, 448
writeCol, 448
writeCsd, 448
writeCsv, 448
writeDat, 448
writeDem, 448
writeDsb, 448
writeDst, 448

writeDsz, 449
 writeDxf, 449
 writeEdr, 449
 writeEmd, 449
 writeExp, 449
 writeExy, 449
 writeEys, 449
 writeFxy, 449
 writeGc, 449
 writeGnc, 449
 writeGt, 449
 writeHus, 450
 writeInb, 450
 writeInf, 450
 writeJef, 450
 writeKsm, 450
 writeMax, 450
 writeMit, 450
 writeNew, 450
 writeOfm, 450
 writePcd, 450
 writePcm, 450
 writePcq, 451
 writePcs, 451
 writePec, 451
 writePecStitches, 451
 writePel, 451
 writePem, 451
 writePes, 451
 writePhb, 451
 writePhc, 451
 writePlt, 451
 writeRgb, 452
 writeSew, 452
 writeShv, 452
 writeSst, 452
 writeStx, 452
 writeSvg, 452
 writeT01, 452
 writeT09, 452
 writeTap, 452
 writeThr, 452
 writeTxt, 452
 writeU00, 453
 writeU01, 453
 writeVip, 453
 writeVp3, 453
 writeXxx, 453
 writeZsk, 453
 YELLOW_TERM_COLOR, 429
EmbSatinOutline
 embroidery.h, 389
EmbSatinOutline_, 117
 length, 117
 side1, 117
 side2, 117
embSatinOutline_generateSatinOutline
 embroidery.h, 400
 main.c, 536
embSatinOutline_renderStitches
 embroidery.h, 400
 main.c, 536
EmbSpline
 embroidery.h, 389
EmbSpline_, 117
 beziers, 117
EmbStitch
 embroidery.h, 390
EmbStitch_, 117
 color, 118
 flags, 118
 x, 118
 y, 118
EmbTextMulti
 embroidery.h, 390
EmbTextMulti_, 118
 position, 118
 text, 118
EmbTextSingle
 embroidery.h, 390
EmbTextSingle_, 118
 position, 118
 text, 118
EmbThread
 embroidery.h, 390
EmbThread_, 119
 catalogNumber, 119
 color, 119
 description, 119
embThread_findNearestColor
 embroidery.h, 400
 main.c, 537
embThread_findNearestThread
 embroidery.h, 401
 main.c, 537
embThread_getRandom
 embroidery.h, 401
 main.c, 537
EmbTime
 embroidery.h, 390
EmbTime_, 119
 day, 119
 hour, 119
 minute, 119
 month, 119
 second, 119
 year, 120
embTime_initNow
 embroidery.h, 401
 main.c, 537
embTime_time
 embroidery.h, 401
 main.c, 537
EmbVector
 embroidery.h, 390
EmbVector_, 120

x, 120
y, 120
embVector_add
 embroidery.h, 401
 vector.c, 526
embVector_angle
 embroidery.h, 401
 vector.c, 526
embVector_average
 embroidery.h, 402
 vector.c, 526
embVector_cross
 embroidery.h, 402
 vector.c, 526
embVector_distance
 embroidery.h, 402
 vector.c, 527
embVector_dot
 embroidery.h, 402
 vector.c, 527
embVector_length
 embroidery.h, 402
 vector.c, 527
embVector_multiply
 embroidery.h, 402
 vector.c, 527
embVector_normalize
 embroidery.h, 403
 vector.c, 527
embVector_print
 main.c, 537
embVector_relativeX
 embroidery.h, 403
 vector.c, 527
embVector_relativeY
 embroidery.h, 403
 vector.c, 528
embVector_subtract
 embroidery.h, 403
 vector.c, 528
embVector_transpose_product
 embroidery.h, 403
 vector.c, 528
embVector_unit
 embroidery.h, 403
 vector.c, 528
EmbView
 embroidermodder.h, 301
EmbView_, 120
 filename, 121
 grid_mode, 121
 grid_type, 121
 lwt_mode, 121
 metric, 121
 n_selected, 121
 origin, 121
 ortho_mode, 121
 pattern, 121
polar_mode, 121
qsnap_mode, 121
qtrack_mode, 121
real_render, 122
rubber_mode, 122
ruler_mode, 122
scale, 122
selected, 122
simulate, 122
simulation_start, 122
snap_mode, 122
text_angle, 122
text_font, 122
text_size, 122
text_style_bold, 122
text_style_italic, 122
text_style_overline, 122
text_style_strikeout, 122
text_style_underline, 122
ui_mode, 122
undo_history, 122
emd, 17, 484
emdDecode
 format_emd.c, 484
enableLwt
 StatusBarButton, 258
enableMoveRapidFire
 MainWindow, 148
 View, 287
enablePromptRapidFire
 MainWindow, 148
enableRapidFire
 CmdPrompt, 78
enableReal
 StatusBarButton, 258
encode_record
 format_dst.c, 482
encode_t01_record
 embroidery_internal.h, 436
 encoding.c, 462
encode_tajima_ternary
 embroidery_internal.h, 436
 encoding.c, 462
encode_tap_record
 format_tap.c, 508
encoding.c
 decode_t01_record, 461
 decode_tajima_ternary, 461
 decodeNewStitch, 461
 embColor_fromHexStr, 461
 emblnt_read, 462
 emblnt_write, 462
 encode_t01_record, 462
 encode_tajima_ternary, 462
 mitDecodeStitch, 462
 mitEncodeStitch, 462
 pfaffDecode, 462
 pfaffEncode, 463

reverse_byte_order, 463
 write_24bit, 463
END
 embroidery.h, 386
end
 EmbArc_, 103
 EmbBezier_, 105
 EmbLine_, 113
endCommand
 CmdPrompt, 78
 CmdPromptInput, 87
ENDIAN_HOST
 embroidery_internal.h, 426
enterEvent
 View, 287
entriesInDifatSector
 embroidery_internal.h, 436
 main.c, 538
Error
 mainwindow.cpp, 348
escapePressed
 CmdPrompt, 78
 CmdPromptInput, 87
 MainWindow, 148
 MdiWindow, 182
 View, 287
event
 Application, 56
eventFilter
 CmdPromptInput, 87
 PropertyEditor, 209
exit
 MainWindow, 148
exitApp
 embroidermodder.cpp, 297
exp, 17, 484
expDecode
 format_exp.c, 484
Exquisite_Polyester
 embroidery.h, 386
extension
 EmbFormatList_, 109
 extern/libembroidery/src/array.c, 369
 extern/libembroidery/src/compress.c, 371
 extern/libembroidery/src/embroidery.h, 373, 406
 extern/libembroidery/src/embroidery_internal.h, 413,
 453
 extern/libembroidery/src/encoding.c, 460
 extern/libembroidery/src/fill.c, 463
 extern/libembroidery/src/formats.c, 468
 extern/libembroidery/src/formats/format_100.c, 472
 extern/libembroidery/src/formats/format_10o.c, 472
 extern/libembroidery/src/formats/format_art.c, 473
 extern/libembroidery/src/formats/format_bmc.c, 473
 extern/libembroidery/src/formats/format_bro.c, 474
 extern/libembroidery/src/formats/format_cnd.c, 474
 extern/libembroidery/src/formats/format_col.c, 475
 extern/libembroidery/src/formats/format_csd.c, 476
 extern/libembroidery/src/formats/format_csv.c, 477
 extern/libembroidery/src/formats/format_dat.c, 478
 extern/libembroidery/src/formats/format_dem.c, 479
 extern/libembroidery/src/formats/format_dsb.c, 479
 extern/libembroidery/src/formats/format_dst.c, 480
 extern/libembroidery/src/formats/format_dsz.c, 482
 extern/libembroidery/src/formats/format_dxf.c, 482
 extern/libembroidery/src/formats/format_edr.c, 483
 extern/libembroidery/src/formats/format_emd.c, 484
 extern/libembroidery/src/formats/format_exp.c, 484
 extern/libembroidery/src/formats/format_exy.c, 485
 extern/libembroidery/src/formats/format_eyc.c, 485
 extern/libembroidery/src/formats/format_fxy.c, 486
 extern/libembroidery/src/formats/format_gc.c, 486
 extern/libembroidery/src/formats/format_gnc.c, 487
 extern/libembroidery/src/formats/format_gt.c, 487
 extern/libembroidery/src/formats/format_hus.c, 488
 extern/libembroidery/src/formats/format_inb.c, 489
 extern/libembroidery/src/formats/format_inf.c, 489
 extern/libembroidery/src/formats/format_jef.c, 490
 extern/libembroidery/src/formats/format_ksm.c, 491
 extern/libembroidery/src/formats/format_max.c, 491
 extern/libembroidery/src/formats/format_mit.c, 492
 extern/libembroidery/src/formats/format_new.c, 493
 extern/libembroidery/src/formats/format_ofm.c, 493
 extern/libembroidery/src/formats/format_pcd.c, 494
 extern/libembroidery/src/formats/format_pcm.c, 495
 extern/libembroidery/src/formats/format_pcq.c, 495
 extern/libembroidery/src/formats/format_pcs.c, 496
 extern/libembroidery/src/formats/format_pec.c, 496
 extern/libembroidery/src/formats/format_pel.c, 498
 extern/libembroidery/src/formats/format_pem.c, 498
 extern/libembroidery/src/formats/format_pes.c, 499
 extern/libembroidery/src/formats/format_phb.c, 501
 extern/libembroidery/src/formats/format_phc.c, 502
 extern/libembroidery/src/formats/format_plt.c, 502
 extern/libembroidery/src/formats/format_rgb.c, 503
 extern/libembroidery/src/formats/format_sew.c, 503
 extern/libembroidery/src/formats/format_shv.c, 504
 extern/libembroidery/src/formats/format_sst.c, 504
 extern/libembroidery/src/formats/format_stx.c, 505
 extern/libembroidery/src/formats/format_svg.c, 505
 extern/libembroidery/src/formats/format_t01.c, 507
 extern/libembroidery/src/formats/format_t09.c, 507
 extern/libembroidery/src/formats/format_tap.c, 508
 extern/libembroidery/src/formats/format_thr.c, 508
 extern/libembroidery/src/formats/format_txt.c, 509
 extern/libembroidery/src/formats/format_u00.c, 509
 extern/libembroidery/src/formats/format_u01.c, 510
 extern/libembroidery/src/formats/format_vip.c, 510
 extern/libembroidery/src/formats/format_vp3.c, 512
 extern/libembroidery/src/formats/format_xxx.c, 513
 extern/libembroidery/src/formats/format_zsk.c, 514
 extern/libembroidery/src/geometry.c, 514
 extern/libembroidery/src/geometry/arc.c, 516
 extern/libembroidery/src/geometry/circle.c, 519
 extern/libembroidery/src/geometry/ellipse.c, 520
 extern/libembroidery/src/geometry/functions.c, 522

extern/libembroidery/src/geometry/line.c, 523
extern/libembroidery/src/geometry/path.c, 523
extern/libembroidery/src/geometry/polygon.c, 523
extern/libembroidery/src/geometry/polyline.c, 523
extern/libembroidery/src/geometry/rect.c, 524
extern/libembroidery/src/geometry/text.c, 524
extern/libembroidery/src/geometry/vector.c, 526
extern/libembroidery/src/image.c, 528
extern/libembroidery/src/main.c, 529
extern/libembroidery/src/pattern.c, 540
extern/libembroidery/src/thread-color.c, 545
exy, 17, 485
eys, 441, 486

F10Pressed
 CmdPrompt, 78
 CmdPromptInput, 87

F11Pressed
 CmdPrompt, 78
 CmdPromptInput, 87

F12Pressed
 CmdPrompt, 78
 CmdPromptInput, 88

F1Pressed
 CmdPrompt, 78
 CmdPromptInput, 88

F2Pressed
 CmdPrompt, 78
 CmdPromptInput, 88

F3Pressed
 CmdPrompt, 78
 CmdPromptInput, 88

F4Pressed
 CmdPrompt, 78
 CmdPromptInput, 88

F5Pressed
 CmdPrompt, 79
 CmdPromptInput, 88

F6Pressed
 CmdPrompt, 79
 CmdPromptInput, 88

F7Pressed
 CmdPrompt, 79
 CmdPromptInput, 88

F8Pressed
 CmdPrompt, 79
 CmdPromptInput, 88

F9Pressed
 CmdPrompt, 79
 CmdPromptInput, 88

factor
 UndoableScaleCommand, 279

fat
 _bcf_file, 50

fatEntries
 _bcf_file_fat, 51

fatEntryCount
 _bcf_file_fat, 51

fatSectorCount
 _bcf_file_fat, 50

fatSectorEntries
 _bcf_file_fat, 50

fieldEdited
 PropertyEditor, 209

fieldNewText
 PropertyEditor, 210

fieldNoText
 PropertyEditor, 211

fieldOffText
 PropertyEditor, 211

fieldOldText
 PropertyEditor, 211

fieldOnText
 PropertyEditor, 211

fieldVariesText
 PropertyEditor, 211

fieldYesText
 PropertyEditor, 211

file_toolbar
 embroidermodder.h, 308
 mainwindow-toolbars.cpp, 345

fileExtension
 MdiWindow, 182

fileMenu
 MainWindow, 168

filename
 EmbView_, 121

fileWasLoaded
 MdiWindow, 186

fill.c
 dragon_curve, 464
 embPattern_combine, 464
 embPattern_convertGeometry, 464
 embPattern_crossstitch, 464
 embPattern_horizontal_fill, 464
 embPattern_stitchArc, 464
 embPattern_stitchCircle, 464
 embPattern_stitchEllipse, 465
 embPattern_stitchPath, 465
 embPattern_stitchPolygon, 465
 embPattern_stitchPolyline, 465
 embPattern_stitchRect, 466
 embPattern_stitchText, 466
 embPolygon_reduceByDistance, 466
 embPolygon_reduceByNth, 466
 generate_dragon_curve, 466
 greedy_algorithm, 466
 hilbert_curve, 466
 hilbert_curve_l_system, 467
 join_short_stitches, 467
 lindenmayer_system, 467
 rules, 467
 save_points_to_pattern, 467
 threshold_method, 467

filled
 DimLeaderObject, 98

findIndex

PolygonObject, 197
 PolylineObject, 201
 findMdiWindow
 MainWindow, 148
 firstDifatSectorLocation
 _bcf_file_header, 51
 firstDirectorySectorLocation
 _bcf_file_header, 51
 firstMiniFATSectorLocation
 _bcf_file_header, 52
 firstRun
 UiObject_, 270
 flag
 EmbGeometry_, 110
 FLAG_CIRCLE
 main.c, 531
 FLAG_CIRCLE_SHORT
 main.c, 532
 FLAG_COMBINE
 main.c, 532
 FLAG_CROSS_STITCH
 main.c, 532
 FLAG_ELLIPSE
 main.c, 532
 FLAG_ELLIPSE_SHORT
 main.c, 532
 FLAG_FILL
 main.c, 532
 FLAG_FILL_SHORT
 main.c, 532
 FLAG_FORMATS
 main.c, 532
 FLAG_FORMATS_SHORT
 main.c, 532
 FLAG_FULL_TEST_SUITE
 main.c, 532
 FLAG_HELP
 main.c, 532
 FLAG_HELP_SHORT
 main.c, 532
 FLAG_HILBERT_CURVE
 main.c, 532
 FLAG_LINE
 main.c, 532
 FLAG_LINE_SHORT
 main.c, 532
 FLAG_POLYGON
 main.c, 532
 FLAG_POLYGON_SHORT
 main.c, 532
 FLAG_POLYLINE
 main.c, 532
 FLAG_POLYLINE_SHORT
 main.c, 533
 FLAG QUIET
 main.c, 533
 FLAG QUIET_SHORT
 main.c, 533
 FLAG_RENDER
 main.c, 533
 FLAG_RENDER_SHORT
 main.c, 533
 FLAG_SATIN
 main.c, 533
 FLAG_SATIN_SHORT
 main.c, 533
 FLAG_SIERPINSKI_TRIANGLE
 main.c, 533
 FLAG_SIMULATE
 main.c, 533
 FLAG_STITCH
 main.c, 533
 FLAG_STITCH_SHORT
 main.c, 533
 FLAG_TEST
 main.c, 533
 FLAG_TO
 main.c, 533
 FLAG_TO_SHORT
 main.c, 533
 FLAG_VERBOSE
 main.c, 533
 FLAG_VERBOSE_SHORT
 main.c, 533
 FLAG_VERSION
 main.c, 533
 FLAG_VERSION_SHORT
 main.c, 533
 flagList
 EmbPath_, 114
 flags
 EmbStitch_, 118
 Flared
 DimLeaderObject, 95
 Fletching
 DimLeaderObject, 95
 floatingChanged
 CmdPrompt, 79
 floatingChangedToolBar
 MainWindow, 148
 fname
 UiObject_, 270
 focusWidget
 PropertyEditor, 211
 UndoEditor, 280
 forceRepaint
 MdiArea, 177
 SelectBox, 223
 format_100.c
 read100, 472
 write100, 472
 format_10o.c
 read10o, 473
 write10o, 473
 format_art.c
 readArt, 473

writeArt, 473
format_bmc.c
 readBmc, 474
 writeBmc, 474
format_bro.c
 readBro, 474
 writeBro, 474
format_cnd.c
 readCnd, 475
 writeCnd, 475
format_col.c
 readCol, 476
 writeCol, 476
format_csd.c
 _subMask, 477
 _xorMask, 477
 BuildDecryptionTable, 476
 csd_decryptArray, 477
 CsdSubMaskSize, 476
 CsdXorMaskSize, 476
 DecodeCsdByte, 477
 readCsd, 477
 writeCsd, 477
format_csv.c
 csvStitchFlagToStr, 478
 csvStrToStitchFlag, 478
 readCsv, 478
 writeCsv, 478
format_dat.c
 readDat, 478
 writeDat, 478
format_dem.c
 readDem, 479
 writeDem, 479
format_dsb.c
 readDsb, 480
 writeDsb, 480
format_dst.c
 cci, 481
 decode_record_flags, 481
 encode_record, 482
 readDst, 482
 set_dst_variable, 482
 writeDst, 482
format_dsz.c
 readDsz, 482
 writeDsz, 482
format_dxf.c
 readDxf, 483
 readLine, 483
 writeDxf, 483
format_edr.c
 readEdr, 483
 writeEdr, 483
format_emd.c
 emdDecode, 484
 readEmd, 484
 writeEmd, 484
format_exp.c
 expDecode, 484
 readExp, 484
 writeExp, 485
format_exy.c
 decode_exy_flags, 485
 readExy, 485
 writeExy, 485
format_eyc.c
 readEys, 485
 writeEys, 486
format_fxy.c
 readFxy, 486
 writeFxy, 486
format_gc.c
 readGc, 486
 writeGc, 487
format_gnc.c
 readGnc, 487
 writeGnc, 487
format_gt.c
 readGt, 487
 writeGt, 488
format_hus.c
 husCompressData, 488
 husDecodeByte, 488
 husDecodeStitchType, 488
 husDecompressData, 488
 husEncodeByte, 488
 husEncodeStitchType, 488
 readHus, 489
 writeHus, 489
format_inb.c
 readInb, 489
 writeInb, 489
format_inf.c
 readInf, 489
 writeInf, 490
format_jef.c
 jefDecode, 490
 jefEncode, 490
 jefGetHoopSize, 490
 jefSetHoopFromId, 490
 read_hoop, 490
 readJef, 491
 writeJef, 491
format_ksm.c
 ksmEncode, 491
 readKsm, 491
 writeKsm, 491
format_max.c
 max_header, 492
 readMax, 492
 writeMax, 492
format_mit.c
 readMit, 492
 writeMit, 493
format_new.c

```

readNew, 493
writeNew, 493
format_ofm.c
ofmDecode, 494
ofmReadBlockHeader, 494
ofmReadClass, 494
ofmReadColorChange, 494
ofmReadExpanded, 494
ofmReadLibrary, 494
ofmReadThreads, 494
readOfm, 494
writeOfm, 494
format_pcd.c
readPcd, 495
writePcd, 495
format_pcm.c
readPcm, 495
writePcm, 495
format_pcq.c
readPcq, 496
writePcq, 496
format_pcs.c
readPcs, 496
writePcs, 496
format_pec.c
pecEncode, 497
pecEncodeJump, 497
pecEncodeStop, 497
readPec, 497
readPecStitches, 497
writeImage, 497
writePec, 497
writePecStitches, 497
format_pel.c
readPel, 498
writePel, 498
format_pem.c
readPem, 498
writePem, 498
format_pes.c
pes_version, 501
pes_version_strings, 501
pesWriteEmbOneSection, 499
pesWriteSewSegSection, 499
readDescriptions, 499
readFeatherPatterns, 499
readHoopName, 499
readImageString, 500
readMotifPatterns, 500
readPes, 500
readPESHeaderV10, 500
readPESHeaderV5, 500
readPESHeaderV6, 500
readPESHeaderV7, 500
readPESHeaderV8, 500
readPESHeaderV9, 500
readProgrammableFills, 500
readThreads, 500
writePes, 501
format_phb.c
readPhb, 501
writePhb, 501
format_phc.c
readPhc, 502
writePhc, 502
format_plt.c
readPlt, 502
writePlt, 502
format_rgb.c
readRgb, 503
writeRgb, 503
format_sew.c
readSew, 503
sewDecode, 503
writeSew, 503
format_shv.c
readShv, 504
shvDecode, 504
shvDecodeShort, 504
writeShv, 504
format_sst.c
readSst, 504
writeSst, 505
format_stx.c
readStx, 505
stxReadThread, 505
writeStx, 505
format_svg.c
attributeList, 506
current_element_id, 506
currentAttribute, 506
currentValue, 506
n_attributes, 506
readSvg, 506
svgCreator, 506
svgExpect, 506
svgMultiValue, 506
writeSvg, 506
format_t01.c
readT01, 507
writeT01, 507
format_t09.c
readT09, 507
writeT09, 507
format_tap.c
decode_tap_record_flags, 508
encode_tap_record, 508
readTap, 508
writeTap, 508
format_thr.c
readThr, 508
writeThr, 509
format_txt.c
readTxt, 509
writeTxt, 509
format_u00.c

```

readU00, 509
writeU00, 510
format_u01.c
readU01, 510
writeU01, 510
format_vip.c
readVip, 511
vipCompressData, 511
vipDecodeByte, 511
vipDecodeStitchType, 511
vipDecodingTable, 511
vipDecompressData, 511
vipEncodeByte, 511
vipEncodeStitchType, 511
writeVip, 511
format_vp3.c
readVp3, 512
vp3Decode, 512
vp3DecodeInt16, 512
vp3PatchByteCount, 512
vp3ReadHoopSection, 512
vp3ReadString, 513
vp3WriteString, 513
vp3WriteStringLen, 513
writeVp3, 513
format_xxx.c
readXxx, 513
writeXxx, 513
xxxDecodeByte, 513
xxxEncodeDesign, 514
xxxEncodeStitch, 514
xxxEncodeStop, 514
format_zsk.c
readZsk, 514
writeZsk, 514
formatFilterOpen
MainWindow, 168
formatFilterSave
MainWindow, 168
formats.c
binaryWriteInt, 469
binaryWriteIntBE, 469
binaryWriteShort, 469
binaryWriteUInt, 469
binaryWriteUIntBE, 469
binaryWriteUShort, 469
binaryWriteUShortBE, 469
emb_identify_format, 470
embFormat_getExtension, 470
embPattern_read, 470
embPattern_readAuto, 470
embPattern_write, 470
embPattern_writeAuto, 470
formatTable, 471
fpad, 471
fread_int16, 471
fread_int32_be, 471
fread_uint16, 471
imageWithFrame, 472
safe_free, 471
formatTable
embroidery.h, 405
formats.c, 471
formatType
SaveObject, 222
Fortron, 441, 486
fpad
embroidery_internal.h, 436
formats.c, 471
fread_int16
embroidery_internal.h, 436
formats.c, 471
fread_int32_be
embroidery_internal.h, 437
formats.c, 471
fread_uint16
embroidery_internal.h, 437
formats.c, 471
fromCenter
UndoableNavCommand, 276
fromTransform
UndoableNavCommand, 276
Fufu_Polyester
embroidery.h, 386
Fufu_Rayon
embroidery.h, 386
full_test_matrix
embroidery.h, 403
functions.c
degrees, 522
emb_round, 522
radians, 522
fx, 17, 441, 486
g
EmbColor_, 106
general_check_for_updates
Settings_, 228
general_current_tip
Settings_, 228
general_mdi_bg_color
Settings_, 228
general_mdi_bg_logo
Settings_, 229
general_mdi_bg_texture
Settings_, 229
general_system_help_browser
Settings_, 229
generate_dragon_curve
fill.c, 466
geometry
EmbArray_, 104
EmbLayer_, 112
EmbPattern_, 115
geometry.c
embGeometry_boundingRect, 515
embGeometry_free, 515

embGeometry_init, 515
 embGeometry_move, 515
 embGeometry_vulcanize, 515
 get_action_index
 embroidermodder.h, 307
 mainwindow-toolbars.cpp, 345
 get_trim_bounds
 main.c, 538
 getAction
 MainWindow, 149
 getApplication
 MainWindow, 149
 getArcCenter
 arc.c, 519
 embroidery.h, 403
 getArcDataFromBulge
 arc.c, 519
 embroidery.h, 404
 getCircleCircleIntersections
 circle.c, 520
 embroidery.h, 404
 getCircleTangentPoints
 circle.c, 520
 embroidery.h, 404
 getCurrentColor
 MainWindow, 149
 MdiWindow, 182
 getCurrentFile
 MdiWindow, 183
 getCurrentLayer
 MainWindow, 149
 MdiWindow, 183
 getCurrentLineType
 MainWindow, 149
 MdiWindow, 183
 getCurrentLineWidth
 MainWindow, 149
 MdiWindow, 183
 getCurrentText
 CmdPrompt, 79
 GetFile
 embroidery_internal.h, 437
 main.c, 538
 getFileSeparator
 MainWindow, 149
 getHistory
 CmdPrompt, 79
 getInfo
 EmbDetailsDialog, 107
 getMdiArea
 MainWindow, 149
 getPrefix
 CmdPrompt, 79
 getScene
 MdiWindow, 183
 getShortCurrentFile
 MdiWindow, 183
 getUndoStack

 View, 287
 getView
 MdiWindow, 183
 gnc, 17, 441, 487
 Gold Thread, 441, 488
 Great Notions, 441, 487
 greedy_algorithm
 fill.c, 466
 GREEN_TERM_COLOR
 embroidery_internal.h, 427
 grid_center
 Settings_, 229
 grid_center_on_origin
 Settings_, 229
 grid_color
 Settings_, 229
 grid_color_match_crosshair
 Settings_, 229
 grid_load_from_file
 Settings_, 229
 grid_mode
 EmbView_, 121
 grid_show_on_load
 Settings_, 229
 grid_show_origin
 Settings_, 229
 grid_size
 Settings_, 229
 grid_size_radius
 Settings_, 229
 grid_spacing
 Settings_, 229
 grid_spacing_angle
 Settings_, 229
 grid_spacing_radius
 Settings_, 229
 grid_type
 EmbView_, 121
 Settings_, 229
 gridColor
 View, 292
 gridPath
 View, 292
 gripBaseObj
 View, 292
 gripColorCool
 View, 292
 gripColorHot
 View, 292
 gripEdit
 ArcObject, 61
 BaseObject, 68
 CircleObject, 74
 DimLeaderObject, 95
 EllipseObject, 101
 ImageObject, 126
 LineObject, 133
 PathObject, 189

PointObject, 193
PolygonObject, 197
PolylineObject, 201
RectObject, 216
TextSingleObject, 264
gripIndex
 PolygonObject, 199
 PolylineObject, 203
grippingActive
 View, 292
gripSize
 View, 292
groupBoxes
 property-editor.cpp, 357
groupBoxGeometryCircle
 property-editor.cpp, 357
groupBoxGeometryDimAligned
 property-editor.cpp, 357
groupBoxGeometryDimAngular
 property-editor.cpp, 357
groupBoxGeometryDimArcLength
 property-editor.cpp, 357
groupBoxGeometryDimDiameter
 property-editor.cpp, 357
groupBoxGeometryDimLeader
 property-editor.cpp, 357
groupBoxGeometryDimLinear
 property-editor.cpp, 357
groupBoxGeometryDimOrdinate
 property-editor.cpp, 358
groupBoxGeometryDimRadius
 property-editor.cpp, 358
groupBoxGeometryEllipse
 property-editor.cpp, 358
groupBoxGeometryImage
 property-editor.cpp, 358
groupBoxGeometryInfiniteLine
 property-editor.cpp, 358
groupBoxGeometryLine
 property-editor.cpp, 358
groupBoxGeometryPath
 property-editor.cpp, 358
groupBoxGeometryPoint
 property-editor.cpp, 358
groupBoxGeometryPolygon
 property-editor.cpp, 358
groupBoxGeometryPolyline
 property-editor.cpp, 358
groupBoxGeometryRay
 property-editor.cpp, 358
groupBoxGeometryRectangle
 property-editor.cpp, 358
groupBoxGeometryTextMulti
 property-editor.cpp, 358
groupBoxGeometryTextSingle
 property-editor.cpp, 358
groupBoxMisclImage
 property-editor.cpp, 358
groupBoxMiscPath
 property-editor.cpp, 358
groupBoxMiscPolyline
 property-editor.cpp, 358
groupBoxMiscTextSingle
 property-editor.cpp, 358
groupBoxTextTextSingle
 property-editor.cpp, 359
gscene
 MdiWindow, 186
 SaveObject, 222
 View, 292
gt, 17, 441, 488
gview
 MdiWindow, 186
 UndoableAddCommand, 272
 UndoableDeleteCommand, 272
 UndoableGripEditCommand, 273
 UndoableMirrorCommand, 274
 UndoableMoveCommand, 275
 UndoableNavCommand, 277
 UndoableRotateCommand, 278
 UndoableScaleCommand, 279
handleMoved
 CmdPromptHandle, 82
handlePressed
 CmdPromptHandle, 82
handleReleased
 CmdPromptHandle, 83
Happy, 508
hash
 Action____, 55
hashDeletedObjects
 View, 292
haveExtraDIFATSectors
 main.c, 538
header
 _bcf_file, 50
HEART_MODE_NUM_POINTS
 embroidermodder.h, 306
HEART_MODE_STYLE
 embroidermodder.h, 306
HEART_MODE_XSCALE
 embroidermodder.h, 306
HEART_MODE_YSCALE
 embroidermodder.h, 306
height
 _vp3Hoop, 53
 EmblImage____, 111
help
 MainWindow, 149
help_toolbar
 mainwindow-toolbars.cpp, 346
helpMenu
 MainWindow, 168
Hemingworth_Polyester
 embroidery.h, 386
hex_code

thread_color_, 267
 hideAllGroups
 PropertyEditor, 209
 hideUnimplemented
 MainWindow, 150
 hilbert_curve
 embroidery.h, 404
 fill.c, 466
 hilbert_curve_l_system
 fill.c, 467
 historyAppended
 CmdPrompt, 79
 CmdPromptHistory, 84
 home
 EmbPattern_, 115
 HOOP_110X110
 embroidery_internal.h, 427
 HOOP_126X110
 embroidery_internal.h, 427
 HOOP_140X200
 embroidery_internal.h, 427
 HOOP_230X200
 embroidery_internal.h, 427
 HOOP_50X50
 embroidery_internal.h, 427
 hoop_height
 EmbPattern_, 115
 hoop_padding, 123
 bottom, 123
 left, 123
 right, 123
 top, 123
 hoop_width
 EmbPattern_, 115
 hoopSize
 ThredHeader_, 268
 hoopX
 ThredExtension_, 268
 hoopY
 ThredExtension_, 268
 hour
 EmbTime_, 119
 Huffman, 123
 default_value, 123
 lengths, 123
 nlengths, 123
 ntable, 123
 table, 124
 table_width, 124
 huffman
 embroidery_internal.h, 430
 huffman_build_table
 compress.c, 373
 embroidery_internal.h, 437
 huffman_lookup
 compress.c, 373
 huffman_lookup_data
 compress.c, 373
 huffman_table_lookup
 embroidery_internal.h, 437
 hus, 17, 488
 hus_compress
 compress.c, 373
 embroidery_internal.h, 437
 hus_decompress
 compress.c, 373
 embroidery_internal.h, 437
 hus_thread
 embroidery.h, 386
 husCompressData
 format_hus.c, 488
 husDecodeByte
 format_hus.c, 488
 husDecodeStitchType
 format_hus.c, 488
 husDecompressData
 format_hus.c, 488
 husEncodeByte
 format_hus.c, 488
 husEncodeStitchType
 format_hus.c, 488
 Husqvarna Viking, 488, 504
 husThreads
 embroidery.h, 405
 thread-color.c, 546
 i_value
 Parameter_, 187
 icon
 Action_, 55
 icon_size
 Settings_, 229
 icon_theme
 Settings_, 229
 icon_toolbar
 mainwindow-toolbars.cpp, 346
 iconDir
 PropertyEditor, 211
 UndoEditor, 281
 iconResize
 MainWindow, 150
 iconSize
 PropertyEditor, 211
 UndoEditor, 281
 id
 UiObject_, 270
 UndoableNavCommand, 276
 image.c
 image_diff, 528
 writeImage, 529
 image_diff
 image.c, 528
 ImageObject, 124
 ~ImageObject, 126
 allGripPoints, 126
 gripEdit, 126
 ImageObject, 126

init, 126
mouseSnapPoint, 126
objectArea, 126
objectBottomLeft, 126
objectBottomRight, 127
objectHeight, 127
objectTopLeft, 127
objectTopRight, 127
objectWidth, 127
paint, 127
setObjectRect, 127
Type, 126
type, 127
updatePath, 127
updateRubber, 127
vulcanize, 127
ImageWidget, 127
~ImageWidget, 128
ImageWidget, 128
img, 129
load, 128
paintEvent, 128
save, 129
imageWithFrame
embroidery_internal.h, 453
formats.c, 472
img
ImageWidget, 129
imgWidget
PreviewDialog, 203
inb, 17, 442, 489
Inbro, 442, 489
Index
embroidermodder.h, 301
inf, 442, 490
init
ArcObject, 61
CircleObject, 74
DimLeaderObject, 96
EllipseObject, 101
ImageObject, 126
LineObject, 133
PathObject, 189
PointObject, 193
PolygonObject, 197
PolylineObject, 201
RectObject, 216
TextSingleObject, 264
input_data
Compress, 91
input_length
Compress, 91
Isacord_Polyester
embroidery.h, 386
Isafil_Rayon
embroidery.h, 386
isBlinking
CmdPromptInput, 89
isCommandActive
CmdPrompt, 79
MainWindow, 150
isLwtEnabled
View, 287
isRapidFireEnabled
CmdPrompt, 79
isRealEnabled
View, 287
isShiftPressed
MainWindow, 150
Janome, 490, 503
jef, 17, 490
jef_thread
embroidery.h, 386
jefDecode
format_jef.c, 490
jefEncode
format_jef.c, 490
jefGetHoopSize
format_jef.c, 490
jefSetHoopFromId
format_jef.c, 490
jefThreads
embroidery.h, 405
thread-color.c, 546
join_short_stitches
fill.c, 467
JUMP
embroidery.h, 386
ksm, 17, 491
ksmEncode
format_ksm.c, 491
L_system
embroidery.h, 390
labelTipOfDay
MainWindow, 168
language
Settings_, 230
lastCmd
CmdPromptInput, 89
lastCommand
CmdPrompt, 79
layer
EmbPattern_, 115
LayerManager, 129
~LayerManager, 130
addLayer, 130
LayerManager, 129
layerModel, 130
layerModelSorted, 130
treeView, 130
layerManager
MainWindow, 150
layerModel
LayerManager, 130

layerModelSorted
 LayerManager, 130
 layerPrevious
 MainWindow, 150
 layerSelector
 MainWindow, 168
 layerSelectorIndexChanged
 MainWindow, 150
 layoutState
 MainWindow, 168
 left
 _vp3Hoop, 53
 EmbRect_, 116
 hoop_padding, 123
 left2
 _vp3Hoop, 53
 leftBrush
 SelectBox, 224
 leftBrushColor
 SelectBox, 224
 leftPen
 SelectBox, 224
 leftPenColor
 SelectBox, 224
 leftSiblingId
 _bcf_directory_entry, 49
 length
 EmbArray_, 104
 EmbSatinOutline_, 117
 ThredHeader_, 268
 lengths
 Huffman, 123
 LIBEMBROIDERY_EMBEDDED_VERSION
 embroidery.h, 386
 lindenmayer_system
 embroidery.h, 404
 fill.c, 467
 line
 BaseObject, 68
 EmbGeometry_, 110
 line.c
 embLine_intersectionPoint, 523
 embLine_normalVector, 523
 embLine_toVector, 523
 lineEditArcArea
 property-editor.cpp, 359
 lineEditArcCenterY
 property-editor.cpp, 359
 lineEditArcChord
 property-editor.cpp, 359
 lineEditArcEndAngle
 property-editor.cpp, 359
 lineEditArcEndX
 property-editor.cpp, 359
 lineEditArcEndY
 property-editor.cpp, 359
 lineEditArcInAngle
 property-editor.cpp, 359
 lineEditArcLength
 property-editor.cpp, 359
 lineEditArcRadius
 property-editor.cpp, 359
 lineEditArcStartAngle
 property-editor.cpp, 359
 lineEditArcStartX
 property-editor.cpp, 359
 lineEditArcStartY
 property-editor.cpp, 359
 lineEditBlockX
 property-editor.cpp, 359
 lineEditBlockY
 property-editor.cpp, 359
 lineEditCircleArea
 property-editor.cpp, 359
 lineEditCircleCenterX
 property-editor.cpp, 359
 lineEditCircleCenterY
 property-editor.cpp, 359
 lineEditCircleCircumference
 property-editor.cpp, 360
 lineEditCircleDiameter
 property-editor.cpp, 360
 lineEditCircleRadius
 property-editor.cpp, 360
 lineEditEllipseCenterX
 property-editor.cpp, 360
 lineEditEllipseCenterY
 property-editor.cpp, 360
 lineEditEllipseDiameterMajor
 property-editor.cpp, 360
 lineEditEllipseDiameterMinor
 property-editor.cpp, 360
 lineEditEllipseRadiusMajor
 property-editor.cpp, 360
 lineEditEllipseRadiusMinor
 property-editor.cpp, 360
 lineEditImageHeight
 property-editor.cpp, 360
 lineEditImageName
 property-editor.cpp, 360
 lineEditImagePath
 property-editor.cpp, 360
 lineEditImageWidth
 property-editor.cpp, 360
 lineEditImageX
 property-editor.cpp, 360
 lineEditImageY
 property-editor.cpp, 360
 lineEditInfiniteLineVectorX
 property-editor.cpp, 360
 lineEditInfiniteLineVectorY
 property-editor.cpp, 360
 lineEditInfiniteLineX2
 property-editor.cpp, 360
 lineEditInfiniteLineY1
 property-editor.cpp, 361

lineEditInfiniteLineY2
 property-editor.cpp, 361
lineEditLineAngle
 property-editor.cpp, 361
lineEditLineDeltaX
 property-editor.cpp, 361
lineEditLineDeltaY
 property-editor.cpp, 361
lineEditLineEndX
 property-editor.cpp, 361
lineEditLineEndY
 property-editor.cpp, 361
lineEditLineLength
 property-editor.cpp, 361
lineEditLineStartX
 property-editor.cpp, 361
lineEditLineStartY
 property-editor.cpp, 361
lineEditPathArea
 property-editor.cpp, 361
lineEditPathLength
 property-editor.cpp, 361
lineEditPathVertexX
 property-editor.cpp, 361
lineEditPathVertexY
 property-editor.cpp, 361
lineEditPointX
 property-editor.cpp, 361
lineEditPointY
 property-editor.cpp, 361
lineEditPolygonCenterX
 property-editor.cpp, 361
lineEditPolygonCenterY
 property-editor.cpp, 361
lineEditPolygonDiameterSide
 property-editor.cpp, 362
lineEditPolygonDiameterVertex
 property-editor.cpp, 362
lineEditPolygonInteriorAngle
 property-editor.cpp, 362
lineEditPolygonRadiusSide
 property-editor.cpp, 362
lineEditPolygonRadiusVertex
 property-editor.cpp, 362
lineEditPolylineArea
 property-editor.cpp, 362
lineEditPolylineLength
 property-editor.cpp, 362
lineEditPolylineVertexX
 property-editor.cpp, 362
lineEditPolylineVertexY
 property-editor.cpp, 362
lineEditRayVectorX
 property-editor.cpp, 362
lineEditRayVectorY
 property-editor.cpp, 362
lineEditRayX1
 property-editor.cpp, 362
lineEditRayX2
 property-editor.cpp, 362
lineEditRayY1
 property-editor.cpp, 362
lineEditRayY2
 property-editor.cpp, 362
lineEditRectangleArea
 property-editor.cpp, 362
lineEditRectangleCorner1X
 property-editor.cpp, 362
lineEditRectangleCorner1Y
 property-editor.cpp, 362
lineEditRectangleCorner2X
 property-editor.cpp, 363
lineEditRectangleCorner2Y
 property-editor.cpp, 363
lineEditRectangleCorner3X
 property-editor.cpp, 363
lineEditRectangleCorner3Y
 property-editor.cpp, 363
lineEditRectangleCorner4X
 property-editor.cpp, 363
lineEditRectangleCorner4Y
 property-editor.cpp, 363
lineEditRectangleHeight
 property-editor.cpp, 363
lineEditRectangleWidth
 property-editor.cpp, 363
lineEdits
 property-editor.cpp, 363
lineEditTextMultiX
 property-editor.cpp, 363
lineEditTextMultiY
 property-editor.cpp, 363
lineEditTextSingleContents
 property-editor.cpp, 363
lineEditTextSingleHeight
 property-editor.cpp, 363
lineEditTextSingleRotation
 property-editor.cpp, 363
lineEditTextSingleX
 property-editor.cpp, 363
lineEditTextSingleY
 property-editor.cpp, 363
LineObject, 130
 ~LineObject, 133
 allGripPoints, 133
 gripEdit, 133
 init, 133
 LineObject, 133
 mouseSnapPoint, 133
 objectAngle, 133
 objectDeltaX, 133
 objectDeltaY, 134
 objectEndPoint1, 134
 objectEndPoint2, 134
 objectLength, 134
 objectMidPoint, 134

objectSavePath, 134
 objectX1, 134
 objectX2, 134
 objectY1, 134
 objectY2, 134
 paint, 134
 setObjectEndPoint1, 134
 setObjectEndPoint2, 134
 setObjectX1, 135
 setObjectX2, 135
 setObjectY1, 135
 setObjectY2, 135
 Type, 133
 type, 135
 updateRubber, 135
 vulcanize, 135
 lineStyle
 DimLeaderObject, 95
 lineStyleAngle
 DimLeaderObject, 98
 lineStyleLength
 DimLeaderObject, 98
 lineStylePath
 DimLeaderObject, 98
 LINETO
 embroidery_internal.h, 427
 lineType
 EmbGeometry_, 110
 EmbLine_, 113
 EmbPath_, 114
 EmbPoint_, 115
 linetypeSelector
 MainWindow, 168
 linetypeSelectorIndexChanged
 MainWindow, 150
 lineWeightPen
 BaseObject, 68
 linewidthSelector
 MainWindow, 168
 lineweightSelectorIndexChanged
 MainWindow, 150
 listMdiWin
 MainWindow, 168
 listTipOfDay
 MainWindow, 168
 load
 ImageWidget, 128
 loadFatFromSector
 embroidery_internal.h, 438
 main.c, 538
 loadFile
 MdiWindow, 183
 loadFormats
 MainWindow, 150
 loadRulerSettings
 View, 287
 logPromptInput
 MainWindow, 150
 MdiWindow, 183
 LSYSTEM, 135
 alphabet, 135
 axiom, 135
 constants, 135
 rules, 136
 lwt_default_lwt
 Settings_, 230
 lwt_mode
 EmbView_, 121
 lwt_real_render
 Settings_, 230
 lwt_show_lwt
 Settings_, 230
 lwtPen
 BaseObject, 71
 Madeira_Polyester
 embroidery.h, 386
 Madeira_Rayon
 embroidery.h, 386
 magicCode
 VipHeader_, 295
 main
 embroidermodder.cpp, 297
 main.c
 bcf_difat_create, 534
 bcf_directory_free, 534
 bcf_file_free, 534
 bcfFile_read, 534
 bcfFileFat_create, 534
 bcfFileHeader_read, 534
 binaryReadString, 534
 binaryReadUnicodeString, 535
 black_thread, 540
 check_header_present, 535
 CompoundFileDirectory, 535
 CompoundFileDirectoryEntry, 535
 copy_trim, 535
 difatEntriesInHeader, 540
 emb_error, 540
 emb_optOut, 535
 emb_readline, 536
 emb_verbose, 540
 embArc_print, 536
 embColor_distance, 536
 embColor_read, 536
 embColor_write, 536
 embConstantPi, 540
 embSatinOutline_generateSatinOutline, 536
 embSatinOutline_renderStitches, 536
 embThread_findNearestColor, 537
 embThread_findNearestThread, 537
 embThread_getRandom, 537
 embTime_initNow, 537
 embTime_time, 537
 embVector_print, 537
 entriesInDifatSector, 538
 FLAG_CIRCLE, 531

FLAG_CIRCLE_SHORT, 532
FLAG_COMBINE, 532
FLAG_CROSS_STITCH, 532
FLAG_ELLIPSE, 532
FLAG_ELLIPSE_SHORT, 532
FLAG_FILL, 532
FLAG_FILL_SHORT, 532
FLAG_FORMATS, 532
FLAG_FORMATS_SHORT, 532
FLAG_FULL_TEST_SUITE, 532
FLAG_HELP, 532
FLAG_HELP_SHORT, 532
FLAG_HILBERT_CURVE, 532
FLAG_LINE, 532
FLAG_LINE_SHORT, 532
FLAG_POLYGON, 532
FLAG_POLYGON_SHORT, 532
FLAG_POLYLINE, 532
FLAG_POLYLINE_SHORT, 533
FLAG QUIET, 533
FLAG QUIET_SHORT, 533
FLAG_RENDER, 533
FLAG_RENDER_SHORT, 533
FLAG_SATIN, 533
FLAG_SATIN_SHORT, 533
FLAG_SIERPINSKI_TRIANGLE, 533
FLAG_SIMULATE, 533
FLAG_STITCH, 533
FLAG_STITCH_SHORT, 533
FLAG_TEST, 533
FLAG_TO, 533
FLAG_TO_SHORT, 533
FLAG_VERBOSE, 533
FLAG_VERBOSE_SHORT, 533
FLAG_VERSION, 533
FLAG_VERSION_SHORT, 533
get_trim_bounds, 538
GetFile, 538
haveExtraDIFATSectors, 538
loadFatFromSector, 538
NUM_FLAGS, 534
parseDIFATSectors, 538
parseDirectoryEntryName, 538
parseTime, 538
readFullSector, 539
readNextSector, 539
sectorSize, 539
seekToSector, 539
sizeOfChainingEntryAtEndOfDifatSector, 540
sizeOfDifatEntry, 540
sizeOfDirectoryEntry, 540
sizeOfFatEntry, 540
stringInArray, 539
WHITE SPACE, 540
write_24bit, 539
mainWidget
 EmbDetailsDialog, 107
mainWin

embroidermodder.h, 307
MainWindow, 168
mainwindow.cpp, 348
MdiArea, 179
MdiWindow, 186
Settings_Dialog, 254
StatusBarButton, 259
View, 292
MainWindow, 136
 ~MainWindow, 145
 about, 145
 actionHash, 167
 activeCommand, 145
 activeMdiWindow, 145
 activeScene, 145
 activeUndoStack, 145
 activeView, 145
 actuator, 145
 buttonTipOfTheDayClicked, 146
 changelog, 146
 checkBoxTipOfTheDay, 167
 checkBoxTipOfTheDayStateChanged, 146
 checkForUpdates, 146
 closeEvent, 146
 closeToolBar, 146
 colorSelector, 167
 colorSelectorIndexChanged, 146
 copy, 146
 create_icon, 146
 create_toolbar, 146
 createAllActions, 146
 createAllMenus, 147
 createAllToolbars, 147
 createEditMenu, 147
 createFileMenu, 147
 createHelpMenu, 147
 createHelpToolbar, 147
 createIconToolbar, 147
 createLayerToolbar, 147
 createPanToolbar, 147
 createPromptToolbar, 147
 createPropertiesToolbar, 147
 createSettingsMenu, 147
 createTextToolbar, 147
 createViewMenu, 147
 createWindowMenu, 148
 cut, 148
 cutCopyObjectList, 167
 dayVision, 148
 deletePressed, 148
 designDetails, 148
 disableMoveRapidFire, 148
 disablePromptRapidFire, 148
 docIndex, 167
 dockPropEdit, 167
 dockUndoEdit, 167
 doNothing, 148
 editMenu, 168

enableMoveRapidFire, 148
enablePromptRapidFire, 148
escapePressed, 148
exit, 148
fileMenu, 168
findMdiWindow, 148
floatingChangedToolBar, 148
formatFilterOpen, 168
formatFilterSave, 168
getAction, 149
getApplication, 149
getCurrentColor, 149
getCurrentLayer, 149
getCurrentLineType, 149
getCurrentLineWeight, 149
getFileSeparator, 149
getMdiArea, 149
help, 149
helpMenu, 168
hideUnimplemented, 150
iconResize, 150
isCommandActive, 150
isShiftPressed, 150
labelTipOfDay, 168
layerManager, 150
layerPrevious, 150
layerSelector, 168
layerSelectorIndexChanged, 150
layoutState, 168
linetypeSelector, 168
linetypeSelectorIndexChanged, 150
lineweightSelector, 168
lineweightSelectorIndexChanged, 150
listMdiWin, 168
listTipOfDay, 168
loadFormats, 150
logPromptInput, 150
mainWin, 168
MainWindow, 145
makeLayerActive, 150
mdiArea, 168
menuHash, 168
myFileSeparator, 168
nativeAddArc, 150
nativeAddCircle, 150
nativeAddDimLeader, 151
nativeAddEllipse, 151
nativeAddHorizontalDimension, 151
nativeAddImage, 151
nativeAddInfiniteLine, 151
nativeAddLine, 151
nativeAddPath, 152
nativeAddPoint, 152
nativeAddPolygon, 152
nativeAddPolyline, 152
nativeAddRay, 152
nativeAddRectangle, 152
nativeAddRegularPolygon, 152
nativeAddRoundedRectangle, 152
nativeAddSlot, 153
nativeAddTextMulti, 153
nativeAddTextSingle, 153
nativeAddToSelection, 153
nativeAddTriangle, 153
nativeAddVerticalDimension, 153
nativeAlert, 154
nativeAllowRubber, 154
nativeAppendPromptHistory, 154
nativeBlinkPrompt, 154
nativeCalculateAngle, 154
nativeCalculateDistance, 154
nativeClearRubber, 154
nativeClearSelection, 154
nativeCopySelected, 154
nativeCutSelected, 154
nativeDeleteSelected, 155
nativeDisableMoveRapidFire, 155
nativeDisablePromptRapidFire, 155
nativeEnableMoveRapidFire, 155
nativeEnablePromptRapidFire, 155
nativeEndCommand, 155
nativeExit, 155
nativeInitCommand, 155
nativeMessageBox, 155
nativeMirrorSelected, 155
nativeMouseX, 156
nativeMouseY, 156
nativeMoveSelected, 156
nativeNewFile, 156
nativeNumSelected, 156
nativeOpenFile, 156
nativePasteSelected, 156
nativePerpendicularDistance, 156
nativePreviewOff, 157
nativePreviewOn, 157
nativePrintArea, 157
nativeQSnapX, 157
nativeQSnapY, 157
nativeRotateSelected, 157
nativeScaleSelected, 158
nativeSelectAll, 158
nativeSetBackgroundColor, 158
nativeSetCrossHairColor, 158
nativeSetCursorShape, 158
nativeSetGridColor, 158
nativeSetPromptPrefix, 158
nativeSetRubberMode, 158
nativeSetRubberPoint, 158
nativeSetRubberText, 159
nativeSpareRubber, 159
nativeTextAngle, 159
nativeTextBold, 159
nativeTextFont, 159
nativeTextItalic, 159
nativeTextOverline, 159
nativeTextSize, 159

nativeTextStrikeOut, 159
nativeTextUnderline, 159
nativeTipOfTheDay, 159
nativeVulcanize, 159
nativeWindowCascade, 159
nativeWindowClose, 159
nativeWindowCloseAll, 159
nativeWindowNext, 159
nativeWindowPrevious, 160
nativeWindowTile, 160
newFile, 160
nightVision, 160
numOfDocs, 168
onCloseMdiWin, 160
onCloseWindow, 160
onWindowActivated, 160
openFile, 160
openFilesPath, 168
openFilesSelected, 160
openrecentfile, 161
panDown, 161
panLeft, 161
panMenu, 169
panpoint, 161
panrealtime, 161
panRight, 161
panUp, 161
paste, 161
pickAddModeToggled, 161
platformString, 161
print, 161
prompt, 169
promptHistoryAppended, 161
promptInputNext, 161
promptInputPrevious, 161
quit, 161
readSettings, 161
recentMenu, 169
recentMenuAboutToShow, 162
redo, 162
resizeEvent, 162
run_script, 162
run_script_file, 162
runCommand, 162
runCommandClick, 162
runCommandContext, 163
runCommandMain, 163
runCommandMove, 163
runCommandPrompt, 163
saveasfile, 163
savefile, 163
selectAll, 163
setShiftPressed, 163
setShiftReleased, 163
setTextAngle, 163
setTextBold, 163
setTextFont, 163
setTextItalic, 163
setTextOverline, 164
setTextSize, 164
setTextStrikeOut, 164
setTextUnderline, 164
settings_display_bg_color, 169
settings_display_crosshair_color, 169
settings_display_crosshair_percent, 169
settings_display_renderhint_aa, 169
settings_display_renderhint_high_aa, 169
settings_display_renderhint_noncosmetic, 169
settings_display_renderhint_smooth_pix, 169
settings_display_renderhint_text_aa, 169
settings_display_scrollbar_widget_num, 169
settings_display_selectbox_alpha, 169
settings_display_selectbox_left_color, 169
settings_display_selectbox_left_fill, 169
settings_display_selectbox_right_color, 169
settings_display_selectbox_right_fill, 169
settings_display_show_scrollbars, 169
settings_display_units, 170
settings_display_use_opengl, 170
settings_display_zoomscale_in, 170
settings_display_zoomscale_out, 170
settings_general_check_for_updates, 170
settings_general_current_tip, 170
settings_general_icon_size, 170
settings_general_icon_theme, 170
settings_general_language, 170
settings_general_mdi_bg_color, 170
settings_general_mdi_bg_logo, 170
settings_general_mdi_bg_texture, 170
settings_general_mdi_bg_use_color, 170
settings_general_mdi_bg_use_logo, 170
settings_general_mdi_bg_use_texture, 170
settings_general_system_help_browser, 170
settings_general_tip_of_the_day, 170
settings_grid_center_on_origin, 170
settings_grid_center_x, 171
settings_grid_center_y, 171
settings_grid_color, 171
settings_grid_color_match_crosshair, 171
settings_grid_load_from_file, 171
settings_grid_show_on_load, 171
settings_grid_show_origin, 171
settings_grid_size_radius, 171
settings_grid_size_x, 171
settings_grid_size_y, 171
settings_grid_spacing_angle, 171
settings_grid_spacing_radius, 171
settings_grid_spacing_x, 171
settings_grid_spacing_y, 171
settings_grid_type, 171
settings_lwt_default_lwt, 171
settings_lwt_real_render, 171
settings_lwt_show_lwt, 171
settings_opensave_custom_filter, 172
settings_opensave_open_format, 172
settings_opensave_open_thumbnail, 172

settings_opensave_recent_directory, 172
settings_opensave_recent_list_of_files, 172
settings_opensave_recent_max_files, 172
settings_opensave_save_format, 172
settings_opensave_save_thumbnail, 172
settings_opensave_trim_dst_num_jumps, 172
settings_printing_default_device, 172
settings_printing_disable_bg, 172
settings_printing_use_last_device, 172
settings_prompt_bg_color, 172
settings_prompt_font_family, 172
settings_prompt_font_size, 172
settings_prompt_font_style, 172
settings_prompt_save_history, 172
settings_prompt_save_history_as_html, 172
settings_prompt_save_history_filename, 173
settings_prompt_text_color, 173
settings_qsnap_aperture_size, 173
settings_qsnap_apparent, 173
settings_qsnap_center, 173
settings_qsnap_enabled, 173
settings_qsnap_endpoint, 173
settings_qsnap_extension, 173
settings_qsnap_insertion, 173
settings_qsnap_intersection, 173
settings_qsnap_locator_color, 173
settings_qsnap_locator_size, 173
settings_qsnap_midpoint, 173
settings_qsnap_nearest, 173
settings_qsnap_node, 173
settings_qsnap_parallel, 173
settings_qsnap_perpendicular, 173
settings_qsnap_quadrant, 173
settings_qsnap_tangent, 174
settings_ruler_color, 174
settings_ruler_metric, 174
settings_ruler_pixel_size, 174
settings_ruler_show_on_load, 174
settings_selection_coolgrip_color, 174
settings_selection_grip_size, 174
settings_selection_hotgrip_color, 174
settings_selection_mode_pickadd, 174
settings_selection_mode_pickdrag, 174
settings_selection_mode_pickfirst, 174
settings_selection_pickbox_size, 174
settings_text_angle, 174
settings_text_font, 174
settings_text_size, 174
settings_text_style_bold, 174
settings_text_style_italic, 174
settings_text_style_overline, 174
settings_text_style_strikeout, 175
settings_text_style_underline, 175
settingsDialog, 164
settingsMenu, 175
settingsPrompt, 164
setUndoCleanIcon, 164
shiftKeyPressedState, 175
statusbar, 175
stub_implement, 164
stub_testing, 164
textAngle, 164
textBold, 164
textFont, 164
textFontSelector, 175
textFontSelectorCurrentFontChanged, 164
textItalic, 165
textOverline, 165
textSize, 165
textSizeSelector, 175
textSizeSelectorIndexChanged, 165
textStrikeOut, 165
textUnderline, 165
tipOfTheDay, 165
toggleGrid, 165
toggleLwt, 165
toggleRuler, 165
toolbarEdit, 175
toolbarFile, 175
toolbarHash, 175
toolbarHelp, 175
toolbarIcon, 175
toolbarLayer, 175
toolbarPan, 175
toolbarPrompt, 175
toolbarProperties, 175
toolbarText, 175
toolbarView, 175
toolbarZoom, 176
undo, 165
updateAllViewBackgroundColors, 165
updateAllViewCrossHairColors, 165
updateAllViewGridColors, 165
updateAllViewRulerColors, 165
updateAllViewScrollBars, 165
updateAllViewSelectBoxColors, 166
updateMenuToolbarStatusbar, 166
updatePickAddMode, 166
validFileFormat, 166
viewMenu, 176
whatsThisContextHelp, 166
windowMenu, 176
windowMenuAboutToShow, 166
windowMenuActivated, 166
wizardTipOfTheDay, 176
writeSettings, 166
zoomAll, 166
zoomCenter, 167
zoomDynamic, 167
zoomExtents, 167
zoomIn, 167
zoomMenu, 176
zoomOut, 167
zoomPrevious, 167
zoomRealtime, 167
zoomScale, 167

zoomSelected, 167
zoomWindow, 167
mainwindow-settings.cpp
 read_configuration, 345
 SettingsDir, 345
 SettingsPath, 345
 to_string_vector, 345
mainwindow-toolbars.cpp
 edit_toolbar, 345
 file_toolbar, 345
 get_action_index, 345
 help_toolbar, 346
 icon_toolbar, 346
 pan_toolbar, 346
 view_toolbar, 346
 zoom_toolbar, 346
mainwindow.cpp
 _mainWin, 349
 action_labels, 349
 action_table, 349
 active_view, 349
 c_split, 348
 CIRCLE_MODE_1P_DIA_, 349
 CIRCLE_MODE_1P_RAD_, 349
 CIRCLE_MODE_2P_, 349
 CIRCLE_MODE_3P_, 349
 CIRCLE_MODE_TTR_, 349
 convert_args_to_type, 348
 dialog, 349
 DOLPHIN_MODE_NUM_POINTS_, 349
 DOLPHIN_MODE_XSCALE_, 350
 DOLPHIN_MODE_YSCALE_, 350
 Error, 348
 mainWin, 348
 menu_layout, 350
 n_views, 350
 Parameter, 348
 preview, 350
 read_settings, 348
 settings, 350
 simplify_path, 349
 SINGLE_LINE_TEXT_MODE_JUSTIFY_, 350
 SINGLE_LINE_TEXT_MODE_RAPID_, 350
 SINGLE_LINE_TEXT_MODE_SETFONT_, 350
 SINGLE_LINE_TEXT_MODE_SETGEOM_, 350
 STAR_MODE_CENTER_PT_, 350
 STAR_MODE_NUM_POINTS_, 350
 STAR_MODE_RAD_INNER_, 350
 STAR_MODE_RAD_OUTER_, 350
 Todo, 349
 toolbar_layout, 350
 translation_table, 350
 validRGB, 349
 views, 350
major_tick_seperation
 Settings_, 230
majorVersion
 _bcf_file_header, 52
makeLayerActive
 MainWindow, 150
manufacturer_code
 thread_color_, 267
mapSignal
 PropertyEditor, 209
Marathon_Polyester
 embroidery.h, 386
Marathon_Rayon
 embroidery.h, 386
max, 492
max_header
 format_max.c, 492
MAX_STITCHES
 embroidery.h, 387
MAX_THREADS
 embroidery.h, 387
maxNumberOfDirectoryEntries
 _bcf_directory, 48
maxPoints
 UiObject_, 270
mdi_bg_use_color
 Settings_, 230
mdi_bg_use_logo
 Settings_, 230
mdi_bg_use_texture
 Settings_, 230
MdiArea, 176
 ~MdiArea, 177
 bgColor, 179
 bgLogo, 179
 bgTexture, 179
 cascade, 177
 forceRepaint, 177
 mainWin, 179
 MdiArea, 177
 mouseDoubleClickEvent, 177
 paintEvent, 177
 setBackgroundColor, 178
 setBackgroundLogo, 178
 setBackgroundTexture, 178
 tile, 178
 useBackgroundColor, 178
 useBackgroundLogo, 178
 useBackgroundTexture, 179
 useColor, 179
 useLogo, 179
 useTexture, 179
 zoomExtentsAllSubWindows, 179
mdiArea
 MainWindow, 168
 MdiWindow, 186
MdiWindow, 179
 ~MdiWindow, 181
 closeEvent, 181
 curColor, 186
 curFile, 186
 curLayer, 186

curLineType, 186
 curLineWeight, 186
 currentColorChanged, 181
 currentLayerChanged, 182
 currentLinetypeChanged, 182
 currentLineweightChanged, 182
 deletePressed, 182
 designDetails, 182
 escapePressed, 182
 fileExtension, 182
 fileWasLoaded, 186
 getCurrentColor, 182
 getCurrentFile, 183
 getCurrentLayer, 183
 getCurrentLineType, 183
 getCurrentLineWeight, 183
 getScene, 183
 getShortCurrentFile, 183
 getView, 183
 gscene, 186
 gview, 186
 loadFile, 183
 logPromptInput, 183
 mainWin, 186
 mdiArea, 186
 MdiWindow, 181
 myIndex, 186
 onWindowActivated, 183
 print, 183
 printer, 186
 promptHistory, 186
 promptHistoryAppended, 183
 promptInputList, 186
 promptInputNext, 184
 promptInputNum, 186
 promptInputPrevious, 184
 promptInputPrevNext, 184
 saveBMC, 184
 saveFile, 184
 sendCloseMdiWin, 184
 setCurrentColor, 184
 setCurrentFile, 184
 setCurrentLayer, 185
 setCurrentLineType, 185
 setCurrentLineWeight, 185
 setViewBackgroundColor, 185
 setViewCrossHairColor, 185
 setViewGridColor, 185
 setViewRulerColor, 185
 setViewSelectBoxColors, 185
 showViewScrollBars, 185
 sizeHint, 185
 updateColorLinetypeLineweight, 185
 Mega 2560 or another board with equal or, 25
 Melco, 475, 479, 484, 494
 menu_action
 Settings_, 230
 menu_layout

mainwindow.cpp, 350
 menuHash
 MainWindow, 168
 mergeWith
 UndoableNavCommand, 276
 metric
 EmbView_, 121
 Metro_Polyester
 embroidery.h, 387
 mid
 EmbArc_, 104
 miniSectorShift
 _bcf_file_header, 52
 miniStreamCutoffSize
 _bcf_file_header, 52
 minorVersion
 _bcf_file_header, 52
 minPoints
 UiObject_, 270
 minute
 EmbTime_, 119
 mirror
 UndoableMirrorCommand, 274
 mirrorLine
 UndoableMirrorCommand, 274
 mirrorSelected
 View, 287
 mit, 442, 493
 mitDecodeStitch
 embroidery_internal.h, 438
 encoding.c, 462
 mitEncodeStitch
 embroidery_internal.h, 438
 encoding.c, 462
 Mitsubishi, 442, 493
 mode
 UiObject_, 270
 modifiedTime
 _bcf_directory_entry, 49
 modifierName
 ThredExtension_, 268
 month
 EmbTime_, 119
 mouseDoubleClickEvent
 MdiArea, 177
 View, 287
 mouseMoveEvent
 CmdPromptHandle, 83
 View, 287
 mousePressEvent
 CmdPromptHandle, 83
 View, 287
 mouseReleaseEvent
 CmdPromptHandle, 83
 View, 287
 mouseSnapPoint
 ArcObject, 62
 BaseObject, 68

CircleObject, 74
DimLeaderObject, 96
EllipseObject, 101
ImageObject, 126
LineObject, 133
PathObject, 190
PointObject, 194
PolygonObject, 197
PolylineObject, 202
RectObject, 216
TextSingleObject, 264
moveAction
 View, 287
movePoint
 View, 292
moveResizeHistory
 CmdPromptSplitter, 90
moveSelected
 View, 287
MOVETO
 embroidery_internal.h, 427
moveY
 CmdPromptHandle, 83
movingActive
 View, 292
myFileSeparator
 MainWindow, 168
myIndex
 MdiWindow, 186
n_attributes
 format_svg.c, 506
n_controlPoints
 UiObject_, 270
N_PES_VERSIONS
 embroidery_internal.h, 427
n_selected
 EmbView_, 121
n_views
 mainwindow.cpp, 350
name
 EmblImage_, 111
 EmblLayer_, 112
 SvgAttribute_, 261
 thread_color_, 267
nativeAddArc
 MainWindow, 150
nativeAddCircle
 MainWindow, 150
nativeAddDimLeader
 MainWindow, 151
nativeAddEllipse
 MainWindow, 151
nativeAddHorizontalDimension
 MainWindow, 151
nativeAddImage
 MainWindow, 151
nativeAddInfiniteLine
 MainWindow, 151
nativeAddLine
 MainWindow, 151
nativeAddPath
 MainWindow, 152
nativeAddPoint
 MainWindow, 152
nativeAddPolygon
 MainWindow, 152
nativeAddPolyline
 MainWindow, 152
nativeAddRay
 MainWindow, 152
nativeAddRectangle
 MainWindow, 152
nativeAddRegularPolygon
 MainWindow, 152
nativeAddRoundedRectangle
 MainWindow, 152
nativeAddSlot
 MainWindow, 153
nativeAddTextMulti
 MainWindow, 153
nativeAddTextSingle
 MainWindow, 153
nativeAddToSelection
 MainWindow, 153
nativeAddTriangle
 MainWindow, 153
nativeAddVerticalDimension
 MainWindow, 153
nativeAlert
 MainWindow, 154
nativeAllowRubber
 MainWindow, 154
nativeAppendPromptHistory
 MainWindow, 154
nativeBlinkPrompt
 MainWindow, 154
nativeCalculateAngle
 MainWindow, 154
nativeCalculateDistance
 MainWindow, 154
nativeClearRubber
 MainWindow, 154
nativeClearSelection
 MainWindow, 154
nativeCopySelected
 MainWindow, 154
nativeCutSelected
 MainWindow, 154
nativeDeleteSelected
 MainWindow, 155
nativeDisableMoveRapidFire
 MainWindow, 155
nativeDisablePromptRapidFire
 MainWindow, 155
nativeEnableMoveRapidFire
 MainWindow, 155

nativeEnablePromptRapidFire
 MainWindow, 155
nativeEndCommand
 MainWindow, 155
nativeExit
 MainWindow, 155
nativeInitCommand
 MainWindow, 155
nativeMessageBox
 MainWindow, 155
nativeMirrorSelected
 MainWindow, 155
nativeMouseX
 MainWindow, 156
nativeMouseY
 MainWindow, 156
nativeMoveSelected
 MainWindow, 156
nativeNewFile
 MainWindow, 156
nativeNumSelected
 MainWindow, 156
nativeOpenFile
 MainWindow, 156
nativePasteSelected
 MainWindow, 156
nativePerpendicularDistance
 MainWindow, 156
nativePreviewOff
 MainWindow, 157
nativePreviewOn
 MainWindow, 157
nativePrintArea
 MainWindow, 157
nativeQSnapX
 MainWindow, 157
nativeQSnapY
 MainWindow, 157
nativeRotateSelected
 MainWindow, 157
nativeScaleSelected
 MainWindow, 158
nativeSelectAll
 MainWindow, 158
nativeSetBackgroundColor
 MainWindow, 158
nativeSetCrossHairColor
 MainWindow, 158
nativeSetCursorShape
 MainWindow, 158
nativeSetGridColor
 MainWindow, 158
nativeSetPromptPrefix
 MainWindow, 158
nativeSetRubberMode
 MainWindow, 158
nativeSetRubberPoint
 MainWindow, 158

nativeSetText
 MainWindow, 159
nativeSpareRubber
 MainWindow, 159
nativeTextAngle
 MainWindow, 159
nativeTextBold
 MainWindow, 159
nativeTextFont
 MainWindow, 159
nativeTextItalic
 MainWindow, 159
nativeTextOverline
 MainWindow, 159
nativeTextSize
 MainWindow, 159
nativeTextStrikeOut
 MainWindow, 159
nativeTextUnderline
 MainWindow, 159
nativeTipOfDay
 MainWindow, 159
nativeVulcanize
 MainWindow, 159
nativeWindowCascade
 MainWindow, 159
nativeWindowClose
 MainWindow, 159
nativeWindowCloseAll
 MainWindow, 159
nativeWindowNext
 MainWindow, 159
nativeWindowPrevious
 MainWindow, 160
nativeWindowTile
 MainWindow, 160
navType
 UndoableNavCommand, 277
needle_speed
 Settings_, 230
negativeXHoopSize
 VipHeader_, 295
negativeYHoopSize
 VipHeader_, 295
new, 442, 493
newFile
 MainWindow, 160
next
 _bcf_directory_entry, 49
nightVision
 MainWindow, 160
nlenghts
 Huffman, 123
NoArrow
 DimLeaderObject, 95
NoLine
 DimLeaderObject, 95
NORMAL

embroidery.h, 387
normalPath
 PathObject, 191
 PolygonObject, 199
 PolylineObject, 203
ntable
 Huffman, 123
NUM_FLAGS
 main.c, 534
numberOfBytesRemaining
 _vp3Hoop, 53
numberOfColors
 _vp3Hoop, 53
 VipHeader_, 295
numberOfDifatSectors
 _bcf_file_header, 52
numberOfDirectorySectors
 _bcf_file_header, 52
numberOfEntriesInDifatSector
 embroidery_internal.h, 438
numberOfEntriesInFatSector
 _bcf_file_fat, 51
numberOfFATSectors
 _bcf_file_header, 52
numberOfFormats
 embroidery.h, 387
numberOfMiniFatSectors
 _bcf_file_header, 52
numberOfStitches
 VipHeader_, 295
numOfDocs
 MainWindow, 168
numPoints
 UiObject_, 270
numSelected
 View, 288
numStiches
 ThredHeader_, 268

OBJ_COLOR
 embroidermodder.h, 302
OBJ_KEYS
 embroidermodder.h, 302
OBJ_LAYER
 embroidermodder.h, 302
OBJ_LTYPE
 embroidermodder.h, 302
OBJ_LTYPE_CENTER
 embroidermodder.h, 302
OBJ_LTYPE_CONT
 embroidermodder.h, 302
OBJ_LTYPE_DOT
 embroidermodder.h, 302
OBJ_LTYPE_FISHBONE
 embroidermodder.h, 303
OBJ_LTYPE_HIDDEN
 embroidermodder.h, 302
OBJ_LTYPE_PHANTOM
 embroidermodder.h, 302
OBJ_LTYPE_RUNNING
 embroidermodder.h, 303
OBJ_LTYPE_SATIN
 embroidermodder.h, 303
OBJ_LTYPE_VALUES
 embroidermodder.h, 302
OBJ_LTYPE_ZIGZAG
 embroidermodder.h, 303
OBJ_LWT
 embroidermodder.h, 302
OBJ_LWT_01
 embroidermodder.h, 303
OBJ_LWT_02
 embroidermodder.h, 303
OBJ_LWT_03
 embroidermodder.h, 303
OBJ_LWT_04
 embroidermodder.h, 303
OBJ_LWT_05
 embroidermodder.h, 303
OBJ_LWT_06
 embroidermodder.h, 303
OBJ_LWT_07
 embroidermodder.h, 303
OBJ_LWT_08
 embroidermodder.h, 303
OBJ_LWT_09
 embroidermodder.h, 303
OBJ_LWT_10
 embroidermodder.h, 303
OBJ_LWT_11
 embroidermodder.h, 303
OBJ_LWT_12
 embroidermodder.h, 303
OBJ_LWT_13
 embroidermodder.h, 303
OBJ_LWT_14
 embroidermodder.h, 303
OBJ_LWT_15
 embroidermodder.h, 303
OBJ_LWT_16
 embroidermodder.h, 303
OBJ_LWT_17
 embroidermodder.h, 303
OBJ_LWT_18
 embroidermodder.h, 303
OBJ_LWT_19
 embroidermodder.h, 303
OBJ_LWT_20
 embroidermodder.h, 303
OBJ_LWT_21
 embroidermodder.h, 303
OBJ_LWT_22
 embroidermodder.h, 303
OBJ_LWT_23
 embroidermodder.h, 303
OBJ_LWT_24
 embroidermodder.h, 303

OBJ_LWT_BYBLOCK
embroidermodder.h, 303
OBJ_LWT_BYLAYER
embroidermodder.h, 303
OBJ_LWT_DEFAULT
embroidermodder.h, 303
OBJ_LWT_VALUES
embroidermodder.h, 303
OBJ_NAME
embroidermodder.h, 302
OBJ_RUBBER
embroidermodder.h, 302
OBJ_RUBBER_CIRCLE_1P_DIA
embroidermodder.h, 304
OBJ_RUBBER_CIRCLE_1P_RAD
embroidermodder.h, 303
OBJ_RUBBER_CIRCLE_2P
embroidermodder.h, 304
OBJ_RUBBER_CIRCLE_3P
embroidermodder.h, 304
OBJ_RUBBER_CIRCLE_TTR
embroidermodder.h, 304
OBJ_RUBBER_CIRCLE_TTT
embroidermodder.h, 304
OBJ_RUBBER_DIMLEADER_LINE
embroidermodder.h, 304
OBJ_RUBBER_ELLIPSE_LINE
embroidermodder.h, 304
OBJ_RUBBER_ELLIPSE_MAJOR_DIAMETER_MINOR_RADIUS_SNAP_VALUES
embroidermodder.h, 304
OBJ_RUBBER_ELLIPSE_MAJOR_RADIUS_MINOR_RADIUS_SNAP_TYPE
embroidermodder.h, 304
OBJ_RUBBER_ELLIPSE_ROTATION
embroidermodder.h, 304
OBJ_RUBBER_GRIP
embroidermodder.h, 304
OBJ_RUBBER_IMAGE
embroidermodder.h, 304
OBJ_RUBBER_LINE
embroidermodder.h, 304
OBJ_RUBBER_OFF
embroidermodder.h, 303
OBJ_RUBBER_ON
embroidermodder.h, 303
OBJ_RUBBER_POLYGON
embroidermodder.h, 304
OBJ_RUBBER_POLYGON_CIRCUMSCRIBE
embroidermodder.h, 304
OBJ_RUBBER_POLYGON_INSCRIBE
embroidermodder.h, 304
OBJ_RUBBER_POLYLINE
embroidermodder.h, 304
OBJ_RUBBER_RECTANGLE
embroidermodder.h, 304
OBJ_RUBBER_TEXTSINGLE
embroidermodder.h, 304
OBJ_RUBBER_VALUES
embroidermodder.h, 303
OBJ_SNAP_APPINTERSECTION
embroidermodder.h, 304
OBJ_SNAP_CENTER
embroidermodder.h, 304
OBJ_SNAP_ENDPOINT
embroidermodder.h, 304
OBJ_SNAP_EXTENSION
embroidermodder.h, 304
OBJ_SNAP_INSERTION
embroidermodder.h, 304
OBJ_SNAP_INTERSECTION
embroidermodder.h, 304
OBJ_SNAP_MIDPOINT
embroidermodder.h, 304
OBJ_SNAP_NEAREST
embroidermodder.h, 304
OBJ_SNAP_NODE
embroidermodder.h, 304
OBJ_SNAP_NULL
embroidermodder.h, 304
OBJ_SNAP_PARALLEL
embroidermodder.h, 304
OBJ_SNAP_PERPENDICULAR
embroidermodder.h, 304
OBJ_SNAP_QUADRANT
embroidermodder.h, 304
OBJ_SNAP_TANGENT
embroidermodder.h, 304
OBJ_TYPE
embroidermodder.h, 302
OBJ_TYPE_ARC
embroidermodder.h, 305
OBJ_TYPE_BASE
embroidermodder.h, 305
OBJ_TYPE_BLOCK
embroidermodder.h, 305
OBJ_TYPE_CIRCLE
embroidermodder.h, 305
OBJ_TYPE_DIMALIGNED
embroidermodder.h, 305
OBJ_TYPE_DIMANGULAR
embroidermodder.h, 305
OBJ_TYPE_DIMARCLENGTH
embroidermodder.h, 305
OBJ_TYPE_DIMDIAMETER
embroidermodder.h, 305
OBJ_TYPE_DIMLEADER
embroidermodder.h, 305
OBJ_TYPE_DIMLINEAR
embroidermodder.h, 305
OBJ_TYPE_DIMORDINATE
embroidermodder.h, 305
OBJ_TYPE_DIMRADIUS
embroidermodder.h, 305
OBJ_TYPE_ELLIPSE
embroidermodder.h, 305

OBJ_TYPE_ELLIPSEARC
embroidermodder.h, 305

OBJ_TYPE_GRID
embroidermodder.h, 305

OBJ_TYPE_HATCH
embroidermodder.h, 305

OBJ_TYPE_IMAGE
embroidermodder.h, 305

OBJ_TYPE_INFINITELINE
embroidermodder.h, 305

OBJ_TYPE_LINE
embroidermodder.h, 305

OBJ_TYPE_NULL
embroidermodder.h, 305

OBJ_TYPE_PATH
embroidermodder.h, 305

OBJ_TYPE_POINT
embroidermodder.h, 305

OBJ_TYPE_POLYGON
embroidermodder.h, 305

OBJ_TYPE_POLYLINE
embroidermodder.h, 305

OBJ_TYPE_RAY
embroidermodder.h, 305

OBJ_TYPE_RECTANGLE
embroidermodder.h, 305

OBJ_TYPE_RUBBER
embroidermodder.h, 305

OBJ_TYPE_SLOT
embroidermodder.h, 305

OBJ_TYPE_SPLINE
embroidermodder.h, 305

OBJ_TYPE_TEXTMULTI
embroidermodder.h, 305

OBJ_TYPE_TEXTSINGLE
embroidermodder.h, 305

OBJ_TYPE_VALUES
embroidermodder.h, 304

object
 EmbGeometry_, 110
 UndoableAddCommand, 272
 UndoableDeleteCommand, 272
 UndoableGripEditCommand, 273
 UndoableMirrorCommand, 274
 UndoableMoveCommand, 275
 UndoableRotateCommand, 278
 UndoableScaleCommand, 279

object-arc.cpp
 rotate_vector, 351

object_index
 UiObject_, 270

objectAngle
 DimLeaderObject, 96
 LineObject, 133

objectArcLength
 ArcObject, 62

objectArea
 ArcObject, 62

CircleObject, 74

ImageObject, 126

RectObject, 216

objectBottomLeft
 ImageObject, 126
 RectObject, 216

objectBottomRight
 ImageObject, 127
 RectObject, 216

objectCenter
 BaseObject, 68

objectCenterX
 BaseObject, 69

objectCenterY
 BaseObject, 69

objectChord
 ArcObject, 62

objectCircumference
 CircleObject, 74

objectClockwise
 ArcObject, 62

objectColor
 BaseObject, 69

objectColorRGB
 BaseObject, 69

objectCopyPath
 PathObject, 190
 PolygonObject, 197
 PolylineObject, 202

objectDeltaX
 DimLeaderObject, 96
 LineObject, 133

objectDeltaY
 DimLeaderObject, 96
 LineObject, 134

objectDiameter
 CircleObject, 74

objectDiameterMajor
 EllipseObject, 101

objectDiameterMinor
 EllipseObject, 101

objectEndAngle
 ArcObject, 62

objectEndPoint
 ArcObject, 63

objectEndPoint1
 DimLeaderObject, 96
 LineObject, 134

objectEndPoint2
 DimLeaderObject, 96
 LineObject, 134

objectEndX
 ArcObject, 63

objectEndY
 ArcObject, 63

objectHeight
 EllipseObject, 101
 ImageObject, 127

RectObject, 216
objectID
 BaseObject, 69
objectIncludedAngle
 ArcObject, 63
objectLength
 DimLeaderObject, 96
 LineObject, 134
objectLineType
 BaseObject, 69
objectLineWidth
 BaseObject, 69
objectMidPoint
 ArcObject, 63
 DimLeaderObject, 96
 LineObject, 134
objectMidX
 ArcObject, 63
objectMidY
 ArcObject, 63
objectPath
 BaseObject, 69
objectPen
 BaseObject, 69
objectPos
 PathObject, 190
 PointObject, 194
 PolygonObject, 198
 PolylineObject, 202
 RectObject, 216
 TextSingleObject, 264
objectQuadrant0
 CircleObject, 74
 EllipseObject, 101
objectQuadrant180
 CircleObject, 74
 EllipseObject, 101
objectQuadrant270
 CircleObject, 74
 EllipseObject, 101
objectQuadrant90
 CircleObject, 74
 EllipseObject, 101
objectRadius
 ArcObject, 64
 CircleObject, 75
objectRadiusMajor
 EllipseObject, 102
objectRadiusMinor
 EllipseObject, 102
objectRubberMode
 BaseObject, 69
objectRubberPoint
 BaseObject, 69
objectRubberText
 BaseObject, 69
objectSavePath
 CircleObject, 75
 EllipseObject, 102
 LineObject, 134
 PathObject, 190
 PointObject, 194
 PolygonObject, 198
 PolylineObject, 202
 RectObject, 216
objectSavePathList
 TextSingleObject, 264
objectStartAngle
 ArcObject, 64
objectStartPoint
 ArcObject, 64
objectStartX
 ArcObject, 64
objectStartY
 ArcObject, 64
objectTextJustifyList
 TextSingleObject, 264
objectTopLeft
 ImageObject, 127
 RectObject, 216
objectTopRight
 ImageObject, 127
 RectObject, 216
objectType
 _bcf_directory_entry, 49
ObjectTypeRootEntry
 embroidery_internal.h, 427
ObjectTypeStorage
 embroidery_internal.h, 427
ObjectTypeStream
 embroidery_internal.h, 427
ObjectTypeUnknown
 embroidery_internal.h, 427
objectWidth
 EllipseObject, 102
 ImageObject, 127
 RectObject, 217
objectX
 PathObject, 190
 PointObject, 194
 PolygonObject, 198
 PolylineObject, 202
 TextSingleObject, 264
objectX1
 DimLeaderObject, 96
 LineObject, 134
objectX2
 DimLeaderObject, 96
 LineObject, 134
objectY
 PathObject, 190
 PointObject, 194
 PolygonObject, 198
 PolylineObject, 202
 TextSingleObject, 264
objectY1

DimLeaderObject, 96
LineObject, 134
objectY2
 DimLeaderObject, 96
 LineObject, 134
objID
 BaseObject, 71
objLine
 BaseObject, 71
objPen
 BaseObject, 71
objRubberMode
 BaseObject, 71
objRubberPoints
 BaseObject, 71
objRubberTexts
 BaseObject, 71
objText
 TextSingleObject, 266
objTextBackward
 TextSingleObject, 266
objTextBold
 TextSingleObject, 266
objTextFont
 TextSingleObject, 266
objTextItalic
 TextSingleObject, 266
objTextJustify
 TextSingleObject, 266
objTextOverline
 TextSingleObject, 266
objTextPath
 TextSingleObject, 266
objTextSize
 TextSingleObject, 266
objTextStrikeOut
 TextSingleObject, 267
objTextUnderline
 TextSingleObject, 267
objTextUpsideDown
 TextSingleObject, 267
ofm, 494
ofmDecode
 format_ofm.c, 494
ofmReadBlockHeader
 format_ofm.c, 494
ofmReadClass
 format_ofm.c, 494
ofmReadColorChange
 format_ofm.c, 494
ofmReadExpanded
 format_ofm.c, 494
ofmReadLibrary
 format_ofm.c, 494
ofmReadThreads
 format_ofm.c, 494
onCloseMdiWin
 MainWindow, 160
onCloseWindow
 MainWindow, 160
onWindowActivated
 MainWindow, 160
 MdiWindow, 183
Open
 DimLeaderObject, 95
openFile
 MainWindow, 160
openFilesPath
 MainWindow, 168
openFilesSelected
 MainWindow, 160
openRecentfile
 MainWindow, 161
opensave_custom_filter
 Settings_, 230
opensave_open_format
 Settings_, 230
opensave_open_thumbnail
 Settings_, 230
opensave_recent_directory
 Settings_, 230
opensave_recent_list_of_files
 Settings_, 230
opensave_recent_max_files
 Settings_, 230
opensave_save_format
 Settings_, 230
opensave_save_thumbnail
 Settings_, 230
opensave_trim_dst_num_jumps
 Settings_, 231
operator+
 embroidermodder.h, 307
operator-
 embroidermodder.h, 307
origin
 EmbView_, 121
originPath
 View, 292
ortho_mode
 EmbView_, 121
paint
 ArcObject, 64
 CircleObject, 75
 DimLeaderObject, 96
 EllipseObject, 102
 ImageObject, 127
 LineObject, 134
 PathObject, 190
 PointObject, 194
 PolygonObject, 198
 PolylineObject, 202
 RectObject, 217
 TextSingleObject, 264
paintEvent
 ImageWidget, 128

MdiArea, 177
 SelectBox, 223
pan_toolbar
 mainwindow-toolbars.cpp, 346
panDistance
 View, 292
panDown
 MainWindow, 161
 View, 288
panLeft
 MainWindow, 161
 View, 288
panMenu
 MainWindow, 169
panningActive
 View, 293
panningPointActive
 View, 293
panningRealTimeActive
 View, 293
panPoint
 View, 288
panpoint
 MainWindow, 161
panRealTime
 View, 288
panrealtime
 MainWindow, 161
panRight
 MainWindow, 161
 View, 288
panStart
 View, 288
panStartX
 View, 293
panStartY
 View, 293
Pantone
 embroidery.h, 387
panUp
 MainWindow, 161
 View, 288
Parameter
 mainwindow.cpp, 348
Parameter_, 187
 i_value, 187
 r_value, 187
 s_value, 187
parseDIFATSectors
 main.c, 538
parseDirectoryEntryName
 main.c, 538
parseTime
 main.c, 538
paste
 MainWindow, 161
 View, 288
pasteClip
 CmdPromptInput, 88
pasteDelta
 View, 293
pasteObjectItemGroup
 View, 293
pastePressed
 CmdPrompt, 79
 CmdPromptInput, 88
pastingActive
 View, 293
path
 EmbGeometry_, 110
 EmbImage_, 111
path_desc
 UiObject_, 270
PathObject, 187
 ~PathObject, 189
 allGripPoints, 189
 gripEdit, 189
 init, 189
 mouseSnapPoint, 190
 normalPath, 191
 objectCopyPath, 190
 objectPos, 190
 objectSavePath, 190
 objectX, 190
 objectY, 190
 paint, 190
 PathObject, 189
 setObjectPos, 190
 setObjectX, 190
 setObjectY, 190
 Type, 189
 type, 190
 updatePath, 190
 updateRubber, 190
 vulcanize, 191
pattern
 EmbView_, 121
pattern.c
 convert, 541
 embPattern_addCircleAbs, 541
 embPattern_addEllipseAbs, 541
 embPattern_addLineAbs, 542
 embPattern_addPathAbs, 542
 embPattern_addPointAbs, 542
 embPattern_addPolygonAbs, 542
 embPattern_addPolylineObjectAbs, 542
 embPattern_addRectAbs, 542
 embPattern_addStitchAbs, 542
 embPattern_addStitchRel, 542
 embPattern_addThread, 542
 embPattern_calcBoundingBox, 543
 embPattern_center, 543
 embPattern_changeColor, 543
 embPattern_color_count, 543
 embPattern_combineJumpStitches, 543
 embPattern_copyPolylinesToStitch_list, 543

embPattern_copystitch_listToPolylines, 543
embPattern_correctForMaxStitchLength, 543
embPattern_create, 543
embPattern_designDetails, 543
embPattern_end, 544
embPattern_fixColorCount, 544
embPattern_flip, 544
embPattern_flipHorizontal, 544
embPattern_flipVertical, 544
embPattern_free, 544
embPattern_hideStitchesOverLength, 544
embPattern_jumpStitches, 544
embPattern_lengthHistogram, 544
embPattern_loadExternalColorFile, 544
embPattern_maximumStitchLength, 544
embPattern_minimumStitchLength, 544
embPattern_movePolylinesToStitch_list, 545
embPattern_movestitch_listToPolylines, 545
embPattern_realStitches, 545
embPattern_scale, 545
embPattern_totalStitchLength, 545
embPattern_trimStitches, 545
pattern_index
 Settings_, 231
 UiObject_, 270
pcd, 17, 443, 495
pcm, 17, 443, 495
pcm_thread
 embroidery.h, 387
pcmThreads
 embroidery.h, 406
 thread-color.c, 546
pcq, 17, 443, 496
pcs, 17, 443, 496
pec, 17, 443, 497
pec_thread
 embroidery.h, 387
pecEncode
 format_pec.c, 497
pecEncodeJump
 format_pec.c, 497
pecEncodeStop
 format_pec.c, 497
pecThreadCount
 embroidery.h, 406
 thread-color.c, 546
pecThreads
 embroidery.h, 406
 thread-color.c, 547
pel, 17, 443, 498
perm, 17, 444, 498
pes, 17, 501
PES0001
 embroidery_internal.h, 427
PES0020
 embroidery_internal.h, 427
PES0022
 embroidery_internal.h, 427
PES0030
 embroidery_internal.h, 427
PES0040
 embroidery_internal.h, 428
PES0050
 embroidery_internal.h, 428
PES0055
 embroidery_internal.h, 428
PES0060
 embroidery_internal.h, 428
PES0070
 embroidery_internal.h, 428
PES0080
 embroidery_internal.h, 428
PES0090
 embroidery_internal.h, 428
PES0100
 embroidery_internal.h, 428
pes_version
 format_pes.c, 501
pes_version_strings
 format_pes.c, 501
pesWriteEmbOneSection
 format_pes.c, 499
pesWriteSewSegSection
 format_pes.c, 499
Pfaff, 406, 443, 445, 491, 492, 495, 496, 507, 512, 513
pfaffDecode
 embroidery_internal.h, 438
 encoding.c, 462
pfaffEncode
 embroidery_internal.h, 438
 encoding.c, 463
phb, 17, 444, 501
phc, 17, 444, 502
pickAdd
 PropertyEditor, 211
pickAddModeToggled
 MainWindow, 161
 PropertyEditor, 209
pickBoxSize
 View, 293
pivotX
 UndoableRotateCommand, 278
pivotY
 UndoableRotateCommand, 278
platformString
 MainWindow, 161
plt, 445, 502
point
 EmbGeometry_, 110
pointList
 EmbPath_, 114
PointObject, 191
 ~PointObject, 193
 allGripPoints, 193

gripEdit, 193
 init, 193
 mouseSnapPoint, 194
 objectPos, 194
 objectSavePath, 194
 objectX, 194
 objectY, 194
 paint, 194
 PointObject, 193
 setObjectPos, 194
 setObjectX, 194
 setObjectY, 194
 Type, 193
 type, 194
 updateRubber, 194
 vulcanize, 194
polar_mode
 EmbView_, 121
polygon
 EmbGeometry_, 110
PolygonObject, 195
 ~PolygonObject, 197
 allGripPoints, 197
 findIndex, 197
 gripEdit, 197
 gripIndex, 199
 init, 197
 mouseSnapPoint, 197
 normalPath, 199
 objectCopyPath, 197
 objectPos, 198
 objectSavePath, 198
 objectX, 198
 objectY, 198
 paint, 198
 PolygonObject, 197
 setObjectPos, 198
 setObjectX, 198
 setObjectY, 198
 Type, 197
 type, 198
 updatePath, 198
 updateRubber, 198
 vulcanize, 198
polyline
 EmbGeometry_, 110
PolylineObject, 199
 ~PolylineObject, 201
 allGripPoints, 201
 findIndex, 201
 gripEdit, 201
 gripIndex, 203
 init, 201
 mouseSnapPoint, 202
 normalPath, 203
 objectCopyPath, 202
 objectPos, 202
 objectSavePath, 202
 objectX, 202
 objectY, 202
 paint, 202
 PolylineObject, 201
 setObjectPos, 202
 setObjectX, 202
 setObjectY, 202
 Type, 201
 type, 202
 updatePath, 202
 updateRubber, 203
 vulcanize, 203
position
 EmbAlignedDim_, 103
 EmbAngularDim_, 103
 EmbArcLengthDim_, 104
 EmbBlock_, 105
 EmbDiameterDim_, 108
 EmblImage_, 111
 EmblInfiniteLine_, 112
 EmbLeaderDim_, 112
 EmbLinearDim_, 113
 EmbOrdinateDim_, 113
 EmbPoint_, 115
 EmbRadiusDim_, 116
 EmbRay_, 116
 EmbTextMulti_, 118
 EmbTextSingle_, 118
 UndoHistory_, 281
positiveXHoopSize
 VipHeader_, 296
positiveYHoopSize
 VipHeader_, 296
precisionAngle
 PropertyEditor, 211
precisionLength
 PropertyEditor, 211
prefix
 CmdPromptInput, 89
pressPoint
 View, 293
pressResizeHistory
 CmdPromptSplitter, 90
pressY
 CmdPromptHandle, 83
preview
 mainwindow.cpp, 350
PREVIEW_CLONE_NULL
 embroidermodder.h, 305
PREVIEW_CLONE_RUBBER
 embroidermodder.h, 305
PREVIEW_CLONE_SELECTED
 embroidermodder.h, 305
PREVIEW_CLONE_VALUES
 embroidermodder.h, 305
preview_display_bg_color
 Settings_Dialog, 254
preview_display_crosshair_color

Settings_Dialog, 254
preview_display_selectbox_alpha
 Settings_Dialog, 254
preview_display_selectbox_left_color
 Settings_Dialog, 254
preview_display_selectbox_left_fill
 Settings_Dialog, 254
preview_display_selectbox_right_color
 Settings_Dialog, 254
preview_display_selectbox_right_fill
 Settings_Dialog, 254
preview_display_show_scrollbars
 Settings_Dialog, 254
preview_general_mdi_bg_color
 Settings_Dialog, 255
preview_general_mdi_bg_use_color
 Settings_Dialog, 255
preview_general_mdi_bg_use_logo
 Settings_Dialog, 255
preview_general_mdi_bg_use_texture
 Settings_Dialog, 255
preview_grid_color
 Settings_Dialog, 255
preview_lwt_real_render
 Settings_Dialog, 255
preview_lwt_show_lwt
 Settings_Dialog, 255
PREVIEW_MODE_MOVE
 embroidermodder.h, 305
PREVIEW_MODE_NULL
 embroidermodder.h, 305
PREVIEW_MODE_ROTATE
 embroidermodder.h, 305
PREVIEW_MODE_SCALE
 embroidermodder.h, 306
PREVIEW_MODE_VALUES
 embroidermodder.h, 305
preview_prompt_bg_color
 Settings_Dialog, 255
preview_prompt_font_family
 Settings_Dialog, 255
preview_prompt_font_size
 Settings_Dialog, 255
preview_prompt_font_style
 Settings_Dialog, 255
preview_prompt_text_color
 Settings_Dialog, 255
preview_ruler_color
 Settings_Dialog, 255
previewActive
 View, 293
previewData
 View, 293
PreviewDialog, 203
 ~PreviewDialog, 203
 imgWidget, 203
 PreviewDialog, 203
previewMode
 View, 293
previewObjectItemGroup
 View, 293
previewObjectList
 View, 293
previewOff
 View, 288
previewOn
 View, 288
previewPoint
 View, 293
print
 MainWindow, 161
 MdiWindow, 183
printArcResults
 embroidery_internal.h, 438
printer
 MdiWindow, 186
printing_default_device
 Settings_, 231
printing_disable_bg
 Settings_, 231
printing_use_last_device
 Settings_, 231
privacy_policy.md, 547
processInput
 CmdPrompt, 79
 CmdPromptInput, 88
prompt
 MainWindow, 169
promptDivider
 CmdPrompt, 81
promptHistory
 CmdPrompt, 81
 MdiWindow, 186
promptHistoryAppended
 MainWindow, 161
 MdiWindow, 183
promptInput
 CmdPrompt, 81
promptInputList
 MdiWindow, 186
promptInputNext
 MainWindow, 161
 MdiWindow, 184
promptInputNum
 MdiWindow, 186
promptInputPrevious
 MainWindow, 161
 MdiWindow, 184
promptInputPrevNext
 MdiWindow, 184
promptSplitter
 CmdPrompt, 81
promptVBoxLayout
 CmdPrompt, 81
property-editor.cpp
 comboBoxArcClockwise, 356

comboBoxes, 356
comboBoxGeneralLineType, 357
comboBoxGeneralLineWidth, 357
comboBoxPathClosed, 357
comboBoxPathVertexNum, 357
comboBoxPolylineClosed, 357
comboBoxPolylineVertexNum, 357
comboBoxTextSingleBackward, 357
comboBoxTextSingleFont, 357
comboBoxTextSingleJustify, 357
comboBoxTextSingleUpsideDown, 357
groupBoxes, 357
groupBoxGeometryCircle, 357
groupBoxGeometryDimAligned, 357
groupBoxGeometryDimAngular, 357
groupBoxGeometryDimArcLength, 357
groupBoxGeometryDimDiameter, 357
groupBoxGeometryDimLeader, 357
groupBoxGeometryDimLinear, 357
groupBoxGeometryDimOrdinate, 358
groupBoxGeometryDimRadius, 358
groupBoxGeometryEllipse, 358
groupBoxGeometryImage, 358
groupBoxGeometryInfiniteLine, 358
groupBoxGeometryLine, 358
groupBoxGeometryPath, 358
groupBoxGeometryPoint, 358
groupBoxGeometryPolygon, 358
groupBoxGeometryPolyline, 358
groupBoxGeometryRay, 358
groupBoxGeometryRectangle, 358
groupBoxGeometryTextMulti, 358
groupBoxGeometryTextSingle, 358
groupBoxMiscImage, 358
groupBoxMiscPath, 358
groupBoxMiscPolyline, 358
groupBoxMiscTextSingle, 358
groupBoxTextTextSingle, 359
lineEditArcArea, 359
lineEditArcCenterY, 359
lineEditArcChord, 359
lineEditArcEndAngle, 359
lineEditArcEndX, 359
lineEditArcEndY, 359
lineEditArcIncAngle, 359
lineEditArcLength, 359
lineEditArcRadius, 359
lineEditArcStartAngle, 359
lineEditArcStartX, 359
lineEditArcStartY, 359
lineEditBlockX, 359
lineEditBlockY, 359
lineEditCircleArea, 359
lineEditCircleCenterX, 359
lineEditCircleCenterY, 359
lineEditCircleCircumference, 360
lineEditCircleDiameter, 360
lineEditCircleRadius, 360
lineEditEllipseCenterX, 360
lineEditEllipseCenterY, 360
lineEditEllipseDiameterMajor, 360
lineEditEllipseDiameterMinor, 360
lineEditEllipseRadiusMajor, 360
lineEditEllipseRadiusMinor, 360
lineEditImageHeight, 360
lineEditImageName, 360
lineEditImagePath, 360
lineEditImageWidth, 360
lineEditImageX, 360
lineEditImageY, 360
lineEditInfiniteLineVectorX, 360
lineEditInfiniteLineVectorY, 360
lineEditInfiniteLineX2, 360
lineEditInfiniteLineY1, 361
lineEditInfiniteLineY2, 361
lineEditLineAngle, 361
lineEditLineDeltaX, 361
lineEditLineDeltaY, 361
lineEditLineEndX, 361
lineEditLineEndY, 361
lineEditLineLength, 361
lineEditLineStartX, 361
lineEditLineStartY, 361
lineEditPathArea, 361
lineEditPathLength, 361
lineEditPathVertexX, 361
lineEditPathVertexY, 361
lineEditPointX, 361
lineEditPointY, 361
lineEditPolygonCenterX, 361
lineEditPolygonCenterY, 361
lineEditPolygonDiameterSide, 362
lineEditPolygonDiameterVertex, 362
lineEditPolygonInteriorAngle, 362
lineEditPolygonRadiusSide, 362
lineEditPolygonRadiusVertex, 362
lineEditPolylineArea, 362
lineEditPolylineLength, 362
lineEditPolylineVertexX, 362
lineEditPolylineVertexY, 362
lineEditRayVectorX, 362
lineEditRayVectorY, 362
lineEditRayX1, 362
lineEditRayX2, 362
lineEditRayY1, 362
lineEditRayY2, 362
lineEditRectangleArea, 362
lineEditRectangleCorner1X, 362
lineEditRectangleCorner1Y, 362
lineEditRectangleCorner2X, 363
lineEditRectangleCorner2Y, 363
lineEditRectangleCorner3X, 363
lineEditRectangleCorner3Y, 363
lineEditRectangleCorner4X, 363
lineEditRectangleCorner4Y, 363
lineEditRectangleHeight, 363

lineEditRectangleWidth, 363
lineEdits, 363
lineEditTextMultiX, 363
lineEditTextMultiY, 363
lineEditTextSingleContents, 363
lineEditTextSingleHeight, 363
lineEditTextSingleRotation, 363
lineEditTextSingleX, 363
lineEditTextSingleY, 363
toolButtonArcClockwise, 363
toolButtonBlockX, 363
toolButtonBlockY, 364
toolButtonCircleArea, 364
toolButtonCircleCenterX, 364
toolButtonCircleCenterY, 364
toolButtonCircleCircumference, 364
toolButtonCircleDiameter, 364
toolButtonCircleRadius, 364
toolButtonEllipseCenterX, 364
toolButtonEllipseCenterY, 364
toolButtonEllipseDiameterMajor, 364
toolButtonEllipseDiameterMinor, 364
toolButtonEllipseRadiusMajor, 364
toolButtonEllipseRadiusMinor, 364
toolButtonImageHeight, 364
toolButtonImageName, 364
toolButtonImagePath, 364
toolButtonImageWidth, 364
toolButtonImageX, 364
toolButtonImageY, 365
toolButtonInfiniteLineVectorX, 365
toolButtonInfiniteLineVectorY, 365
toolButtonInfiniteLineX2, 365
toolButtonInfiniteLineY1, 365
toolButtonInfiniteLineY2, 365
toolButtonLineAngle, 365
toolButtonLineDeltaX, 365
toolButtonLineDeltaY, 365
toolButtonLineEndX, 365
toolButtonLineEndY, 365
toolButtonLineLength, 365
toolButtonLineStartX, 365
toolButtonLineStartY, 365
toolButtonPathArea, 365
toolButtonPathClosed, 365
toolButtonPathLength, 365
toolButtonPathVertexNum, 365
toolButtonPathVertexX, 366
toolButtonPathVertexY, 366
toolButtonPointX, 366
toolButtonPointY, 366
toolButtonPolygonCenterX, 366
toolButtonPolygonCenterY, 366
toolButtonPolygonDiameterSide, 366
toolButtonPolygonDiameterVertex, 366
toolButtonPolygonInteriorAngle, 366
toolButtonPolygonRadiusSide, 366
toolButtonPolygonRadiusVertex, 366
toolButtonPolylineArea, 366
toolButtonPolylineClosed, 366
toolButtonPolylineLength, 366
toolButtonPolylineVertexNum, 366
toolButtonPolylineVertexX, 366
toolButtonPolylineVertexY, 366
toolButtonRayVectorX, 366
toolButtonRayVectorY, 367
toolButtonRayX1, 367
toolButtonRayX2, 367
toolButtonRayY1, 367
toolButtonRayY2, 367
toolButtonRectangleArea, 367
toolButtonRectangleCorner1X, 367
toolButtonRectangleCorner1Y, 367
toolButtonRectangleCorner2X, 367
toolButtonRectangleCorner2Y, 367
toolButtonRectangleCorner3X, 367
toolButtonRectangleCorner3Y, 367
toolButtonRectangleCorner4X, 367
toolButtonRectangleCorner4Y, 367
toolButtonRectangleHeight, 367
toolButtonRectangleWidth, 367
toolButtons, 367
toolButtonTextMultiX, 367
toolButtonTextMultiY, 368
toolButtonTextSingleBackward, 368
toolButtonTextSingleContents, 368
toolButtonTextSingleFont, 368
toolButtonTextSingleHeight, 368
toolButtonTextSingleJustify, 368
toolButtonTextSingleRotation, 368
toolButtonTextSingleUpsideDown, 368
toolButtonTextSingleX, 368
toolButtonTextSingleY, 368
PropertyEditor, 204
~PropertyEditor, 206
clearAllFields, 206
comboBoxSelected, 210
createComboBox, 206
createComboBoxSelected, 206
createFontComboBox, 206
createGroupBoxGeneral, 206
createGroupBoxGeometryArc, 207
createGroupBoxGeometryBlock, 207
createGroupBoxGeometryCircle, 207
createGroupBoxGeometryDimAligned, 207
createGroupBoxGeometryDimAngular, 207
createGroupBoxGeometryDimArcLength, 207
createGroupBoxGeometryDimDiameter, 207
createGroupBoxGeometryDimLeader, 207
createGroupBoxGeometryDimLinear, 207
createGroupBoxGeometryDimOrdinate, 207
createGroupBoxGeometryDimRadius, 207
createGroupBoxGeometryEllipse, 207
createGroupBoxGeometryImage, 208
createGroupBoxGeometryInfiniteLine, 208
createGroupBoxGeometryLine, 208

createGroupBoxGeometryPath, 208
createGroupBoxGeometryPoint, 208
createGroupBoxGeometryPolygon, 208
createGroupBoxGeometryPolyline, 208
createGroupBoxGeometryRay, 208
createGroupBoxGeometryRectangle, 208
createGroupBoxGeometryTextMulti, 208
createGroupBoxGeometryTextSingle, 208
createGroupBoxMiscArc, 208
createGroupBoxMisclImage, 208
createGroupBoxMiscPath, 209
createGroupBoxMiscPolyline, 209
createGroupBoxMiscTextSingle, 209
createGroupBoxTextTextSingle, 209
createLineEdit, 209
createToolButton, 209
createToolButtonPickAdd, 209
createToolButtonQSelect, 209
eventFilter, 209
fieldEdited, 209
fieldNewText, 210
fieldNoText, 211
fieldOffText, 211
fieldOldText, 211
fieldOnText, 211
fieldVariesText, 211
fieldYesText, 211
focusWidget, 211
hideAllGroups, 209
iconDir, 211
iconSize, 211
mapSignal, 209
pickAdd, 211
pickAddModeToggled, 209
precisionAngle, 211
precisionLength, 211
PropertyEditor, 206
propertyEditorButtonStyle, 211
selectedItemList, 211
setSelectedItems, 210
showGroups, 210
showOneType, 210
signalMapper, 211
tempArcObj, 211
tempBlockObj, 211
tempCircleObj, 211
tempDimAlignedObj, 212
tempDimAngularObj, 212
tempDimArcLenObj, 212
tempDimDiamObj, 212
tempDimLeaderObj, 212
tempDimLinearObj, 212
tempDimOrdObj, 212
tempDimRadiusObj, 212
tempEllipseArcObj, 212
tempEllipseObj, 212
tempHatchObj, 212
templImageObj, 212
tempInflLineObj, 212
tempLineObj, 212
tempPathObj, 212
tempPointObj, 212
tempPolygonObj, 212
tempPolylineObj, 212
tempRayObj, 213
tempRectObj, 213
tempSplineObj, 213
tempTextMultiObj, 213
tempTextSingleObj, 213
togglePickAddMode, 210
toolButtonPickAdd, 213
toolButtonQSelect, 213
updateComboBoxBoolIfVaries, 210
updateComboBoxStrIfVaries, 210
updateFontComboBoxStrIfVaries, 210
updateLineEditNumIfVaries, 210
updateLineEditStrIfVaries, 210
updatePickAddModeButton, 210
propertyEditorButtonStyle
 PropertyEditor, 211

qsnap_aperture_size
 Settings_, 231
qsnap_apparent
 Settings_, 231
qsnap_center
 Settings_, 231
qsnap_enabled
 Settings_, 231
qsnap_endpoint
 Settings_, 231
qsnap_extension
 Settings_, 231
qsnap_insertion
 Settings_, 231
qsnap_intersection
 Settings_, 231
qsnap_locator_color
 Settings_, 231
qsnap_locator_size
 Settings_, 231
qsnap_midpoint
 Settings_, 231
qsnap_mode
 EmbView_, 121
qsnap_nearest
 Settings_, 231
qsnap_node
 Settings_, 231
qsnap_parallel
 Settings_, 232
qsnap_perpendicular
 Settings_, 232
qsnap_quadrant
 Settings_, 232
qsnap_tangent
 Settings_, 232

qSnapActive
 View, 293
qsnapApertureSize
 View, 293
qsnapLocatorColor
 View, 294
qsnapLocatorSize
 View, 294
qSnapToggle
 View, 294
qtrack_mode
 EmbView_, 121
QUADTOCONTROL
 embroidery_internal.h, 428
QUADTOEND
 embroidery_internal.h, 428
quit
 MainWindow, 161

r
 EmbColor_, 106
r_value
 Parameter_, 187
radians
 embroidermodder.h, 307
 embroidery.h, 404
 functions.c, 522
radius
 EmbCircle_, 106
 EmbEllipse_, 108
 EmbRect_, 116
rapidFireEnabled
 CmdPromptInput, 89
rapidMoveActive
 View, 294
read100
 embroidery_internal.h, 439
 format_100.c, 472
read10o
 embroidery_internal.h, 439
 format_10o.c, 473
read_configuration
 mainwindow-settings.cpp, 345
read_hoop
 format_jef.c, 490
read_settings
 embroidermodder.h, 307
 mainwindow.cpp, 348
readArt
 embroidery_internal.h, 439
 format_art.c, 473
readBmc
 embroidery_internal.h, 439
 format_bmc.c, 474
readBro
 embroidery_internal.h, 439
 format_bro.c, 474
readCnd
 embroidery_internal.h, 439
 format_cnd.c, 475
readCol
 embroidery_internal.h, 439
 format_col.c, 476
readCsd
 embroidery_internal.h, 439
 format_csd.c, 477
readCsv
 embroidery_internal.h, 439
 format_csv.c, 478
readDat
 embroidery_internal.h, 439
 format_dat.c, 478
readDem
 embroidery_internal.h, 439
 format_dem.c, 479
readDescriptions
 embroidery_internal.h, 440
 format_pes.c, 499
readDsb
 embroidery_internal.h, 440
 format_dsb.c, 480
readDst
 embroidery_internal.h, 440
 format_dst.c, 482
readDsz
 embroidery_internal.h, 440
 format_dsz.c, 482
readDxf
 embroidery_internal.h, 440
 format_dxf.c, 483
readEdr
 embroidery_internal.h, 440
 format_edr.c, 483
readEmd
 embroidery_internal.h, 440
 format_emd.c, 484
reader_state
 EmbFormatList_, 109
readExp
 embroidery_internal.h, 440
 format_exp.c, 484
readExy
 embroidery_internal.h, 440
 format_exy.c, 485
readEys
 embroidery_internal.h, 440
 format_eys.c, 485
readFeatherPatterns
 embroidery_internal.h, 441
 format_pes.c, 499
readFullSector
 embroidery_internal.h, 441
 main.c, 539
readFxy
 embroidery_internal.h, 441
 format_fxy.c, 486
readGc

embroidery_internal.h, 441
format_gc.c, 486

readGnc
 embroidery_internal.h, 441
 format_gnc.c, 487

readGt
 embroidery_internal.h, 441
 format_gt.c, 487

readHoopName
 embroidery_internal.h, 441
 format_pes.c, 499

readHus
 embroidery_internal.h, 441
 format_hus.c, 489

readImageString
 embroidery_internal.h, 441
 format_pes.c, 500

readInb
 embroidery_internal.h, 442
 format_inb.c, 489

readInf
 embroidery_internal.h, 442
 format_inf.c, 489

readJef
 embroidery_internal.h, 442
 format_jef.c, 491

readKsm
 embroidery_internal.h, 442
 format_ksm.c, 491

readLine
 format_dxf.c, 483

readMax
 embroidery_internal.h, 442
 format_max.c, 492

readMit
 embroidery_internal.h, 442
 format_mit.c, 492

readMotifPatterns
 embroidery_internal.h, 442
 format_pes.c, 500

readNew
 embroidery_internal.h, 442
 format_new.c, 493

readNextSector
 embroidery_internal.h, 442
 main.c, 539

readOfm
 embroidery_internal.h, 442
 format_ofm.c, 494

readPcd
 embroidery_internal.h, 443
 format_pcd.c, 495

readPcm
 embroidery_internal.h, 443
 format_pcm.c, 495

readPcq
 embroidery_internal.h, 443
 format_pcq.c, 496

readPcs
 embroidery_internal.h, 443
 format_pcs.c, 496

readPec
 embroidery_internal.h, 443
 format_pec.c, 497

readPecStitches
 embroidery_internal.h, 443
 format_pec.c, 497

readPel
 embroidery_internal.h, 443
 format_pel.c, 498

readPem
 embroidery_internal.h, 443
 format_pem.c, 498

readPes
 embroidery_internal.h, 444
 format_pes.c, 500

readPESHeaderV10
 embroidery_internal.h, 444
 format_pes.c, 500

readPESHeaderV5
 embroidery_internal.h, 444
 format_pes.c, 500

readPESHeaderV6
 embroidery_internal.h, 444
 format_pes.c, 500

readPESHeaderV7
 embroidery_internal.h, 444
 format_pes.c, 500

readPESHeaderV8
 embroidery_internal.h, 444
 format_pes.c, 500

readPESHeaderV9
 embroidery_internal.h, 444
 format_pes.c, 500

readPhb
 embroidery_internal.h, 444
 format_phb.c, 501

readPhc
 embroidery_internal.h, 444
 format_phc.c, 502

readPlt
 embroidery_internal.h, 444
 format_plt.c, 502

readProgrammableFills
 embroidery_internal.h, 445
 format_pes.c, 500

readRgb
 embroidery_internal.h, 445
 format_rgb.c, 503

readSettings
 MainWindow, 161

readSew
 embroidery_internal.h, 445
 format_sew.c, 503

readShv
 embroidery_internal.h, 445

format_shv.c, 504
readSst
embroidery_internal.h, 445
format_sst.c, 504
readStx
embroidery_internal.h, 445
format_stx.c, 505
readSvg
embroidery_internal.h, 445
format_svg.c, 506
readT01
embroidery_internal.h, 445
format_t01.c, 507
readT09
embroidery_internal.h, 445
format_t09.c, 507
readTap
embroidery_internal.h, 445
format_tap.c, 508
readThr
embroidery_internal.h, 446
format_thr.c, 508
readThreads
embroidery_internal.h, 446
format_pes.c, 500
readTxt
embroidery_internal.h, 446
format_txt.c, 509
readU00
embroidery_internal.h, 446
format_u00.c, 509
readU01
embroidery_internal.h, 446
format_u01.c, 510
readVip
embroidery_internal.h, 446
format_vip.c, 511
readVp3
embroidery_internal.h, 446
format_vp3.c, 512
readXxx
embroidery_internal.h, 446
format_xxx.c, 513
readZsk
embroidery_internal.h, 446
format_zsk.c, 514
real_render
EmbView_, 122
realRender
BaseObject, 69
recalculateLimits
View, 288
recentMenu
MainWindow, 169
recentMenuAboutToShow
MainWindow, 162
rect
BaseObject, 69
EmbGeometry_, 110
rect.c
embRect_area, 524
embRect_init, 524
RectObject, 213
~RectObject, 215
allGripPoints, 215
gripEdit, 216
init, 216
mouseSnapPoint, 216
objectArea, 216
objectBottomLeft, 216
objectBottomRight, 216
objectHeight, 216
objectPos, 216
objectSavePath, 216
objectTopLeft, 216
objectTopRight, 216
objectWidth, 217
paint, 217
RectObject, 215
setObjectRect, 217
Type, 215
type, 217
updatePath, 217
updateRubber, 217
vulcanize, 217
RED_TERM_COLOR
embroidery_internal.h, 428
redo
MainWindow, 162
UndoableAddCommand, 271
UndoableDeleteCommand, 272
UndoableGripEditCommand, 273
UndoableMirrorCommand, 274
UndoableMoveCommand, 275
UndoableNavCommand, 276
UndoableRotateCommand, 277
UndoableScaleCommand, 279
UndoEditor, 280
redoPressed
CmdPrompt, 79
CmdPromptInput, 88
redoText
UndoEditor, 280
rejectChanges
Settings_Dialog, 246
releasePoint
View, 294
releaseResizeHistory
CmdPromptSplitter, 90
releaseY
CmdPromptHandle, 83
repeatAction
View, 288
report
embroidery.h, 404
reserved

ThredExtension_, 268
 ThredHeader_, 268
 reserved1
 _bcf_file_header, 52
 reserved2
 _bcf_file_header, 52
 RESET_TERM_COLOR
 embroidery_internal.h, 428
 resizeEvent
 MainWindow, 162
 resizeHistory
 CmdPromptHistory, 84
 resizeTheHistory
 CmdPrompt, 79
 reverse_byte_order
 encoding.c, 463
 rgb, 17, 445, 503
 right
 _vp3Hoop, 54
 EmbRect_, 116
 hoop_padding, 123
 right2
 _vp3Hoop, 54
 rightBrush
 SelectBox, 224
 rightBrushColor
 SelectBox, 224
 rightPen
 SelectBox, 224
 rightPenColor
 SelectBox, 224
 rightSiblingId
 _bcf_directory_entry, 49
 RobisonAnton_Polyester
 embroidery.h, 387
 RobisonAnton_Rayon
 embroidery.h, 387
 rotate
 UndoableRotateCommand, 277
 ROTATE_MODE_NORMAL
 embroidermodder.h, 306
 ROTATE_MODE_REFERENCE
 embroidermodder.h, 306
 rotate_vector
 object-arc.cpp, 351
 rotateAction
 View, 288
 rotateSelected
 View, 288
 rotation
 EmbEllipse_, 108
 EmbRect_, 116
 UiObject_, 270
 roundToMultiple
 View, 289
 rubber_mode
 EmbView_, 122
 rubberRoomList

View, 294
 ruler_color
 Settings_, 232
 ruler_metric
 Settings_, 232
 ruler_mode
 EmbView_, 122
 ruler_pixel_size
 Settings_, 232
 ruler_show_on_load
 Settings_, 232
 ruler_width
 Settings_, 232
 rulerColor
 View, 294
 rulerMetric
 View, 294
 rulerPixelSize
 View, 294
 rules
 fill.c, 467
 LSYSTEM, 136
 run_script
 MainWindow, 162
 run_script_file
 MainWindow, 162
 runCommand
 CmdPrompt, 80
 CmdPromptInput, 88
 MainWindow, 162
 runCommandClick
 MainWindow, 162
 runCommandContext
 MainWindow, 163
 runCommandMain
 MainWindow, 163
 runCommandMove
 MainWindow, 163
 runCommandPrompt
 MainWindow, 163
 running
 Settings_, 232
 s_value
 Parameter_, 187
 safe_free
 embroidery_internal.h, 446
 formats.c, 471
 save
 ImageWidget, 129
 SaveObject, 222
 save_points_to_pattern
 fill.c, 467
 saveasfile
 MainWindow, 163
 saveBMC
 MdiWindow, 184
 saveFile
 MdiWindow, 184

savefile
 MainWindow, 163

saveHistory
 CmdPrompt, 80

SaveObject, 217
 ~SaveObject, 218

addArc, 218

addBlock, 219

addCircle, 219

addDimAligned, 219

addDimAngular, 219

addDimArcLength, 219

addDimDiameter, 220

addDimLeader, 220

addDimLinear, 220

addDimOrdinate, 220

addDimRadius, 220

addEllipse, 220

addEllipseArc, 220

addGrid, 221

addHatch, 221

addImage, 221

addInfiniteLine, 221

addLine, 221

addPath, 221

addPoint, 221

addPolygon, 221

addPolyline, 221

addRay, 221

addRectangle, 221

addSlot, 222

addSpline, 222

addTextMulti, 222

addTextSingle, 222

formatType, 222

gscene, 222

save, 222

SaveObject, 218

toPolyline, 222

scale
 EmbView_, 122

 UiObject_, 270

SCALE_MODE_NORMAL
 embroidermodder.h, 306

SCALE_MODE_REFERENCE
 embroidermodder.h, 306

scaleAction
 View, 289

scaleSelected
 View, 289

sceneGripPoint
 View, 294

sceneMousePoint
 View, 294

sceneMovePoint
 View, 294

scenePressPoint
 View, 294

sceneReleasePoint
 View, 294

script
 Action__, 55

second
 EmbTime_, 119

sectionName
 StxThread_, 260

sectorShift
 _bcf_file_header, 52

sectorSize
 _bcf_file_difat, 50

 main.c, 539

seekToSector
 main.c, 539

selectable
 UiObject_, 270

selectAll
 MainWindow, 163

 View, 289

selectAllPressed
 CmdPrompt, 80

 CmdPromptInput, 88

SelectBox, 223
 alpha, 224

 boxDir, 224

 dirBrush, 224

 dirPen, 224

 forceRepaint, 223

 leftBrush, 224

 leftBrushColor, 224

 leftPen, 224

 leftPenColor, 224

 paintEvent, 223

 rightBrush, 224

 rightBrushColor, 224

 rightPen, 224

 rightPenColor, 224

 SelectBox, 223

 setColors, 224

 setDirection, 224

selectBox
 View, 294

selected
 EmbView_, 122

selectedItemList
 PropertyEditor, 211

selectingActive
 View, 294

selection_coolgrip_color
 Settings_, 232

selection_grip_size
 Settings_, 232

selection_hotgrip_color
 Settings_, 232

selection_mode_pickadd
 Settings_, 232

selection_mode_pickdrag

Settings_, 232
 selection_mode_pickfirst
 Settings_, 232
 selection_pickbox_size
 Settings_, 232
 selectionChanged
 View, 289
 sendCloseMdiWin
 MdiWindow, 184
 SEQUIN
 embroidery.h, 387
 set_dst_variable
 format_dst.c, 482
 set_object_color
 arc.c, 519
 setBackgroundColor
 MdiArea, 178
 View, 289
 setBackgroundLogo
 MdiArea, 178
 setBackgroundTexture
 MdiArea, 178
 setColors
 SelectBox, 224
 setCornerButton
 View, 289
 setCrossHairColor
 View, 289
 setCrossHairSize
 View, 289
 setCurrentColor
 MdiWindow, 184
 setCurrentFile
 MdiWindow, 184
 setCurrentLayer
 MdiWindow, 185
 setCurrentLineType
 MdiWindow, 185
 setCurrentLineWidth
 MdiWindow, 185
 setCurrentText
 CmdPrompt, 80
 setDirection
 SelectBox, 224
 setGridColor
 View, 289
 setHistory
 CmdPrompt, 80
 setLine
 BaseObject, 69
 setMainWin
 Application, 56
 setMouseCoord
 StatusBar, 256
 setObjectArea
 CircleObject, 75
 setObjectCenter
 BaseObject, 70
 setObjectCenterX
 BaseObject, 70
 setObjectCenterY
 BaseObject, 70
 setObjectCircumference
 CircleObject, 75
 setObjectColor
 BaseObject, 70
 setObjectColorRGB
 BaseObject, 70
 setObjectDiameter
 CircleObject, 75
 setObjectDiameterMajor
 EllipseObject, 102
 setObjectDiameterMinor
 EllipseObject, 102
 setObjectEndAngle
 ArcObject, 64
 setObjectEndPoint
 ArcObject, 65
 setObjectEndPoint1
 DimLeaderObject, 97
 LineObject, 134
 setObjectEndPoint2
 DimLeaderObject, 97
 LineObject, 134
 setObjectLineType
 BaseObject, 70
 setObjectLineWidth
 BaseObject, 70
 setObjectMidPoint
 ArcObject, 65
 setObjectPath
 BaseObject, 70
 setObjectPos
 PathObject, 190
 PointObject, 194
 PolygonObject, 198
 PolylineObject, 202
 TextSingleObject, 264, 265
 setObjectRadius
 ArcObject, 65
 CircleObject, 75
 setObjectRadiusMajor
 EllipseObject, 102
 setObjectRadiusMinor
 EllipseObject, 102
 setObjectRect
 ImageObject, 127
 RectObject, 217
 setObjectRubberMode
 BaseObject, 70
 setObjectRubberPoint
 BaseObject, 70
 setObjectRubberText
 BaseObject, 70
 setObjectSize
 EllipseObject, 102

setObjectStartAngle
 ArcObject, 65
setObjectStartPoint
 ArcObject, 65
setObjectText
 TextSingleObject, 265
setObjectTextBackward
 TextSingleObject, 265
setObjectTextBold
 TextSingleObject, 265
setObjectTextFont
 TextSingleObject, 265
setObjectTextItalic
 TextSingleObject, 265
setObjectTextJustify
 TextSingleObject, 265
setObjectTextOverline
 TextSingleObject, 265
setObjectTextSize
 TextSingleObject, 265
setObjectTextStrikeOut
 TextSingleObject, 265
setObjectTextStyle
 TextSingleObject, 265
setObjectTextUnderline
 TextSingleObject, 265
setObjectTextUpsideDown
 TextSingleObject, 265
setObjectX
 PathObject, 190
 PointObject, 194
 PolygonObject, 198
 PolylineObject, 202
 TextSingleObject, 266
setObjectX1
 DimLeaderObject, 97
 LineObject, 135
setObjectX2
 DimLeaderObject, 97
 LineObject, 135
setObjectY
 PathObject, 190
 PointObject, 194
 PolygonObject, 198
 PolylineObject, 202
 TextSingleObject, 266
setObjectY1
 DimLeaderObject, 97
 LineObject, 135
setObjectY2
 DimLeaderObject, 97
 LineObject, 135
setPrefix
 CmdPrompt, 80
setPromptBackgroundColor
 CmdPrompt, 80
setPromptFontFamily
 CmdPrompt, 80
setPromptFontSize
 CmdPrompt, 80
setPromptFontStyle
 CmdPrompt, 80
setPromptTextColor
 CmdPrompt, 80
setRect
 BaseObject, 70
setRubberMode
 View, 289
setRubberPoint
 View, 289
setRubberText
 View, 289
setRulerColor
 View, 290
setSelectBoxColors
 View, 290
setSelectedItems
 PropertyEditor, 210
setShiftPressed
 MainWindow, 163
setShiftReleased
 MainWindow, 163
setTextAngle
 MainWindow, 163
setTextBold
 MainWindow, 163
setTextFont
 MainWindow, 163
setTextItalic
 MainWindow, 163
setTextOverline
 MainWindow, 164
setTextSize
 MainWindow, 164
setTextStrikeOut
 MainWindow, 164
setTextUnderline
 MainWindow, 164
Settings
 embroidermodder.h, 301
settings
 embroidermodder.h, 308
 mainwindow.cpp, 350
Settings_
 assets_dir, 227
 current_directory, 227
 debug_mode, 227
 display_bg_color, 227
 display_crosshair_color, 227
 display_crosshair_percent, 227
 display_renderhint_aa, 227
 display_renderhint_high_aa, 228
 display_renderhint_noncosmetic, 228
 display_renderhint_smooth_pix, 228
 display_renderhint_text_aa, 228
 display_scrollbar_widget_num, 228

display_selectbox_alpha, 228
 display_selectbox_left_color, 228
 display_selectbox_left_fill, 228
 display_selectbox_right_color, 228
 display_selectbox_right_fill, 228
 display_show_scrollbars, 228
 display_units, 228
 display_use_opengl, 228
 display_zoomscale_in, 228
 display_zoomscale_out, 228
 general_check_for_updates, 228
 general_current_tip, 228
 general_mdi_bg_color, 228
 general_mdi_bg_logo, 229
 general_mdi_bg_texture, 229
 general_system_help_browser, 229
 grid_center, 229
 grid_center_on_origin, 229
 grid_color, 229
 grid_color_match_crosshair, 229
 grid_load_from_file, 229
 grid_show_on_load, 229
 grid_show_origin, 229
 grid_size, 229
 grid_size_radius, 229
 grid_spacing, 229
 grid_spacing_angle, 229
 grid_spacing_radius, 229
 grid_type, 229
 icon_size, 229
 icon_theme, 229
 language, 230
 lwt_default_lwt, 230
 lwt_real_render, 230
 lwt_show_lwt, 230
 major_tick_seperation, 230
 mdi_bg_use_color, 230
 mdi_bg_use_logo, 230
 mdi_bg_use_texture, 230
 menu_action, 230
 needle_speed, 230
 opensave_custom_filter, 230
 opensave_open_format, 230
 opensave_open_thumbnail, 230
 opensave_recent_directory, 230
 opensave_recent_list_of_files, 230
 opensave_recent_max_files, 230
 opensave_save_format, 230
 opensave_save_thumbnail, 230
 opensave_trim_dst_num_jumps, 231
 pattern_index, 231
 printing_default_device, 231
 printing_disable_bg, 231
 printing_use_last_device, 231
 qsnap_aperture_size, 231
 qsnap_apparent, 231
 qsnap_center, 231
 qsnap_enabled, 231
 qsnap_endpoint, 231
 qsnap_extension, 231
 qsnap_insertion, 231
 qsnap_intersection, 231
 qsnap_locator_color, 231
 qsnap_locator_size, 231
 qsnap_midpoint, 231
 qsnap_nearest, 231
 qsnap_node, 231
 qsnap_parallel, 232
 qsnap_perpendicular, 232
 qsnap_quadrant, 232
 qsnap_tangent, 232
 ruler_color, 232
 ruler_metric, 232
 ruler_pixel_size, 232
 ruler_show_on_load, 232
 ruler_width, 232
 running, 232
 selection_coolgrip_color, 232
 selection_grip_size, 232
 selection_hotgrip_color, 232
 selection_mode_pickadd, 232
 selection_mode_pickdrag, 232
 selection_mode_pickfirst, 232
 selection_pickbox_size, 232
 shift_held, 232
 shine_color, 233
 show_about_dialog, 233
 show_details_dialog, 233
 show_editor, 233
 show_open_file_dialog, 233
 show_settings_editor, 233
 stitch_time, 233
 testing, 233
 text_angle, 233
 text_font, 233
 text_size, 233
 text_style_bold, 233
 text_style_italic, 233
 text_style_overline, 233
 text_style_strikeout, 233
 text_style_underline, 233
 texture_list, 233
 tick_depth, 233
 ticks_color, 234
 tip_of_the_day, 234
 to_open, 234
 use_translation, 234
 version, 234
 zoomInLimit, 234
 zoomOutLimit, 234
 Settings_Dialog, 234
 ~Settings_Dialog, 239
 accept_display_bg_color, 248
 accept_display_crosshair_color, 248
 accept_display_selectbox_left_color, 248
 accept_display_selectbox_left_fill, 248

accept_display_selectbox_right_color, 248
accept_display_selectbox_right_fill, 248
accept_general_mdi_bg_color, 248
accept_general_mdi_bg_logo, 248
accept_general_mdi_bg_texture, 248
accept_grid_color, 248
accept_prompt_bg_color, 249
accept_prompt_text_color, 249
accept_ruler_color, 249
acceptChanges, 239
addColorsToComboBox, 239
buttonBox, 249
buttonCustomFilterClearAll, 239
buttonCustomFilterClearAllClicked, 239
buttonCustomFilterSelectAll, 239
buttonCustomFilterSelectAllClicked, 239
buttonQSnapClearAll, 240
buttonQSnapClearAllClicked, 240
buttonQSnapSelectAll, 240
buttonQSnapSelectAllClicked, 240
checkBoxCustomFilterStateChanged, 240
checkBoxDisableBGStateChanged, 240
checkBoxGeneralMdiBGUseColorStateChanged,
 240
checkBoxGeneralMdiBGUseLogoStateChanged,
 240
checkBoxGeneralMdiBGUseTextureStateChanged,
 240
checkBoxGridCenterOnOriginStateChanged, 240
checkBoxGridColorMatchCrossHairStateChanged,
 240
checkBoxGridLoadFromFileStateChanged, 240
checkBoxGridShowOnLoadStateChanged, 240
checkBoxGridShowOriginStateChanged, 241
checkBoxLwtRealRenderStateChanged, 241
checkBoxLwtShowLwtStateChanged, 241
checkBoxPromptSaveHistoryAsHtmlStateChanged,
 241
checkBoxPromptSaveHistoryStateChanged, 241
checkBoxQSnapApparentStateChanged, 241
checkBoxQSnapCenterStateChanged, 241
checkBoxQSnapEndPointStateChanged, 241
checkBoxQSnapExtensionStateChanged, 241
checkBoxQSnapInsertionStateChanged, 241
checkBoxQSnapIntersectionStateChanged, 241
checkBoxQSnapMidPointStateChanged, 241
checkBoxQSnapNearestStateChanged, 241
checkBoxQSnapNodeStateChanged, 242
checkBoxQSnapParallelStateChanged, 242
checkBoxQSnapPerpendicularStateChanged, 242
checkBoxQSnapQuadrantStateChanged, 242
checkBoxQSnapTangentStateChanged, 242
checkBoxRenderHintAAStateChanged, 242
checkBoxRenderHintHighAAStateChanged, 242
checkBoxRenderHintNonCosmeticStateChanged,
 242
checkBoxRenderHintSmoothPixStateChanged,
 242
checkBoxRenderHintTextAAStateChanged, 242
checkBoxRulerShowOnLoadStateChanged, 242
checkBoxSelectionModePickAddStateChanged,
 242
checkBoxSelectionModePickDragStateChanged,
 242
checkBoxSelectionModePickFirstStateChanged,
 243
checkBoxShowScrollBarsStateChanged, 243
checkBoxTipOfTheDayStateChanged, 243
checkBoxUseOpenGLStateChanged, 243
chooseDisplayBackgroundColor, 243
chooseDisplayCrossHairColor, 243
chooseDisplaySelectBoxLeftColor, 243
chooseDisplaySelectBoxLeftFill, 243
chooseDisplaySelectBoxRightColor, 243
chooseDisplaySelectBoxRightFill, 243
chooseGeneralMdiBackgroundColor, 243
chooseGeneralMdiBackgroundLogo, 243
chooseGeneralMdiBackgroundTexture, 243
chooseGridColor, 244
choosePromptBackgroundColor, 244
choosePromptTextColor, 244
chooseRulerColor, 244
comboBoxGridTypeCurrentIndexChanged, 244
comboBoxIconSizeCurrentIndexChanged, 244
comboBoxIconThemeCurrentIndexChanged, 244
comboBoxLanguageCurrentIndexChanged, 244
comboBoxPromptFontFamilyCurrentIndexChanged,
 244
comboBoxPromptFontSizeCurrentIndexChanged,
 244
comboBoxQSnapLocatorColorCurrentIndex-
 Changed, 244
comboBoxRulerMetricCurrentIndexChanged, 244
comboBoxScrollBarWidgetCurrentIndexChanged,
 244
comboBoxSelectionCoolGripColorCurrentIndex-
 Changed, 245
comboBoxSelectionHotGripColorCurrentIndex-
 Changed, 245
createTabDisplay, 245
createTabFilePaths, 245
createTabGeneral, 245
createTabGridRuler, 245
createTabLineWeight, 245
createTabOpenSave, 245
createTabOrthoPolar, 245
createTabPrinting, 245
createTabPrompt, 245
createTabQuickSnap, 245
createTabQuickTrack, 245
createTabSelection, 245
createTabSnap, 245
currentDisplayBackgroundColorChanged, 245
currentDisplayCrossHairColorChanged, 246
currentDisplaySelectBoxLeftColorChanged, 246
currentDisplaySelectBoxLeftFillChanged, 246

currentDisplaySelectBoxRightColorChanged, 246
currentDisplaySelectBoxRightFillChanged, 246
currentGeneralMdiBackgroundColorChanged, 246
currentGridColorChanged, 246
currentPromptBackgroundColorChanged, 246
currentPromptTextColorChanged, 246
currentRulerColorChanged, 246
dialog_display_bg_color, 249
dialog_display_crosshair_color, 249
dialog_display_crosshair_percent, 249
dialog_display_renderhint_aa, 249
dialog_display_renderhint_high_aa, 249
dialog_display_renderhint_noncosmetic, 249
dialog_display_renderhint_smooth_pix, 249
dialog_display_renderhint_text_aa, 249
dialog_display_scrollbar_widget_num, 249
dialog_display_selectbox_alpha, 249
dialog_display_selectbox_left_color, 249
dialog_display_selectbox_left_fill, 249
dialog_display_selectbox_right_color, 249
dialog_display_selectbox_right_fill, 249
dialog_display_show_scrollbars, 250
dialog_display_units, 250
dialog_display_use_opengl, 250
dialog_display_zoomscale_in, 250
dialog_display_zoomscale_out, 250
dialog_general_icon_size, 250
dialog_general_icon_theme, 250
dialog_general_language, 250
dialog_general_mdi_bg_color, 250
dialog_general_mdi_bg_logo, 250
dialog_general_mdi_bg_texture, 250
dialog_general_mdi_bg_use_color, 250
dialog_general_mdi_bg_use_logo, 250
dialog_general_mdi_bg_use_texture, 250
dialog_general_system_help_browser, 250
dialog_general_tip_of_the_day, 250
dialog_grid_center_on_origin, 250
dialog_grid_center_x, 250
dialog_grid_center_y, 251
dialog_grid_color, 251
dialog_grid_color_match_crosshair, 251
dialog_grid_load_from_file, 251
dialog_grid_show_on_load, 251
dialog_grid_show_origin, 251
dialog_grid_size_radius, 251
dialog_grid_size_x, 251
dialog_grid_size_y, 251
dialog_grid_spacing_angle, 251
dialog_grid_spacing_radius, 251
dialog_grid_spacing_x, 251
dialog_grid_spacing_y, 251
dialog_grid_type, 251
dialog_lwt_default_lwt, 251
dialog_lwt_real_render, 251
dialog_lwt_show_lwt, 251
dialog_opensave_custom_filter, 251
dialog_opensave_open_format, 252
dialog_opensave_open_thumbnail, 252
dialog_opensave_recent_max_files, 252
dialog_opensave_save_format, 252
dialog_opensave_save_thumbnail, 252
dialog_opensave_trim_dst_num_jumps, 252
dialog_printing_default_device, 252
dialog_printing_disable_bg, 252
dialog_printing_use_last_device, 252
dialog_prompt_bg_color, 252
dialog_prompt_font_family, 252
dialog_prompt_font_size, 252
dialog_prompt_font_style, 252
dialog_prompt_save_history, 252
dialog_prompt_save_history_as_html, 252
dialog_prompt_save_history_filename, 252
dialog_prompt_text_color, 252
dialog_qsnap_aperture_size, 252
dialog_qsnap_apparent, 253
dialog_qsnap_center, 253
dialog_qsnap_enabled, 253
dialog_qsnap_endpoint, 253
dialog_qsnap_extension, 253
dialog_qsnap_insertion, 253
dialog_qsnap_intersection, 253
dialog_qsnap_locator_color, 253
dialog_qsnap_locator_size, 253
dialog_qsnap_midpoint, 253
dialog_qsnap_nearest, 253
dialog_qsnap_node, 253
dialog_qsnap_parallel, 253
dialog_qsnap_perpendicular, 253
dialog_qsnap_quadrant, 253
dialog_qsnap_tangent, 253
dialog_ruler_color, 253
dialog_ruler_metric, 253
dialog_ruler_pixel_size, 254
dialog_ruler_show_on_load, 254
dialog_selection_coolgrip_color, 254
dialog_selection_grip_size, 254
dialog_selection_hotgrip_color, 254
dialog_selection_mode_pickadd, 254
dialog_selection_mode_pickdrag, 254
dialog_selection_mode_pickfirst, 254
dialog_selection_pickbox_size, 254
mainWin, 254
preview_display_bg_color, 254
preview_display_crosshair_color, 254
preview_display_selectbox_alpha, 254
preview_display_selectbox_left_color, 254
preview_display_selectbox_left_fill, 254
preview_display_selectbox_right_color, 254
preview_display_selectbox_right_fill, 254
preview_display_show_scrollbars, 254
preview_general_mdi_bg_color, 255
preview_general_mdi_bg_use_color, 255
preview_general_mdi_bg_use_logo, 255
preview_general_mdi_bg_use_texture, 255
preview_grid_color, 255

preview_lwt_real_render, 255
preview_lwt_show_lwt, 255
preview_prompt_bg_color, 255
preview_prompt_font_family, 255
preview_prompt_font_size, 255
preview_prompt_font_style, 255
preview_prompt_text_color, 255
preview_ruler_color, 255
rejectChanges, 246
Settings_Dialog, 239
sliderQSnapApertureSizeValueChanged, 246
sliderQSnapLocatorSizeValueChanged, 246
sliderSelectionGripSizeValueChanged, 247
sliderSelectionPickBoxSizeValueChanged, 247
spinBoxDisplaySelectBoxAlphaValueChanged,
 247
spinBoxGridCenterXValueChanged, 247
spinBoxGridCenterYValueChanged, 247
spinBoxGridRadiusValueChanged, 247
spinBoxGridSizeXValueChanged, 247
spinBoxGridSizeYValueChanged, 247
spinBoxGridSpacingAngleValueChanged, 247
spinBoxGridSpacingRadiusValueChanged, 247
spinBoxGridSpacingXValueChanged, 247
spinBoxGridSpacingYValueChanged, 247
spinBoxPromptFontSizeValueChanged, 247
spinBoxRecentMaxFilesValueChanged, 248
spinBoxRulerPixelSizeValueChanged, 248
spinBoxTrimDstNumJumpsValueChanged, 248
spinBoxZoomScaleInValueChanged, 248
spinBoxZoomScaleOutValueChanged, 248
tabWidget, 255
settings_display_bg_color
 MainWindow, 169
settings_display_crosshair_color
 MainWindow, 169
settings_display_crosshair_percent
 MainWindow, 169
settings_display_renderhint_aa
 MainWindow, 169
settings_display_renderhint_high_aa
 MainWindow, 169
settings_display_renderhint_noncosmetic
 MainWindow, 169
settings_display_renderhint_smooth_pix
 MainWindow, 169
settings_display_renderhint_text_aa
 MainWindow, 169
settings_display_scrollbar_widget_num
 MainWindow, 169
settings_display_selectbox_alpha
 MainWindow, 169
settings_display_selectbox_left_color
 MainWindow, 169
settings_display_selectbox_left_fill
 MainWindow, 169
settings_display_selectbox_right_color
 MainWindow, 169
settings_display_selectbox_right_fill
 MainWindow, 169
settings_display_show_scrollbars
 MainWindow, 169
settings_display_units
 MainWindow, 170
settings_display_use_opengl
 MainWindow, 170
settings_display_zoomscale_in
 MainWindow, 170
settings_display_zoomscale_out
 MainWindow, 170
settings_general_check_for_updates
 MainWindow, 170
settings_general_current_tip
 MainWindow, 170
settings_general_icon_size
 MainWindow, 170
settings_general_icon_theme
 MainWindow, 170
settings_general_language
 MainWindow, 170
settings_general_mdi_bg_color
 MainWindow, 170
settings_general_mdi_bg_logo
 MainWindow, 170
settings_general_mdi_bg_texture
 MainWindow, 170
settings_general_mdi_bg_use_color
 MainWindow, 170
settings_general_mdi_bg_use_logo
 MainWindow, 170
settings_general_mdi_bg_use_texture
 MainWindow, 170
settings_general_system_help_browser
 MainWindow, 170
settings_general_tip_of_the_day
 MainWindow, 170
settings_grid_center_on_origin
 MainWindow, 170
settings_grid_center_x
 MainWindow, 171
settings_grid_center_y
 MainWindow, 171
settings_grid_color
 MainWindow, 171
settings_grid_color_match_crosshair
 MainWindow, 171
settings_grid_load_from_file
 MainWindow, 171
settings_grid_show_on_load
 MainWindow, 171
settings_grid_show_origin
 MainWindow, 171
settings_grid_size_radius
 MainWindow, 171
settings_grid_size_x
 MainWindow, 171

settings_grid_size_y
 MainWindow, 171
settings_grid_spacing_angle
 MainWindow, 171
settings_grid_spacing_radius
 MainWindow, 171
settings_grid_spacing_x
 MainWindow, 171
settings_grid_spacing_y
 MainWindow, 171
settings_grid_type
 MainWindow, 171
settings_lwt_default_lwt
 MainWindow, 171
settings_lwt_real_render
 MainWindow, 171
settings_lwt_show_lwt
 MainWindow, 171
settings_opensave_custom_filter
 MainWindow, 172
settings_opensave_open_format
 MainWindow, 172
settings_opensave_open_thumbnail
 MainWindow, 172
settings_opensave_recent_directory
 MainWindow, 172
settings_opensave_recent_list_of_files
 MainWindow, 172
settings_opensave_recent_max_files
 MainWindow, 172
settings_opensave_save_format
 MainWindow, 172
settings_opensave_save_thumbnail
 MainWindow, 172
settings_opensave_trim_dst_num_jumps
 MainWindow, 172
settings_printing_default_device
 MainWindow, 172
settings_printing_disable_bg
 MainWindow, 172
settings_printing_use_last_device
 MainWindow, 172
settings_prompt_bg_color
 MainWindow, 172
settings_prompt_font_family
 MainWindow, 172
settings_prompt_font_size
 MainWindow, 172
settings_prompt_font_style
 MainWindow, 172
settings_prompt_save_history
 MainWindow, 172
settings_prompt_save_history_as_html
 MainWindow, 172
settings_prompt_save_history_filename
 MainWindow, 173
settings_prompt_text_color
 MainWindow, 173

settings_qsnap_aperture_size
 MainWindow, 173
settings_qsnap_apparent
 MainWindow, 173
settings_qsnap_center
 MainWindow, 173
settings_qsnap_enabled
 MainWindow, 173
settings_qsnap_endpoint
 MainWindow, 173
settings_qsnap_extension
 MainWindow, 173
settings_qsnap_insertion
 MainWindow, 173
settings_qsnap_intersection
 MainWindow, 173
settings_qsnap_locator_color
 MainWindow, 173
settings_qsnap_locator_size
 MainWindow, 173
settings_qsnap_midpoint
 MainWindow, 173
settings_qsnap_nearest
 MainWindow, 173
settings_qsnap_node
 MainWindow, 173
settings_qsnap_parallel
 MainWindow, 173
settings_qsnap_perpendicular
 MainWindow, 173
settings_qsnap_quadrant
 MainWindow, 173
settings_qsnap_tangent
 MainWindow, 174
settings_ruler_color
 MainWindow, 174
settings_ruler_metric
 MainWindow, 174
settings_ruler_pixel_size
 MainWindow, 174
settings_ruler_show_on_load
 MainWindow, 174
settings_selection_coolgrip_color
 MainWindow, 174
settings_selection_grip_size
 MainWindow, 174
settings_selection_hotgrip_color
 MainWindow, 174
settings_selection_mode_pickadd
 MainWindow, 174
settings_selection_mode_pickdrag
 MainWindow, 174
settings_selection_mode_pickfirst
 MainWindow, 174
settings_selection_pickbox_size
 MainWindow, 174
settings_text_angle
 MainWindow, 174

settings_text_font
 MainWindow, 174
settings_text_size
 MainWindow, 174
settings_text_style_bold
 MainWindow, 174
settings_text_style_italic
 MainWindow, 174
settings_text_style_overline
 MainWindow, 174
settings_text_style_strikeout
 MainWindow, 175
settings_text_style_underline
 MainWindow, 175
settingsDialog
 MainWindow, 164
SettingsDir
 mainwindow-settings.cpp, 345
settingsGrid
 StatusBarButton, 258
settingsLwt
 StatusBarButton, 258
settingsMenu
 MainWindow, 175
settingsOrtho
 StatusBarButton, 258
SettingsPath
 mainwindow-settings.cpp, 345
settingsPolar
 StatusBarButton, 258
settingsPrompt
 MainWindow, 164
settingsQSnap
 StatusBarButton, 258
settingsQTrack
 StatusBarButton, 258
settingsRuler
 StatusBarButton, 258
settingsSnap
 StatusBarButton, 258
setUndoCleanIcon
 MainWindow, 164
setViewBackgroundColor
 MdiWindow, 185
setViewCrossHairColor
 MdiWindow, 185
setViewGridColor
 MdiWindow, 185
setViewRulerColor
 MdiWindow, 185
setViewSelectBoxColors
 MdiWindow, 185
sew, 17, 503
sewDecode
 format_sew.c, 503
shape
 BaseObject, 71
shift_held

 Settings_, 232
shiftKeyPressedState
 MainWindow, 175
shiftPressed
 CmdPrompt, 80
 CmdPromptInput, 88
shiftReleased
 CmdPrompt, 80
 CmdPromptInput, 89
shine_color
 Settings_, 233
shortcut
 Action__, 55
show_about_dialog
 Settings_, 233
show_details_dialog
 Settings_, 233
show_editor
 Settings_, 233
show_open_file_dialog
 Settings_, 233
show_settings_editor
 Settings_, 233
showGroups
 PropertyEditor, 210
showOneType
 PropertyEditor, 210
showScrollBars
 View, 290
showSettings
 CmdPrompt, 80
 CmdPromptInput, 89
showViewScrollBars
 MdiWindow, 185
shv, 17, 504
shv_thread
 embroidery.h, 387
shvDecode
 format_shv.c, 504
shvDecodeShort
 format_shv.c, 504
shvThreadCount
 embroidery.h, 406
 thread-color.c, 547
shvThreads
 embroidery.h, 406
 thread-color.c, 547
side1
 EmbSatinOutline_, 117
side2
 EmbSatinOutline_, 117
Sierra Expanded, 441, 486
Sigma_Polyester
 embroidery.h, 387
signalMapper
 PropertyEditor, 211
signature
 _bcf_file_header, 52

sigVersion
 ThredHeader_, 268
 simplify_path
 mainwindow.cpp, 349
 simulate
 EmbView_, 122
 simulation_start
 EmbView_, 122
 Singer, 476, 514
 SINGLE_LINE_TEXT_MODE_JUSTIFY
 embroidermodder.h, 306
 SINGLE_LINE_TEXT_MODE_JUSTIFY_
 mainwindow.cpp, 350
 SINGLE_LINE_TEXT_MODE_RAPID
 embroidermodder.h, 306
 SINGLE_LINE_TEXT_MODE_RAPID_
 mainwindow.cpp, 350
 SINGLE_LINE_TEXT_MODE_SETFONT
 embroidermodder.h, 306
 SINGLE_LINE_TEXT_MODE_SETFONT_
 mainwindow.cpp, 350
 SINGLE_LINE_TEXT_MODE_SETGEOM
 embroidermodder.h, 306
 SINGLE_LINE_TEXT_MODE_SETGEOM_
 mainwindow.cpp, 350
 sizeHint
 MdiWindow, 185
 sizeOfChainingEntryAtEndOfDifatSector
 main.c, 540
 sizeOfDifatEntry
 main.c, 540
 sizeOfDirectoryEntry
 main.c, 540
 sizeOfFatEntry
 main.c, 540
 sliderQSnapApertureSizeValueChanged
 Settings_Dialog, 246
 sliderQSnapLocatorSizeValueChanged
 Settings_Dialog, 246
 sliderSelectionGripSizeValueChanged
 Settings_Dialog, 247
 sliderSelectionPickBoxSizeValueChanged
 Settings_Dialog, 247
 snap_mode
 EmbView_, 122
 SNOWFLAKE_MODE_NUM_POINTS
 embroidermodder.h, 306
 SNOWFLAKE_MODE_XSCALE
 embroidermodder.h, 306
 SNOWFLAKE_MODE_YSCALE
 embroidermodder.h, 306
 someInt
 SubDescriptor_, 260
 someNum
 SubDescriptor_, 260
 someOtherInt
 SubDescriptor_, 260
 SPARE_RUBBER_OFF
 embroidermodder.h, 306
 SPARE_RUBBER_PATH
 embroidermodder.h, 306
 SPARE_RUBBER_POLYGON
 embroidermodder.h, 306
 SPARE_RUBBER_POLYLINE
 embroidermodder.h, 306
 SPARE_RUBBER_VALUES
 embroidermodder.h, 306
 spareRubber
 View, 290
 spareRubberList
 View, 294
 spinBoxDisplaySelectBoxAlphaValueChanged
 Settings_Dialog, 247
 spinBoxGridCenterXValueChanged
 Settings_Dialog, 247
 spinBoxGridCenterYValueChanged
 Settings_Dialog, 247
 spinBoxGridSizeRadiusValueChanged
 Settings_Dialog, 247
 spinBoxGridSizeXValueChanged
 Settings_Dialog, 247
 spinBoxGridSizeYValueChanged
 Settings_Dialog, 247
 spinBoxGridSpacingAngleValueChanged
 Settings_Dialog, 247
 spinBoxGridSpacingRadiusValueChanged
 Settings_Dialog, 247
 spinBoxGridSpacingXValueChanged
 Settings_Dialog, 247
 spinBoxGridSpacingYValueChanged
 Settings_Dialog, 247
 spinBoxPromptFontSizeValueChanged
 Settings_Dialog, 247
 spinBoxRecentMaxFilesValueChanged
 Settings_Dialog, 248
 spinBoxRulerPixelSizeValueChanged
 Settings_Dialog, 248
 spinBoxTrimDstNumJumpsValueChanged
 Settings_Dialog, 248
 spinBoxZoomScaleInValueChanged
 Settings_Dialog, 248
 spinBoxZoomScaleOutValueChanged
 Settings_Dialog, 248
 spline
 EmbGeometry_, 111
 sst, 17, 445, 505
 STAR_MODE_CENTER_PT
 embroidermodder.h, 306
 STAR_MODE_CENTER_PT_
 mainwindow.cpp, 350
 STAR_MODE_NUM_POINTS
 embroidermodder.h, 306
 STAR_MODE_NUM_POINTS_
 mainwindow.cpp, 350
 STAR_MODE_RAD_INNER
 embroidermodder.h, 306

STAR_MODE_RAD_INNER_
mainwindow.cpp, 350
STAR_MODE_RAD_OUTER
embroidermodder.h, 306
STAR_MODE_RAD_OUTER_
mainwindow.cpp, 350
start
 EmbArc_, 104
 EmbBezier_, 105
 EmbLine_, 113
startBlinking
 CmdPrompt, 81
startCommand
 CmdPrompt, 81
 CmdPromptInput, 89
startGripping
 View, 290
startingSectorLocation
 _bcf_directory_entry, 49
startResizeHistory
 CmdPromptHistory, 84
startResizingTheHistory
 CmdPrompt, 81
stateBits
 _bcf_directory_entry, 49
StatusBar, 255
 setMouseCoord, 256
 StatusBar, 256
 statusBarGridButton, 256
 statusBarLwtButton, 256
 statusBarMouseCoord, 256
 statusBarOrthoButton, 256
 statusBarPolarButton, 256
 statusBarQSnapButton, 256
 statusBarQTrackButton, 256
 statusBarRulerButton, 256
 statusBarSnapButton, 257
statusbar
 MainWindow, 175
 StatusBarButton, 259
StatusBarButton, 257
 contextMenuEvent, 258
 disableLwt, 258
 disableReal, 258
 enableLwt, 258
 enableReal, 258
 mainWin, 259
 settingsGrid, 258
 settingsLwt, 258
 settingsOrtho, 258
 settingsPolar, 258
 settingsQSnap, 258
 settingsQTrack, 258
 settingsRuler, 258
 settingsSnap, 258
 statusbar, 259
 StatusBarButton, 257
 toggleGrid, 258
toggleLwt, 259
toggleOrtho, 259
togglePolar, 259
toggleQSnap, 259
toggleQTrack, 259
toggleRuler, 259
toggleSnap, 259
statusBarGridButton
 StatusBar, 256
statusBarLwtButton
 StatusBar, 256
statusBarMouseCoord
 StatusBar, 256
statusBarOrthoButton
 StatusBar, 256
statusBarPolarButton
 StatusBar, 256
statusBarQSnapButton
 StatusBar, 256
statusBarQTrackButton
 StatusBar, 256
statusBarRulerButton
 StatusBar, 256
statusBarSnapButton
 StatusBar, 257
statustip
 Action__, 55
stitch
 EmbArray_, 104
 EmbGeometry_, 111
stitch_list
 EmbPattern_, 115
stitch_time
 Settings_, 233
stitchesJump
 EmbDetailsDialog, 107
stitchesReal
 EmbDetailsDialog, 108
stitchesTotal
 EmbDetailsDialog, 108
stitchesTrim
 EmbDetailsDialog, 108
stitchGranularity
 ThredExtension_, 268
STOP
 embroidery.h, 387
stopBlinking
 CmdPrompt, 81
 CmdPromptInput, 89
stopGripping
 View, 290
stopResizeHistory
 CmdPromptHistory, 84
stopResizingTheHistory
 CmdPrompt, 81
streamSize
 _bcf_directory_entry, 49
streamSizeHigh

_bcf_directory_entry, 49
 stringInArray
 embroidery_internal.h, 447
 main.c, 539
 stringVal
 VipHeader_, 296
 stub_implement
 MainWindow, 164
 stub_testing
 MainWindow, 164
 stx, 505
 stxColor
 StxThread_, 260
 stxReadThread
 format_stx.c, 505
 StxThread
 embroidery_internal.h, 430
 StxThread_, 259
 colorCode, 260
 colorName, 260
 sectionName, 260
 stxColor, 260
 subDescriptors, 260
 styleHash
 CmdPrompt, 82
 SubDescriptor
 embroidery_internal.h, 430
 SubDescriptor_, 260
 colorCode, 260
 colorName, 260
 someInt, 260
 someNum, 260
 someOtherInt, 260
 subDescriptors
 StxThread_, 260
 subPathList
 TextSingleObject, 266
 Sulky_Rayon
 embroidery.h, 387
 Sunstar, 445, 505
 svg, 17, 506
 SVG_ATTRIBUTE
 embroidery_internal.h, 428
 SVG_CATCH_ALL
 embroidery_internal.h, 428
 SVG_Colors
 embroidery.h, 387
 SVG_CREATOR_EMBROIDERMODDER
 embroidery_internal.h, 428
 SVG_CREATOR_ILLUSTRATOR
 embroidery_internal.h, 428
 SVG_CREATOR_INKSCAPE
 embroidery_internal.h, 428
 SVG_CREATOR_NULL
 embroidery_internal.h, 429
 SVG_ELEMENT
 embroidery_internal.h, 429
 SVG_EXPECT_ATTRIBUTE
 embroidery_internal.h, 429
 SVG_EXPECT_ELEMENT
 embroidery_internal.h, 429
 SVG_EXPECT_NULL
 embroidery_internal.h, 429
 SVG_EXPECT_VALUE
 embroidery_internal.h, 429
 SVG_MEDIA_PROPERTY
 embroidery_internal.h, 429
 SVG_NULL
 embroidery_internal.h, 429
 SVG_PROPERTY
 embroidery_internal.h, 429
 SvgAttribute
 embroidery_internal.h, 430
 SvgAttribute_, 261
 name, 261
 value, 261
 svgCreator
 format_svg.c, 506
 svgExpect
 format_svg.c, 506
 svgMultiValue
 format_svg.c, 506
 t01, 445, 507
 t09, 445, 507
 table
 Huffman, 124
 table_width
 Huffman, 124
 tabPressed
 CmdPrompt, 81
 CmdPromptInput, 89
 tabWidget
 Settings_Dialog, 255
 Tajima, 480
 tap, 17, 508
 tempArcObj
 PropertyEditor, 211
 tempBaseObj
 View, 294
 tempBlockObj
 PropertyEditor, 211
 tempCircleObj
 PropertyEditor, 211
 tempDimAlignedObj
 PropertyEditor, 212
 tempDimAngularObj
 PropertyEditor, 212
 tempDimArcLenObj
 PropertyEditor, 212
 tempDimDiamObj
 PropertyEditor, 212
 tempDimLeaderObj
 PropertyEditor, 212
 tempDimLinearObj
 PropertyEditor, 212
 tempDimOrdObj

PropertyEditor, 212
tempDimRadiusObj
 PropertyEditor, 212
tempEllipseArcObj
 PropertyEditor, 212
tempEllipseObj
 PropertyEditor, 212
tempHatchObj
 PropertyEditor, 212
tempImageObj
 PropertyEditor, 212
tempInflLineObj
 PropertyEditor, 212
tempLineObj
 PropertyEditor, 212
tempPathObj
 PropertyEditor, 212
tempPointObj
 PropertyEditor, 212
tempPolygonObj
 PropertyEditor, 212
tempPolylineObj
 PropertyEditor, 212
tempRayObj
 PropertyEditor, 213
tempRectObj
 PropertyEditor, 213
tempSplineObj
 PropertyEditor, 213
tempTextMultiObj
 PropertyEditor, 213
tempTextSingleObj
 PropertyEditor, 213
testEmbCircle
 embroidery_internal.h, 447
testEmbCircle_2
 embroidery_internal.h, 447
testEmbFormat
 embroidery_internal.h, 447
testGeomArc
 embroidery_internal.h, 447
testing
 Settings_, 233
testMain
 embroidery.h, 405
testTangentPoints
 embroidery_internal.h, 447
testThreadColor
 embroidery_internal.h, 447
text
 EmbTextMulti_, 118
 EmbTextSingle_, 118
 UiObject_, 271
text.c
 textSingle_gripEdit, 524
 textSingle_mouseSnapPoint, 524
 textSingle_paint, 525
 textSingle_setJustify, 525
textSingleSetTextBackward, 525
textSingleSetTextBold, 525
textSingleSetTextFont, 525
textSingleSetTextItalic, 525
textSingleSetTextOverline, 525
textSingleSetTextSize, 525
textSingleSetTextStrikeOut, 525
textSingleSetTextStyle, 525
textSingleSetTextUnderline, 525
textSingleSetTextUpsideDown, 525
textSingle_updateRubber, 525
text_angle
 EmbView_, 122
 Settings_, 233
text_font
 EmbView_, 122
 Settings_, 233
text_size
 EmbView_, 122
 Settings_, 233
text_style_bold
 EmbView_, 122
 Settings_, 233
text_style_italic
 EmbView_, 122
 Settings_, 233
text_style_overline
 EmbView_, 122
 Settings_, 233
text_style_strikeout
 EmbView_, 122
 Settings_, 233
text_style_underline
 EmbView_, 122
 Settings_, 233
textAngle
 MainWindow, 164
textBold
 MainWindow, 164
textFont
 MainWindow, 164
 UiObject_, 271
textFontSelector
 MainWindow, 175
textFontSelectorCurrentFontChanged
 MainWindow, 164
textHeight
 UiObject_, 271
textItalic
 MainWindow, 165
textJustify
 UiObject_, 271
textOverline
 MainWindow, 165
textRotation
 UiObject_, 271
textSingle_gripEdit
 text.c, 524

textSingle_mouseSnapPoint
 text.c, 524
 textSingle_paint
 text.c, 525
 textSingle_setJustify
 text.c, 525
 textSingle_setTextBackward
 text.c, 525
 textSingle_setTextBold
 text.c, 525
 textSingle_setTextFont
 text.c, 525
 textSingle_setTextItalic
 text.c, 525
 textSingle_setTextOverline
 text.c, 525
 textSingle_setTextSize
 text.c, 525
 textSingle_setTextStrikeOut
 text.c, 525
 textSingle_setTextStyle
 text.c, 525
 textSingle_setTextUnderline
 text.c, 525
 textSingle_setTextUpsideDown
 text.c, 525
 textSingle_updateRubber
 text.c, 525
 TextSingleObject, 261
 ~TextSingleObject, 264
 allGripPoints, 264
 gripEdit, 264
 init, 264
 mouseSnapPoint, 264
 objectPos, 264
 objectSavePathList, 264
 objectTextJustifyList, 264
 objectX, 264
 objectY, 264
 objText, 266
 objTextBackward, 266
 objTextBold, 266
 objTextFont, 266
 objTextItalic, 266
 objTextJustify, 266
 objTextOverline, 266
 objTextPath, 266
 objTextSize, 266
 objTextStrikeOut, 267
 objTextUnderline, 267
 objTextUpsideDown, 267
 paint, 264
 setObjectPos, 264, 265
 setObjectText, 265
 setObjectTextBackward, 265
 setObjectTextBold, 265
 setObjectTextFont, 265
 setObjectTextItalic, 265
 setObjectTextJustify, 265
 setObjectTextOverline, 265
 setObjectTextSize, 265
 setObjectTextStrikeOut, 265
 setObjectTextStyle, 265
 setObjectTextUnderline, 265
 setObjectTextUpsideDown, 265
 setObjectX, 266
 setObjectY, 266
 subPathList, 266
 TextSingleObject, 263
 Type, 263
 type, 266
 updateRubber, 266
 vulcanize, 266
 textSize
 MainWindow, 165
 textSizeSelector
 MainWindow, 175
 textSizeSelectorIndexChanged
 MainWindow, 165
 textStrikeOut
 MainWindow, 165
 textUnderline
 MainWindow, 165
 texture_list
 Settings_, 233
 thr, 446, 509
 thread
 EmbArray_, 104
 EmbGeometry_, 111
 thread-color.c
 _dxIColorTable, 546
 brand_codes, 546
 brand_codes_files, 546
 husThreads, 546
 jetThreads, 546
 pcmThreads, 546
 pecThreadCount, 546
 pecThreads, 547
 shvThreadCount, 547
 shvThreads, 547
 threadColor, 546
 threadColorName, 546
 threadColorNum, 546
 thread_color
 embroidery.h, 390
 thread_color_
 hex_code, 267
 manufacturer_code, 267
 name, 267
 thread_list
 EmbPattern_, 115
 ThreadArt_Polyester
 embroidery.h, 387
 ThreadArt_Rayon
 embroidery.h, 387
 threadColor

embroidery.h, 405
thread-color.c, 546
threadColorName
 embroidery.h, 405
 thread-color.c, 546
threadColorNum
 embroidery.h, 405
 thread-color.c, 546
ThreaDelight_Polyester
 embroidery.h, 388
threadLength
 _vp3Hoop, 54
ThreadWorks, 446, 509
ThredExtension
 embroidery_internal.h, 430
ThredExtension_, 267
 auxFormat, 267
 creatorName, 268
 hoopX, 268
 hoopY, 268
 modifierName, 268
 reserved, 268
 stitchGranularity, 268
ThredHeader
 embroidery_internal.h, 430
ThredHeader_, 268
 hoopSize, 268
 length, 268
 numStiches, 268
 reserved, 268
 sigVersion, 268
threshold_method
 fill.c, 467
Tick
 DimLeaderObject, 95
tick_depth
 Settings_, 233
ticks_color
 Settings_, 234
tile
 MdiArea, 178
tip_of_the_day
 Settings_, 234
tipOfTheDay
 MainWindow, 165
tmpHeight
 CmdPromptHistory, 84
to_EmbVector
 embroidermodder.h, 307
to_open
 Settings_, 234
to_QPointF
 embroidermodder.h, 307
to_string_vector
 mainwindow-settings.cpp, 345
toCenter
 UndoableNavCommand, 277
Todo
 mainwindow.cpp, 349
toggleGrid
 MainWindow, 165
 StatusBarButton, 258
 View, 290
toggleLwt
 MainWindow, 165
 StatusBarButton, 259
 View, 290
toggleOrtho
 StatusBarButton, 259
 View, 290
togglePickAddMode
 PropertyEditor, 210
togglePolar
 StatusBarButton, 259
 View, 290
toggleQSnap
 StatusBarButton, 259
 View, 290
toggleQTrack
 StatusBarButton, 259
 View, 290
toggleReal
 View, 290
toggleRuler
 MainWindow, 165
 StatusBarButton, 259
 View, 291
toggleSnap
 StatusBarButton, 259
 View, 291
toolbar_layout
 mainwindow.cpp, 350
toolbarEdit
 MainWindow, 175
toolbarFile
 MainWindow, 175
toolbarHash
 MainWindow, 175
toolbarHelp
 MainWindow, 175
toolbarIcon
 MainWindow, 175
toolbarLayer
 MainWindow, 175
toolbarPan
 MainWindow, 175
toolbarPrompt
 MainWindow, 175
toolbarProperties
 MainWindow, 175
toolbarText
 MainWindow, 175
toolbarView
 MainWindow, 175
toolbarZoom
 MainWindow, 176

toolButtonArcClockwise
 property-editor.cpp, 363
toolButtonBlockX
 property-editor.cpp, 363
toolButtonBlockY
 property-editor.cpp, 364
toolButtonCircleArea
 property-editor.cpp, 364
toolButtonCircleCenterX
 property-editor.cpp, 364
toolButtonCircleCenterY
 property-editor.cpp, 364
toolButtonCircleCircumference
 property-editor.cpp, 364
toolButtonCircleDiameter
 property-editor.cpp, 364
toolButtonCircleRadius
 property-editor.cpp, 364
toolButtonEllipseCenterX
 property-editor.cpp, 364
toolButtonEllipseCenterY
 property-editor.cpp, 364
toolButtonEllipseDiameterMajor
 property-editor.cpp, 364
toolButtonEllipseDiameterMinor
 property-editor.cpp, 364
toolButtonEllipseRadiusMajor
 property-editor.cpp, 364
toolButtonEllipseRadiusMinor
 property-editor.cpp, 364
toolButtonImageHeight
 property-editor.cpp, 364
toolButtonImageName
 property-editor.cpp, 364
toolButtonImagePath
 property-editor.cpp, 364
toolButtonImageWidth
 property-editor.cpp, 364
toolButtonImageX
 property-editor.cpp, 364
toolButtonImageY
 property-editor.cpp, 365
toolButtonInfiniteLineVectorX
 property-editor.cpp, 365
toolButtonInfiniteLineVectorY
 property-editor.cpp, 365
toolButtonInfiniteLineX2
 property-editor.cpp, 365
toolButtonInfiniteLineY1
 property-editor.cpp, 365
toolButtonInfiniteLineY2
 property-editor.cpp, 365
toolButtonLineAngle
 property-editor.cpp, 365
toolButtonLineDeltaX
 property-editor.cpp, 365
toolButtonLineDeltaY
 property-editor.cpp, 365
toolButtonLineEndX
 property-editor.cpp, 365
toolButtonLineEndY
 property-editor.cpp, 365
toolButtonLineLength
 property-editor.cpp, 365
toolButtonLineStartX
 property-editor.cpp, 365
toolButtonLineStartY
 property-editor.cpp, 365
toolButtonPathArea
 property-editor.cpp, 365
toolButtonPathClosed
 property-editor.cpp, 365
toolButtonPathLength
 property-editor.cpp, 365
toolButtonPathVertexNum
 property-editor.cpp, 365
toolButtonPathVertexX
 property-editor.cpp, 366
toolButtonPathVertexY
 property-editor.cpp, 366
toolButtonPickAdd
 PropertyEditor, 213
toolButtonPointX
 property-editor.cpp, 366
toolButtonPointY
 property-editor.cpp, 366
toolButtonPolygonCenterX
 property-editor.cpp, 366
toolButtonPolygonCenterY
 property-editor.cpp, 366
toolButtonPolygonDiameterSide
 property-editor.cpp, 366
toolButtonPolygonDiameterVertex
 property-editor.cpp, 366
toolButtonPolygonInteriorAngle
 property-editor.cpp, 366
toolButtonPolygonRadiusSide
 property-editor.cpp, 366
toolButtonPolygonRadiusVertex
 property-editor.cpp, 366
toolButtonPolylineArea
 property-editor.cpp, 366
toolButtonPolylineClosed
 property-editor.cpp, 366
toolButtonPolylineLength
 property-editor.cpp, 366
toolButtonPolylineVertexNum
 property-editor.cpp, 366
toolButtonPolylineVertexX
 property-editor.cpp, 366
toolButtonPolylineVertexY
 property-editor.cpp, 366
toolButtonQSelect
 PropertyEditor, 213
toolButtonRayVectorX
 property-editor.cpp, 366

toolButtonRayVectorY
 property-editor.cpp, 367

toolButtonRayX1
 property-editor.cpp, 367

toolButtonRayX2
 property-editor.cpp, 367

toolButtonRayY1
 property-editor.cpp, 367

toolButtonRayY2
 property-editor.cpp, 367

toolButtonRectangleArea
 property-editor.cpp, 367

toolButtonRectangleCorner1X
 property-editor.cpp, 367

toolButtonRectangleCorner1Y
 property-editor.cpp, 367

toolButtonRectangleCorner2X
 property-editor.cpp, 367

toolButtonRectangleCorner2Y
 property-editor.cpp, 367

toolButtonRectangleCorner3X
 property-editor.cpp, 367

toolButtonRectangleCorner3Y
 property-editor.cpp, 367

toolButtonRectangleCorner4X
 property-editor.cpp, 367

toolButtonRectangleCorner4Y
 property-editor.cpp, 367

toolButtonRectangleHeight
 property-editor.cpp, 367

toolButtonRectangleWidth
 property-editor.cpp, 367

toolButtons
 property-editor.cpp, 367

toolButtonTextMultiX
 property-editor.cpp, 367

toolButtonTextMultiY
 property-editor.cpp, 368

toolButtonTextSingleBackward
 property-editor.cpp, 368

toolButtonTextSingleContents
 property-editor.cpp, 368

toolButtonTextSingleFont
 property-editor.cpp, 368

toolButtonTextSingleHeight
 property-editor.cpp, 368

toolButtonTextSingleJustify
 property-editor.cpp, 368

toolButtonTextSingleRotation
 property-editor.cpp, 368

toolButtonTextSingleUpsideDown
 property-editor.cpp, 368

toolButtonTextSingleX
 property-editor.cpp, 368

toolButtonTextSingleY
 property-editor.cpp, 368

tooltip
 Action___, 55

top
 _vp3Hoop, 54

EmbRect_, 117

hoop_padding, 123

top2
 _vp3Hoop, 54

toPolyline
 SaveObject, 222

toTransform
 UndoableNavCommand, 277

Toyota, 472

transactionSignatureNumber
 _bcf_file_header, 52

translation_table
 mainwindow.cpp, 350

treeView
 LayerManager, 130

TRIM
 embroidery.h, 388

txt, 446, 509

Type
 ArcObject, 59

BaseObject, 68

CircleObject, 73

DimLeaderObject, 95

EllipseObject, 100

ImageObject, 126

LineObject, 133

PathObject, 189

PointObject, 193

PolygonObject, 197

PolylineObject, 201

RectObject, 215

TextSingleObject, 263

type
 ArcObject, 65

BaseObject, 71

CircleObject, 75

DimLeaderObject, 97

EllipseObject, 102

EmbArray_, 105

EmbFormatList_, 109

EmbGeometry_, 111

ImageObject, 127

LineObject, 135

PathObject, 190

PointObject, 194

PolygonObject, 198

PolylineObject, 202

RectObject, 217

TextSingleObject, 266

UiObject_, 271

u00, 446, 510

u01, 17, 446, 510

ui_mode
 EmbView_, 122

UiMode
 embroidermodder.h, 306

UiObject
 embryoermodder.h, 302

UiObject_, 269
 center, 269
 color, 270
 command, 270
 controlPointLabels, 270
 controlPoints, 270
 firstRun, 270
 fname, 270
 id, 270
 maxPoints, 270
 minPoints, 270
 mode, 270
 n_controlPoints, 270
 numPoints, 270
 object_index, 270
 path_desc, 270
 pattern_index, 270
 rotation, 270
 scale, 270
 selectable, 270
 text, 271
 textFont, 271
 textHeight, 271
 textJustify, 271
 textRotation, 271
 type, 271

undo
 MainWindow, 165
 UndoableAddCommand, 271
 UndoableDeleteCommand, 272
 UndoableGripEditCommand, 273
 UndoableMirrorCommand, 274
 UndoableMoveCommand, 275
 UndoableNavCommand, 276
 UndoableRotateCommand, 278
 UndoableScaleCommand, 279
 UndoEditor, 280

undo_history
 EmbView_, 122

UndoableAddCommand, 271
 gview, 272
 object, 272
 redo, 271
 undo, 271

UndoableAddCommand, 271

UndoableDeleteCommand, 272
 gview, 272
 object, 272
 redo, 272
 undo, 272

UndoableDeleteCommand, 272

UndoableGripEditCommand, 273
 after, 273
 before, 273
 gview, 273
 object, 273

redo, 273
 undo, 273
 UndoableGripEditCommand, 273

UndoableMirrorCommand, 274
 gview, 274
 mirror, 274
 mirrorLine, 274
 object, 274
 redo, 274
 undo, 274
 UndoableMirrorCommand, 274

UndoableMoveCommand, 275
 dx, 275
 dy, 275
 gview, 275
 object, 275
 redo, 275
 undo, 275
 UndoableMoveCommand, 275

UndoableNavCommand, 276
 done, 276
 fromCenter, 276
 fromTransform, 276
 gview, 277
 id, 276
 mergeWith, 276
 navType, 277
 redo, 276
 toCenter, 277
 toTransform, 277
 undo, 276
 UndoableNavCommand, 276

UndoableRotateCommand, 277
 angle, 278
 gview, 278
 object, 278
 pivotX, 278
 pivotY, 278
 redo, 277
 rotate, 277
 undo, 278
 UndoableRotateCommand, 277

UndoableScaleCommand, 278
 dx, 279
 dy, 279
 factor, 279
 gview, 279
 object, 279
 redo, 279
 undo, 279

UndoableScaleCommand, 278

UndoEditor, 279
 ~UndoEditor, 280
 addStack, 280
 canRedo, 280
 canUndo, 280
 focusWidget, 280
 iconDir, 281

iconSize, 281
redo, 280
redoText, 280
undo, 280
UndoEditor, 280
undoGroup, 281
undoText, 280
undoView, 281
updateCleanIcon, 280
undoGroup
 UndoEditor, 281
UndoHistory
 embroidermodder.h, 302
UndoHistory_, 281
 data, 281
 position, 281
undoPressed
 CmdPrompt, 81
 CmdPromptInput, 89
undoStack
 View, 295
undoText
 UndoEditor, 280
undoView
 UndoEditor, 281
unknown
 VipHeader_, 296
unknown2
 _vp3Hoop, 54
unknown3
 _vp3Hoop, 54
unknown4
 _vp3Hoop, 54
updateAllViewBackgroundColors
 MainWindow, 165
updateAllViewCrossHairColors
 MainWindow, 165
updateAllViewGridColors
 MainWindow, 165
updateAllViewRulerColors
 MainWindow, 165
updateAllViewScrollBars
 MainWindow, 165
updateAllViewSelectBoxColors
 MainWindow, 166
updateArcRect
 ArcObject, 65
updateCleanIcon
 UndoEditor, 280
updateColorLinetypeLineweight
 MdiWindow, 185
updateComboBoxBoolIfVaries
 PropertyEditor, 210
updateComboBoxStrIfVaries
 PropertyEditor, 210
updateCurrentText
 CmdPromptInput, 89
updateFontComboBoxStrIfVaries
 PropertyEditor, 210
PropertyEditor, 210
updateLeader
 DimLeaderObject, 97
updateLineEditNumIfVaries
 PropertyEditor, 210
updateLineEditStrIfVaries
 PropertyEditor, 210
updateMenuToolbarStatusbar
 MainWindow, 166
updateMouseCoords
 View, 291
updatePath
 ArcObject, 66
 CircleObject, 75
 EllipseObject, 102
 ImageObject, 127
 PathObject, 190
 PolygonObject, 198
 PolylineObject, 202
 RectObject, 217
updatePickAddMode
 MainWindow, 166
updatePickAddModeButton
 PropertyEditor, 210
updateRubber
 ArcObject, 66
 CircleObject, 75
 DimLeaderObject, 97
 EllipseObject, 102
 ImageObject, 127
 LineObject, 135
 PathObject, 190
 PointObject, 194
 PolygonObject, 198
 PolylineObject, 203
 RectObject, 217
 TextSingleObject, 266
updateStyle
 CmdPrompt, 81
upPressed
 CmdPrompt, 81
 CmdPromptInput, 89
usage
 embroidermodder.cpp, 297
use_translation
 Settings_, 234
useBackgroundColor
 MdiArea, 178
useBackgroundLogo
 MdiArea, 178
useBackgroundTexture
 MdiArea, 179
useColor
 MdiArea, 179
useLogo
 MdiArea, 179
useTexture
 MdiArea, 179

validFileFormat
 MainWindow, 166
validRGB
 mainwindow.cpp, 349
value
 SvgAttribute_, 261
vector
 EmbGeometry_, 111
vector.c
 embVector_add, 526
 embVector_angle, 526
 embVector_average, 526
 embVector_cross, 526
 embVector_distance, 527
 embVector_dot, 527
 embVector_length, 527
 embVector_multiply, 527
 embVector_normalize, 527
 embVector_relativeX, 527
 embVector_relativeY, 528
 embVector_subtract, 528
 embVector_transpose_product, 528
 embVector_unit, 528
version
 embroidermodder.cpp, 297
 Settings_, 234
View, 281
 ~View, 285
 addObject, 285
 addToRubberRoom, 285
 alignScenePointWithViewPoint, 285
 allowRubber, 285
 allowZoomIn, 285
 allowZoomOut, 285
 center, 285
 centerAt, 285
 clearRubberRoom, 285
 clearSelection, 285
 contextMenuEvent, 285
 copy, 285
 copySelected, 285
 cornerButtonClicked, 286
 createGrid, 286
 createGridIso, 286
 createGridPolar, 286
 createGridRect, 286
 createObjectList, 286
 createOrigin, 286
 createRulerTextPath, 286
 crosshairColor, 292
 crosshairSize, 292
 cut, 286
 cutCopyMousePoint, 292
 deleteObject, 286
 deletePressed, 286
 deleteSelected, 286
 disableMoveRapidFire, 286
 drawBackground, 286
 drawForeground, 286
 enableMoveRapidFire, 287
 enterEvent, 287
 escapePressed, 287
 getUndoStack, 287
 gridColor, 292
 gridPath, 292
 gripBaseObj, 292
 gripColorCool, 292
 gripColorHot, 292
 grippingActive, 292
 gripSize, 292
 gscene, 292
 hashDeletedObjects, 292
 isLwtEnabled, 287
 isRealEnabled, 287
 loadRulerSettings, 287
 mainWin, 292
 mirrorSelected, 287
 mouseDoubleClickEvent, 287
 mouseMoveEvent, 287
 mousePressEvent, 287
 mouseReleaseEvent, 287
 moveAction, 287
 movePoint, 292
 moveSelected, 287
 movingActive, 292
 numSelected, 288
 originPath, 292
 panDistance, 292
 panDown, 288
 panLeft, 288
 panningActive, 293
 panningPointActive, 293
 panningRealTimeActive, 293
 panPoint, 288
 panRealTime, 288
 panRight, 288
 panStart, 288
 panStartX, 293
 panStartY, 293
 panUp, 288
 paste, 288
 pasteDelta, 293
 pasteObjectItemGroup, 293
 pastingActive, 293
 pickBoxSize, 293
 pressPoint, 293
 previewActive, 293
 previewData, 293
 previewMode, 293
 previewObjectItemGroup, 293
 previewObjectList, 293
 previewOff, 288
 previewOn, 288
 previewPoint, 293
 qSnapActive, 293
 qsnapApertureSize, 293

qsnapLocatorColor, 294
qsnapLocatorSize, 294
qSnapToggle, 294
rapidMoveActive, 294
recalculateLimits, 288
releasePoint, 294
repeatAction, 288
rotateAction, 288
rotateSelected, 288
roundToMultiple, 289
rubberRoomList, 294
rulerColor, 294
rulerMetric, 294
rulerPixelSize, 294
scaleAction, 289
scaleSelected, 289
sceneGripPoint, 294
sceneMousePoint, 294
sceneMovePoint, 294
scenePressPoint, 294
sceneReleasePoint, 294
selectAll, 289
selectBox, 294
selectingActive, 294
selectionChanged, 289
setBackgroundColor, 289
setCornerButton, 289
setCrossHairColor, 289
setCrossHairSize, 289
setGridColor, 289
setRubberMode, 289
setRubberPoint, 289
setRubberText, 289
setRulerColor, 290
setSelectBoxColors, 290
showScrollBars, 290
spareRubber, 290
spareRubberList, 294
startGripping, 290
stopGripping, 290
tempBaseObj, 294
toggleGrid, 290
toggleLwt, 290
toggleOrtho, 290
togglePolar, 290
toggleQSnap, 290
toggleQTrack, 290
toggleReal, 290
toggleRuler, 291
toggleSnap, 291
undoStack, 295
updateMouseCoords, 291
View, 284
viewMousePoint, 295
vulcanizeObject, 291
vulcanizeRubberRoom, 291
wheelEvent, 291
willOverflowInt32, 291
willUnderflowInt32, 291
zoomExtents, 291
zoomIn, 291
zoomOut, 291
zoomSelected, 291
zoomToPoint, 291
zoomWindow, 291
zoomWindowActive, 295
view_toolbar
embroidermodder.h, 308
mainwindow-toolbars.cpp, 346
viewMenu
 MainWindow, 176
viewMousePoint
 View, 295
views
 mainwindow.cpp, 350
vip, 17, 406, 512
vipCompressData
 format_vip.c, 511
vipDecodeByte
 format_vip.c, 511
vipDecodeStitchType
 format_vip.c, 511
vipDecodingTable
 embroidery.h, 406
 format_vip.c, 511
vipDecompressData
 format_vip.c, 511
vipEncodeByte
 format_vip.c, 511
vipEncodeStitchType
 format_vip.c, 511
VipHeader
 embroidery_internal.h, 430
VipHeader_-, 295
 attributeOffset, 295
 colorLength, 295
 magicCode, 295
 negativeXHoopSize, 295
 negativeYHoopSize, 295
 numberOfColors, 295
 numberOfStitches, 295
 positiveXHoopSize, 296
 positiveYHoopSize, 296
 stringVal, 296
 unknown, 296
 xOffset, 296
 yOffset, 296
vp3, 17, 513
vp3Decode
 format_vp3.c, 512
vp3DecodeInt16
 format_vp3.c, 512
vp3Hoop
 embroidery_internal.h, 430
vp3PatchByteCount
 format_vp3.c, 512

vp3ReadHoopSection
 format_vp3.c, 512
 vp3ReadString
 format_vp3.c, 513
 vp3WriteString
 format_vp3.c, 513
 vp3WriteStringLen
 format_vp3.c, 513
 vulcanize
 ArcObject, 66
 BaseObject, 71
 CircleObject, 75
 DimLeaderObject, 97
 EllipseObject, 102
 ImageObject, 127
 LineObject, 135
 PathObject, 191
 PointObject, 194
 PolygonObject, 198
 PolylineObject, 203
 RectObject, 217
 TextSingleObject, 266
 vulcanizeObject
 View, 291
 vulcanizeRubberRoom
 View, 291

 whatsThisContextHelp
 MainWindow, 166
 wheelEvent
 View, 291
 WHITESPACE
 main.c, 540
 width
 _vp3Hoop, 54
 EmblImage_, 111
 willOverflowInt32
 View, 291
 willUnderflowInt32
 View, 291
 windowMenu
 MainWindow, 176
 windowMenuAboutToShow
 MainWindow, 166
 windowMenuActivated
 MainWindow, 166
 wizardTipOfDay
 MainWindow, 176
 write100
 embroidery_internal.h, 447
 format_100.c, 472
 write10o
 embroidery_internal.h, 447
 format_10o.c, 473
 write_24bit
 embroidery_internal.h, 447
 encoding.c, 463
 main.c, 539
 write_external_color_file
 EmbFormatList_, 109
 write_settings
 embroidermodder.h, 307
 writeArt
 embroidery_internal.h, 448
 format_art.c, 473
 writeBmc
 embroidery_internal.h, 448
 format_bmc.c, 474
 writeBro
 embroidery_internal.h, 448
 format_bro.c, 474
 writeCnd
 embroidery_internal.h, 448
 format_cnd.c, 475
 writeCol
 embroidery_internal.h, 448
 format_col.c, 476
 writeCsd
 embroidery_internal.h, 448
 format_csd.c, 477
 writeCsv
 embroidery_internal.h, 448
 format_csv.c, 478
 writeDat
 embroidery_internal.h, 448
 format_dat.c, 478
 writeDem
 embroidery_internal.h, 448
 format_dem.c, 479
 writeDsb
 embroidery_internal.h, 448
 format_dsb.c, 480
 writeDst
 embroidery_internal.h, 448
 format_dst.c, 482
 writeDsz
 embroidery_internal.h, 449
 format_dsz.c, 482
 writeDxf
 embroidery_internal.h, 449
 format_dxf.c, 483
 writeEdr
 embroidery_internal.h, 449
 format_edr.c, 483
 writeEmd
 embroidery_internal.h, 449
 format_emd.c, 484
 writeExp
 embroidery_internal.h, 449
 format_exp.c, 485
 writeExy
 embroidery_internal.h, 449
 format_exy.c, 485
 writeEys
 embroidery_internal.h, 449
 format_eys.c, 486
 writeFxy

embroidery_internal.h, 449
format_fxy.c, 486
writeGc
 embroidery_internal.h, 449
 format_gc.c, 487
writeGnc
 embroidery_internal.h, 449
 format_gnc.c, 487
writeGt
 embroidery_internal.h, 449
 format_gt.c, 488
writeHus
 embroidery_internal.h, 450
 format_hus.c, 489
writelImage
 format_pec.c, 497
 image.c, 529
writelnb
 embroidery_internal.h, 450
 format_inb.c, 489
writeInf
 embroidery_internal.h, 450
 format_inf.c, 490
writeJef
 embroidery_internal.h, 450
 format_jef.c, 491
writeKsm
 embroidery_internal.h, 450
 format_ksm.c, 491
writeMax
 embroidery_internal.h, 450
 format_max.c, 492
writeMit
 embroidery_internal.h, 450
 format_mit.c, 493
writeNew
 embroidery_internal.h, 450
 format_new.c, 493
writeOfm
 embroidery_internal.h, 450
 format_ofm.c, 494
writePcd
 embroidery_internal.h, 450
 format_pcd.c, 495
writePcm
 embroidery_internal.h, 450
 format_pcm.c, 495
writePcq
 embroidery_internal.h, 451
 format_pcq.c, 496
writePcs
 embroidery_internal.h, 451
 format_pcs.c, 496
writePec
 embroidery_internal.h, 451
 format_pec.c, 497
writePecStitches
 embroidery_internal.h, 451
 format_pec.c, 497
 format_pec.c, 497
format_pec.c, 497
writePel
 embroidery_internal.h, 451
 format_pel.c, 498
writePem
 embroidery_internal.h, 451
 format_pem.c, 498
writePes
 embroidery_internal.h, 451
 format_pes.c, 501
writePhb
 embroidery_internal.h, 451
 format_phb.c, 501
writePhc
 embroidery_internal.h, 451
 format_phc.c, 502
writePlt
 embroidery_internal.h, 451
 format_plt.c, 502
writer_state
 EmbFormatList_, 109
writeRgb
 embroidery_internal.h, 452
 format_rgb.c, 503
writeSettings
 MainWindow, 166
writeSew
 embroidery_internal.h, 452
 format_sew.c, 503
writeShv
 embroidery_internal.h, 452
 format_shv.c, 504
writeSst
 embroidery_internal.h, 452
 format_sst.c, 505
writeStx
 embroidery_internal.h, 452
 format_stx.c, 505
writeSvg
 embroidery_internal.h, 452
 format_svg.c, 506
writeT01
 embroidery_internal.h, 452
 format_t01.c, 507
writeT09
 embroidery_internal.h, 452
 format_t09.c, 507
writeTap
 embroidery_internal.h, 452
 format.tap.c, 508
writeThr
 embroidery_internal.h, 452
 format_thr.c, 509
writeTxt
 embroidery_internal.h, 452
 format_txt.c, 509
writeU00
 embroidery_internal.h, 453

format_u00.c, 510
writeU01
 embroidery_internal.h, 453
 format_u01.c, 510
writeVip
 embroidery_internal.h, 453
 format_vip.c, 511
writeVp3
 embroidery_internal.h, 453
 format_vp3.c, 513
writeXxx
 embroidery_internal.h, 453
 format_xxx.c, 513
writeZsk
 embroidery_internal.h, 453
 format_zsk.c, 514

x
 EmbStitch_, 118
 EmbVector_, 120
xOffset
 _vp3Hoop, 54
 VipHeader_, 296
xxx, 17, 514
xxxDecodeByte
 format_xxx.c, 513
xxxEncodeDesign
 format_xxx.c, 514
xxxEncodeStitch
 format_xxx.c, 514
xxxEncodeStop
 format_xxx.c, 514

y
 EmbStitch_, 118
 EmbVector_, 120
year
 EmbTime_, 120
YELLOW_TERM_COLOR
 embroidery_internal.h, 429
yOffset
 _vp3Hoop, 54
 VipHeader_, 296

Z102_Isacord_Polyester
 embroidery.h, 388
zoom_toolbar
 embroidermodder.h, 308
 mainwindow-toolbars.cpp, 346
zoomAll
 MainWindow, 166
zoomCenter
 MainWindow, 167
zoomDynamic
 MainWindow, 167
zoomExtents
 MainWindow, 167
 View, 291
zoomExtentsAllSubWindows
 MdiArea, 179
zoomIn
 MainWindow, 167
 View, 291
zoomInLimit
 Settings_, 234
zoomMenu
 MainWindow, 176
zoomOut
 MainWindow, 167
 View, 291
zoomOutLimit
 Settings_, 234
zoomPrevious
 MainWindow, 167
zoomRealtime
 MainWindow, 167
zoomScale
 MainWindow, 167
zoomSelected
 MainWindow, 167
 View, 291
zoomToPoint
 View, 291
zoomWindow
 MainWindow, 167
 View, 291
zoomWindowActive
 View, 295
zsk, 17, 514
ZSK USA, 440, 482, 514