

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
2.2.4 History	5
2.3 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
8 Formats	14

8.1 Overview	14
8.1.1 Read/Write Support Levels	14
8.1.2 Table of Format Support Levels	16
8.1.3 Format Support	17
9 Geometry and Algorithms	18
9.1 To Do	18
9.1.1 Development	18
9.1.2 Testing	18
9.1.3 Contributing	19
9.1.4 Embroidermodder Project Coding Standards	19
9.1.5 Version Control	20
9.1.6 Donations	20
9.1.7 Embroidermodder Project Coding Standards	20
9.1.8 Ideas	21
9.1.9 Electronics development	23
9.1.10 Development	24
9.2 Embroiderbot and Libembroidery on Embedded Systems	24
9.2.1 Compatible Boards	24
9.2.2 Arduino Considerations	24
9.2.3 Space	25
9.2.4 Tables	25
9.2.5 Current Pattern Memory Management	25
9.2.6 Special Notes	25
9.2.7 The Assembly Split	25
9.3 The Embroider Command Line Program	26
9.3.1 Embroider pipeline	26
9.3.2 embroider CLI	26
10 GNU Free Documentation License	26
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	32
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	34

12 Privacy Policy for Embroidery Viewer	34
12.0.1 CONTACT US	34
13 Todo List	34
14 Hierarchical Index	39
14.1 Class Hierarchy	39
15 Class Index	43
15.1 Class List	43
16 File Index	46
16.1 File List	46
17 Class Documentation	50
17.1 _bcf_directory Struct Reference	50
17.1.1 Detailed Description	51
17.1.2 Member Data Documentation	51
17.2 _bcf_directory_entry Struct Reference	51
17.2.1 Member Data Documentation	51
17.3 _bcf_file Struct Reference	53
17.3.1 Member Data Documentation	53
17.4 _bcf_file_difat Struct Reference	54
17.4.1 Member Data Documentation	54
17.5 _bcf_file_fat Struct Reference	54
17.5.1 Member Data Documentation	55
17.6 _bcf_file_header Struct Reference	55
17.6.1 Detailed Description	55
17.6.2 Member Data Documentation	56
17.7 _vp3Hoop Struct Reference	57
17.7.1 Member Data Documentation	58
17.8 Action__ Struct Reference	60
17.8.1 Detailed Description	60
17.8.2 Member Data Documentation	60
17.9 Application Class Reference	62
17.9.1 Detailed Description	62
17.9.2 Constructor & Destructor Documentation	62
17.9.3 Member Function Documentation	63
17.9.4 Member Data Documentation	63
17.10 ArcObject Class Reference	63
17.10.1 Member Enumeration Documentation	66
17.10.2 Constructor & Destructor Documentation	67
17.10.3 Member Function Documentation	68
17.10.4 Member Data Documentation	75

17.11 BaseObject Class Reference	76
17.11.1 Member Enumeration Documentation	77
17.11.2 Constructor & Destructor Documentation	77
17.11.3 Member Function Documentation	77
17.11.4 Member Data Documentation	82
17.12 CircleObject Class Reference	82
17.12.1 Member Enumeration Documentation	84
17.12.2 Constructor & Destructor Documentation	85
17.12.3 Member Function Documentation	85
17.13 CmdPrompt Class Reference	88
17.13.1 Detailed Description	90
17.13.2 Constructor & Destructor Documentation	90
17.13.3 Member Function Documentation	90
17.13.4 Member Data Documentation	96
17.14 CmdPromptHandle Class Reference	96
17.14.1 Detailed Description	97
17.14.2 Constructor & Destructor Documentation	97
17.14.3 Member Function Documentation	97
17.14.4 Member Data Documentation	98
17.15 CmdPromptHistory Class Reference	98
17.15.1 Detailed Description	99
17.15.2 Constructor & Destructor Documentation	99
17.15.3 Member Function Documentation	99
17.15.4 Member Data Documentation	100
17.16 CmdPromptInput Class Reference	101
17.16.1 Constructor & Destructor Documentation	102
17.16.2 Member Function Documentation	102
17.16.3 Member Data Documentation	107
17.17 CmdPromptSplitter Class Reference	108
17.17.1 Detailed Description	108
17.17.2 Constructor & Destructor Documentation	108
17.17.3 Member Function Documentation	108
17.18 Compress Struct Reference	109
17.18.1 Member Data Documentation	109
17.19 DimLeaderObject Class Reference	110
17.19.1 Member Enumeration Documentation	113
17.19.2 Constructor & Destructor Documentation	113
17.19.3 Member Function Documentation	114
17.19.4 Member Data Documentation	117
17.20 EllipseObject Class Reference	118
17.20.1 Member Enumeration Documentation	120
17.20.2 Constructor & Destructor Documentation	120

17.20.3 Member Function Documentation	121
17.21 EmbAlignedDim_Struct Reference	124
17.21.1 Member Data Documentation	124
17.22 EmbAngularDim_Struct Reference	124
17.22.1 Member Data Documentation	124
17.23 EmbArc_Struct Reference	124
17.23.1 Detailed Description	125
17.23.2 Member Data Documentation	125
17.24 EmbArcLengthDim_Struct Reference	125
17.24.1 Member Data Documentation	125
17.25 EmbArray_Struct Reference	126
17.25.1 Member Data Documentation	126
17.26 EmbBezier_Struct Reference	127
17.26.1 Member Data Documentation	127
17.27 EmbBlock_Struct Reference	127
17.27.1 Member Data Documentation	127
17.28 EmbCircle_Struct Reference	128
17.28.1 Member Data Documentation	128
17.29 EmbColor_Struct Reference	128
17.29.1 Detailed Description	128
17.29.2 Member Data Documentation	129
17.30 EmbDetailsDialog Class Reference	129
17.30.1 Detailed Description	129
17.30.2 Constructor & Destructor Documentation	129
17.30.3 Member Function Documentation	130
17.30.4 Member Data Documentation	130
17.31 EmbDiameterDim_Struct Reference	131
17.31.1 Member Data Documentation	131
17.32 EmbEllipse_Struct Reference	131
17.32.1 Member Data Documentation	132
17.33 EmbFormatList_Struct Reference	132
17.33.1 Member Data Documentation	132
17.34 EmbGeometry_Struct Reference	133
17.34.1 Member Data Documentation	134
17.35 EmblImage_Struct Reference	135
17.35.1 Member Data Documentation	136
17.36 EmblInfiniteLine_Struct Reference	136
17.36.1 Member Data Documentation	137
17.37 EmbLayer_Struct Reference	137
17.37.1 Member Data Documentation	137
17.38 EmbLeaderDim_Struct Reference	137
17.38.1 Member Data Documentation	138

17.39 EmbLine_ Struct Reference	138
17.39.1 Member Data Documentation	138
17.40 EmbLinearDim_ Struct Reference	139
17.40.1 Member Data Documentation	139
17.41 EmbOrdinateDim_ Struct Reference	139
17.41.1 Member Data Documentation	139
17.42 EmbPath_ Struct Reference	139
17.42.1 Member Data Documentation	140
17.43 EmbPattern_ Struct Reference	140
17.43.1 Member Data Documentation	141
17.44 EmbPoint_ Struct Reference	141
17.44.1 Member Data Documentation	142
17.45 EmbRadiusDim_ Struct Reference	142
17.45.1 Member Data Documentation	142
17.46 EmbRay_ Struct Reference	143
17.46.1 Member Data Documentation	143
17.47 EmbRect_ Struct Reference	143
17.47.1 Member Data Documentation	143
17.48 EmbSatinOutline_ Struct Reference	144
17.48.1 Member Data Documentation	144
17.49 EmbSpline_ Struct Reference	145
17.49.1 Member Data Documentation	145
17.50 EmbStitch_ Struct Reference	145
17.50.1 Member Data Documentation	145
17.51 EmbTextMulti_ Struct Reference	146
17.51.1 Member Data Documentation	146
17.52 EmbTextSingle_ Struct Reference	146
17.52.1 Member Data Documentation	147
17.53 EmbThread_ Struct Reference	147
17.53.1 Member Data Documentation	147
17.54 EmbTime_ Struct Reference	148
17.54.1 Member Data Documentation	148
17.55 EmbVector_ Struct Reference	149
17.55.1 Detailed Description	149
17.55.2 Member Data Documentation	149
17.56 EmbView_ Struct Reference	149
17.56.1 Detailed Description	150
17.56.2 EmbViews	150
17.56.3 Member Data Documentation	150
17.57 GroupBoxData_ Struct Reference	153
17.57.1 Member Data Documentation	153
17.58 hoop_padding Struct Reference	154

17.58.1 Member Data Documentation	154
17.59 Huffman Struct Reference	155
17.59.1 Member Data Documentation	155
17.60 ImageObject Class Reference	156
17.60.1 Member Enumeration Documentation	158
17.60.2 Constructor & Destructor Documentation	158
17.60.3 Member Function Documentation	159
17.61 ImageWidget Class Reference	161
17.61.1 Detailed Description	161
17.61.2 Constructor & Destructor Documentation	161
17.61.3 Member Function Documentation	162
17.61.4 Member Data Documentation	163
17.62 LayerManager Class Reference	163
17.62.1 Detailed Description	163
17.62.2 Constructor & Destructor Documentation	163
17.62.3 Member Function Documentation	164
17.62.4 Member Data Documentation	164
17.63 LineObject Class Reference	165
17.63.1 Member Enumeration Documentation	167
17.63.2 Constructor & Destructor Documentation	167
17.63.3 Member Function Documentation	168
17.64 LSYSTEM Struct Reference	171
17.64.1 Member Data Documentation	171
17.65 MainWindow Class Reference	171
17.65.1 Detailed Description	178
17.65.2 Constructor & Destructor Documentation	178
17.65.3 Member Function Documentation	179
17.65.4 Member Data Documentation	205
17.66 MdiArea Class Reference	210
17.66.1 Constructor & Destructor Documentation	211
17.66.2 Member Function Documentation	211
17.66.3 Member Data Documentation	214
17.67 MdiWindow Class Reference	214
17.67.1 Constructor & Destructor Documentation	216
17.67.2 Member Function Documentation	217
17.67.3 Member Data Documentation	223
17.68 Parameter_Struct Reference	224
17.68.1 Member Data Documentation	224
17.69 PathObject Class Reference	225
17.69.1 Member Enumeration Documentation	227
17.69.2 Constructor & Destructor Documentation	227
17.69.3 Member Function Documentation	227

17.69.4 Member Data Documentation	230
17.70 PointObject Class Reference	230
17.70.1 Member Enumeration Documentation	232
17.70.2 Constructor & Destructor Documentation	232
17.70.3 Member Function Documentation	232
17.71 PolygonObject Class Reference	234
17.71.1 Member Enumeration Documentation	236
17.71.2 Constructor & Destructor Documentation	237
17.71.3 Member Function Documentation	237
17.71.4 Member Data Documentation	239
17.72 PolylineObject Class Reference	240
17.72.1 Member Enumeration Documentation	242
17.72.2 Constructor & Destructor Documentation	242
17.72.3 Member Function Documentation	243
17.72.4 Member Data Documentation	245
17.73 PreviewDialog Class Reference	245
17.73.1 Constructor & Destructor Documentation	245
17.73.2 Member Data Documentation	246
17.74 PropertyEditor Class Reference	246
17.74.1 Constructor & Destructor Documentation	248
17.74.2 Member Function Documentation	248
17.74.3 Member Data Documentation	256
17.75 RectObject Class Reference	258
17.75.1 Member Enumeration Documentation	260
17.75.2 Constructor & Destructor Documentation	261
17.75.3 Member Function Documentation	261
17.76 SaveObject Class Reference	264
17.76.1 Constructor & Destructor Documentation	265
17.76.2 Member Function Documentation	265
17.76.3 Member Data Documentation	270
17.77 SelectBox Class Reference	270
17.77.1 Constructor & Destructor Documentation	271
17.77.2 Member Function Documentation	271
17.77.3 Member Data Documentation	272
17.78 Settings_Struct Reference	273
17.78.1 Detailed Description	275
17.78.2 Member Data Documentation	275
17.79 Settings_Dialog Class Reference	282
17.79.1 Constructor & Destructor Documentation	285
17.79.2 Member Function Documentation	285
17.79.3 Member Data Documentation	292
17.80 StatusBar Class Reference	293

17.80.1 Constructor & Destructor Documentation	293
17.80.2 Member Function Documentation	293
17.80.3 Member Data Documentation	293
17.81 StatusBarButton Class Reference	294
17.81.1 Constructor & Destructor Documentation	295
17.81.2 Member Function Documentation	295
17.81.3 Member Data Documentation	296
17.82 StxThread_ Struct Reference	297
17.82.1 Member Data Documentation	297
17.83 SubDescriptor_ Struct Reference	297
17.83.1 Member Data Documentation	297
17.84 SvgAttribute_ Struct Reference	298
17.84.1 Member Data Documentation	298
17.85 TextSingleObject Class Reference	298
17.85.1 Member Enumeration Documentation	301
17.85.2 Constructor & Destructor Documentation	301
17.85.3 Member Function Documentation	301
17.85.4 Member Data Documentation	303
17.86 thread_color_ Struct Reference	304
17.86.1 Member Data Documentation	304
17.87 ThredExtension_ Struct Reference	305
17.87.1 Member Data Documentation	305
17.88 ThredHeader_ Struct Reference	305
17.88.1 Member Data Documentation	306
17.89 UiObject_ Struct Reference	306
17.89.1 Detailed Description	306
17.89.2 Member Data Documentation	307
17.90 UndoableAddCommand Class Reference	308
17.90.1 Constructor & Destructor Documentation	308
17.90.2 Member Function Documentation	308
17.90.3 Member Data Documentation	309
17.91 UndoableDeleteCommand Class Reference	309
17.91.1 Constructor & Destructor Documentation	309
17.91.2 Member Function Documentation	309
17.91.3 Member Data Documentation	309
17.92 UndoableGripEditCommand Class Reference	310
17.92.1 Constructor & Destructor Documentation	310
17.92.2 Member Function Documentation	310
17.92.3 Member Data Documentation	310
17.93 UndoableMirrorCommand Class Reference	311
17.93.1 Constructor & Destructor Documentation	311
17.93.2 Member Function Documentation	311

17.93.3 Member Data Documentation	311
17.94 UndoableMoveCommand Class Reference	312
17.94.1 Constructor & Destructor Documentation	312
17.94.2 Member Function Documentation	312
17.94.3 Member Data Documentation	312
17.95 UndoableNavCommand Class Reference	313
17.95.1 Constructor & Destructor Documentation	313
17.95.2 Member Function Documentation	313
17.95.3 Member Data Documentation	313
17.96 UndoableRotateCommand Class Reference	314
17.96.1 Constructor & Destructor Documentation	314
17.96.2 Member Function Documentation	314
17.96.3 Member Data Documentation	315
17.97 UndoableScaleCommand Class Reference	315
17.97.1 Constructor & Destructor Documentation	315
17.97.2 Member Function Documentation	316
17.97.3 Member Data Documentation	316
17.98 UndoEditor Class Reference	316
17.98.1 Constructor & Destructor Documentation	317
17.98.2 Member Function Documentation	317
17.98.3 Member Data Documentation	317
17.99 View Class Reference	318
17.99.1 Constructor & Destructor Documentation	321
17.99.2 Member Function Documentation	321
17.99.3 Member Data Documentation	328
17.100 VipHeader_ Struct Reference	332
17.100.1 Member Data Documentation	332
18 File Documentation	333
18.1 CODE_OF_CONDUCT.md File Reference	333
18.2 embroidermodder2/cmdprompt.cpp File Reference	333
18.3 embroidermodder2/data.cpp File Reference	333
18.3.1 Detailed Description	333
18.3.2 Variable Documentation	333
18.4 embroidermodder2/embdetails-dialog.cpp File Reference	334
18.5 embroidermodder2/embroidermodder.cpp File Reference	334
18.5.1 Function Documentation	335
18.5.2 Variable Documentation	335
18.6 embroidermodder2/embroidermodder.h File Reference	336
18.6.1 Detailed Description	340
18.6.2 Typedef Documentation	340
18.6.3 EmbViews	340

18.6.4 Enumeration Type Documentation	340
18.6.5 Function Documentation	345
18.6.6 Variable Documentation	346
18.7 embroidermodder.h	347
18.8 embroidermodder2/imagewidget.cpp File Reference	380
18.9 embroidermodder2/interface.cpp File Reference	380
18.9.1 Detailed Description	380
18.9.2 Function Documentation	380
18.10 embroidermodder2/layer-manager.cpp File Reference	380
18.10.1 Detailed Description	381
18.11 embroidermodder2/mainwindow-commands.cpp File Reference	381
18.12 embroidermodder2/mainwindow-menus.cpp File Reference	381
18.13 embroidermodder2/mainwindow-settings.cpp File Reference	381
18.13.1 Function Documentation	381
18.14 embroidermodder2/mainwindow-toolbars.cpp File Reference	382
18.14.1 Function Documentation	382
18.14.2 Variable Documentation	382
18.15 embroidermodder2/mainwindow.cpp File Reference	384
18.15.1 Typedef Documentation	384
18.15.2 Function Documentation	384
18.15.3 Variable Documentation	385
18.16 embroidermodder2/mdiarea.cpp File Reference	386
18.17 embroidermodder2 mdiwindow.cpp File Reference	386
18.18 embroidermodder2/object-arc.cpp File Reference	386
18.18.1 Function Documentation	386
18.19 embroidermodder2/object-base.cpp File Reference	386
18.20 embroidermodder2/object-circle.cpp File Reference	386
18.21 embroidermodder2/object-dimleader.cpp File Reference	386
18.22 embroidermodder2/object-ellipse.cpp File Reference	386
18.23 embroidermodder2/object-image.cpp File Reference	386
18.24 embroidermodder2/object-line.cpp File Reference	387
18.25 embroidermodder2/object-path.cpp File Reference	387
18.26 embroidermodder2/object-point.cpp File Reference	387
18.27 embroidermodder2/object-polygon.cpp File Reference	387
18.28 embroidermodder2/object-polyline.cpp File Reference	387
18.29 embroidermodder2/object-rect.cpp File Reference	387
18.29.1 Detailed Description	387
18.30 embroidermodder2/object-save.cpp File Reference	387
18.31 embroidermodder2/object-textsingle.cpp File Reference	387
18.32 embroidermodder2/preview-dialog.cpp File Reference	387
18.33 embroidermodder2/property-editor.cpp File Reference	387
18.33.1 Variable Documentation	390

18.34 embroidermodder2/README.md File Reference	398
18.35 embroidermodder2/selectbox.cpp File Reference	398
18.36 embroidermodder2/settings-dialog.cpp File Reference	399
18.36.1 Variable Documentation	399
18.37 embroidermodder2/statusbar-button.cpp File Reference	399
18.38 embroidermodder2/statusbar.cpp File Reference	399
18.39 embroidermodder2/undo-commands.cpp File Reference	400
18.40 embroidermodder2/undo-editor.cpp File Reference	400
18.40.1 Detailed Description	400
18.41 embroidermodder2/view.cpp File Reference	400
18.41.1 Detailed Description	400
18.42 extern/libembroidery/src/array.c File Reference	400
18.42.1 Function Documentation	400
18.43 extern/libembroidery/src/compress.c File Reference	402
18.43.1 Detailed Description	403
18.43.2 Function Documentation	403
18.43.3 Variable Documentation	404
18.44 extern/libembroidery/src/embroidery.h File Reference	404
18.44.1 Macro Definition Documentation	411
18.44.2 Typedef Documentation	419
18.44.3 Function Documentation	421
18.44.4 Variable Documentation	436
18.45 embroidery.h	437
18.46 extern/libembroidery/src/embroidery_internal.h File Reference	444
18.46.1 Macro Definition Documentation	452
18.46.2 Typedef Documentation	460
18.46.3 Enumeration Type Documentation	461
18.46.4 Function Documentation	462
18.46.5 Variable Documentation	484
18.47 embroidery_internal.h	484
18.48 extern/libembroidery/src/encoding.c File Reference	491
18.48.1 Detailed Description	492
18.48.2 Function Documentation	492
18.49 extern/libembroidery/src/fill.c File Reference	494
18.49.1 Function Documentation	495
18.49.2 Variable Documentation	498
18.50 extern/libembroidery/src/formats.c File Reference	499
18.50.1 Function Documentation	500
18.50.2 Variable Documentation	502
18.51 extern/libembroidery/src/formats/format_100.c File Reference	503
18.51.1 Detailed Description	503
18.51.2 Function Documentation	503

18.52 extern/libembroidery/src/formats/format_10o.c File Reference	503
18.52.1 Detailed Description	503
18.52.2 Function Documentation	504
18.53 extern/libembroidery/src/formats/format_art.c File Reference	504
18.53.1 Detailed Description	504
18.53.2 Function Documentation	504
18.54 extern/libembroidery/src/formats/format_bmc.c File Reference	504
18.54.1 Detailed Description	505
18.54.2 Function Documentation	505
18.55 extern/libembroidery/src/formats/format_bro.c File Reference	505
18.55.1 Detailed Description	505
18.55.2 Function Documentation	505
18.56 extern/libembroidery/src/formats/format_cnd.c File Reference	505
18.56.1 Detailed Description	506
18.56.2 Function Documentation	506
18.57 extern/libembroidery/src/formats/format_col.c File Reference	506
18.57.1 Detailed Description	506
18.57.2 Function Documentation	507
18.58 extern/libembroidery/src/formats/format_csd.c File Reference	507
18.58.1 Detailed Description	507
18.58.2 Macro Definition Documentation	507
18.58.3 Function Documentation	507
18.58.4 Variable Documentation	508
18.59 extern/libembroidery/src/formats/format_csv.c File Reference	508
18.59.1 Detailed Description	509
18.59.2 Function Documentation	509
18.60 extern/libembroidery/src/formats/format_dat.c File Reference	509
18.60.1 Function Documentation	509
18.61 extern/libembroidery/src/formats/format_dem.c File Reference	510
18.61.1 Detailed Description	510
18.61.2 Function Documentation	510
18.62 extern/libembroidery/src/formats/format_dsb.c File Reference	510
18.62.1 Detailed Description	510
18.62.2 Function Documentation	511
18.63 extern/libembroidery/src/formats/format_dst.c File Reference	511
18.63.1 Detailed Description	511
18.63.2 Macro Definition Documentation	512
18.63.3 Function Documentation	512
18.64 extern/libembroidery/src/formats/format_dsz.c File Reference	513
18.64.1 Function Documentation	513
18.65 extern/libembroidery/src/formats/format_dxf.c File Reference	513
18.65.1 Function Documentation	514

18.66 extern/libembroidery/src/formats/format_edr.c File Reference	514
18.66.1 Function Documentation	514
18.67 extern/libembroidery/src/formats/format_emd.c File Reference	515
18.67.1 Detailed Description	515
18.67.2 Function Documentation	515
18.68 extern/libembroidery/src/formats/format_exp.c File Reference	515
18.68.1 Function Documentation	515
18.69 extern/libembroidery/src/formats/format_exy.c File Reference	516
18.69.1 Function Documentation	516
18.70 extern/libembroidery/src/formats/format_eyc.c File Reference	516
18.70.1 Function Documentation	516
18.71 extern/libembroidery/src/formats/format_fxy.c File Reference	517
18.71.1 Function Documentation	517
18.72 extern/libembroidery/src/formats/format_gc.c File Reference	517
18.72.1 Function Documentation	517
18.73 extern/libembroidery/src/formats/format_gnc.c File Reference	518
18.73.1 Function Documentation	518
18.74 extern/libembroidery/src/formats/format_gt.c File Reference	518
18.74.1 Function Documentation	518
18.75 extern/libembroidery/src/formats/format_hus.c File Reference	519
18.75.1 Function Documentation	519
18.76 extern/libembroidery/src/formats/format_inb.c File Reference	520
18.76.1 Function Documentation	520
18.77 extern/libembroidery/src/formats/format_inf.c File Reference	520
18.77.1 Function Documentation	520
18.78 extern/libembroidery/src/formats/format_jef.c File Reference	521
18.78.1 Function Documentation	521
18.79 extern/libembroidery/src/formats/format_ksm.c File Reference	522
18.79.1 Function Documentation	522
18.80 extern/libembroidery/src/formats/format_max.c File Reference	522
18.80.1 Function Documentation	523
18.80.2 Variable Documentation	523
18.81 extern/libembroidery/src/formats/format_mit.c File Reference	523
18.81.1 Function Documentation	523
18.82 extern/libembroidery/src/formats/format_new.c File Reference	524
18.82.1 Function Documentation	524
18.83 extern/libembroidery/src/formats/format_ofm.c File Reference	524
18.83.1 Function Documentation	525
18.84 extern/libembroidery/src/formats/format_pcd.c File Reference	525
18.84.1 Function Documentation	526
18.85 extern/libembroidery/src/formats/format_pcm.c File Reference	526
18.85.1 Function Documentation	526

18.86 extern/libembroidery/src/formats/format_pcq.c File Reference	526
18.86.1 Function Documentation	527
18.87 extern/libembroidery/src/formats/format_pcs.c File Reference	527
18.87.1 Function Documentation	527
18.88 extern/libembroidery/src/formats/format_pec.c File Reference	527
18.88.1 Function Documentation	528
18.89 extern/libembroidery/src/formats/format_pel.c File Reference	529
18.89.1 Function Documentation	529
18.90 extern/libembroidery/src/formats/format_pem.c File Reference	529
18.90.1 Function Documentation	529
18.91 extern/libembroidery/src/formats/format_pes.c File Reference	530
18.91.1 Function Documentation	530
18.91.2 Variable Documentation	532
18.92 extern/libembroidery/src/formats/format_phb.c File Reference	532
18.92.1 Function Documentation	532
18.93 extern/libembroidery/src/formats/format_phc.c File Reference	533
18.93.1 Function Documentation	533
18.94 extern/libembroidery/src/formats/format_plt.c File Reference	533
18.94.1 Function Documentation	533
18.95 extern/libembroidery/src/formats/format_rgb.c File Reference	534
18.95.1 Function Documentation	534
18.96 extern/libembroidery/src/formats/format_sew.c File Reference	534
18.96.1 Function Documentation	534
18.97 extern/libembroidery/src/formats/format_shv.c File Reference	535
18.97.1 Function Documentation	535
18.98 extern/libembroidery/src/formats/format_sst.c File Reference	535
18.98.1 Function Documentation	535
18.99 extern/libembroidery/src/formats/format_stx.c File Reference	536
18.99.1 Function Documentation	536
18.100 extern/libembroidery/src/formats/format_svg.c File Reference	536
18.100.1 Function Documentation	537
18.100.2 Variable Documentation	537
18.101 extern/libembroidery/src/formats/format_t01.c File Reference	538
18.101.1 Function Documentation	538
18.102 extern/libembroidery/src/formats/format_t09.c File Reference	538
18.102.1 Function Documentation	538
18.103 extern/libembroidery/src/formats/format.tap.c File Reference	539
18.103.1 Function Documentation	539
18.104 extern/libembroidery/src/formats/format_thr.c File Reference	539
18.104.1 Function Documentation	539
18.105 extern/libembroidery/src/formats/format.txt.c File Reference	540
18.105.1 Function Documentation	540

18.106 extern/libembroidery/src/formats/format_u00.c File Reference	540
18.106.1 Function Documentation	540
18.107 extern/libembroidery/src/formats/format_u01.c File Reference	541
18.107.1 Function Documentation	541
18.108 extern/libembroidery/src/formats/format_vip.c File Reference	541
18.108.1 Function Documentation	542
18.108.2 Variable Documentation	542
18.109 extern/libembroidery/src/formats/format_vp3.c File Reference	543
18.109.1 Function Documentation	543
18.110 extern/libembroidery/src/formats/format_xxx.c File Reference	544
18.110.1 Function Documentation	544
18.111 extern/libembroidery/src/formats/format_zsk.c File Reference	545
18.111.1 Detailed Description	545
18.111.2 Function Documentation	545
18.112 extern/libembroidery/src/geometry.c File Reference	545
18.112.1 Function Documentation	546
18.113 extern/libembroidery/src/geometry/arc.c File Reference	547
18.113.1 Function Documentation	547
18.114 extern/libembroidery/src/geometry/circle.c File Reference	550
18.114.1 Function Documentation	551
18.115 extern/libembroidery/src/geometry/ellipse.c File Reference	551
18.115.1 Function Documentation	552
18.116 extern/libembroidery/src/geometry/functions.c File Reference	553
18.116.1 Function Documentation	553
18.117 extern/libembroidery/src/geometry/line.c File Reference	554
18.117.1 Function Documentation	554
18.118 extern/libembroidery/src/geometry/path.c File Reference	554
18.119 extern/libembroidery/src/geometry/polygon.c File Reference	554
18.120 extern/libembroidery/src/geometry/polyline.c File Reference	554
18.121 extern/libembroidery/src/geometry/rect.c File Reference	555
18.121.1 Function Documentation	555
18.122 extern/libembroidery/src/geometry/text.c File Reference	555
18.122.1 Function Documentation	555
18.123 extern/libembroidery/src/geometry/vector.c File Reference	557
18.123.1 Function Documentation	557
18.124 extern/libembroidery/src/image.c File Reference	559
18.124.1 Detailed Description	559
18.124.2 Function Documentation	559
18.125 extern/libembroidery/src/main.c File Reference	560
18.125.1 Macro Definition Documentation	562
18.125.2 Function Documentation	565
18.125.3 Variable Documentation	571

18.126 extern/libembroidery/src/pattern.c File Reference	571
18.126.1 Detailed Description	572
18.126.2 Function Documentation	572
18.127 extern/libembroidery/src/thread-color.c File Reference	576
18.127.1 Function Documentation	577
18.127.2 Variable Documentation	577
18.128 privacy_policy.md File Reference	578
Bibliography	579
Index	581

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.

2.2.4 History

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

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

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

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

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

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

2.3 Contact us

For general questions email: [embroidermodder at gmail.com](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 [news0.html::fast-forward](#) Permanent link to this article.

6 Changelog

7 Ideas

Stuff that is now supposed to be generated by Doxygen:

Todo Bibliography style to plainnat.

Todo Serif font for printed docs.

Todo US letter paper version of printed docs.

8 Formats

8.1 Overview

8.1.1 Read/Write Support Levels

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

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

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

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

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

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

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

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

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

Status Label	Description
read-only	If write support is none and read support is not none.

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

8.1.2 Table of Format Support Levels

Overview of documentation support by format.

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

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

8.1.3 Format Support

```
| FORMAT | READ | WRITE | NOTES | -----|-----|-----|-----| | 10o | YES || read (need to fix external color
loading) (maybe find out what ctrl | code flags of 0x10, 0x08, 0x04, and 0x02 mean) || 100 || | none (4 byte codes)
61 00 10 09 (type, type2, x, y ?) x | y (signed char) || 100 || | none (4 byte codes) 61 00 10 09 (type, type2, x, y ?)
x & y (signed char) || art || | none || bro | YES || read (complete)(maybe figure out detail of header) || cnd ||
| none || col || | (color file no design) read(final) write(final) || csd | YES || read (complete) || dat || | read () |
| dem || | none (looks like just encrypted cnd) || dsb | YES || read (unknown how well) (stitch data looks same as
10o) || dst | YES || read (complete) / write(unknown) || dsz | YES || read (unknown) || dxf || | read (Port to
C. needs refactored) || edr || | read (C version is broken) / write (complete) || emd || | read (unknown) || exp |
YES || read (unknown) / write(unknown) || exy | YES || read (need to fix external color loading) || fxy | YES ||
read (need to fix external color loading) || gnc || | none || gt || | read (need to fix external color loading) || hus
| YES || | read (unknown) / write (C version is broken) || inb | YES || read (buggy?) || jef | YES || write (need
to fix the offsets when it is moving to another spot) || ksm | YES || | read (unknown) / write (unknown) || pcd ||
|| pcm || | pcq || | read (Port to C) || pcs | BUGGY || read (buggy / colors are not correct / after reading,
writing any other format is messed up) || pec || | read / write (without embedded images, sometimes overlooks
some stitches leaving a gap) || pel || | none || pem || | none || pes | YES || | phb || | phc || | | rgb ||
| | sew | YES || | shv || | read (C version is broken) || sst || | none || svg || YES || | tap | YES || | read
(unknown) || u01 || | | | vip | YES || | vp3 | YES || | | | xxx | YES || | | | zsk || | read (complete) |
```

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

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

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

9 Geometry and Algorithms

9.1 To Do

Todo (Arduino) Fix emb-outline files

Todo (Arduino) Fix thread-color files

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

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

Todo (Arduino) Smoothieboard experiments

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

9.1.1 Development

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

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

9.1.2 Testing

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

```
$ embroidermodder --test
```

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

9.1.3 Contributing

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

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

- Low level C developers are needed for the base library `libembroidery`.
- Low level assembly programmers are needed for translating some of `libembroidery` to Embroider←Bot.
- Hardware Engineers to help design our own kitbashed embroidery machine EmbroiderBot, one of the original project aims in 2013.
- Scheme developers and C/SDL developers to help build the GUI.
- Scheme developers to help add designs for generating of custom stitch-filled emblems like the heart or dolphin. Note that this happens in Embroidermudder 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 Embroidermudder Project Coding Standards

A basic set of guidelines to use when submitting code.

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

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

9.1.4.2 Non-compiled Files Go

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

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

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

9.1.5 Version Control

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

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

9.1.6 Donations

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

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

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

9.1.7 Embroidermodder Project Coding Standards

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

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

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

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

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

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

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

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

9.1.8 Ideas

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

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

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

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

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

9.1.8.5 Identify the meaning of these TODO items

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

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

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

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

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

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

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

9.1.8.10 To Do We currently need help with:

- Thorough descriptions of each embroidery format.
- Finding resources for each of the branded thread libraries (along with a full citation for documentation).
- Finding resources for each geometric algorithm used (along with a full citation for documentation).
- Completing the full `--full-test-suite` with no segfaults and at least a clear error message (for example not implemented yet).
- Identifying best guesses for filling in missing information when going from, say `.csv` to a late `.pes` version. What should the default be when the data doesn't clarify?
- Improving the written documentation.
- Funding, see the Sponsor button above. We can treat this as `work` and put far more hours in with broad support in small donations from people who want specific features.

Beyond this the development targets are categories sorted into:

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

9.1.8.11 Basic Features

- Incorporate `#if` `0ed` parts of `libembroidery.c`.
- Interpret how to write formats that have a read mode from the source code and vice versa.
- Document the specifics of the file formats here for embroidery machine specific formats. Find websites and other sources that break down the binary formats we currently don't understand.
- Find more and better documentation of the structure of the headers for the formats we do understand.

9.1.8.12 Code quality and user friendliness

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

9.1.8.13 Documentation

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

9.1.8.14 GUI

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

9.1.9 Electronics development

- Currently experimenting with Fritzing8, upload netlists to embroiderbot when they can run simulations using the asm in `libembroidery`.
- Create a common assembly for data that is the same across chipsets `libembroidery_data_\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.

9.1.10 Development

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

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

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

```
#!/bin/bash

rm -fr libembroidery-debug

git clone http://github.com/embroidermodder/libembroidery libembroidery-debug
cd libembroidery-debug

cmake -DCMAKE_BUILD_TYPE=DEBUG .
cmake --build . --config=DEBUG

valgrind ./embroider --full-test-suite
```

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

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

9.2 Embroiderbot and Libembroidery on Embedded Systems

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

9.2.1 Compatible Boards

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

9.2.2 Arduino Considerations

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

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

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

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

9.2.3 Space

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

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

Todo Replace EmbArray functions with embPattern load functions.

9.2.4 Tables

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

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

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

9.2.5 Current Pattern Memory Management

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

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

9.2.6 Special Notes

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

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

9.2.7 The Assembly Split

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

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

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

9.3 The Embroider Command Line Program

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

9.3.1 Embroider pipeline

Adjectives apply to every following noun so

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

Creates:

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

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

9.3.2 `embroider` CLI

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

10.0.0.0.7 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.

10.0.0.0.8 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.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

11 Contributor Covenant Code of Conduct

11.1 Our Pledge

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

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

11.2 Our Standards

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

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

Examples of unacceptable behavior include:

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

11.3 Enforcement Responsibilities

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

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

11.4 Scope

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

11.5 Enforcement

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

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

11.6 Enforcement Guidelines

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

11.6.1 1. Correction

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

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

11.6.2 2. Warning

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

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

11.6.3 3. Temporary Ban

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

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

11.6.4 4. Permanent Ban

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

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

11.7 Attribution

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

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

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

12 Privacy Policy for Embroidery Viewer

Last updated December 15, 2021

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

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

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

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

12.0.1 CONTACT US

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

Embroidermodder@gmail.com

13 Todo List

Member [ArcObject::gripEdit \(const QPointF &before, const QPointF &after\)](#)
gripEdit() for ArcObject

Member [ArcObject::updateRubber \(QPainter *painter=0\)](#)
Arc Rubber Modes
updateRubber() gripping for ArcObject

Member [bcf_directory](#)
possibly add a directory tree in the future.

Member `bcf_file_header`

CLSID should be a separate type.

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

replace with emblnt_read

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

replace with emblnt_read

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

replace with emblnt_read

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

replace with emblnt_read

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

replace with emblnt_read

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

replace with emblnt_read

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

replace with emblnt_read

Member `copy_trim (char const *s)`

description

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

remove the unused return argument.

Member `DimLeaderObject::mouseSnapPoint (const QPointF &mousePoint)`

generic closest point from list to point x.

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

gripEdit() for EllipseObject

Member `embArc_print (EmbArc arc)`

move to arc.c

Member `embGeometry_vulcanize (EmbGeometry *obj)`

Review. This could be controlled by a simple flag.

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

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

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

finish stitchEllipse

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

finish stitch path

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

finish stitch polygon

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

finish stitch polyline

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

make result return argument.

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

make result return argument.

File `format_art.c`

Find a source.

File [format_bmc.c](#)

Find a source.

File [format_cnd.c](#)

Find a source.

Page [Formats](#)

Support for Singer FHE, CHE (Compucon) formats?

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

Member [formatTable \[numberOfFormats\]](#)

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

Member [fread_int32_be \(FILE *\)](#)

replace with emblnt_read

Member [fread_uint16 \(FILE *\)](#)

replace with emblnt_read

Member [generate_dragon_curve \(char *state, int iterations\)](#)

find citation for paper folding method

Page [Geometry and Algorithms](#)

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

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.

(Arduino) Fix emb-outline files

(Arduino) Fix thread-color files

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

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

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

(Arduino) Smoothieboard experiments

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

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

Replace EmbArray functions with embPattern load functions.

Member [group_box_ellipse_geometry \[\]](#)

use proper icons for toolButtons

Page [Ideas](#)

Bibliography style to plainnat.

Serif font for printed docs.

US letter paper version of printed docs.

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 ()

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

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?

Member OBJ_LTYPE

Use color chart in formats/format-dxf.h for this

Member PropertyEditor::clearAllFields ()

DimAligned

DimAngular

DimArcLength

DimDiameter

DimLeader

DimLinear

DimOrdinate

DimRadius

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::createGroupBoxGeometryDimAligned ()

toolButtons and lineEdits for DimAligned

Member PropertyEditor::createGroupBoxGeometryDimAngular ()

toolButtons and lineEdits for DimAngular

Member PropertyEditor::createGroupBoxGeometryDimArcLength ()

toolButtons and lineEdits for DimArcLength

Member PropertyEditor::createGroupBoxGeometryDimDiameter ()

toolButtons and lineEdits for DimDiameter

Member PropertyEditor::createGroupBoxGeometryDimLeader ()

toolButtons and lineEdits for DimLeader

Member PropertyEditor::createGroupBoxGeometryDimLinear ()

toolButtons and lineEdits for DimLinear

Member PropertyEditor::createGroupBoxGeometryDimOrdinate ()

toolButtons and lineEdits for DimOrdinate

Member `PropertyEditor::createGroupBoxGeometryDimRadius ()`

toolButtons and lineEdits for DimRadius

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::createGroupBoxGeometryLine ()`

use proper icons for toolButtons

Member `PropertyEditor::createGroupBoxGeometryPath ()`

use proper icons for toolButtons

Member `PropertyEditor::createGroupBoxGeometryPoint ()`

use proper icons for toolButtons

Member `PropertyEditor::createGroupBoxGeometryPolygon ()`

use proper icons for toolButtons

mapSignal for polygons

Member `PropertyEditor::createGroupBoxGeometryPolyline ()`

use proper icons for toolButtons

Member `PropertyEditor::createGroupBoxGeometryRay ()`

use proper icons for toolButtons

Member `PropertyEditor::createGroupBoxGeometryRectangle ()`

use proper icons for toolButtons

Member `PropertyEditor::createGroupBoxGeometryTextMulti ()`

use proper icons for toolButtons

mapSignal for multiline text

Member `PropertyEditor::createGroupBoxGeometryTextSingle ()`

use proper icons for toolButtons

Member `PropertyEditor::createGroupBoxMiscArc ()`

use proper icons for toolButtons

Member `PropertyEditor::createGroupBoxMiscImage ()`

use proper icons for toolButtons

Member `PropertyEditor::createGroupBoxMiscPath ()`

use proper icons for toolButtons

mapSignal for paths

Member `PropertyEditor::createGroupBoxMiscPolyline ()`

use proper icons for toolButtons

mapSignal for polylines.

Member `PropertyEditor::createGroupBoxMiscTextSingle ()`

use proper icons for toolButtons

Member `PropertyEditor::createGroupBoxTextTextSingle ()`

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 [Settings_Dialog::comboBoxQSnapLocatorColorCurrentIndexChanged \(int\)](#)

Figure out how to abstract the slot in a way that it can be used for comboBoxes in general Currently comboBoxQSnapLocatorColorCurrentIndexChanged(int index) comboBoxSelectionCoolGripColorCurrentIndexChanged(int index) comboBoxSelectionHotGripColorCurrentIndexChanged(int index) are all similar except the dialog. variable being worked on and the QVariant.

Member [Settings_Dialog::set_label_visibility \(QObject *parent, const char *name, bool visibility\)](#)

error reporting.

Member [Settings_Dialog::set_spinbox_visibility \(QObject *parent, const char *name, bool visibility\)](#)

error reporting.

Member [SubDescriptor_::colorCode](#)

better variable naming

Member [SubDescriptor_::someInt](#)

better variable naming

Member [SubDescriptor_::someOtherInt](#)

better variable naming

Member [View::mouseMoveEvent \(QMouseEvent *event\)](#)

turn move into an actuator call.

14 Hierarchical Index

14.1 Class Hierarchy

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

_bcf_directory	50
_bcf_directory_entry	51
_bcf_file	53
_bcf_file_difat	54
_bcf_file_fat	54

_bcf_file_header	55
_vp3Hoop	57
Action_	60
Compress	109
EmbAlignedDim_	124
EmbAngularDim_	124
EmbArc_	124
EmbArcLengthDim_	125
EmbArray_	126
EmbBezier_	127
EmbBlock_	127
EmbCircle_	128
EmbColor_	128
EmbDiameterDim_	131
EmbEllipse_	131
EmbFormatList_	132
EmbGeometry_	133
EmblImage_	135
EmblInfiniteLine_	136
EmbLayer_	137
EmbLeaderDim_	137
EmbLine_	138
EmbLinearDim_	139
EmbOrdinateDim_	139
EmbPath_	139
EmbPattern_	140
EmbPoint_	141
EmbRadiusDim_	142
EmbRay_	143
EmbRect_	143
EmbSatinOutline_	144
EmbSpline_	145

EmbStitch_	145
EmbTextMulti_	146
EmbTextSingle_	146
EmbThread_	147
EmbTime_	148
EmbVector_	149
EmbView_	149
GroupBoxData_	153
hoop_padding	154
Huffman	155
LSYSTEM	171
Parameter_	224
QApplication	
Application	62
QDialog	
EmbDetailsDialog	129
LayerManager	
Settings_Dialog	282
QDockWidget	
PropertyEditor	
UndoEditor	316
QFileDialog	
PreviewDialog	
QGraphicsPathItem	245
BaseObject	
ArcObject	76
CircleObject	63
DimLeaderObject	82
EllipseObject	110
ImageObject	118
LineObject	156
PathObject	165
PointObject	225
PolygonObject	230
PolylineObject	234

PolylineObject	240
RectObject	258
TextSingleObject	298
QGraphicsView	
View	318
QLineEdit	
CmdPromptInput	101
QMainWindow	
MainWindow	171
QMdiArea	
MdiArea	210
QMdiSubWindow	
MdiWindow	214
QObject	
SaveObject	264
QRubberBand	
SelectBox	270
QSplitter	
CmdPromptSplitter	108
QSplitterHandle	
CmdPromptHandle	96
QStatusBar	
Statusbar	293
QTextBrowser	
CmdPromptHistory	98
QToolButton	
StatusBarButton	294
QUndoCommand	
UndoableAddCommand	308
UndoableDeleteCommand	309
UndoableGripEditCommand	310
UndoableMirrorCommand	311
UndoableMoveCommand	312
UndoableNavCommand	313
UndoableRotateCommand	314
UndoableScaleCommand	315
QWidget	
CmdPrompt	88

ImageWidget	161
Settings_	273
StxThread_	297
SubDescriptor_	297
SvgAttribute_	298
thread_color_	304
ThredExtension_	305
ThredHeader_	305
UiObject_	306
VipHeader_	332

15 Class Index

15.1 Class List

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

_bcf_directory	50
_bcf_directory_entry	51
_bcf_file	53
_bcf_file_difat	54
_bcf_file_fat	54
_bcf_file_header	55
_vp3Hoop	57
Action_	60
Application	62
ArcObject	63
BaseObject	76
CircleObject	82
CmdPrompt 88	
CmdPromptHandle 96	
CmdPromptHistory The Command Prompt History class	98

CmdPromptInput	101
CmdPromptSplitter	108
Compress	109
DimLeaderObject	110
EllipseObject	118
EmbAlignedDim_	124
EmbAngularDim_	124
EmbArc_	
Absolute position (not relative)	124
EmbArcLengthDim_	125
EmbArray_	126
EmbBezier_	127
EmbBlock_	127
EmbCircle_	128
EmbColor_	128
EmbDetailsDialog	
129	
EmbDiameterDim_	131
EmbEllipse_	131
EmbFormatList_	132
EmbGeometry_	133
EmblImage_	135
EmblInfiniteLine_	136
EmbLayer_	137
EmbLeaderDim_	137
EmbLine_	138
EmbLinearDim_	139
EmbOrdinateDim_	139
EmbPath_	139
EmbPattern_	140
EmbPoint_	141
EmbRadiusDim_	142

EmbRay_	143
EmbRect_	143
EmbSatinOutline_	144
EmbSpline_	145
EmbStitch_	145
EmbTextMulti_	146
EmbTextSingle_	146
EmbThread_	147
EmbTime_	148
EmbVector_	149
EmbView_	149
GroupBoxData_	153
hoop_padding	154
Huffman	155
ImageObject	156
ImageWidget	
161	
LayerManager	
163	
LineObject	165
LSYSTEM	171
MainWindow	
The MainWindow class	171
MdiArea	210
MdiWindow	214
Parameter_	224
PathObject	225
PointObject	230
PolygonObject	234
PolylineObject	240
PreviewDialog	245
PropertyEditor	246
RectObject	258

SaveObject	264
SelectBox	270
Settings	
Settings System	273
Settings Dialog	282
StatusBar	293
StatusBarButton	294
StxThread	
SubDescriptor	297
SvgAttribute	298
TextSingleObject	298
thread_color	304
ThredExtension	305
ThredHeader	305
UiObject	
This covers the inbuilt designs: Dolphin, Snowflake and Heart. Covers Rotate, Scale and Point UI events	306
UndoableAddCommand	308
UndoableDeleteCommand	309
UndoableGripEditCommand	310
UndoableMirrorCommand	311
UndoableMoveCommand	312
UndoableNavCommand	313
UndoableRotateCommand	314
UndoableScaleCommand	315
UndoEditor	316
View	318
VipHeader	
_	332

16 File Index

16.1 File List

Here is a list of all files with brief descriptions:

embroidermodder2/cmdprompt.cpp	333
embroidermodder2/data.cpp	333
embroidermodder2/embdetails-dialog.cpp	334
embroidermodder2/embroidermodder.cpp	334
embroidermodder2/embroidermodder.h	336
embroidermodder2/imagewidget.cpp	380
embroidermodder2/interface.cpp	380
embroidermodder2/layer-manager.cpp	380
embroidermodder2/mainwindow-commands.cpp	381
embroidermodder2/mainwindow-menus.cpp	381
embroidermodder2/mainwindow-settings.cpp	381
embroidermodder2/mainwindow-toolbars.cpp	382
embroidermodder2/mainwindow.cpp	384
embroidermodder2/mdiarea.cpp	386
embroidermodder2/mdindow.cpp	386
embroidermodder2/object-arc.cpp	386
embroidermodder2/object-base.cpp	386
embroidermodder2/object-circle.cpp	386
embroidermodder2/object-dimleader.cpp	386
embroidermodder2/object-ellipse.cpp	386
embroidermodder2/object-image.cpp	386
embroidermodder2/object-line.cpp	387
embroidermodder2/object-path.cpp	387
embroidermodder2/object-point.cpp	387
embroidermodder2/object-polygon.cpp	387
embroidermodder2/object-polyline.cpp	387
embroidermodder2/object-rect.cpp	387
embroidermodder2/object-save.cpp	387
embroidermodder2/object-textsingle.cpp	387
embroidermodder2/preview-dialog.cpp	387
embroidermodder2/property-editor.cpp	387
embroidermodder2/selectbox.cpp	398

embroidermodder2/settings-dialog.cpp	399
embroidermodder2/statusbar-button.cpp	399
embroidermodder2/statusbar.cpp	399
embroidermodder2/undo-commands.cpp	400
embroidermodder2/undo-editor.cpp	400
embroidermodder2/view.cpp	400
extern/libembroidery/src/array.c	400
extern/libembroidery/src/compress.c	402
extern/libembroidery/src/embroidery.h	404
extern/libembroidery/src/embroidery_internal.h	444
extern/libembroidery/src/encoding.c	491
extern/libembroidery/src/fill.c	494
extern/libembroidery/src/formats.c	499
extern/libembroidery/src/geometry.c	545
extern/libembroidery/src/image.c	559
extern/libembroidery/src/main.c	560
extern/libembroidery/src/pattern.c	571
extern/libembroidery/src/thread-color.c	576
extern/libembroidery/src/formats/format_100.c	503
extern/libembroidery/src/formats/format_10o.c	503
extern/libembroidery/src/formats/format_art.c	504
extern/libembroidery/src/formats/format_bmc.c	504
extern/libembroidery/src/formats/format_bro.c	505
extern/libembroidery/src/formats/format_cnd.c	505
extern/libembroidery/src/formats/format_col.c	506
extern/libembroidery/src/formats/format_csd.c	507
extern/libembroidery/src/formats/format_csv.c	508
extern/libembroidery/src/formats/format_dat.c	509
extern/libembroidery/src/formats/format_dem.c	510
extern/libembroidery/src/formats/format_dsb.c	510
extern/libembroidery/src/formats/format_dst.c	511
extern/libembroidery/src/formats/format_dsz.c	513

extern/libembroidery/src/formats/format_dxf.c	513
extern/libembroidery/src/formats/format_edr.c	514
extern/libembroidery/src/formats/format_emd.c	515
extern/libembroidery/src/formats/format_exp.c	515
extern/libembroidery/src/formats/format_exy.c	516
extern/libembroidery/src/formats/format_eyc.c	516
extern/libembroidery/src/formats/format_fxy.c	517
extern/libembroidery/src/formats/format_gc.c	517
extern/libembroidery/src/formats/format_gnc.c	518
extern/libembroidery/src/formats/format_gt.c	518
extern/libembroidery/src/formats/format_hus.c	519
extern/libembroidery/src/formats/format_inb.c	520
extern/libembroidery/src/formats/format_inf.c	520
extern/libembroidery/src/formats/format_jef.c	521
extern/libembroidery/src/formats/format_ksm.c	522
extern/libembroidery/src/formats/format_max.c	522
extern/libembroidery/src/formats/format_mit.c	523
extern/libembroidery/src/formats/format_new.c	524
extern/libembroidery/src/formats/format_ofm.c	524
extern/libembroidery/src/formats/format_pcd.c	525
extern/libembroidery/src/formats/format_pcm.c	526
extern/libembroidery/src/formats/format_pcq.c	526
extern/libembroidery/src/formats/format_pcs.c	527
extern/libembroidery/src/formats/format_pec.c	527
extern/libembroidery/src/formats/format_pel.c	529
extern/libembroidery/src/formats/format_pem.c	529
extern/libembroidery/src/formats/format_pes.c	530
extern/libembroidery/src/formats/format_phb.c	532
extern/libembroidery/src/formats/format_phc.c	533
extern/libembroidery/src/formats/format_plt.c	533
extern/libembroidery/src/formats/format_rgb.c	534
extern/libembroidery/src/formats/format_sew.c	534

extern/libembroidery/src/formats/ format_shv.c	535
extern/libembroidery/src/formats/ format_sst.c	535
extern/libembroidery/src/formats/ format_stx.c	536
extern/libembroidery/src/formats/ format_svg.c	536
extern/libembroidery/src/formats/ format_t01.c	538
extern/libembroidery/src/formats/ format_t09.c	538
extern/libembroidery/src/formats/ format_tap.c	539
extern/libembroidery/src/formats/ format_thr.c	539
extern/libembroidery/src/formats/ format_txt.c	540
extern/libembroidery/src/formats/ format_u00.c	540
extern/libembroidery/src/formats/ format_u01.c	541
extern/libembroidery/src/formats/ format_vip.c	541
extern/libembroidery/src/formats/ format_vp3.c	543
extern/libembroidery/src/formats/ format_xxx.c	544
extern/libembroidery/src/formats/ format_zsk.c	545
extern/libembroidery/src/geometry/ arc.c	547
extern/libembroidery/src/geometry/ circle.c	550
extern/libembroidery/src/geometry/ ellipse.c	551
extern/libembroidery/src/geometry/ functions.c	553
extern/libembroidery/src/geometry/ line.c	554
extern/libembroidery/src/geometry/ path.c	554
extern/libembroidery/src/geometry/ polygon.c	554
extern/libembroidery/src/geometry/ polyline.c	554
extern/libembroidery/src/geometry/ rect.c	555
extern/libembroidery/src/geometry/ text.c	555
extern/libembroidery/src/geometry/ vector.c	557

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

17.8.1 Detailed Description

17.8.2 Member Data Documentation

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

17.8.2.2 command std::string command

17.8.2.3 hash int hash

17.8.2.4 icon std::string icon

17.8.2.5 menu_name std::string menu_name

17.8.2.6 menu_position int menu_position

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

17.8.2.8 shortcut std::string shortcut

17.8.2.9 statustip std::string statustip

17.8.2.10 toolbar_name std::string toolbar_name

17.8.2.11 toolbar_position int toolbar_position

17.8.2.12 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

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

Returns

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

17.9.4 Member Data Documentation

17.9.4.1 _mainWin `MainWindow* _mainWin`

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

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

17.10 ArcObject Class Reference

`#include <embroidermodder.h>`

Public Types

- enum { `Type` = `OBJ_TYPE_ARC` }

Public Types inherited from `BaseObject`

- enum { `Type` = `OBJ_TYPE_BASE` }

Public Member Functions

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

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

Public Member Functions inherited from `BaseObject`

- `BaseObject (QGraphicsItem *parent=0)`
- `virtual ~BaseObject ()`
- `virtual int type () const`
- `qint64 objectID () const`
- `QPen objectPen () const`
- `QColor objectColor () const`
- `QRgb objectColorRGB () const`
- `Qt::PenStyle objectLineType () const`
- `EmbReal objectLineWidth () const`
- `QPainterPath objectPath () const`
- `int objectRubberMode () const`
- `QPointF objectRubberPoint (const QString &key) const`
- `QString objectRubberText (const QString &key) const`
- `QPointF objectCenter () const`
- `EmbReal objectCenterX () const`
- `EmbReal objectCenterY () const`
- `void setObjectCenter (EmbVector center)`
- `void setObjectCenterX (EmbReal centerX)`
- `void setObjectCenterY (EmbReal centerY)`
- `QRectF rect () const`
- `void setRect (const QRectF &r)`
- `void setRect (EmbReal x, EmbReal y, EmbReal w, EmbReal h)`
- `QLineF line () const`
- `void setLine (const QLineF &li)`

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

Public Attributes

- 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	
------	--

17.10.2 Constructor & Destructor Documentation

17.10.2.1 ArcObject() [1/3] `ArcObject (`
 `EmbArc arc,`
 `QRgb rgb,`
 `QGraphicsItem * parent = 0)`

`ArcObject::ArcObject.`

Parameters

<i>arc</i>	
<i>rgb</i>	
<i>parent</i>	

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 (`

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

[ArcObject::ArcObject](#).

Parameters

<code>obj</code>	
<code>parent</code>	

17.10.2.4 ~ArcObject() `~ArcObject ()`

[ArcObject::~ArcObject](#).

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

[ArcObject::allGripPoints](#).

Returns

Implements [BaseObject](#).

17.10.3.2 calculateArcData() `void calculateArcData (`

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

[ArcObject::calculateArcData](#).

Parameters

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

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

[ArcObject::gripEdit](#).

Parameters

<i>before</i>	
<i>after</i>	

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

Implements [BaseObject](#).

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

[ArcObject::init](#).

Parameters

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

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

[ArcObject::mouseSnapPoint](#).

Parameters

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

Returns

the closest snap point to the mouse point.

Implements [BaseObject](#).

17.10.3.6 objectArcLength() `EmbReal objectArcLength () const`

[ArcObject::objectArcLength](#).

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

[ArcObject::objectArea](#).

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

[ArcObject::objectChord](#).

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

[ArcObject::objectClockwise](#).

Returns

17.10.3.10 objectEndAngle() `EmbReal objectEndAngle() const`

[ArcObject::objectEndAngle](#).

Returns

17.10.3.11 objectEndPoint() `QPointF objectEndPoint() const`

[ArcObject::objectEndPoint](#).

Returns

17.10.3.12 objectEndX() `EmbReal objectEndX() const`

[ArcObject::objectEndX](#).

Returns

17.10.3.13 objectEndY() `EmbReal objectEndY() const`

[ArcObject::objectEndY](#).

Returns

17.10.3.14 objectIncludedAngle() `EmbReal objectIncludedAngle() const`

[ArcObject::objectIncludedAngle](#).

Returns

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

Returns

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

Returns

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

Returns

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

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

Returns

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

Returns

17.10.3.21 objectStartX() `EmbReal objectStartX () const`

[ArcObject::objectStartX](#).

Returns

17.10.3.22 objectStartY() `EmbReal objectStartY () const`

[ArcObject::objectStartY](#).

Returns

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

[ArcObject::paint](#).

Parameters

<code>painter</code>	
<code>option</code>	

17.10.3.24 setObjectEndAngle() `void setObjectEndAngle (`
 `EmbReal angle)`

17.10.3.25 setObjectEndPoint() `[1/2] void setObjectEndPoint (`
 `const QPointF & point)`

[ArcObject::setObjectEndPoint](#).

Parameters

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

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

EmReal pointX,
EmReal pointY)

ArcObject::setObjectEndPoint.

Parameters

<i>pointX</i>	
<i>pointY</i>	

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

const QPointF & point)

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

EmReal pointX,
EmReal pointY)

17.10.3.29 setObjectRadius() void setObjectRadius (

EmReal radius)

17.10.3.30 setObjectStartAngle() void setObjectStartAngle (

EmReal angle)

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

const QPointF & point)

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

EmReal pointX,
EmReal pointY)

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

17.10.3.34 updateArcRect() void updateArcRect (

EmReal radius)

ArcObject::updateArcRect.

Parameters

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

17.10.3.35 updatePath() void updatePath ()

[ArcObject::updatePath](#).

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

[ArcObject::updateRubber](#).

Parameters

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

Todo Arc Rubber Modes

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

17.10.3.37 vulcanize() void vulcanize () [virtual]

[ArcObject::vulcanize](#).

Implements [BaseObject](#).

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

17.10.4.3 **arcStartPoint** QPointF arcStartPoint

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

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

17.11 BaseObject Class Reference

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

Public Types

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

Public Member Functions

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

Public Attributes

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

Protected Member Functions

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

17.11.1 Member Enumeration Documentation

17.11.1.1 anonymous enum anonymous enum

Enumerator

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

17.11.2 Constructor & Destructor Documentation

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

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

17.11.3 Member Function Documentation

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

17.11.3.29 setObjectColorRGB() void setObjectColorRGB (QRgb rgb)

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

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

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

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

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

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

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

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

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

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

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

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

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

17.11.4 Member Data Documentation

17.11.4.1 lwtPen QPen lwtPen

17.11.4.2 objID qint64 objID

17.11.4.3 objLine QLineF objLine

17.11.4.4 objPen QPen objPen

17.11.4.5 objRubberMode int objRubberMode

17.11.4.6 objRubberPoints QHash<QString, QPointF> objRubberPoints

17.11.4.7 objRubberTexts QHash<QString, QString> objRubberTexts

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

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

17.12 CircleObject Class Reference

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

Public Types

- enum { `Type` = `OBJ_TYPE_CIRCLE` }

Public Types inherited from `BaseObject`

- enum { `Type` = `OBJ_TYPE_BASE` }

Public Member Functions

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

Public Member Functions inherited from `BaseObject`

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

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

Protected Member Functions

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

Protected Member Functions inherited from `BaseObject`

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

Additional Inherited Members

Public Attributes inherited from `BaseObject`

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

17.12.1 Member Enumeration Documentation

17.12.1.1 anonymous enum anonymous enum

Enumerator

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

17.12.2 Constructor & Destructor Documentation

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

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

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

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

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

17.12.3 Member Function Documentation

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

Implements [BaseObject](#).

17.12.3.2 `gripEdit()` `void gripEdit (`

```
    const QPointF & before,  
    const QPointF & after ) [virtual]
```

Implements [BaseObject](#).

```
17.12.3.3 init() void init (
    EmbReal centerX,
    EmbReal centerY,
    EmbReal radius,
    QRgb rgb,
    Qt::PenStyle lineType )
```

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

Implements [BaseObject](#).

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Reimplemented from [BaseObject](#).

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

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

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

Implements [BaseObject](#).

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

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

17.13 CmdPrompt Class Reference

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

Public Slots

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

Signals

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

Public Member Functions

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

Public Attributes

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

17.13.1 Detailed Description

17.13.2 Constructor & Destructor Documentation

17.13.2.1 CmdPrompt() `CmdPrompt (QWidget * parent = 0)`

17.13.2.2 ~CmdPrompt() `~CmdPrompt ()`

17.13.3 Member Function Documentation

17.13.3.1 activeCommand `QString activeCommand () [inline], [slot]`

17.13.3.2 addCommand `void addCommand (const QString & alias, const QString & cmd) [inline], [slot]`

17.13.3.3 alert `void alert (const QString & txt) [slot]`

17.13.3.4 appendHistory `void appendHistory (const QString & txt) [slot]`

17.13.3.5 appendTheHistory `void appendTheHistory (const QString & txt, int prefixLength) [signal]`

17.13.3.6 blink `void blink () [slot]`

17.13.3.7 copyPressed void copyPressed () [signal]

17.13.3.8 cutPressed void cutPressed () [signal]

17.13.3.9 deletePressed void deletePressed () [signal]

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

17.13.3.11 downPressed void downPressed () [signal]

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

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

17.13.3.14 escapePressed void escapePressed () [signal]

17.13.3.15 F10Pressed void F10Pressed () [signal]

17.13.3.16 F11Pressed void F11Pressed () [signal]

17.13.3.17 F12Pressed void F12Pressed () [signal]

17.13.3.18 F1Pressed void F1Pressed () [signal]

17.13.3.19 F2Pressed void F2Pressed () [signal]

17.13.3.20 F3Pressed void F3Pressed () [signal]

17.13.3.21 F4Pressed void F4Pressed () [signal]

17.13.3.22 F5Pressed void F5Pressed () [signal]

17.13.3.23 F6Pressed void F6Pressed () [signal]

17.13.3.24 F7Pressed void F7Pressed () [signal]

17.13.3.25 F8Pressed void F8Pressed () [signal]

17.13.3.26 F9Pressed void F9Pressed () [signal]

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

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

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

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

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

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

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

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

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

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

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

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

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

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

17.13.3.41 `selectAllPressed` void selectAllPressed () [signal]

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

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

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

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

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

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

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

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

17.13.3.50 shiftPressed void shiftPressed () [signal]

17.13.3.51 shiftReleased void shiftReleased () [signal]

17.13.3.52 showSettings void showSettings () [signal]

17.13.3.53 startBlinking void startBlinking () [slot]

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

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

17.13.3.56 stopBlinking void stopBlinking () [slot]

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

17.13.3.58 tabPressed void tabPressed () [signal]

17.13.3.59 undoPressed void undoPressed () [signal]

17.13.3.60 updateStyle() void updateStyle ()

17.13.3.61 upPressed void upPressed () [signal]

17.13.4 Member Data Documentation

17.13.4.1 blinkState bool blinkState

17.13.4.2 blinkTimer QTimer* blinkTimer

17.13.4.3 promptDivider QFrame* promptDivider

17.13.4.4 promptHistory CmdPromptHistory* promptHistory

17.13.4.5 promptInput CmdPromptInput* promptInput

17.13.4.6 promptSplitter CmdPromptSplitter* promptSplitter

17.13.4.7 promptVBoxLayout QVBoxLayout* promptVBoxLayout

17.13.4.8 styleHash QHash<QString, QString>* styleHash

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

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

17.14 CmdPromptHandle Class Reference

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

Signals

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

Public Member Functions

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

Public Attributes

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

Protected Member Functions

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

17.14.1 Detailed Description

17.14.2 Constructor & Destructor Documentation

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

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

17.14.3 Member Function Documentation

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

17.14.3.2 handlePressed void handlePressed (int y) [signal]

17.14.3.3 handleReleased void handleReleased (int y) [signal]

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

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

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

17.14.4 Member Data Documentation

17.14.4.1 moveY int moveY

17.14.4.2 pressY int pressY

17.14.4.3 releaseY int releaseY

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

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

17.15 CmdPromptHistory Class Reference

The Command Prompt History class.

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

Public Slots

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

Signals

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

Public Member Functions

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

Public Attributes

- int [tmpHeight](#)

Protected Member Functions

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

17.15.1 Detailed Description

The Command Prompt History class.

17.15.2 Constructor & Destructor Documentation

17.15.2.1 [CmdPromptHistory\(\)](#) [CmdPromptHistory](#) (QWidget * parent = 0)

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

17.15.3 Member Function Documentation

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

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

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

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

17.15.3.5 resizeHistory void resizeHistory (int y) [slot]

17.15.3.6 startResizeHistory void startResizeHistory (int y) [slot]

17.15.3.7 stopResizeHistory void stopResizeHistory (int y) [slot]

17.15.4 Member Data Documentation

17.15.4.1 tmpHeight int tmpHeight

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

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

17.16 CmdPromptInput Class Reference

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

Public Slots

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

Signals

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

Public Member Functions

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

Public Attributes

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

Protected Member Functions

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

Private Slots

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

17.16.1 Constructor & Destructor Documentation

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

17.16.1.2 ~CmdPromptInput() `~CmdPromptInput ()`

17.16.2 Member Function Documentation

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

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

17.16.2.3 applyFormatting() void applyFormatting ()

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

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

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

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

17.16.2.8 checkSelection void checkSelection () [slot]

17.16.2.9 clearFormatting() void clearFormatting ()

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

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

17.16.2.12 `copyPressed` void copyPressed () [signal]

17.16.2.13 `cutPressed` void cutPressed () [signal]

17.16.2.14 `deletePressed` void deletePressed () [signal]

17.16.2.15 `downPressed` void downPressed () [signal]

17.16.2.16 `endCommand` void endCommand () [slot]

17.16.2.17 `escapePressed` void escapePressed () [signal]

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

17.16.2.19 `F10Pressed` void F10Pressed () [signal]

17.16.2.20 `F11Pressed` void F11Pressed () [signal]

17.16.2.21 `F12Pressed` void F12Pressed () [signal]

17.16.2.22 F1Pressed void F1Pressed () [signal]

17.16.2.23 F2Pressed void F2Pressed () [signal]

17.16.2.24 F3Pressed void F3Pressed () [signal]

17.16.2.25 F4Pressed void F4Pressed () [signal]

17.16.2.26 F5Pressed void F5Pressed () [signal]

17.16.2.27 F6Pressed void F6Pressed () [signal]

17.16.2.28 F7Pressed void F7Pressed () [signal]

17.16.2.29 F8Pressed void F8Pressed () [signal]

17.16.2.30 F9Pressed void F9Pressed () [signal]

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

17.16.2.32 pastePressed void pastePressed () [signal]

17.16.2.33 processInput void processInput (void) [slot]

17.16.2.34 redoPressed void redoPressed () [signal]

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

17.16.2.36 selectAllPressed void selectAllPressed () [signal]

17.16.2.37 shiftPressed void shiftPressed () [signal]

17.16.2.38 shiftReleased void shiftReleased () [signal]

17.16.2.39 showSettings void showSettings () [signal]

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

17.16.2.41 stopBlinking void stopBlinking () [signal]

17.16.2.42 tabPressed void tabPressed () [signal]

17.16.2.43 undoPressed void undoPressed () [signal]

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

17.16.2.45 upPressed void upPressed () [signal]

17.16.3 Member Data Documentation

17.16.3.1 aliasHash QHash<QString, QString>* aliasHash

17.16.3.2 cmdActive bool cmdActive

17.16.3.3 curCmd QString curCmd

17.16.3.4 curText QString curText

17.16.3.5 defaultPrefix QString defaultPrefix

17.16.3.6 isBlinking bool isBlinking

17.16.3.7 lastCmd QString lastCmd

17.16.3.8 prefix QString prefix

17.16.3.9 rapidFireEnabled bool rapidFireEnabled

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

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

17.17 CmdPromptSplitter Class Reference

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

Signals

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

Public Member Functions

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

Protected Member Functions

- [QSplitterHandle * createHandle](#) ()

17.17.1 Detailed Description

17.17.2 Constructor & Destructor Documentation

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

17.17.3 Member Function Documentation

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

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

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

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

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

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

17.18 Compress Struct Reference

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

Public Attributes

- [int bit_position](#)
- [char * input_data](#)
- [int input_length](#)
- [int bits_total](#)
- [int block_elements](#)
- [huffman character_length_huffman](#)
- [huffman character_huffman](#)
- [huffman distance_huffman](#)

17.18.1 Member Data Documentation

17.18.1.1 `bit_position` `int bit_position`

17.18.1.2 bits_total int bits_total

17.18.1.3 block_elements int block_elements

17.18.1.4 character_huffman [huffman](#) character_huffman

17.18.1.5 character_length_huffman [huffman](#) character_length_huffman

17.18.1.6 distance_huffman [huffman](#) distance_huffman

17.18.1.7 input_data char* input_data

17.18.1.8 input_length int input_length

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

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

17.19 DimLeaderObject Class Reference

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

Public Types

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

Public Types inherited from [BaseObject](#)

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

Public Member Functions

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

Public Member Functions inherited from `BaseObject`

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

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

Public Attributes

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

Public Attributes inherited from `BaseObject`

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

Protected Member Functions

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

Protected Member Functions inherited from BaseObject

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

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

Enumerator

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

17.19.1.2 ArrowStyle [enum ArrowStyle](#)

Enumerator

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

17.19.1.3 lineStyle [enum lineStyle](#)

Enumerator

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

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

Returns the closest snap point to the mouse point

Todo generic closest point from list to point x.

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` (Painter *, const QStyleOptionGraphicsItem *, QWidget *)

Protected Member Functions inherited from BaseObject

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

Additional Inherited Members

Public Attributes inherited from [BaseObject](#)

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

17.20.1 Member Enumeration Documentation

17.20.1.1 anonymous enum [anonymous enum](#)

Enumerator

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

17.20.2 Constructor & Destructor Documentation

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

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

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

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

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

17.20.3 Member Function Documentation

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

Implements [BaseObject](#).

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

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

Implements [BaseObject](#).

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

Warning

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

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

Returns the closest snap point to the mouse point.

Implements [BaseObject](#).

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

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

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

17.20.3.8 objectQuadrant0() `QPointF` `objectQuadrant0 () const`

17.20.3.9 objectQuadrant180() `QPointF` `objectQuadrant180 () const`

17.20.3.10 objectQuadrant270() `QPointF` `objectQuadrant270 () const`

17.20.3.11 objectQuadrant90() `QPointF` `objectQuadrant90 () const`

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

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

17.20.3.14 objectSavePath() `QPainterPath` `objectSavePath () const`

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

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

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

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

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

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

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

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

Reimplemented from [BaseObject](#).

17.20.3.23 `updatePath()` void updatePath ()

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

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

Implements [BaseObject](#).

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

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

17.21 EmbAlignedDim_ Struct Reference

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

Public Attributes

- EmbVector position

17.21.1 Member Data Documentation

17.21.1.1 position EmbVector position

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

- extern/libembroidery/src/embroidery.h

17.22 EmbAngularDim_ Struct Reference

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

Public Attributes

- EmbVector position

17.22.1 Member Data Documentation

17.22.1.1 position EmbVector position

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

- extern/libembroidery/src/embroidery.h

17.23 EmbArc_ Struct Reference

absolute position (not relative)

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

Public Attributes

- `EmbVector start`
- `EmbVector mid`
- `EmbVector end`

17.23.1 Detailed Description

absolute position (not relative)

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

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

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

17.24 EmbArcLengthDim_ Struct Reference

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

Public Attributes

- `EmbVector position`

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

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

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

17.25 EmbArray_ Struct Reference

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

Public Attributes

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

17.25.1 Member Data Documentation

17.25.1.1 count `int count`

17.25.1.2 geometry `EmbGeometry* geometry`

17.25.1.3 length `int length`

17.25.1.4 stitch `EmbStitch* stitch`

17.25.1.5 thread `EmbThread* thread`

17.25.1.6 type `int type`

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

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

17.26 EmbBezier_Struct Reference

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

Public Attributes

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

17.26.1 Member Data Documentation

17.26.1.1 control1 EmbVector control1

17.26.1.2 control2 EmbVector control2

17.26.1.3 end EmbVector end

17.26.1.4 start EmbVector start

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

- extern/libembroidery/src/embroidery.h

17.27 EmbBlock_Struct Reference

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

Public Attributes

- EmbVector position

17.27.1 Member Data Documentation

17.27.1.1 **position** `EmbVector` position

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

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

17.28 EmbCircle_ Struct Reference

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

Public Attributes

- `EmbVector center`
- `EmbReal radius`

17.28.1 Member Data Documentation

17.28.1.1 **center** `EmbVector` center

17.28.1.2 **radius** `EmbReal` radius

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

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

17.29 EmbColor_ Struct Reference

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

Public Attributes

- `unsigned char r`
- `unsigned char g`
- `unsigned char b`

17.29.1 Detailed Description

EmbColor uses the light primaries: red, green, blue in that order.

17.29.2 Member Data Documentation

17.29.2.1 b unsigned char b

17.29.2.2 g unsigned char g

17.29.2.3 r unsigned char r

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

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

17.30 EmbDetailsDialog Class Reference

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

Public Member Functions

- `EmbDetailsDialog (QGraphicsScene *theScene, QWidget *parent=0)`
- `~EmbDetailsDialog ()`
- `void getInfo ()`
- `QWidget * createMainWidget ()`
- `QWidget * createHistogram ()`

Public Attributes

- `QWidget * mainWidget`
- `QDialogButtonBox * buttonBox`
- `quint32 stitchesTotal`
- `quint32 stitchesReal`
- `quint32 stitchesJump`
- `quint32 stitchesTrim`
- `quint32 colorTotal`
- `quint32 colorChanges`
- `QRectF boundingRect`

17.30.1 Detailed Description

17.30.2 Constructor & Destructor Documentation

17.30.2.1 EmbDetailsDialog() `EmbDetailsDialog (`
 `QGraphicsScene * theScene,`
 `QWidget * parent = 0)`

17.30.2.2 ~EmbDetailsDialog() `~EmbDetailsDialog ()`

17.30.3 Member Function Documentation

17.30.3.1 createHistogram() `QWidget * createHistogram ()`

17.30.3.2 createMainWidget() `QWidget * createMainWidget ()`

17.30.3.3 getInfo() `void getInfo ()`

17.30.4 Member Data Documentation

17.30.4.1 boundingRect `QRectF boundingRect`

17.30.4.2 buttonBox `QDialogButtonBox* buttonBox`

17.30.4.3 colorChanges `quint32 colorChanges`

17.30.4.4 colorTotal `quint32 colorTotal`

17.30.4.5 mainWidget QWidget* mainWidget

17.30.4.6 stitchesJump quint32 stitchesJump

17.30.4.7 stitchesReal quint32 stitchesReal

17.30.4.8 stitchesTotal quint32 stitchesTotal

17.30.4.9 stitchesTrim quint32 stitchesTrim

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

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

17.31 EmbDiameterDim_ Struct Reference

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

Public Attributes

- [EmbVector position](#)

17.31.1 Member Data Documentation

17.31.1.1 position [EmbVector](#) position

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

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

17.32 EmbEllipse_ Struct Reference

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

Public Attributes

- EmbVector center
- EmbVector radius
- EmbReal rotation

17.32.1 Member Data Documentation**17.32.1.1 center EmbVector center****17.32.1.2 radius EmbVector radius****17.32.1.3 rotation EmbReal rotation**

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

- extern/libembroidery/src/embroidery.h

17.33 EmbFormatList_ Struct Reference

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

Public Attributes

- char extension [2+EMBFORMAT_MAXEXT]
- char description [EMBFORMAT_MAXDESC]
- char reader_state
- char writer_state
- int type
- int color_only
- int check_for_color_file
- int write_external_color_file

17.33.1 Member Data Documentation**17.33.1.1 check_for_color_file int check_for_color_file**

17.33.1.2 color_only int color_only

17.33.1.3 description char description[EMBFORMAT_MAXDESC]

17.33.1.4 extension char extension[2+EMBFORMAT_MAXEXT]

17.33.1.5 reader_state char reader_state

17.33.1.6 type int type

17.33.1.7 write_external_color_file int write_external_color_file

17.33.1.8 writer_state char writer_state

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

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

17.34 EmbGeometry_ Struct Reference

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

Public Attributes

- union {
 - EmbArc arc
 - EmbCircle circle
 - EmbColor color
 - EmbEllipse ellipse
 - EmbLine line
 - EmbPath path
 - EmbPoint point
 - EmbPolygon polygon
 - EmbPolyline polyline
 - EmbRect rect
 - EmbSpline spline
 - EmbVector vector}
- EmbStitch stitch
- EmbThread thread
- int flag
- int type
- int lineType

17.34.1 Member Data Documentation

17.34.1.1 arc `EmbArc` arc

17.34.1.2 circle `EmbCircle` circle

17.34.1.3 color `EmbColor` color

17.34.1.4 ellipse `EmbEllipse` ellipse

17.34.1.5 flag int flag

17.34.1.6 line `EmbLine` line

17.34.1.7 lineType int lineType

17.34.1.8 union { ... } object

17.34.1.9 path `EmbPath` path

17.34.1.10 point `EmbPoint` point

17.34.1.11 polygon [EmbPolygon](#) polygon

17.34.1.12 polyline [EmbPolyline](#) polyline

17.34.1.13 rect [EmbRect](#) rect

17.34.1.14 spline [EmbSpline](#) spline

17.34.1.15 stitch [EmbStitch](#) stitch

17.34.1.16 thread [EmbThread](#) thread

17.34.1.17 type int type

17.34.1.18 vector [EmbVector](#) vector

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

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

17.35 EmbImage_ Struct Reference

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

Public Attributes

- [EmbVector position](#)
- [EmbVector dimensions](#)
- [unsigned char * data](#)
- [int width](#)
- [int height](#)
- [char path \[200\]](#)
- [char name \[200\]](#)

17.35.1 Member Data Documentation

17.35.1.1 data `unsigned char* data`

17.35.1.2 dimensions `EmbVector dimensions`

17.35.1.3 height `int height`

17.35.1.4 name `char name[200]`

17.35.1.5 path `char path[200]`

17.35.1.6 position `EmbVector position`

17.35.1.7 width `int width`

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

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

17.36 EmbInfiniteLine_ Struct Reference

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

Public Attributes

- `EmbVector position`

17.36.1 Member Data Documentation

17.36.1.1 position `EmbVector` position

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

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

17.37 EmbLayer_ Struct Reference

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

Public Attributes

- `char name [100]`
- `EmbArray * geometry`

17.37.1 Member Data Documentation

17.37.1.1 geometry `EmbArray*` geometry

17.37.1.2 name `char name[100]`

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

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

17.38 EmbLeaderDim_ Struct Reference

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

Public Attributes

- `EmbVector position`

17.38.1 Member Data Documentation

17.38.1.1 position `EmbVector` position

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

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

17.39 EmbLine_ Struct Reference

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

Public Attributes

- `EmbVector start`
- `EmbVector end`
- `int lineType`
- `EmbColor color`

17.39.1 Member Data Documentation

17.39.1.1 color `EmbColor` color

17.39.1.2 end `EmbVector` end

17.39.1.3 lineType `int` lineType

17.39.1.4 start `EmbVector` start

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

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

17.40 EmbLinearDim_ Struct Reference

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

Public Attributes

- EmbVector position

17.40.1 Member Data Documentation

17.40.1.1 position EmbVector position

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

- extern/libembroidery/src/embroidery.h

17.41 EmbOrdinateDim_ Struct Reference

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

Public Attributes

- EmbVector position

17.41.1 Member Data Documentation

17.41.1.1 position EmbVector position

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

- extern/libembroidery/src/embroidery.h

17.42 EmbPath_ Struct Reference

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

Public Attributes

- `EmbArray * pointList`
- `EmbArray * flagList`
- `int lineType`
- `EmbColor color`

17.42.1 Member Data Documentation**17.42.1.1 color `EmbColor` color****17.42.1.2 flagList `EmbArray*` flagList****17.42.1.3 lineType `int` lineType****17.42.1.4 pointList `EmbArray*` pointList**

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

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

17.43 EmbPattern_ Struct Reference

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

Public Attributes

- `unsigned int dstJumpsPerTrim`
- `EmbVector home`
- `EmbReal hoop_width`
- `EmbReal hoop_height`
- `EmbArray * thread_list`
- `EmbArray * stitch_list`
- `EmbArray * geometry`
- `EmbLayer layer [EMB_MAX_LAYERS]`
- `int currentColorIndex`

17.43.1 Member Data Documentation

17.43.1.1 currentColorIndex int currentColorIndex

17.43.1.2 dstJumpsPerTrim unsigned int dstJumpsPerTrim

17.43.1.3 geometry EmbArray* geometry

17.43.1.4 home EmbVector home

17.43.1.5 hoop_height EmbReal hoop_height

17.43.1.6 hoop_width EmbReal hoop_width

17.43.1.7 layer EmbLayer layer[EMB_MAX_LAYERS]

17.43.1.8 stitch_list EmbArray* stitch_list

17.43.1.9 thread_list EmbArray* thread_list

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

- extern/libembroidery/src/embroidery.h

17.44 EmbPoint_ Struct Reference

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

Public Attributes

- EmbVector position
- int lineType
- EmbColor color

17.44.1 Member Data Documentation

17.44.1.1 color EmbColor color

17.44.1.2 lineType int lineType

17.44.1.3 position EmbVector position

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

- extern/libembroidery/src/embroidery.h

17.45 EmbRadiusDim_ Struct Reference

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

Public Attributes

- EmbVector position

17.45.1 Member Data Documentation

17.45.1.1 position EmbVector position

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

- extern/libembroidery/src/embroidery.h

17.46 EmbRay_ Struct Reference

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

Public Attributes

- EmbVector position

17.46.1 Member Data Documentation

17.46.1.1 position EmbVector position

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

- extern/libembroidery/src/embroidery.h

17.47 EmbRect_ Struct Reference

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

Public Attributes

- EmbReal top
- EmbReal left
- EmbReal bottom
- EmbReal right
- EmbReal rotation
- EmbReal radius

17.47.1 Member Data Documentation

17.47.1.1 bottom EmbReal bottom

17.47.1.2 left EmbReal left

17.47.1.3 radius `EmbReal` `radius`

17.47.1.4 right `EmbReal` `right`

17.47.1.5 rotation `EmbReal` `rotation`

17.47.1.6 top `EmbReal` `top`

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

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

17.48 EmbSatinOutline_ Struct Reference

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

Public Attributes

- `int length`
- `EmbArray * side1`
- `EmbArray * side2`

17.48.1 Member Data Documentation

17.48.1.1 length `int` `length`

17.48.1.2 side1 `EmbArray*` `side1`

17.48.1.3 side2 `EmbArray*` `side2`

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

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

17.49 EmbSpline_Struct Reference

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

Public Attributes

- `EmbArray * beziers`

17.49.1 Member Data Documentation

17.49.1.1 `beziers` `EmbArray* beziers`

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

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

17.50 EmbStitch_Struct Reference

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

Public Attributes

- `int flags`
- `EmbReal x`
- `EmbReal y`
- `int color`

17.50.1 Member Data Documentation

17.50.1.1 `color` `int color`

positive is up, units are in mm

17.50.1.2 `flags` `int flags`

17.50.1.3 x EmbReal x

uses codes defined above

17.50.1.4 y EmbReal y

absolute position (not relative)

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

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

17.51 EmbTextMulti_ Struct Reference

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

Public Attributes

- `EmbVector position`
- `char text [200]`

17.51.1 Member Data Documentation

17.51.1.1 position EmbVector position

17.51.1.2 text char text[200]

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

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

17.52 EmbTextSingle_ Struct Reference

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

Public Attributes

- `EmbVector position`
- `char text [200]`

17.52.1 Member Data Documentation**17.52.1.1 position** `EmbVector position`**17.52.1.2 text** `char text[200]`

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

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

17.53 EmbThread_Struct Reference

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

Public Attributes

- `EmbColor color`
- `char description [50]`
- `char catalogNumber [30]`

17.53.1 Member Data Documentation**17.53.1.1 catalogNumber** `char catalogNumber[30]`**17.53.1.2 color** `EmbColor color`**17.53.1.3 description** `char description[50]`

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

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

17.54 EmbTime_ Struct Reference

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

Public Attributes

- unsigned int **year**
- unsigned int **month**
- unsigned int **day**
- unsigned int **hour**
- unsigned int **minute**
- unsigned int **second**

17.54.1 Member Data Documentation

17.54.1.1 day unsigned int day

17.54.1.2 hour unsigned int hour

17.54.1.3 minute unsigned int minute

17.54.1.4 month unsigned int month

17.54.1.5 second unsigned int second

17.54.1.6 year unsigned int year

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

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

17.55 EmbVector_ Struct Reference

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

Public Attributes

- EmbReal x
- EmbReal y

17.55.1 Detailed Description

The basic type to represent points absolutely or represent directions.

Positive y is up, units are in mm.

17.55.2 Member Data Documentation

17.55.2.1 x EmbReal x

17.55.2.2 y EmbReal y

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

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

17.56 EmbView_ Struct Reference

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

Public Attributes

- `EmbPattern * pattern`
- `EmbVector origin`
- `EmbReal scale`
- `QString grid_type`
- `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`
- `QString text_font`
- `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`
- `QString filename`
- `QStringList 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` `QString filename`

17.56.3.2 `grid_mode` `bool grid_mode`

17.56.3.3 grid_type `QString grid_type`

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 QString text_font

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` QStringList `undo_history`

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

- embroidermodder2/[embroidermodder.h](#)

17.57 GroupBoxData_ Struct Reference

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

Public Attributes

- int `object`
- char `key` [200]
- char `icon_name` [200]
- char `label` [200]
- char `type` [200]
- char `map_signal` [200]

17.57.1 Member Data Documentation

17.57.1.1 `icon_name` char `icon_name`[200]

17.57.1.2 key char key[200]

17.57.1.3 label char label[200]

17.57.1.4 map_signal char map_signal[200]

17.57.1.5 object int object

17.57.1.6 type char type[200]

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

- embroidermodder2/[embroidermodder.h](#)

17.58 hoop_padding Struct Reference

Public Attributes

- int [left](#)
- int [right](#)
- int [top](#)
- int [bottom](#)

17.58.1 Member Data Documentation

17.58.1.1 bottom int bottom

17.58.1.2 left int left

17.58.1.3 right int right

17.58.1.4 top int top

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

- [extern/libembroidery/src/formats/format_jef.c](#)

17.59 Huffman Struct Reference

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

Public Attributes

- int [default_value](#)
- int [lengths](#) [1000]
- int [nlenghts](#)
- int [table](#) [1000]
- int [table_width](#)
- int [ntable](#)

17.59.1 Member Data Documentation

17.59.1.1 **default_value** int default_value

17.59.1.2 **lengths** int lengths[1000]

17.59.1.3 **nlenghts** int nlenghts

17.59.1.4 **ntable** int ntable

17.59.1.5 **table** int table[1000]

17.59.1.6 table_width int table_width

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

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

17.60 ImageObject Class Reference

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

Public Types

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

Public Types inherited from [BaseObject](#)

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

Public Member Functions

- [ImageObject \(EmbReal x, EmbReal y, EmbReal w, EmbReal h, QRgb rgb, QGraphicsItem *parent=0\)](#)
- [ImageObject \(ImageObject *obj, QGraphicsItem *parent=0\)](#)
- [~ImageObject \(\)](#)
- void [init \(EmbReal x, EmbReal y, EmbReal w, EmbReal h, QRgb rgb, Qt::PenStyle lineType\)](#)
- void [updatePath \(\)](#)
- virtual int [type \(\) const](#)
- QPointF [objectTopLeft \(\) const](#)
- QPointF [objectTopRight \(\) const](#)
- QPointF [objectBottomLeft \(\) const](#)
- QPointF [objectBottomRight \(\) const](#)
- EmbReal [objectWidth \(\) const](#)
- EmbReal [objectHeight \(\) const](#)
- EmbReal [objectArea \(\) const](#)
- void [setObjectRect \(EmbReal x, EmbReal y, EmbReal w, EmbReal h\)](#)
- void [updateRubber \(QPainter *painter=0\)](#)
- virtual void [vulcanize \(\)](#)
- virtual QPointF [mouseSnapPoint \(const QPointF &mousePoint\)](#)
- virtual QList< QPointF > [allGripPoints \(\)](#)
- virtual void [gripEdit \(const QPointF &before, const QPointF &after\)](#)

Public Member Functions inherited from BaseObject

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

Protected Member Functions

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

Protected Member Functions inherited from BaseObject

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

Additional Inherited Members

Public Attributes inherited from [BaseObject](#)

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

17.60.1 Member Enumeration Documentation

17.60.1.1 anonymous enum [anonymous enum](#)

Enumerator

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

17.60.2 Constructor & Destructor Documentation

17.60.2.1 [ImageObject\(\)](#) [1/2] [ImageObject](#) (

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

17.60.2.2 [ImageObject\(\)](#) [2/2] [ImageObject](#) (

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

17.60.2.3 [~ImageObject\(\)](#) [~ImageObject](#) ()

17.60.3 Member Function Documentation

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

Implements [BaseObject](#).

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

Implements [BaseObject](#).

17.60.3.3 init() `void init (EmbReal x, EmbReal y, EmbReal w, EmbReal h, QRgb rgb, Qt::PenStyle lineType)`

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

Implements [BaseObject](#).

17.60.3.5 objectArea() `EmbReal objectArea () const [inline]`

17.60.3.6 objectBottomLeft() `QPointF objectBottomLeft () const`

17.60.3.7 objectBottomRight() `QPointF objectBottomRight () const`

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

17.60.3.9 objectTopLeft() `QPointF objectTopLeft () const`

17.60.3.10 objectTopRight() `QPointF objectTopRight () const`

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

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

17.60.3.13 setObjectRect() `void setObjectRect (`
 `EmbReal x,`
 `EmbReal y,`
 `EmbReal w,`
 `EmbReal h)`

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

Reimplemented from [BaseObject](#).

17.60.3.15 updatePath() `void updatePath ()`

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

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

Implements [BaseObject](#).

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

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

17.61 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.

17.61.1 Detailed Description

17.61.2 Constructor & Destructor Documentation

17.61.2.1 `ImageWidget()` `ImageWidget (`
 `const QString & filename,`
 `QWidget * parent = 0)`

ImageWidget::ImageWidget.

Parameters

<i>filename</i>	<input type="text"/>
<i>parent</i>	<input type="text"/>

17.61.2.2 ~ImageWidget() ~ImageWidget ()

ImageWidget::~ImageWidget.

17.61.3 Member Function Documentation**17.61.3.1 load()** bool load (const QString & *fileName*)

ImageWidget::load.

Parameters

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

Returns**17.61.3.2 paintEvent()** void paintEvent (QPaintEvent * *event*) [protected]

ImageWidget::paintEvent.

17.61.3.3 save() bool save (const QString & *fileName*)

ImageWidget::save.

Parameters

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

Returns

17.61.4 Member Data Documentation

17.61.4.1 img QImage img

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

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

17.62 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](#)

17.62.1 Detailed Description

17.62.2 Constructor & Destructor Documentation

17.62.2.1 LayerManager() [LayerManager \(](#)

```
MainWindow * mw,
QWidget * parent = 0 )
```

LayerManager::LayerManager.

Parameters

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

17.62.2.2 ~LayerManager() ~[LayerManager](#) ()

[LayerManager](#)::~[LayerManager](#).

17.62.3 Member Function Documentation

17.62.3.1 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>	

17.62.4 Member Data Documentation**17.62.4.1 layerModel** QStandardItemModel* layerModel

17.62.4.2 layerModelSorted QSortFilterProxyModel* layerModelSorted

17.62.4.3 treeView QTreeView* treeView

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

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

17.63 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 [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](#)

17.63.1 Member Enumeration Documentation

17.63.1.1 anonymous enum [anonymous enum](#)

Enumerator

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

17.63.2 Constructor & Destructor Documentation

17.63.2.1 [LineObject\(\)](#) [1/2] [LineObject](#) (

```
    EmbReal x1,  
    EmbReal y1,  
    EmbReal x2,  
    EmbReal y2,  
    QRgb rgb,  
    QGraphicsItem * parent = 0 )
```

17.63.2.2 [LineObject\(\)](#) [2/2] [LineObject](#) (

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

17.63.2.3 [~LineObject\(\)](#) [~LineObject](#) ()

17.63.3 Member Function Documentation

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

Implements [BaseObject](#).

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

Implements [BaseObject](#).

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

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

Implements [BaseObject](#).

17.63.3.5 objectAngle() `EmbReal objectAngle () const`

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

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

17.63.3.8 objectEndPoint1() QPointF objectEndPoint1 () const [inline]

17.63.3.9 objectEndPoint2() QPointF objectEndPoint2 () const

17.63.3.10 objectLength() EmbReal objectLength () const [inline]

17.63.3.11 objectMidPoint() QPointF objectMidPoint () const

17.63.3.12 objectSavePath() QPainterPath objectSavePath () const

17.63.3.13 objectX1() EmbReal objectX1 () const [inline]

17.63.3.14 objectX2() EmbReal objectX2 () const [inline]

17.63.3.15 objectY1() EmbReal objectY1 () const [inline]

17.63.3.16 objectY2() EmbReal objectY2 () const [inline]

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

17.63.3.18 setObjectEndPoint1() [1/2] void setObjectEndPoint1 (
 const QPointF & endPtl)

17.63.3.19 setObjectEndPoint1() [2/2] void setObjectEndPoint1 (EmbReal x1, EmbReal y1)

17.63.3.20 setObjectEndPoint2() [1/2] void setObjectEndPoint2 (const QPointF & endPt2)

17.63.3.21 setObjectEndPoint2() [2/2] void setObjectEndPoint2 (EmbReal x2, EmbReal y2)

17.63.3.22 setObjectX1() void setObjectX1 (EmbReal x) [inline]

17.63.3.23 setObjectX2() void setObjectX2 (EmbReal x) [inline]

17.63.3.24 setObjectY1() void setObjectY1 (EmbReal y) [inline]

17.63.3.25 setObjectY2() void setObjectY2 (EmbReal y) [inline]

17.63.3.26 type() virtual int type () const [inline], [virtual]

Reimplemented from [BaseObject](#).

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

17.63.3.28 vulcanize() void vulcanize () [virtual]

Implements [BaseObject](#).

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

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

17.64 LSYSTEM Struct Reference

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

Public Attributes

- char [axiom](#)
- char * [alphabet](#)
- char * [constants](#)
- char ** [rules](#)

17.64.1 Member Data Documentation

17.64.1.1 **alphabet** char* alphabet

17.64.1.2 **axiom** char axiom

17.64.1.3 **constants** char* constants

17.64.1.4 **rules** char** rules

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

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

17.65 MainWindow Class Reference

The [MainWindow](#) class.

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

Public Slots

- void `enablePromptRapidFire ()`
- void `disablePromptRapidFire ()`
- void `enableMoveRapidFire ()`
- void `disableMoveRapidFire ()`
- void `onCloseWindow ()`
`MainWindow::onCloseWindow.`
- virtual void `onCloseMdiWin (MdiWindow *)`
`MainWindow::onCloseMdiWin.`
- void `recentMenuAboutToShow ()`
`MainWindow::recentMenuAboutToShow.`
- void `onWindowActivated (QMdiSubWindow *w)`
`MainWindow::onWindowActivated.`
- void `windowMenuAboutToShow ()`
`MainWindow::windowMenuAboutToShow.`
- void `windowMenuActivated (bool checked)`
`MainWindow::windowMenuActivated.`
- QAction * `getAction (int actionEnum)`
`MainWindow::getAction.`
- void `updateAllViewScrollBars (bool val)`
- void `updateAllViewCrossHairColors (QRgb color)`
- void `updateAllViewBackgroundColors (QRgb color)`
- void `updateAllViewSelectBoxColors (QRgb colorL, QRgb fillL, QRgb colorR, QRgb fillR, int alpha)`
- void `updateAllViewGridColors (QRgb color)`
- void `updateAllViewRulerColors (QRgb color)`
- void `updatePickAddMode (bool val)`
- void `pickAddModeToggled ()`
- void `settingsPrompt ()`
- void `settingsDialog (const QString &showTab=QString())`
- void `readSettings ()`
`MainWindow::readSettings.`
- void `writeSettings ()`
`MainWindow::writeSettings.`
- static bool `validFileFormat (const QString &fileName)`
`MainWindow::validFileFormat.`
- void `stub_implement (QString txt)`
`MainWindow::stub_implement txt.`
- void `stub_testing ()`
`MainWindow::stub_testing.`
- void `promptHistoryAppended (const QString &txt)`
- void `logPromptInput (const QString &txt)`
- void `promptInputPrevious ()`
- void `promptInputNext ()`
- void `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 `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)`
- 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 `getCurrentLineWeight ()`
- void `undo ()`
- void `redo ()`
- bool `isShiftPressed ()`
- void `setShiftPressed ()`
- void `setShiftReleased ()`
- void `deletePressed ()`
- void `escapePressed ()`

- void `makeLayerActive` ()
- void `layerManager` ()
- void `layerPrevious` ()
- void `zoomRealtime` ()
- void `zoomPrevious` ()
- void `zoomWindow` ()
- void `zoomDynamic` ()
- void `zoomScale` ()
- void `zoomCenter` ()
- void `zoomIn` ()
- void `zoomOut` ()
- void `zoomSelected` ()
- void `zoomAll` ()
- void `zoomExtents` ()
- void `panrealtime` ()
- void `panpoint` ()
- void `panLeft` ()
- void `panRight` ()
- void `panUp` ()
- void `panDown` ()
 MainWindow::panDown.
- void `dayVision` ()
 MainWindow::dayVision.
- void `nightVision` ()
 MainWindow::nightVision.
- void `doNothing` ()

Public Member Functions

- `MainWindow` ()
 MainWindow::MainWindow.
- `~MainWindow` ()
 MainWindow::~MainWindow.
- `MdiArea * getMdiArea` ()
 MainWindow::getMdiArea.
- `MainWindow * getApplication` ()
 MainWindow::getApplication.
- `MdiWindow * activeMdiWindow` ()
- `View * activeView` ()
- `QGraphicsScene * activeScene` ()
- `QUndoStack * activeUndoStack` ()
- void `setUndoCleanIcon` (bool opened)
- virtual void `updateMenuToolbarStatusbar` ()
 MainWindow::updateMenuToolbarStatusbar.
- std::string `actuator` (std::string command)
 MainWindow::actuator.
- std::string `run_script_file` (std::string fname)
 MainWindow::run_script_file.
- std::string `run_script` (std::vector< std::string > script)
 A basic line-by-line script processor to allow for extensions to the program.
- void `LoadCommand` (QString cmdName)
- bool `isCommandActive` ()

- `QString activeCommand ()`
- `QIcon create_icon (QString stub)`
- `void create_toolbar (QToolBar *toolbar, std::string label, std::vector< std::string > entries)`
- `QString platformString ()`
- `void nativeAlert (const QString &txt)`
- `void nativeBlinkPrompt ()`
- `void nativeSetPromptPrefix (const QString &txt)`
- `void nativeAppendPromptHistory (const QString &txt)`
- `void nativeEnablePromptRapidFire ()`
- `void nativeDisablePromptRapidFire ()`
- `void nativeInitCommand ()`
- `void nativeEndCommand ()`
- `void nativeEnableMoveRapidFire ()`
- `void nativeDisableMoveRapidFire ()`
- `void nativeExit ()`
- `void nativeTipOfTheDay ()`
- `void nativeMessageBox (const QString &type, const QString &title, const QString &text)`
MainWindow::nativeMessageBox type title text.
- `void nativePrintArea (EmbReal x, EmbReal y, EmbReal w, EmbReal h)`
MainWindow::nativePrintArea x y w h.
- `void nativeSetBackgroundColor (uint8_t r, uint8_t g, uint8_t b)`
- `void nativeSetCrossHairColor (uint8_t r, uint8_t g, uint8_t b)`
- `void nativeSetGridColor (uint8_t r, uint8_t g, uint8_t b)`
- `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 nativeAddToSelection (const QPainterPath path, Qt::ItemSelectionMode mode)
- void nativeClearSelection ()
- void nativeDeleteSelected ()

MainWindow::nativeDeleteSelected.
- void nativeCutSelected (EmbReal x, EmbReal y)

MainWindow::nativeCutSelected x y.
- void nativeCopySelected (EmbReal x, EmbReal y)

MainWindow::nativeCopySelected x y.
- void nativePasteSelected (EmbReal x, EmbReal y)

MainWindow::nativePasteSelected x y.
- void nativeMoveSelected (EmbReal dx, EmbReal dy)

MainWindow::nativeMoveSelected dx dy.
- void nativeScaleSelected (EmbReal x, EmbReal y, EmbReal factor)

MainWindow::nativeScaleSelected x y factor.
- void nativeRotateSelected (EmbReal x, EmbReal y, EmbReal rot)

MainWindow::nativeRotateSelected x y rot.
- void nativeMirrorSelected (EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2)

MainWindow::nativeMirrorSelected x1 y1 x2 y2.
- EmbReal nativeQSnapX ()

MainWindow::nativeQSnapX.
- EmbReal nativeQSnapY ()

MainWindow::nativeQSnapY.
- EmbReal nativeMouseX ()

MainWindow::nativeMouseX.
- EmbReal nativeMouseY ()

MainWindow::nativeMouseY.

Public Attributes

- 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.

17.65.1 Detailed Description

The `MainWindow` class.

17.65.2 Constructor & Destructor Documentation

17.65.2.1 `MainWindow()` *MainWindow ()*

`MainWindow::MainWindow.`

17.65.2.2 `~MainWindow()` *~MainWindow ()*

`MainWindow::~MainWindow.`

17.65.3 Member Function Documentation

17.65.3.1 `about` `void about () [slot]`

17.65.3.2 `activeCommand()` `QString activeCommand () [inline]`

17.65.3.3 `activeMdiWindow()` `MdiWindow * activeMdiWindow ()`

17.65.3.4 `activeScene()` `QGraphicsScene * activeScene ()`

17.65.3.5 `activeUndoStack()` `QUndoStack * activeUndoStack ()`

17.65.3.6 `activeView()` `View * activeView ()`

17.65.3.7 `actuator()` `std::string actuator (std::string line)`

MainWindow::actuator.

Parameters

`command`

17.65.3.8 RUN COMMAND `QAction* act = qobject_cast<QAction*>(sender()); if (act) { qDebug("runCommand(%s)", qPrintable(act->objectName())); prompt->endCommand(); prompt->setCurrentText(act->objectName()); prompt->processInput(); }`

17.65.3.9 INIT `qDebug("runCommandMain(%s)", qPrintable(cmd)); QString fileName = "commands/" + cmd + "/" + cmd + ".js"; if (!getSettingsSelectionModePickFirst()) { nativeClearSelection(); } TODO: Uncomment this line when post-selection is available engine->evaluate(cmd + "_main()", fileName);`

17.65.3.10 `PROMPT` `qDebug("runCommandPrompt(%s, %s)", qPrintable(cmd), qPrintable(str));` `QString fileName = "commands/" + cmd + "/" + cmd + ".js";` NOTE: Replace any special characters that will cause a syntax error `QString safeStr = str; safeStr.replace("\\\\", "\\"); safeStr.replace("\\", "\\");` if (`prompt->isRapidFireEnabled()`) { `engine->evaluate(cmd + "_prompt(\"" + safeStr + "\")", fileName);`} else { `engine->evaluate(cmd + "_prompt(\"" + safeStr.toUpperCase() + "\")", fileName);`}

17.65.3.11 `buttonTipOfTheDayClicked` `void buttonTipOfTheDayClicked (`
`int button) [slot]`

17.65.3.12 `changelog` `void changelog () [slot]`

17.65.3.13 `checkBoxTipOfTheDayStateChanged` `void checkBoxTipOfTheDayStateChanged (`
`int checked) [slot]`

17.65.3.14 `checkForUpdates` `void checkForUpdates () [slot]`

17.65.3.15 `closeEvent()` `void closeEvent (`
`QCloseEvent * event) [protected]`

MainWindow::closeEvent.

Parameters

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

17.65.3.16 `closeToolBar` `void closeToolBar (`
`QAction * action) [slot]`

MainWindow::closeToolBar.

Parameters

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

17.65.3.17 `colorSelectorIndexChanged` `void colorSelectorIndexChanged (`
`int index) [slot]`

17.65.3.18 copy void copy () [slot]

17.65.3.19 create_icon() QIcon create_icon (QString *stub*)

17.65.3.20 create_toolbar() void create_toolbar (QToolBar * *toolbar*, std::string *label*, std::vector< std::string > *entries*)

17.65.3.21 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 ... <.<

17.65.3.22 createAllMenus() void createAllMenus () [protected]

MainWindow::createAllMenus.

17.65.3.24 createEditMenu() void createEditMenu () [protected]

17.65.3.25 `createFileMenu()` void createFileMenu () [protected]

[MainWindow::createFileMenu](#).

17.65.3.26 `createHelpMenu()` void createHelpMenu () [protected]

17.65.3.27 `createHelpToolbar()` void createHelpToolbar () [protected]

17.65.3.28 `createIconToolbar()` void createIconToolbar () [protected]

17.65.3.29 `createLayerToolbar()` void createLayerToolbar () [protected]

[MainWindow::createLayerToolbar](#).

17.65.3.30 `createPanToolbar()` void createPanToolbar () [protected]

17.65.3.31 `createPromptToolbar()` void createPromptToolbar () [protected]

[MainWindow::createPromptToolbar](#).

17.65.3.32 `createPropertiesToolbar()` void createPropertiesToolbar () [protected]

[MainWindow::createPropertiesToolbar](#).

17.65.3.33 `createSettingsMenu()` void createSettingsMenu () [protected]

17.65.3.34 `createTextToolbar()` void createTextToolbar () [protected]

[MainWindow::createTextToolbar](#).

17.65.3.35 `createViewMenu()` void createViewMenu () [protected]

17.65.3.36 `createWindowMenu()` void createWindowMenu () [protected]

17.65.3.37 `cut` void cut () [slot]

17.65.3.38 `dayVision` void dayVision () [slot]

[MainWindow::dayVision](#).

17.65.3.39 `deletePressed` void deletePressed () [slot]

17.65.3.40 `designDetails` void designDetails () [slot]

17.65.3.41 `disableMoveRapidFire` void disableMoveRapidFire () [slot]

17.65.3.42 `disablePromptRapidFire` void disablePromptRapidFire () [slot]

17.65.3.43 `doNothing` void doNothing () [slot]

17.65.3.44 `enableMoveRapidFire` void enableMoveRapidFire () [slot]

17.65.3.45 `enablePromptRapidFire` void enablePromptRapidFire () [slot]

17.65.3.46 `escapePressed` void escapePressed () [slot]

17.65.3.47 `exit` void exit () [slot]

[MainWindow::exit](#).

17.65.3.48 `findMdiWindow()` QMdiSubWindow * findMdiWindow (const QString & fileName) [protected]

[MainWindow::findMdiWindow](#).

Parameters

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

Returns

17.65.3.49 floatingChangedToolBar void floatingChangedToolBar (bool *isFloating*) [slot]

[MainWindow::floatingChangedToolBar.](#)

Parameters

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

17.65.3.50 getAction QAction * getAction (int *actionEnum*) [slot]

[MainWindow::getAction.](#)

Parameters

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

Returns

17.65.3.51 getApplication() MainWindow * getApplication ()

[MainWindow::getApplication.](#)

Returns

17.65.3.52 getCurrentColor QRgb getCurrentColor () [slot]

17.65.3.53 `getCurrentLayer` `QString getCurrentLayer () [slot]`

17.65.3.54 `getCurrentLineType` `QString getCurrentLineType () [slot]`

17.65.3.55 `getCurrentLineWeight` `QString getCurrentLineWeight () [slot]`

17.65.3.56 `getFileSeparator()` `QAction * getFileSeparator () [protected]`

[MainWindow::getFileSeparator](#).

Returns

17.65.3.57 `getMdiArea()` `MdiArea * getMdiArea ()`

[MainWindow::getMdiArea](#).

Returns

17.65.3.58 `help` `void help () [slot]`

17.65.3.59 `hideUnimplemented` `void hideUnimplemented () [private], [slot]`

[MainWindow::hideUnimplemented](#).

17.65.3.60 `iconResize` `void iconResize (int iconSize) [slot]`

17.65.3.61 `isCommandActive()` `bool isCommandActive () [inline]`

17.65.3.62 `isShiftPressed` `bool isShiftPressed () [slot]`

17.65.3.63 `layerManager` `void layerManager () [slot]`

17.65.3.64 `layerPrevious` `void layerPrevious () [slot]`

17.65.3.65 `layerSelectorIndexChanged` `void layerSelectorIndexChanged (int index) [slot]`

17.65.3.66 `linetypeSelectorIndexChanged` `void linetypeSelectorIndexChanged (int index) [slot]`

17.65.3.67 `lineweightSelectorIndexChanged` `void linewidthSelectorIndexChanged (int index) [slot]`

17.65.3.68 `LoadCommand()` `void LoadCommand (QString cmdName)`

NOTE: Every QScriptProgram must have a unique function name to call. If every function was called `main()`, then the QScriptEngine would only call the last script evaluated (which happens to be `main()` in another script). Thus, by adding the cmdName before `main()`, it becomes `line_main()`, `circle_main()`, etc... Do not change this code unless you really know what you are doing. I mean it.

17.65.3.69 `loadFormats()` `void loadFormats () [protected]`

[MainWindow::loadFormats](#).

17.65.3.70 `logPromptInput` `void logPromptInput (const QString & txt) [slot]`

17.65.3.71 makeLayerActive void makeLayerActive () [slot]

17.65.3.72 nativeAddArc() void nativeAddArc (

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

17.65.3.73 nativeAddCircle() void nativeAddCircle (

```
EmbReal centerX,
EmbReal centerY,
EmbReal radius,
bool fill,
int rubberMode )
```

17.65.3.74 nativeAddDimLeader() void nativeAddDimLeader (

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

17.65.3.75 nativeAddEllipse() void nativeAddEllipse (

```
EmbReal centerX,
EmbReal centerY,
EmbReal width,
EmbReal height,
EmbReal rot,
bool fill,
int rubberMode )
```

17.65.3.76 nativeAddHorizontalDimension() void nativeAddHorizontalDimension (

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

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

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

```
17.65.3.79 nativeAddLine() void nativeAddLine (
    EmbReal x1,
    EmbReal y1,
    EmbReal x2,
    EmbReal y2,
    EmbReal rot,
    int rubberMode )
```

```
17.65.3.80 nativeAddPath() void nativeAddPath (
    EmbReal startX,
    EmbReal startY,
    const QPainterPath & p,
    int rubberMode )
```

```
17.65.3.81 nativeAddPoint() void nativeAddPoint (
    EmbReal x,
    EmbReal y )
```

```
17.65.3.82 nativeAddPolygon() void nativeAddPolygon (
    EmbReal startX,
    EmbReal startY,
    const QPainterPath & p,
    int rubberMode )
```

```
17.65.3.83 nativeAddPolyline() void nativeAddPolyline (
    EmbReal startX,
    EmbReal startY,
    const QPainterPath & p,
    int rubberMode )
```

```
17.65.3.84 nativeAddRay() void nativeAddRay (
    EmbReal x1,
    EmbReal y1,
    EmbReal x2,
    EmbReal y2,
    EmbReal rot )
```

```
17.65.3.85 nativeAddRectangle() void nativeAddRectangle (
    EmbReal x,
    EmbReal y,
    EmbReal w,
    EmbReal h,
    EmbReal rot,
    bool fill,
    int rubberMode )
```

```
17.65.3.86 nativeAddRegularPolygon() void nativeAddRegularPolygon (
    EmbReal centerX,
    EmbReal centerY,
    quint16 sides,
    uint8_t mode,
    EmbReal rad,
    EmbReal rot,
    bool fill )
```

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

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

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

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

```
17.65.3.91 nativeAddToSelection() void nativeAddToSelection (  
    const QPainterPath path,  
    Qt::ItemSelectionMode mode )
```

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

17.65.3.93 nativeAddVerticalDimension() void nativeAddVerticalDimension (

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

17.65.3.94 nativeAlert() void nativeAlert (

```
const QString & txt )
```

17.65.3.95 nativeAllowRubber() bool nativeAllowRubber ()

17.65.3.96 nativeAppendPromptHistory() void nativeAppendPromptHistory (

```
const QString & txt )
```

17.65.3.97 nativeBlinkPrompt() void nativeBlinkPrompt ()

17.65.3.98 nativeCalculateAngle() EmbReal nativeCalculateAngle (

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

17.65.3.99 nativeCalculateDistance() EmbReal nativeCalculateDistance (

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

17.65.3.100 nativeClearRubber() void nativeClearRubber ()

17.65.3.101 nativeClearSelection() void nativeClearSelection ()

17.65.3.102 nativeCopySelected() void nativeCopySelected (

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

MainWindow::nativeCopySelected x y.

17.65.3.103 nativeCutSelected() void nativeCutSelected (

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

MainWindow::nativeCutSelected x y.

17.65.3.104 nativeDeleteSelected() void nativeDeleteSelected ()

MainWindow::nativeDeleteSelected.

17.65.3.105 nativeDisableMoveRapidFire() void nativeDisableMoveRapidFire ()

17.65.3.106 nativeDisablePromptRapidFire() void nativeDisablePromptRapidFire ()

17.65.3.107 nativeEnableMoveRapidFire() void nativeEnableMoveRapidFire ()

17.65.3.108 nativeEnablePromptRapidFire() void nativeEnablePromptRapidFire ()

17.65.3.109 nativeEndCommand() void nativeEndCommand ()

17.65.3.110 nativeExit() void nativeExit ()

17.65.3.111 nativeInitCommand() void nativeInitCommand ()

17.65.3.112 nativeMessageBox() void nativeMessageBox (
 const QString & type,
 const QString & title,
 const QString & text)

MainWindow::nativeMessageBox *type title text.*

17.65.3.113 nativeMirrorSelected() void nativeMirrorSelected (
 EmbReal x1,
 EmbReal y1,
 EmbReal x2,
 EmbReal y2)

MainWindow::nativeMirrorSelected *x1 y1 x2 y2.*

17.65.3.114 nativeMouseX() EmbReal nativeMouseX ()

MainWindow::nativeMouseX.

Returns

17.65.3.115 nativeMouseY() EmbReal nativeMouseY ()

MainWindow::nativeMouseY.

Returns

17.65.3.116 nativeMoveSelected() void nativeMoveSelected (
 EmbReal dx,
 EmbReal dy)

MainWindow::nativeMoveSelected *dx dy.*

17.65.3.117 nativeNumSelected() int nativeNumSelected ()

17.65.3.118 nativePasteSelected() void nativePasteSelected (EmbReal x,
EmbReal y)

MainWindow::nativePasteSelected x y.

17.65.3.119 nativePerpendicularDistance() EmbReal nativePerpendicularDistance (EmbReal px,
EmbReal py,
EmbReal x1,
EmbReal y1,
EmbReal x2,
EmbReal y2)

17.65.3.120 nativePreviewOff() void nativePreviewOff ()

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

17.65.3.122 nativePrintArea() void nativePrintArea (EmbReal x,
EmbReal y,
EmbReal w,
EmbReal h)

MainWindow::nativePrintArea x y w h.

17.65.3.123 nativeQSnapX() EmbReal nativeQSnapX ()

MainWindow::nativeQSnapX.

Returns

17.65.3.124 nativeQSnapY() `EmbReal nativeQSnapY ()`

`MainWindow::nativeQSnapY.`

Returns

17.65.3.125 nativeRotateSelected() `void nativeRotateSelected (`

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

`MainWindow::nativeRotateSelected x y rot.`

17.65.3.126 nativeScaleSelected() `void nativeScaleSelected (`

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

`MainWindow::nativeScaleSelected x y factor.`

17.65.3.127 nativeSetBackgroundColor() `void nativeSetBackgroundColor (`

```
    uint8_t r,  
    uint8_t g,  
    uint8_t b )
```

17.65.3.128 nativeSetCrossHairColor() `void nativeSetCrossHairColor (`

```
    uint8_t r,  
    uint8_t g,  
    uint8_t b )
```

17.65.3.129 nativeSetCursorShape() `void nativeSetCursorShape (`

```
    const QString & str )
```

17.65.3.130 nativeSetGridColor() `void nativeSetGridColor (`

```
    uint8_t r,  
    uint8_t g,  
    uint8_t b )
```

17.65.3.131 nativeSetPromptPrefix() void nativeSetPromptPrefix (const QString & txt)

17.65.3.132 nativeSetRubberMode() void nativeSetRubberMode (int mode)

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

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

17.65.3.135 nativeSpareRubber() void nativeSpareRubber (qint64 id)

17.65.3.136 nativeTipOfDay() void nativeTipOfDay ()

17.65.3.137 nativeVulcanize() void nativeVulcanize ()

17.65.3.138 newFile void newFile () [slot]

MainWindow::newFile.

17.65.3.139 nightVision void nightVision () [slot]

MainWindow::nightVision.

17.65.3.140 onCloseMdiWin void onCloseMdiWin (MdiWindow * theMdiWin) [virtual], [slot]

MainWindow::onCloseMdiWin.

Parameters

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

17.65.3.141 onCloseWindow void onCloseWindow () [slot][MainWindow::onCloseWindow.](#)**17.65.3.142 onWindowActivated** void onWindowActivated (QMdiSubWindow * *w*) [slot][MainWindow::onWindowActivated.](#)**Parameters**

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

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

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

17.65.3.144 openFilesSelected void openFilesSelected (const QStringList & *filesToOpen*) [slot][MainWindow::openFilesSelected.](#)**Parameters**

<i>filesToOpen</i>	<input type="button" value=""/>
--------------------	---------------------------------

17.65.3.145 openrecentfile void openrecentfile () [slot]

[MainWindow::openrecentfile](#).

17.65.3.146 panDown void panDown () [slot]

[MainWindow::panDown](#).

17.65.3.147 panLeft void panLeft () [slot]

17.65.3.148 panpoint void panpoint () [slot]

17.65.3.149 panrealtime void panrealtime () [slot]

17.65.3.150 panRight void panRight () [slot]

17.65.3.151 panUp void panUp () [slot]

17.65.3.152 paste void paste () [slot]

17.65.3.153 pickAddModeToggled void pickAddModeToggled () [slot]

17.65.3.154 platformString() QString platformString ()

17.65.3.155 print void print () [slot]

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

17.65.3.157 promptInputNext void promptInputNext () [slot]

17.65.3.158 promptInputPrevious void promptInputPrevious () [slot]

17.65.3.159 quit void quit () [slot]

[MainWindow::quit](#).

17.65.3.160 readSettings void readSettings () [slot]

[MainWindow::readSettings](#).

17.65.3.161 recentMenuAboutToShow void recentMenuAboutToShow () [slot]

[MainWindow::recentMenuAboutToShow](#).

17.65.3.162 redo void redo () [slot]

17.65.3.163 resizeEvent() void resizeEvent (QResizeEvent * e) [protected], [virtual]

[MainWindow::resizeEvent](#).

Parameters

e	
---	--

17.65.3.164 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
-----
```

17.65.3.165 run_script_file() `std::string run_script_file (std::string fname)`

[MainWindow::run_script_file](#).

Parameters

<code>fname</code>	The path of the script to run.
--------------------	--------------------------------

17.65.3.166 saveasfile `void saveasfile () [slot]`

[MainWindow::saveasfile](#).

17.65.3.167 savefile `void savefile () [slot]`

[MainWindow::savefile](#).

17.65.3.168 setShiftPressed `void setShiftPressed () [slot]`

17.65.3.169 setShiftReleased void setShiftReleased () [slot]

17.65.3.170 setTextAngle void setTextAngle (EmbReal num) [slot]

17.65.3.171 setTextBold void setTextBold (bool val) [slot]

17.65.3.172 setTextFont void setTextFont (const QString & str) [slot]

17.65.3.173 setTextItalic void setTextItalic (bool val) [slot]

17.65.3.174 setTextOverline void setTextOverline (bool val) [slot]

17.65.3.175 setTextSize void setTextSize (EmbReal num) [slot]

17.65.3.176 setTextStrikeOut void setTextStrikeOut (bool val) [slot]

17.65.3.177 setTextUnderline void setTextUnderline (bool val) [slot]

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

17.65.3.179 settingsPrompt void settingsPrompt () [slot]

17.65.3.180 setUndoCleanIcon() void setUndoCleanIcon (bool opened)

17.65.3.181 stub_implement void stub_implement (QString txt) [slot]

[MainWindow::stub_implement](#) txt.

17.65.3.182 stub_testing void stub_testing () [slot]

[MainWindow::stub_testing](#).

17.65.3.183 textFontSelectorCurrentFontChanged void textFontSelectorCurrentFontChanged (const QFont & font) [slot]

17.65.3.184 textSizeSelectorIndexChanged void textSizeSelectorIndexChanged (int index) [slot]

17.65.3.185 tipOfDay void tipOfDay () [slot]

17.65.3.186 toggleGrid void toggleGrid () [slot]

17.65.3.187 toggleLwt void toggleLwt () [slot]

17.65.3.188 toggleRuler void toggleRuler () [slot]

17.65.3.189 undo void undo () [slot]

17.65.3.190 updateAllViewBackgroundColor void updateAllViewBackgroundColor (QRgb color) [slot]

17.65.3.191 updateAllViewCrossHairColors void updateAllViewCrossHairColors (QRgb color) [slot]

17.65.3.192 updateAllViewGridColors void updateAllViewGridColors (QRgb color) [slot]

17.65.3.193 updateAllViewRulerColors void updateAllViewRulerColors (QRgb color) [slot]

17.65.3.194 updateAllViewScrollBars void updateAllViewScrollBars (bool val) [slot]

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

17.65.3.196 updateMenuToolbarStatusbar() void updateMenuToolbarStatusbar () [virtual]

MainWindow::updateMenuToolbarStatusbar.

17.65.3.197 updatePickAddMode void updatePickAddMode (bool val) [slot]

17.65.3.198 validFileFormat bool validFileFormat (const QString & fileName) [static], [slot]

MainWindow::validFileFormat.

Parameters

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

Returns

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

17.65.3.199 whatsThisContextHelp void whatsThisContextHelp () [slot]

17.65.3.200 windowMenuAboutToShow void windowMenuAboutToShow () [slot]

[MainWindow::windowMenuAboutToShow](#).

17.65.3.201 windowMenuActivated void windowMenuActivated (bool *checked*) [slot]

[MainWindow::windowMenuActivated](#).

Parameters

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

17.65.3.202 writeSettings void writeSettings () [slot]

[MainWindow::writeSettings](#).

This file needs to be read from the users home directory to ensure it is writable

17.65.3.203 zoomAll void zoomAll () [slot]

17.65.3.204 zoomCenter void zoomCenter () [slot]

17.65.3.205 zoomDynamic void zoomDynamic () [slot]

17.65.3.206 zoomExtents void zoomExtents () [slot]

17.65.3.207 zoomIn void zoomIn () [slot]

17.65.3.208 zoomOut void zoomOut () [slot]

17.65.3.209 zoomPrevious void zoomPrevious () [slot]

17.65.3.210 zoomRealtime void zoomRealtime () [slot]

17.65.3.211 zoomScale void zoomScale () [slot]

17.65.3.212 zoomSelected void zoomSelected () [slot]

17.65.3.213 zoomWindow void zoomWindow () [slot]

17.65.4 Member Data Documentation

17.65.4.1 actionHash QAction* actionHash[200]

17.65.4.2 checkBoxTipOfDay QCheckBox* checkBoxTipOfDay [protected]

17.65.4.3 colorSelector `QComboBox* colorSelector [protected]`

17.65.4.4 cutCopyObjectList `QList<QGraphicsItem*> cutCopyObjectList`

17.65.4.5 docIndex `int docIndex [protected]`

17.65.4.6 dockPropEdit `PropertyEditor* dockPropEdit`

17.65.4.7 dockUndoEdit `UndoEditor* dockUndoEdit`

17.65.4.8 editMenu `QMenu* editMenu [protected]`

17.65.4.9 fileMenu `QMenu* fileMenu [protected]`

17.65.4.10 formatFilterOpen `QString formatFilterOpen`

17.65.4.11 formatFilterSave `QString formatFilterSave`

17.65.4.12 helpMenu `QMenu* helpMenu [protected]`

17.65.4.13 labelTipOfDay `QLabel* labelTipOfDay [protected]`

17.65.4.14 layerSelector QComboBox* layerSelector [protected]

17.65.4.15 layoutState QByteArray layoutState [protected]

17.65.4.16 linetypeSelector QComboBox* linetypeSelector [protected]

17.65.4.17 linewidthSelector QComboBox* linewidthSelector [protected]

17.65.4.18 listMdiWin QList<MdiWindow*> listMdiWin [protected]

17.65.4.19 listTipOfDay QStringList listTipOfDay [protected]

17.65.4.20 mainWin MainWindow* mainWin

17.65.4.21 mdiArea MdiArea* mdiArea

17.65.4.22 menuHash QHash<QString, QMenu*> menuHash

17.65.4.23 myFileSeparator QAction* myFileSeparator [protected]

17.65.4.24 numOfDocs int numOfDocs [protected]

17.65.4.25 openFilesPath `QString openFilesPath [protected]`

17.65.4.26 panMenu `QMenu* panMenu [protected]`

17.65.4.27 prompt `CmdPrompt* prompt`

17.65.4.28 recentMenu `QMenu* recentMenu [protected]`

17.65.4.29 settingsMenu `QMenu* settingsMenu [protected]`

17.65.4.30 shiftKeyPressedState `bool shiftKeyPressedState [protected]`

17.65.4.31 statusbar `StatusBar* statusbar`

17.65.4.32 textFontSelector `QFontComboBox* textFontSelector [protected]`

17.65.4.33 textSizeSelector `QComboBox* textSizeSelector [protected]`

17.65.4.34 toolbarEdit `QToolBar* toolbarEdit [protected]`

17.65.4.35 toolbarFile `QToolBar* toolbarFile [protected]`

17.65.4.36 toolbarHash QHash<QString, QToolBar*> toolbarHash

17.65.4.37 toolbarHelp QToolBar* toolbarHelp [protected]

17.65.4.38 toolbarIcon QToolBar* toolbarIcon [protected]

17.65.4.39 toolbarLayer QToolBar* toolbarLayer [protected]

17.65.4.40 toolbarPan QToolBar* toolbarPan [protected]

17.65.4.41 toolbarPrompt QToolBar* toolbarPrompt [protected]

17.65.4.42 toolbarProperties QToolBar* toolbarProperties [protected]

17.65.4.43 toolbarText QToolBar* toolbarText [protected]

17.65.4.44 toolbarView QToolBar* toolbarView [protected]

17.65.4.45 toolbarZoom QToolBar* toolbarZoom [protected]

17.65.4.46 viewMenu QMenu* viewMenu [protected]

17.65.4.47 windowMenu QMenu* windowMenu [protected]

17.65.4.48 wizardTipOfDay QWizard* wizardTipOfDay [protected]

17.65.4.49 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](#)

17.66 MdiArea Class Reference

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

Public Slots

- void **cascade** ()
MdiArea::cascade.
- void **tile** ()
MdiArea::tile.

Public Member Functions

- void **zoomExtentsAllSubWindows** ()
MdiArea::zoomExtentsAllSubWindows.
- void **forceRepaint** ()
MdiArea::forceRepaint.
- **MdiArea** (**MainWindow** *mw, **QWidget** *parent=0)
MdiArea::MdiArea.
- **~MdiArea** ()
MdiArea::~MdiArea.
- void **useBackgroundLogo** (bool use)
MdiArea::useBackgroundLogo.
- void **useBackgroundTexture** (bool use)
MdiArea::useBackgroundTexture.
- void **useBackgroundColor** (bool use)
- void **setBackgroundLogo** (const **QString** &fileName)
MdiArea::setBackgroundLogo.
- void **setBackgroundTexture** (const **QString** &fileName)
MdiArea::setBackgroundTexture.
- void **setBackgroundColor** (const **QColor** &color)
MdiArea::setBackgroundColor.

Public Attributes

- `MainWindow * mainWin`
- `bool useLogo`
- `bool useTexture`
- `bool useColor`
- `QPixmap bgLogo`
- `QPixmap bgTexture`
- `QColor bgColor`

Protected Member Functions

- `virtual void mouseDoubleClickEvent (QMouseEvent *e)`
MdiArea::mouseDoubleClickEvent.
- `virtual void paintEvent (QPaintEvent *e)`
MdiArea::paintEvent.

17.66.1 Constructor & Destructor Documentation

17.66.1.1 MdiArea() `MdiArea (`
 `MainWindow * mw,`
 `QWidget * parent = 0)`

`MdiArea::MdiArea.`

Parameters

<code>mw</code>	
<code>parent</code>	

17.66.1.2 ~MdiArea() `~MdiArea ()`

`MdiArea::~MdiArea.`

17.66.2 Member Function Documentation

17.66.2.1 cascade `void cascade () [slot]`

`MdiArea::cascade.`

17.66.2.2 forceRepaint() void forceRepaint ()

MdiArea::forceRepaint.

17.66.2.3 mouseDoubleClickEvent() void mouseDoubleClickEvent (QMouseEvent * e) [protected], [virtual]

MdiArea::mouseDoubleClickEvent.

17.66.2.4 paintEvent() void paintEvent (QPaintEvent * e) [protected], [virtual]

MdiArea::paintEvent.

17.66.2.5 setBackgroundColor() void setBackgroundColor (const QColor & color)

MdiArea::setBackgroundColor.

Parameters

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

17.66.2.6 setBackgroundLogo() void setBackgroundLogo (const QString & fileName)

MdiArea::setBackgroundLogo.

Parameters

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

17.66.2.7 setBackgroundTexture() void setBackgroundTexture (const QString & fileName)

MdiArea::setBackgroundTexture.

Parameters

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

17.66.2.8 tile void tile () [slot]

MdiArea::tile.

17.66.2.9 useBackgroundColor() void useBackgroundColor (bool use)**Parameters**

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

17.66.2.10 useBackgroundLogo() void useBackgroundLogo (bool use)

MdiArea::useBackgroundLogo.

Parameters

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

17.66.2.11 useBackgroundTexture() void useBackgroundTexture (bool use)

MdiArea::useBackgroundTexture.

Parameters

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

17.66.2.12 zoomExtentsAllSubWindows() void zoomExtentsAllSubWindows ()

MdiArea::zoomExtentsAllSubWindows.

17.66.3 Member Data Documentation

17.66.3.1 `bgColor` `QColor bgColor`

17.66.3.2 `bgLogo` `QPixmap bgLogo`

17.66.3.3 `bgTexture` `QPixmap bgTexture`

17.66.3.4 `mainWin` `MainWindow* mainWin`

17.66.3.5 `useColor` `bool useColor`

17.66.3.6 `useLogo` `bool useLogo`

17.66.3.7 `useTexture` `bool useTexture`

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

- `embroidermodder2/embroidermodder.h`
- `embroidermodder2/mdiarea.cpp`

17.67 MdiWindow Class Reference

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

Public Slots

- void `closeEvent` (QCloseEvent *e)
MdiWindow::closeEvent.
- void `onWindowActivated` ()
MdiWindow::onWindowActivated.
- void `currentLayerChanged` (const QString &layer)
MdiWindow::currentLayerChanged.
- void `currentColorChanged` (const QRgb &color)
MdiWindow::currentColorChanged.
- void `currentLinetypeChanged` (const QString &type)
MdiWindow::currentLinetypeChanged.
- void `currentLineweightChanged` (const QString &weight)
MdiWindow::currentLineweightChanged.
- void `updateColorLinetypeLinewidth` ()
- void `deletePressed` ()
- void `escapePressed` ()
- void `showViewScrollBars` (bool val)
- void `setViewCrossHairColor` (QRgb color)
- void `setViewBackgroundColor` (QRgb color)
- void `setViewSelectBoxColors` (QRgb colorL, QRgb fillL, QRgb colorR, QRgb fillR, int alpha)
- void `setViewGridColor` (QRgb color)
- void `setViewRulerColor` (QRgb color)
- void `print` ()
MdiWindow::print.
- void `saveBMC` ()
MdiWindow::saveBMC.
- void `promptHistoryAppended` (const QString &txt)
- void `logPromptInput` (const QString &txt)
- void `promptInputPrevious` ()
- void `promptInputNext` ()
MdiWindow::promptInputNext.

Signals

- void `sendCloseMdiWin` (MdiWindow *)

Public Member Functions

- MdiWindow (const int theIndex, MainWindow *mw, QMdiArea *parent, Qt::WindowFlags wflags)
- ~MdiWindow ()
MdiWindow::~MdiWindow.
- virtual QSize `sizeHint` () const
MdiWindow::sizeHint.
- QString `getCurrentFile` ()
- QString `getShortCurrentFile` ()
MdiWindow::getShortCurrentFile.
- View * `getView` ()
- QGraphicsScene * `getScene` ()
- QString `getCurrentLayer` ()
- QRgb `getCurrentColor` ()

- `QString getCurrentLineType ()`
- `QString getCurrentLineWeight ()`
- `void setCurrentLayer (const QString &layer)`
- `void setCurrentColor (const QRgb &color)`
- `void setCurrentLineType (const QString &lineType)`
- `void setCurrentLineWeight (const QString &lineWeight)`
- `void designDetails ()`
- `bool loadFile (const QString &fileName)`
MdiWindow::loadFile.
- `bool saveFile (const QString &fileName)`
MdiWindow::saveFile.

Private Member Functions

- `void setCurrentFile (const QString &fileName)`
MdiWindow::setCurrentFile.
- `QString fileExtension (const QString &fileName)`
MdiWindow::fileExtension.
- `void promptInputPrevNext (bool prev)`
MdiWindow::promptInputPrevNext.

Private Attributes

- `MainWindow * mainWin`
- `QMdiArea * mdiArea`
- `QGraphicsScene * gscene`
- `View * gview`
- `bool fileWasLoaded`
- `QString promptHistory`
- `QList<QString> promptInputList`
- `int promptInputNum`
- `QPrinter printer`
- `QString curFile`
- `int myIndex`
- `QString curLayer`
- `QRgb curColor`
- `QString curLineType`
- `QString curLineWeight`

17.67.1 Constructor & Destructor Documentation

```
17.67.1.1 MdiWindow() MdiWindow (
    const int theIndex,
    MainWindow * mw,
    QMdiArea * parent,
    Qt::WindowFlags wflags )
```

17.67.1.2 ~MdiWindow() ~MdiWindow()

MdiWindow::~MdiWindow.

17.67.2 Member Function Documentation**17.67.2.1 closeEvent void closeEvent (**
 QCLOSEEvent * e) [slot]

MdiWindow::closeEvent.

17.67.2.2 currentColorChanged void currentColorChanged (
 const QRgb & color) [slot]

MdiWindow::currentColorChanged.

Parameters

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

17.67.2.3 currentLayerChanged void currentLayerChanged (
 const QString & layer) [slot]

MdiWindow::currentLayerChanged.

Parameters

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

17.67.2.4 currentLinetypeChanged void currentLinetypeChanged (
 const QString & type) [slot]

MdiWindow::currentLinetypeChanged.

Parameters

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

17.67.2.5 currentLineweightChanged void currentLineweightChanged (const QString & weight) [slot]

MdiWindow::currentLineweightChanged.

Parameters

<i>weight</i>	
---------------	--

17.67.2.6 deletePressed void deletePressed () [slot]

17.67.2.7 designDetails() void designDetails ()

17.67.2.8 escapePressed void escapePressed () [slot]

17.67.2.9 fileExtension() QString fileExtension (const QString & fileName) [private]

MdiWindow::fileExtension.

Parameters

<i>fileName</i>	
-----------------	--

Returns

17.67.2.10 getCurrentColor() QRgb getCurrentColor () [inline]

17.67.2.11 getCurrentFile() QString getCurrentFile () [inline]

17.67.2.12 getCurrentLayer() `QString getCurrentLayer () [inline]`

17.67.2.13 getCurrentLineType() `QString getCurrentLineType () [inline]`

17.67.2.14 getCurrentLineWeight() `QString getCurrentLineWeight () [inline]`

17.67.2.15 getScene() `QGraphicsScene * getScene () [inline]`

17.67.2.16 getShortCurrentFile() `QString getShortCurrentFile ()`

[MdiWindow::getShortCurrentFile.](#)

Returns

17.67.2.17 getView() `View * getView () [inline]`

17.67.2.18 loadFile() `bool loadFile (const QString & fileName)`

[MdiWindow::loadFile.](#)

Parameters

<code>fileName</code>

Returns

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

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

17.67.2.20 onWindowActivated void onWindowActivated () [slot]

[MdiWindow::onWindowActivated.](#)

17.67.2.21 print void print () [slot]

[MdiWindow::print.](#)

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

17.67.2.23 promptInputNext void promptInputNext () [slot]

[MdiWindow::promptInputNext.](#)

17.67.2.24 promptInputPrevious void promptInputPrevious () [slot]

17.67.2.25 promptInputPrevNext() void promptInputPrevNext (bool prev) [private]

[MdiWindow::promptInputPrevNext.](#)

Parameters

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

17.67.2.26 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)

17.67.2.27 saveFile() `bool saveFile (`
 `const QString & fileName)`

[MdiWindow::saveFile.](#)

Parameters

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

Returns

17.67.2.28 sendCloseMdiWin `void sendCloseMdiWin (`
 `MdiWindow *) [signal]`

17.67.2.29 setCurrentColor() `void setCurrentColor (`
 `const QRgb & color) [inline]`

17.67.2.30 setCurrentFile() `void setCurrentFile (`
 `const QString & fileName) [private]`

[MdiWindow::setCurrentFile.](#)

Parameters

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

17.67.2.31 `setCurrentLayer()` void setCurrentLayer (const QString & *layer*) [inline]

17.67.2.32 `setCurrentLineType()` void setCurrentLineType (const QString & *lineType*) [inline]

17.67.2.33 `setCurrentLineWeight()` void setCurrentLineWeight (const QString & *lineWeight*) [inline]

17.67.2.34 `setViewBackgroundColor` void setViewBackgroundColor (QRgb *color*) [slot]

17.67.2.35 `setViewCrossHairColor` void setViewCrossHairColor (QRgb *color*) [slot]

17.67.2.36 `setViewGridColor` void setViewGridColor (QRgb *color*) [slot]

17.67.2.37 `setViewRulerColor` void setViewRulerColor (QRgb *color*) [slot]

17.67.2.38 `setViewSelectBoxColors` void setViewSelectBoxColors (QRgb *colorL*, QRgb *fillL*, QRgb *colorR*, QRgb *fillR*, int *alpha*) [slot]

17.67.2.39 `showViewScrollBars` void showViewScrollBars (bool *val*) [slot]

17.67.2.40 sizeHint() QSize sizeHint () const [virtual]

MdiWindow::sizeHint.

Returns

17.67.2.41 updateColorLinetypeLinewidth void updateColorLinetypeLinewidth () [slot]

17.67.3 Member Data Documentation

17.67.3.1 curColor QRgb curColor [private]

17.67.3.2 curFile QString curFile [private]

17.67.3.3 curLayer QString curLayer [private]

17.67.3.4 curLineType QString curLineType [private]

17.67.3.5 curLineWeight QString curLineWeight [private]

17.67.3.6 fileWasLoaded bool fileWasLoaded [private]

17.67.3.7 gscene QGraphicsScene* gscene [private]

17.67.3.8 gview `View* gview [private]`

17.67.3.9 mainWin `MainWindow* mainWin [private]`

17.67.3.10 mdiArea `QMdiArea* mdiArea [private]`

17.67.3.11 myIndex `int myIndex [private]`

17.67.3.12 printer `QPrinter printer [private]`

17.67.3.13 promptHistory `QString promptHistory [private]`

17.67.3.14 promptInputList `QList<QString> promptInputList [private]`

17.67.3.15 promptInputNum `int promptInputNum [private]`

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

- `embroidermodder2/embroidermodder.h`
- `embroidermodder2/mdiwindow.cpp`

17.68 Parameter_Struct Reference

Public Attributes

- `std::string s_value`
- `EmbReal r_value`
- `int i_value`

17.68.1 Member Data Documentation

17.68.1.1 i_value int i_value

17.68.1.2 r_value EmbReal r_value

17.68.1.3 s_value std::string s_value

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

- embroidermodder2/mainwindow.cpp

17.69 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 [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 [setObjectLineWeight](#) (EmbReal lineWeight)
- void [setObjectPath](#) (const QPainterPath &p)
- void [setObjectRubberMode](#) (int mode)
- void [setObjectRubberPoint](#) (const QString &key, const QPointF &point)
- void [setObjectRubberText](#) (const QString &key, const QString &txt)
- virtual QRectF [boundingRect](#) () const
- virtual QPainterPath [shape](#) () const
- void [drawRubberLine](#) (const QLineF &rubLine, QPainter *painter=0, const char *colorFromScene=0)
- virtual void [vulcanize](#) ()=0
- virtual QPointF [mouseSnapPoint](#) (const QPointF &mousePoint)=0
- virtual QList<QPointF> [allGripPoints](#) ()=0
- virtual void [gripEdit](#) (const QPointF &before, const QPointF &after)=0

Public Attributes

- QPainterPath [normalPath](#)

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)

17.69.1 Member Enumeration Documentation**17.69.1.1 anonymous enum** anonymous enum

Enumerator

Type	
------	--

17.69.2 Constructor & Destructor Documentation**17.69.2.1 PathObject() [1/2]** [PathObject](#) (

```
EmbReal x,  
EmbReal y,  
const QPainterPath p,  
QRgb rgb,  
QGraphicsItem * parent = 0 )
```

17.69.2.2 PathObject() [2/2] [PathObject](#) (

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

17.69.2.3 ~PathObject() [~PathObject](#) ()**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, const QPainterPath & p, QRgb rgb, Qt::PenStyle lineType)

17.69.3.4 mouseSnapPoint() QPointF mouseSnapPoint (const QPointF & mousePoint) [virtual]

Implements [BaseObject](#).

17.69.3.5 objectCopyPath() QPainterPath objectCopyPath () const

17.69.3.6 objectPos() QPointF objectPos () const [inline]

17.69.3.7 objectSavePath() QPainterPath objectSavePath () const

17.69.3.8 objectX() EmbReal objectX () const [inline]

17.69.3.9 objectY() EmbReal objectY () const [inline]

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

17.69.3.11 `setObjectPos()` [1/2] void setObjectPos (const QPointF & point) [inline]

17.69.3.12 `setObjectPos()` [2/2] void setObjectPos (EmbReal x, EmbReal y) [inline]

17.69.3.13 `setObjectX()` void setObjectX (EmbReal x) [inline]

17.69.3.14 `setObjectY()` void setObjectY (EmbReal y) [inline]

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

Reimplemented from [BaseObject](#).

17.69.3.16 `updatePath()` void updatePath (const QPainterPath & p)

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

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

Implements [BaseObject](#).

17.69.4 Member Data Documentation

17.69.4.1 **normalPath** QPainterPath normalPath

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

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

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

Protected Member Functions

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

Protected Member Functions inherited from BaseObject

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

Additional Inherited Members

Public Attributes inherited from [BaseObject](#)

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

17.70.1 Member Enumeration Documentation

17.70.1.1 anonymous enum [anonymous enum](#)

Enumerator

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

17.70.2 Constructor & Destructor Documentation

17.70.2.1 [PointObject\(\)](#) [1/2] [PointObject](#) (

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

17.70.2.2 [PointObject\(\)](#) [2/2] [PointObject](#) (

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

17.70.2.3 [~PointObject\(\)](#) [~PointObject](#) ()

17.70.3 Member Function Documentation

17.70.3.1 allGripPoints() QList< QPointF > allGripPoints () [virtual]

Implements [BaseObject](#).

17.70.3.2 gripEdit() void gripEdit (const QPointF & *before*, const QPointF & *after*) [virtual]

Implements [BaseObject](#).

17.70.3.3 init() void init (EmbReal *x*, EmbReal *y*, QRgb *rgb*, Qt::PenStyle *lineType*)

17.70.3.4 mouseSnapPoint() QPointF mouseSnapPoint (const QPointF & *mousePoint*) [virtual]

Implements [BaseObject](#).

17.70.3.5 objectPos() QPointF objectPos () const [inline]

17.70.3.6 objectSavePath() QPainterPath objectSavePath () const

17.70.3.7 objectX() EmbReal objectX () const [inline]

17.70.3.8 objectY() EmbReal objectY () const [inline]

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

17.70.3.10 `setObjectPos()` [1/2] void setObjectPos (const QPointF & point) [inline]

17.70.3.11 `setObjectPos()` [2/2] void setObjectPos (EmbReal x, EmbReal y) [inline]

17.70.3.12 `setObjectX()` void setObjectX (EmbReal x) [inline]

17.70.3.13 `setObjectY()` void setObjectY (EmbReal y) [inline]

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

Reimplemented from [BaseObject](#).

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

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

Implements [BaseObject](#).

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

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

17.71 PolygonObject Class Reference

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

Public Types

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

Public Types inherited from BaseObject

- enum { `Type` = `OBJ_TYPE_BASE` }

Public Member Functions

- `PolygonObject (EmbReal x, EmbReal y, const QPainterPath &p, QRgb rgb, QGraphicsItem *parent=0)`
- `PolygonObject (PolygonObject *obj, QGraphicsItem *parent=0)`
- `~PolygonObject ()`
- virtual int `type () const`
- void `init (EmbReal x, EmbReal y, const QPainterPath &p, QRgb rgb, Qt::PenStyle lineType)`
- void `updatePath (const QPainterPath &p)`
- int `findIndex (const QPointF &point)`
- `QPainterPath objectCopyPath () const`
- `QPainterPath objectSavePath () const`
- `QPointF objectPos () const`
- `EmbReal objectX () const`
- `EmbReal objectY () const`
- void `setObjectPos (const QPointF &point)`
- void `setObjectPos (EmbReal x, EmbReal y)`
- void `setObjectX (EmbReal x)`
- void `setObjectY (EmbReal y)`
- void `updateRubber (QPainter *painter=0)`
- virtual void `vulcanize ()`
- virtual `QPointF mouseSnapPoint (const QPointF &mousePoint)`
- virtual `QList< QPointF > allGripPoints ()`
- virtual void `gripEdit (const QPointF &before, const QPointF &after)`

Public Member Functions inherited from BaseObject

- `BaseObject (QGraphicsItem *parent=0)`
- virtual `~BaseObject ()`
- virtual int `type () const`
- qint64 `objectID () const`
- `QPen objectPen () const`
- `QColor objectColor () const`
- `QRgb objectColorRGB () const`
- `Qt::PenStyle objectLineType () const`
- `EmbReal objectLineWidth () const`
- `QPainterPath objectPath () const`
- int `objectRubberMode () const`
- `QPointF objectRubberPoint (const QString &key) const`
- `QString objectRubberText (const QString &key) const`
- `QPointF objectCenter () const`
- `EmbReal objectCenterX () const`
- `EmbReal objectCenterY () const`
- void `setObjectCenter (EmbVector center)`
- void `setObjectCenterX (EmbReal centerX)`
- void `setObjectCenterY (EmbReal centerY)`
- `QRectF rect () const`
- void `setRect (const QRectF &r)`
- void `setRect (EmbReal x, EmbReal y, EmbReal w, EmbReal h)`

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

Public Attributes

- `QPainterPath normalPath`
- `int gripIndex`

Public Attributes inherited from [BaseObject](#)

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

Protected Member Functions

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

Protected Member Functions inherited from [BaseObject](#)

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

17.71.1 Member Enumeration Documentation

17.71.1.1 anonymous enum anonymous enum

Enumerator

Type	
------	--

17.71.2 Constructor & Destructor Documentation

17.71.2.1 **PolygonObject()** [1/2] [PolygonObject](#) (

```
EmbReal x,  
EmbReal y,  
const QPainterPath & p,  
QRgb rgb,  
QGraphicsItem * parent = 0 )
```

17.71.2.2 **PolygonObject()** [2/2] [PolygonObject](#) (

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

17.71.2.3 **~PolygonObject()** [~PolygonObject](#) ()

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-polygon.cpp](#)

17.72 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)

17.72.1 Member Enumeration Documentation**17.72.1.1 anonymous enum** anonymous enum**Enumerator**

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

17.72.2 Constructor & Destructor Documentation**17.72.2.1 PolylineObject() [1/2]** `PolylineObject (`

```
    EmbReal x,
    EmbReal y,
    const QPainterPath & p,
    QRgb rgb,
    QGraphicsItem * parent = 0 )
```

17.72.2.2 PolylineObject() [2/2] `PolylineObject (`

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

17.72.2.3 ~PolylineObject() ~[PolylineObject](#) ()**17.72.3 Member Function Documentation****17.72.3.1 allGripPoints()** QList< [QPointF](#) > allGripPoints () [virtual]

Implements [BaseObject](#).

17.72.3.2 findIndex() int findIndex (const [QPointF](#) & point)**17.72.3.3 gripEdit()** void gripEdit (const [QPointF](#) & before, const [QPointF](#) & after) [virtual]

Implements [BaseObject](#).

17.72.3.4 init() void init ([EmbReal](#) x, [EmbReal](#) y, const [QPainterPath](#) & p, [QRgb](#) rgb, [Qt::PenStyle](#) lineType)**17.72.3.5 mouseSnapPoint()** [QPointF](#) mouseSnapPoint (const [QPointF](#) & mousePoint) [virtual]

Implements [BaseObject](#).

17.72.3.6 objectCopyPath() [QPainterPath](#) objectCopyPath () const**17.72.3.7 objectPos()** [QPointF](#) objectPos () const [inline]

17.72.3.8 `objectSavePath()` `QPainterPath objectSavePath () const`

17.72.3.9 `objectX()` `EmbReal objectX () const [inline]`

17.72.3.10 `objectY()` `EmbReal objectY () const [inline]`

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

17.72.3.12 `setObjectPos()` [1/2] `void setObjectPos (`
 `const QPointF & point) [inline]`

17.72.3.13 `setObjectPos()` [2/2] `void setObjectPos (`
 `EmbReal x,`
 `EmbReal y) [inline]`

17.72.3.14 `setObjectX()` `void setObjectX (`
 `EmbReal x) [inline]`

17.72.3.15 `setObjectY()` `void setObjectY (`
 `EmbReal y) [inline]`

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

Reimplemented from [BaseObject](#).

17.72.3.17 updatePath() void updatePath (const QPainterPath & p)

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

17.72.3.19 vulcanize() void vulcanize () [virtual]

Implements [BaseObject](#).

17.72.4 Member Data Documentation

17.72.4.1 gripIndex int gripIndex

17.72.4.2 normalPath QPainterPath normalPath

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

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

17.73 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

17.73.1 Constructor & Destructor Documentation

```
17.73.1.1 PreviewDialog() PreviewDialog (   
    QWidget * parent = 0,  
    const QString & caption = QString(),  
    const QString & directory = QString(),  
    const QString & filter = QString() )
```

```
17.73.1.2 ~PreviewDialog() ~PreviewDialog ( )
```

17.73.2 Member Data Documentation

```
17.73.2.1 imgWidget ImageWidget* imgWidget
```

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

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

17.74 PropertyEditor Class Reference

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

Public Slots

- void [setSelectedItems](#) (QList< QGraphicssItem * > itemList)
- void [updatePickAddModeButton](#) (bool pickAddMode)

Signals

- void [pickAddModeToggled](#) ()

Public Member Functions

- [PropertyEditor](#) (const QString &iconDirectory=QString(), bool pickAddMode=true, QWidget *widgetToFocus=0, QWidget *parent=0)
- [~PropertyEditor](#) ()

Protected Member Functions

- bool [eventFilter](#) (QObject *obj, QEvent *event)

Private Slots

- void `fieldEdited` (QObject *fieldObj)
- void `showGroups` (int objType)
- void `showOneType` (int index)
- void `hideAllGroups` ()
- void `clearAllFields` ()
- void `togglePickAddMode` ()

Private Member Functions

- QToolButton * `createToolButton` (const QString &iconName, const QString &txt)
- QLineEdit * `createLineEdit` (const QString &validatorType=QString(), bool readOnly=false)
- QComboBox * `createComboBox` (bool disable=false)
- QFontComboBox * `createFontComboBox` (bool disable=false)
- void `updateLineEditStrIfVaries` (QLineEdit *lineEdit, const QString &str)
- void `updateLineEditNumIfVaries` (QLineEdit *lineEdit, `EmbReal` num, bool useAnglePrecision)
- void `updateFontComboBoxStrIfVaries` (QFontComboBox *fontComboBox, const QString &str)
- void `updateComboBoxStrIfVaries` (QComboBox *comboBox, const QString &str, const QStringList &strList)
- void `updateComboBoxBoolIfVaries` (QComboBox *comboBox, bool val, bool yesOrNoText)
- void `mapSignal` (QObject *fieldObj, const QString &name, QVariant value)
- QComboBox * `createComboBoxSelected` ()
- QToolButton * `createToolButtonQSelect` ()
- QToolButton * `createToolButtonPickAdd` ()
- void `createGroupBox` (std::string group_box_key, const char *title, `GroupBoxData` data[], int lines)
- 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`
- 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`

17.74.1 Constructor & Destructor Documentation

17.74.1.1 `PropertyEditor()` `PropertyEditor (`

```
    const QString & iconDirectory = QString(),
    bool pickAddMode = true,
    QWidget * widgetToFocus = 0,
    QWidget * parent = 0 )
```

17.74.1.2 `~PropertyEditor()` `~PropertyEditor ()`

Todo document this

17.74.2 Member Function Documentation

17.74.2.1 clearAllFields void clearAllFields () [private], [slot]

Todo DimAligned
DimAngular
DimArcLength
DimDiameter
DimLeader
DimLinear
DimOrdinate
DimRadius

17.74.2.2 createComboBox() QComboBox * createComboBox (bool disable = false) [private]**17.74.2.3 createComboBoxSelected()** QComboBox * createComboBoxSelected () [private]

Todo document this

17.74.2.4 createFontComboBox() QFontComboBox * createFontComboBox (bool disable = false) [private]**17.74.2.5 createGroupBox()** void createGroupBox (std::string group_box_key, const char * title, GroupBoxData data[], int lines) [private]**17.74.2.6 createGroupBoxGeneral()** QGroupBox * createGroupBoxGeneral () [private]

Todo use proper icons for toolButtons

17.74.2.7 `createGroupBoxGeometryArc()` `QGroupBox * createGroupBoxGeometryArc () [private]`

Todo use proper icons for toolButtons

17.74.2.8 `createGroupBoxGeometryBlock()` `QGroupBox * createGroupBoxGeometryBlock () [private]`

Todo use proper icons for toolButtons

mapSignal for blocks

17.74.2.9 `createGroupBoxGeometryCircle()` `QGroupBox * createGroupBoxGeometryCircle () [private]`

Todo use proper icons for toolButtons

17.74.2.10 `createGroupBoxGeometryDimAligned()` `QGroupBox * createGroupBoxGeometryDimAligned () [private]`

Todo toolButtons and lineEdits for DimAligned

17.74.2.11 `createGroupBoxGeometryDimAngular()` `QGroupBox * createGroupBoxGeometryDimAngular () [private]`

Todo toolButtons and lineEdits for DimAngular

17.74.2.12 `createGroupBoxGeometryDimArcLength()` `QGroupBox * createGroupBoxGeometryDimArcLength () [private]`

Todo toolButtons and lineEdits for DimArcLength

17.74.2.13 `createGroupBoxGeometryDimDiameter()` `QGroupBox * createGroupBoxGeometryDimDiameter()` [private]

Todo toolButtons and lineEdits for DimDiameter

17.74.2.14 `createGroupBoxGeometryDimLeader()` `QGroupBox * createGroupBoxGeometryDimLeader()` [private]

Todo toolButtons and lineEdits for DimLeader

17.74.2.15 `createGroupBoxGeometryDimLinear()` `QGroupBox * createGroupBoxGeometryDimLinear()` [private]

Todo toolButtons and lineEdits for DimLinear

17.74.2.16 `createGroupBoxGeometryDimOrdinate()` `QGroupBox * createGroupBoxGeometryDimOrdinate()` [private]

Todo toolButtons and lineEdits for DimOrdinate

17.74.2.17 `createGroupBoxGeometryDimRadius()` `QGroupBox * createGroupBoxGeometryDimRadius()` [private]

Todo toolButtons and lineEdits for DimRadius

17.74.2.18 `createGroupBoxGeometryEllipse()` `QGroupBox * createGroupBoxGeometryEllipse()` [private]

17.74.2.19 `createGroupBoxGeometryImage()` `QGroupBox * createGroupBoxGeometryImage()` [private]

Todo use proper icons for toolButtons

mapSignal for images

17.74.2.20 `createGroupBoxGeometryInfiniteLine()` `QGroupBox * createGroupBoxGeometryInfiniteLine()` [private]

Todo use proper icons for toolButtons
mapSignal for infinite lines

17.74.2.21 `createGroupBoxGeometryLine()` `QGroupBox * createGroupBoxGeometryLine()` [private]

Todo use proper icons for toolButtons

17.74.2.22 `createGroupBoxGeometryPath()` `QGroupBox * createGroupBoxGeometryPath()` [private]

Todo use proper icons for toolButtons

17.74.2.23 `createGroupBoxGeometryPoint()` `QGroupBox * createGroupBoxGeometryPoint()` [private]

Todo use proper icons for toolButtons

17.74.2.24 `createGroupBoxGeometryPolygon()` `QGroupBox * createGroupBoxGeometryPolygon()` [private]

Todo use proper icons for toolButtons
mapSignal for polygons

17.74.2.25 `createGroupBoxGeometryPolyline()` `QGroupBox * createGroupBoxGeometryPolyline()` [private]

Todo use proper icons for toolButtons

17.74.2.26 `createGroupBoxGeometryRay()` `QGroupBox * createGroupBoxGeometryRay ()` [private]

Todo use proper icons for toolButtons

17.74.2.27 `createGroupBoxGeometryRectangle()` `QGroupBox * createGroupBoxGeometryRectangle ()` [private]

Todo use proper icons for toolButtons

17.74.2.28 `createGroupBoxGeometryTextMulti()` `QGroupBox * createGroupBoxGeometryTextMulti ()` [private]

Todo use proper icons for toolButtons
mapSignal for multiline text

17.74.2.29 `createGroupBoxGeometryTextSingle()` `QGroupBox * createGroupBoxGeometryTextSingle ()` [private]

Todo use proper icons for toolButtons

17.74.2.30 `createGroupBoxMiscArc()` `QGroupBox * createGroupBoxMiscArc ()` [private]

Todo use proper icons for toolButtons

17.74.2.31 `createGroupBoxMiscImage()` `QGroupBox * createGroupBoxMiscImage ()` [private]

Todo use proper icons for toolButtons

17.74.2.32 `createGroupBoxMiscPath()` `QGroupBox * createGroupBoxMiscPath () [private]`

Todo use proper icons for toolButtons
mapSignal for paths

17.74.2.33 `createGroupBoxMiscPolyline()` `QGroupBox * createGroupBoxMiscPolyline () [private]`

Todo use proper icons for toolButtons
mapSignal for polylines.

17.74.2.34 `createGroupBoxMiscTextSingle()` `QGroupBox * createGroupBoxMiscTextSingle () [private]`

Todo use proper icons for toolButtons

17.74.2.35 `createGroupBoxTextTextSingle()` `QGroupBox * createGroupBoxTextTextSingle () [private]`

Todo use proper icons for toolButtons

17.74.2.36 `createLineEdit()` `QLineEdit * createLineEdit (`
`const QString & validatorType = QString(),`
`bool readOnly = false) [private]`

17.74.2.37 `createToolButton()` `QToolButton * createToolButton (`
`const QString & iconName,`
`const QString & txt) [private]`

17.74.2.38 `createToolButtonPickAdd()` `QToolButton * createToolButtonPickAdd () [private]`

17.74.2.39 `createToolButtonQSelect()` `QToolButton * createToolButtonQSelect () [private]`

Todo document this

17.74.2.40 `eventFilter()` `bool eventFilter (`
 `QObject * obj,`
 `QEvent * event) [protected]`

Todo document this

17.74.2.41 `fieldEdited()` `void fieldEdited (`
 `QObject * fieldObj) [private], [slot]`

17.74.2.42 `hideAllGroups()` `void hideAllGroups () [private], [slot]`

Note

General group will never be hidden.

17.74.2.43 `mapSignal()` `void mapSignal (`
 `QObject * fieldObj,`
 `const QString & name,`
 `QVariant value) [private]`

17.74.2.44 `pickAddModeToggled()` `void pickAddModeToggled () [signal]`

17.74.2.45 `setSelectedItems()` `void setSelectedItems (`
 `QList< QGraphicsItem * > itemList) [slot]`

17.74.2.46 `showGroups()` `void showGroups (`
 `int objType) [private], [slot]`

17.74.2.47 showOneType void showOneType (int index) [private], [slot]

17.74.2.48 togglePickAddMode void togglePickAddMode () [private], [slot]

17.74.2.49 updateComboBoxBoolIfVaries() void updateComboBoxBoolIfVaries (QComboBox * comboBox, bool val, bool yesOrNoText) [private]

17.74.2.50 updateComboBoxStrIfVaries() void updateComboBoxStrIfVaries (QComboBox * comboBox, const QString & str, const QStringList & strList) [private]

17.74.2.51 updateFontComboBoxStrIfVaries() void updateFontComboBoxStrIfVaries (QFontComboBox * fontComboBox, const QString & str) [private]

17.74.2.52 updateLineEditNumIfVaries() void updateLineEditNumIfVaries (QLineEdit * lineEdit, EmbReal num, bool useAnglePrecision) [private]

17.74.2.53 updateLineEditStrIfVaries() void updateLineEditStrIfVaries (QLineEdit * lineEdit, const QString & str) [private]

17.74.2.54 updatePickAddModeButton void updatePickAddModeButton (bool pickAddMode) [slot]

17.74.3 Member Data Documentation

17.74.3.1 comboBoxSelected QComboBox* comboBoxSelected [private]

17.74.3.2 fieldNewText QString fieldNewText [private]

17.74.3.3 fieldNoText QString fieldNoText [private]

17.74.3.4 fieldOffText QString fieldOffText [private]

17.74.3.5 fieldOldText QString fieldOldText [private]

17.74.3.6 fieldOnText QString fieldOnText [private]

17.74.3.7 fieldVariesText QString fieldVariesText [private]

17.74.3.8 fieldYesText QString fieldYesText [private]

17.74.3.9 focusWidget QWidget* focusWidget [private]

17.74.3.10 iconDir QString iconDir [private]

17.74.3.11 iconSize int iconSize [private]

17.74.3.12 pickAdd bool pickAdd [private]

17.74.3.13 precisionAngle int precisionAngle [private]

17.74.3.14 precisionLength int precisionLength [private]

17.74.3.15 propertyEditorButtonStyle Qt::ToolButtonStyle propertyEditorButtonStyle [private]

17.74.3.16 selectedItemList QList<QGraphicsItem*> selectedItemList [private]

17.74.3.17 signalMapper QSignalMapper* signalMapper [private]

17.74.3.18 toolButtonPickAdd QToolButton* toolButtonPickAdd [private]

17.74.3.19 toolButtonQSelect QToolButton* toolButtonQSelect [private]

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

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

17.75 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 (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 `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`

17.75.1 Member Enumeration Documentation

17.75.1.1 anonymous enum anonymous enum

Enumerator

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

17.75.2 Constructor & Destructor Documentation

17.75.2.1 RectObject() [1/2] `RectObject (`

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

17.75.2.2 RectObject() [2/2] `RectObject (`

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

17.75.2.3 ~RectObject() `~RectObject ()`

`RectObject` destructor.

17.75.3 Member Function Documentation

17.75.3.1 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`.

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

Implements [BaseObject](#).

```
17.75.3.3 init() void init (
    EmbReal x,
    EmbReal y,
    EmbReal w,
    EmbReal h,
    QRgb rgb,
    Qt::PenStyle lineType )
```

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

Returns

The closest snap point to the mouse point.

Implements [BaseObject](#).

```
17.75.3.5 objectArea() EmbReal objectArea ( ) const [inline]
```

```
17.75.3.6 objectBottomLeft() QPointF objectBottomLeft ( ) const
```

```
17.75.3.7 objectBottomRight() QPointF objectBottomRight ( ) const
```

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

```
17.75.3.9 objectPos() QPointF objectPos ( ) const [inline]
```

17.75.3.10 objectSavePath() QPainterPath objectSavePath () const

17.75.3.11 objectTopLeft() QPointF objectTopLeft () const

Returns

The top left corner location as a QPointF.

17.75.3.12 objectTopRight() QPointF objectTopRight () const

17.75.3.13 objectWidth() EmbReal objectWidth () const [inline]

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

17.75.3.15 setObjectRect() void setObjectRect (
 EmbReal x,
 EmbReal y,
 EmbReal w,
 EmbReal h)

17.75.3.16 type() virtual int type () const [inline], [virtual]

Reimplemented from [BaseObject](#).

17.75.3.17 updatePath() void updatePath ()

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

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

Implements [BaseObject](#).

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

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

17.76 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`

17.76.1 Constructor & Destructor Documentation

17.76.1.1 `SaveObject()` `SaveObject (`
 `QGraphicsScene * theScene,`
 `QObject * parent = 0)`

17.76.1.2 `~SaveObject()` `~SaveObject ()`

17.76.2 Member Function Documentation

17.76.2.1 `addArc()` `void addArc (`
 `EmbPattern * pattern,`
 `QGraphicsItem * item)`

`SaveObject::addArc.`

Parameters

<code>pattern</code>	
<code>item</code>	

17.76.2.2 `addBlock()` `void addBlock (`
 `EmbPattern * pattern,`
 `QGraphicsItem * item)`

`SaveObject::addBlock.`

Parameters

<code>pattern</code>	
<code>item</code>	

17.76.2.3 addCircle() void addCircle (

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

[SaveObject::addCircle](#).

Parameters

<i>pattern</i>	<input type="button" value=""/>
<i>item</i>	<input type="button" value=""/>

17.76.2.4 addDimAligned() void addDimAligned (

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

[SaveObject::addDimAligned](#).

Parameters

<i>pattern</i>	<input type="button" value=""/>
<i>item</i>	<input type="button" value=""/>

17.76.2.5 addDimAngular() void addDimAngular (

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

[SaveObject::addDimAngular](#).

Parameters

<i>pattern</i>	<input type="button" value=""/>
<i>item</i>	<input type="button" value=""/>

17.76.2.6 addDimArcLength() void addDimArcLength (

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

[SaveObject::addDimArcLength](#).

Parameters

<i>pattern</i>	<input type="button" value=""/>
<i>item</i>	<input type="button" value=""/>

```
17.76.2.7 addDimDiameter() void addDimDiameter (
    EmbPattern * pattern,
    QGraphicsItem * item )
```

[SaveObject::addDimDiameter](#).

Parameters

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

```
17.76.2.8 addDimLeader() void addDimLeader (
    EmbPattern * pattern,
    QGraphicsItem * item )
```

[SaveObject::addDimLeader](#).

Parameters

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

```
17.76.2.9 addDimLinear() void addDimLinear (
    EmbPattern * pattern,
    QGraphicsItem * item )
```

```
17.76.2.10 addDimOrdinate() void addDimOrdinate (
    EmbPattern * pattern,
    QGraphicsItem * item )
```

```
17.76.2.11 addDimRadius() void addDimRadius (
    EmbPattern * pattern,
    QGraphicsItem * item )
```

```
17.76.2.12 addEllipse() void addEllipse (
    EmbPattern * pattern,
    QGraphicsItem * item )
```

17.76.2.13 addEllipseArc() void addEllipseArc (

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

17.76.2.14 addGrid() void addGrid (

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

17.76.2.15 addHatch() void addHatch (

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

17.76.2.16 addImage() void addImage (

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

17.76.2.17 addInfiniteLine() void addInfiniteLine (

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

17.76.2.18 addLine() void addLine (

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

17.76.2.19 addPath() void addPath (

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

17.76.2.20 addPoint() void addPoint (

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

17.76.2.21 addPolygon() void addPolygon (

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

17.76.2.22 addPolyline() void addPolyline (

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

17.76.2.23 addRay() void addRay (

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

17.76.2.24 addRectangle() void addRectangle (

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

17.76.2.25 addSlot() void addSlot (

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

17.76.2.26 addSpline() void addSpline (

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

17.76.2.27 addTextMulti() void addTextMulti (

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

17.76.2.28 addTextSingle() void addTextSingle (

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

```
17.76.2.29 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.

```
17.76.2.30 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);

17.76.3 Member Data Documentation

17.76.3.1 **formatType** int formatType

17.76.3.2 **gscene** QGraphicsScene* gscene

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

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

17.77 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`
- `uint8_t alpha`
- `QBrush dirBrush`
- `QBrush leftBrush`
- `QBrush rightBrush`
- `QPen dirPen`
- `QPen leftPen`
- `QPen rightPen`
- `bool boxDir`

Protected Member Functions

- `void paintEvent (QPaintEvent *)`

17.77.1 Constructor & Destructor Documentation

```
17.77.1.1 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/>

17.77.2 Member Function Documentation

```
17.77.2.1 forceRepaint() void forceRepaint ( )
```

```
17.77.2.2 paintEvent() void paintEvent (  
    QPaintEvent * ) [protected]
```

```
17.77.2.3 setColors void setColors (
    const QColor & colorL,
    const QColor & fillL,
    const QColor & colorR,
    const QColor & fillR,
    int newAlpha ) [slot]
```

```
17.77.2.4 setDirection void setDirection (
    int dir ) [slot]
```

17.77.3 Member Data Documentation

17.77.3.1 alpha uint8_t alpha

17.77.3.2 boxDir bool boxDir

17.77.3.3 dirBrush QBrush dirBrush

17.77.3.4 dirPen QPen dirPen

17.77.3.5 leftBrush QBrush leftBrush

17.77.3.6 leftBrushColor QColor leftBrushColor

17.77.3.7 leftPen QPen leftPen

17.77.3.8 leftPenColor QColor leftPenColor

17.77.3.9 rightBrush QBrush rightBrush

17.77.3.10 rightBrushColor QColor rightBrushColor

17.77.3.11 rightPen QPen rightPen

17.77.3.12 rightPenColor QColor rightPenColor

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

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

17.78 Settings_Struct Reference

Settings System.

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

Public Attributes

- `QString general_language`
- `QString general_icon_theme`
- `int general_icon_size`
- `QString version`
- `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 pattern_index`
- `QString assets_dir`
- `bool use_translation`
- `bool general_mdi_bg_use_logo`
- `bool general_mdi_bg_use_texture`
- `bool general_mdi_bg_use_color`
- `QString general_mdi_bg_logo`
- `QString general_mdi_bg_texture`
- `QRgb general_mdi_bg_color`
- `bool general_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`
- `QString display_units`
- `QString opensave_custom_filter`
- `QString opensave_open_format`
- `bool opensave_open_thumbnail`
- `QString opensave_save_format`

- bool `opensave_save_thumbnail`
- uint8_t `opensave_recent_max_files`
- QStringList `opensave_recent_list_of_files`
- QString `opensave_recent_directory`
- uint8_t `opensave_trim_dst_num_jumps`
- QString `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`
- QString `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`
- QString `text_font`
- 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`
- `uint32_t ticks_color`
- `uint32_t shine_color`
- `QString to_open`
- `QString current_directory`
- `EmbReal zoomInLimit`
- `EmbReal zoomOutLimit`
- `EmbReal ruler_width`
- `EmbReal tick_depth`
- `EmbReal major_tick_seperation`
- `EmbReal needle_speed`
- `EmbReal stitch_time`
- `QRgb prompt_text_color`
- `QRgb prompt_bg_color`
- `QString prompt_font_family`
- `QString prompt_font_style`
- `uint8_t prompt_font_size`
- `bool prompt_save_history`
- `bool prompt_save_history_as_html`
- `QString prompt_save_history_filename`

17.78.1 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.

17.78.2 Member Data Documentation

17.78.2.1 assets_dir `QString assets_dir`

17.78.2.2 current_directory `QString current_directory`

17.78.2.3 debug_mode `int debug_mode`

17.78.2.4 display_bg_color `uint32_t display_bg_color`

17.78.2.5 display_crosshair_color `uint32_t display_crosshair_color`

17.78.2.6 display_crosshair_percent `uint8_t display_crosshair_percent`

17.78.2.7 **display_renderhint_aa** bool display_renderhint_aa

17.78.2.8 **display_renderhint_high_aa** bool display_renderhint_high_aa

17.78.2.9 **display_renderhint_noncosmetic** bool display_renderhint_noncosmetic

17.78.2.10 **display_renderhint_smooth_pix** bool display_renderhint_smooth_pix

17.78.2.11 **display_renderhint_text_aa** bool display_renderhint_text_aa

17.78.2.12 **display_scrollbar_widget_num** int display_scrollbar_widget_num

17.78.2.13 **display_selectbox_alpha** uint8_t display_selectbox_alpha

17.78.2.14 **display_selectbox_left_color** uint32_t display_selectbox_left_color

17.78.2.15 **display_selectbox_left_fill** uint32_t display_selectbox_left_fill

17.78.2.16 **display_selectbox_right_color** uint32_t display_selectbox_right_color

17.78.2.17 **display_selectbox_right_fill** uint32_t display_selectbox_right_fill

17.78.2.18 **display_show_scrollbars** bool display_show_scrollbars

17.78.2.19 **display_units** QString display_units

17.78.2.20 **display_use_opengl** bool display_use_opengl

17.78.2.21 **display_zoomscale_in** EmbReal display_zoomscale_in

17.78.2.22 **display_zoomscale_out** EmbReal display_zoomscale_out

17.78.2.23 **general_check_for_updates** bool general_check_for_updates

17.78.2.24 **general_current_tip** uint32_t general_current_tip

17.78.2.25 **general_icon_size** int general_icon_size

17.78.2.26 **general_icon_theme** QString general_icon_theme

17.78.2.27 **general_language** QString general_language

17.78.2.28 **general_mdi_bg_color** QRgb general_mdi_bg_color

17.78.2.29 **general_mdi_bg_logo** QString general_mdi_bg_logo

17.78.2.30 **general_mdi_bg_texture** QString general_mdi_bg_texture

17.78.2.31 **general_mdi_bg_use_color** bool general_mdi_bg_use_color

17.78.2.32 **general_mdi_bg_use_logo** bool general_mdi_bg_use_logo

17.78.2.33 **general_mdi_bg_use_texture** bool general_mdi_bg_use_texture

17.78.2.34 **general_system_help_browser** bool general_system_help_browser

17.78.2.35 **general_tip_of_the_day** bool general_tip_of_the_day

17.78.2.36 **grid_center** EmbVector grid_center

17.78.2.37 **grid_center_on_origin** bool grid_center_on_origin

17.78.2.38 **grid_color** uint32_t grid_color

17.78.2.39 **grid_color_match_crosshair** bool grid_color_match_crosshair

17.78.2.40 **grid_load_from_file** bool grid_load_from_file

17.78.2.41 **grid_show_on_load** bool grid_show_on_load

17.78.2.42 **grid_show_origin** bool grid_show_origin

17.78.2.43 **grid_size** `EmbVector` `grid_size`

17.78.2.44 **grid_size_radius** `EmbReal` `grid_size_radius`

17.78.2.45 **grid_spacing** `EmbVector` `grid_spacing`

17.78.2.46 **grid_spacing_angle** `EmbReal` `grid_spacing_angle`

17.78.2.47 **grid_spacing_radius** `EmbReal` `grid_spacing_radius`

17.78.2.48 **grid_type** `QString` `grid_type`

17.78.2.49 **lwt_default_lwt** `EmbReal` `lwt_default_lwt`

17.78.2.50 **lwt_real_render** `bool` `lwt_real_render`

17.78.2.51 **lwt_show_lwt** `bool` `lwt_show_lwt`

17.78.2.52 **major_tick_seperation** `EmbReal` `major_tick_seperation`

17.78.2.53 **needle_speed** `EmbReal` `needle_speed`

17.78.2.54 **opensave_custom_filter** `QString` `opensave_custom_filter`

17.78.2.55 **opensave_open_format** `QString` `opensave_open_format`

17.78.2.56 **opensave_open_thumbnail** `bool` `opensave_open_thumbnail`

17.78.2.57 **opensave_recent_directory** `QString` `opensave_recent_directory`

17.78.2.58 **opensave_recent_list_of_files** `QStringList` `opensave_recent_list_of_files`

17.78.2.59 **opensave_recent_max_files** `uint8_t` `opensave_recent_max_files`

17.78.2.60 **opensave_save_format** `QString` `opensave_save_format`

17.78.2.61 opensave_save_thumbnail bool opensave_save_thumbnail

17.78.2.62 opensave_trim_dst_num_jumps uint8_t opensave_trim_dst_num_jumps

17.78.2.63 pattern_index int pattern_index

17.78.2.64 printing_default_device QString printing_default_device

17.78.2.65 printing_disable_bg bool printing_disable_bg

17.78.2.66 printing_use_last_device bool printing_use_last_device

17.78.2.67 prompt_bg_color QRgb prompt_bg_color

17.78.2.68 prompt_font_family QString prompt_font_family

17.78.2.69 prompt_font_size uint8_t prompt_font_size

17.78.2.70 prompt_font_style QString prompt_font_style

17.78.2.71 prompt_save_history bool prompt_save_history

17.78.2.72 prompt_save_history_as_html bool prompt_save_history_as_html

17.78.2.73 prompt_save_history_filename QString prompt_save_history_filename

17.78.2.74 prompt_text_color QRgb prompt_text_color

17.78.2.75 qsnap_aperture_size uint8_t qsnap_aperture_size

17.78.2.76 qsnap_apparent bool qsnap_apparent

17.78.2.77 qsnap_center bool qsnap_center

17.78.2.78 qsnap_enabled bool qsnap_enabled

17.78.2.79 `qsnap_endpoint` `bool qsnap_endpoint`

17.78.2.80 `qsnap_extension` `bool qsnap_extension`

17.78.2.81 `qsnap_insertion` `bool qsnap_insertion`

17.78.2.82 `qsnap_intersection` `bool qsnap_intersection`

17.78.2.83 `qsnap_locator_color` `uint32_t qsnap_locator_color`

17.78.2.84 `qsnap_locator_size` `uint8_t qsnap_locator_size`

17.78.2.85 `qsnap_midpoint` `bool qsnap_midpoint`

17.78.2.86 `qsnap_nearest` `bool qsnap_nearest`

17.78.2.87 `qsnap_node` `bool qsnap_node`

17.78.2.88 `qsnap_parallel` `bool qsnap_parallel`

17.78.2.89 `qsnap_perpendicular` `bool qsnap_perpendicular`

17.78.2.90 `qsnap_quadrant` `bool qsnap_quadrant`

17.78.2.91 `qsnap_tangent` `bool qsnap_tangent`

17.78.2.92 `ruler_color` `uint32_t ruler_color`

17.78.2.93 `ruler_metric` `bool ruler_metric`

17.78.2.94 `ruler_pixel_size` `uint8_t ruler_pixel_size`

17.78.2.95 `ruler_show_on_load` `bool ruler_show_on_load`

17.78.2.96 `ruler_width` `EmbReal ruler_width`

17.78.2.97 **running** bool running

17.78.2.98 **selection_coolgrip_color** uint32_t selection_coolgrip_color

17.78.2.99 **selection_grip_size** uint8_t selection_grip_size

17.78.2.100 **selection_hotgrip_color** uint32_t selection_hotgrip_color

17.78.2.101 **selection_mode_pickadd** bool selection_mode_pickadd

17.78.2.102 **selection_mode_pickdrag** bool selection_mode_pickdrag

17.78.2.103 **selection_mode_pickfirst** bool selection_mode_pickfirst

17.78.2.104 **selection_pickbox_size** uint8_t selection_pickbox_size

17.78.2.105 **shift_held** bool shift_held

17.78.2.106 **shine_color** uint32_t shine_color

17.78.2.107 **show_about_dialog** bool show_about_dialog

17.78.2.108 **show_details_dialog** bool show_details_dialog

17.78.2.109 **show_editor** bool show_editor

17.78.2.110 **show_open_file_dialog** bool show_open_file_dialog

17.78.2.111 **show_settings_editor** bool show_settings_editor

17.78.2.112 **stitch_time** EmbReal stitch_time

17.78.2.113 **testing** bool testing

17.78.2.114 **text_angle** EmbReal text_angle

17.78.2.115 **text_font** `QString text_font`

17.78.2.116 **text_size** `EmbReal text_size`

17.78.2.117 **text_style_bold** `bool text_style_bold`

17.78.2.118 **text_style_italic** `bool text_style_italic`

17.78.2.119 **text_style_overline** `bool text_style_overline`

17.78.2.120 **text_style_strikeout** `bool text_style_strikeout`

17.78.2.121 **text_style_underline** `bool text_style_underline`

17.78.2.122 **tick_depth** `EmbReal tick_depth`

17.78.2.123 **ticks_color** `uint32_t ticks_color`

17.78.2.124 **to_open** `QString to_open`

17.78.2.125 **use_translation** `bool use_translation`

17.78.2.126 **version** `QString version`

17.78.2.127 **zoomInLimit** `EmbReal zoomInLimit`

17.78.2.128 **zoomOutLimit** `EmbReal zoomOutLimit`

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

- [embroidermodder2/embroidermodder.h](#)

17.79 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)`
- `void create_float_spinbox (QGroupBox *gb, QGridLayout *gridLayout, const char *label_in, EmbReal single_step, EmbReal lower, EmbReal upper, EmbReal *ptr, int row)`
- `QCheckBox * create_checkbox (QGroupBox *groupbox, std::string label)`
- `void set_label_visibility (QObject *parent, const char *name, bool visibility)`
- `void set_spinbox_visibility (QObject *parent, const char *name, bool visibility)`

Public Attributes

- `MainWindow * mainWin`
- `QTabWidget * tabWidget`
- `QDialogButtonBox * buttonBox`

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 checkBoxTipOfTheDayStateChanged (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 `checkBoxRulerShowOnLoadStateChanged (int)`
- void `comboBoxRulerMetricCurrentIndexChanged (int)`
- void `chooseRulerColor ()`
- void `currentRulerColorChanged (const QColor &)`
- void `spinBoxRulerPixelSizeValueChanged (double)`
- 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 ()`

17.79.1 Constructor & Destructor Documentation

17.79.1.1 **Settings_Dialog()** `Settings_Dialog (`

```
    MainWindow * mw,  
    const QString & showTab = QString(),  
    QWidget * parent = 0 )
```

17.79.1.2 **~Settings_Dialog()** `~Settings_Dialog ()`

17.79.2 Member Function Documentation

17.79.2.1 **acceptChanges** `void acceptChanges () [private], [slot]`

17.79.2.2 **addColorsToComboBox()** `void addColorsToComboBox (`

```
    QComboBox * comboBox )
```

17.79.2.3 **buttonCustomFilterClearAll** `void buttonCustomFilterClearAll (`

```
    bool ) [signal]
```

17.79.2.4 **buttonCustomFilterClearAllClicked** `void buttonCustomFilterClearAllClicked () [private], [slot]`

17.79.2.5 **buttonCustomFilterSelectAll** `void buttonCustomFilterSelectAll (`

```
    bool ) [signal]
```

17.79.2.6 **buttonCustomFilterSelectAllClicked** `void buttonCustomFilterSelectAllClicked () [private], [slot]`

17.79.2.7 **buttonQSnapClearAll** `void buttonQSnapClearAll (`

```
    bool ) [signal]
```

17.79.2.8 **buttonQSnapClearAllClicked** `void buttonQSnapClearAllClicked () [private], [slot]`

17.79.2.9 **buttonQSnapSelectAll** `void buttonQSnapSelectAll (`

```
    bool ) [signal]
```

17.79.2.10 **buttonQSnapSelectAllClicked** `void buttonQSnapSelectAllClicked () [private], [slot]`

17.79.2.11 **checkBoxCustomFilterStateChanged** `void checkBoxCustomFilterStateChanged (`

```
    int checked ) [private], [slot]
```

17.79.2.12 `checkBoxDisableBGStateChanged` void checkBoxDisableBGStateChanged (int *checked*) [private], [slot]

17.79.2.13 `checkBoxGeneralMdiBGUseColorStateChanged` void checkBoxGeneralMdiBGUseColorStateChanged (int *checked*) [private], [slot]

17.79.2.14 `checkBoxGeneralMdiBGUseLogoStateChanged` void checkBoxGeneralMdiBGUseLogoStateChanged (int *checked*) [private], [slot]

17.79.2.15 `checkBoxGeneralMdiBGUseTextureStateChanged` void checkBoxGeneralMdiBGUseTextureStateChanged (int *checked*) [private], [slot]

17.79.2.16 `checkBoxGridCenterOnOriginStateChanged` void checkBoxGridCenterOnOriginStateChanged (int *checked*) [private], [slot]

17.79.2.17 `checkBoxGridColorMatchCrossHairStateChanged` void checkBoxGridColorMatchCrossHairStateChanged (int *checked*) [private], [slot]

17.79.2.18 `checkBoxGridLoadFromFileStateChanged` void checkBoxGridLoadFromFileStateChanged (int *checked*) [private], [slot]

17.79.2.19 `checkBoxGridShowOnLoadStateChanged` void checkBoxGridShowOnLoadStateChanged (int *checked*) [private], [slot]

17.79.2.20 `checkBoxGridShowOriginStateChanged` void checkBoxGridShowOriginStateChanged (int *checked*) [private], [slot]

17.79.2.21 `checkBoxLwtRealRenderStateChanged` void checkBoxLwtRealRenderStateChanged (int *checked*) [private], [slot]

17.79.2.22 `checkBoxLwtShowLwtStateChanged` void checkBoxLwtShowLwtStateChanged (int *checked*) [private], [slot]

17.79.2.23 `checkBoxPromptSaveHistoryAsHtmlStateChanged` void checkBoxPromptSaveHistoryAsHtmlStateChanged (int *checked*) [private], [slot]

```
17.79.2.24 checkBoxPromptSaveHistoryStateChanged void checkBoxPromptSaveHistoryStateChanged  
(  
    int checked ) [private], [slot]  
  
17.79.2.25 checkBoxRenderHintAAStateChanged void checkBoxRenderHintAAStateChanged (   
    int checked ) [private], [slot]  
  
17.79.2.26 checkBoxRenderHintHighAAStateChanged void checkBoxRenderHintHighAAStateChanged (   
    int checked ) [private], [slot]  
  
17.79.2.27 checkBoxRenderHintNonCosmeticStateChanged void checkBoxRenderHintNonCosmetic←  
StateChanged (   
    int checked ) [private], [slot]  
  
17.79.2.28 checkBoxRenderHintSmoothPixStateChanged void checkBoxRenderHintSmoothPixState←  
Changed (   
    int checked ) [private], [slot]  
  
17.79.2.29 checkBoxRenderHintTextAAStateChanged void checkBoxRenderHintTextAAStateChanged (   
    int checked ) [private], [slot]  
  
17.79.2.30 checkBoxRulerShowOnLoadStateChanged void checkBoxRulerShowOnLoadStateChanged (   
    int checked ) [private], [slot]  
  
17.79.2.31 checkBoxSelectionModePickAddStateChanged void checkBoxSelectionModePickAddState←  
Changed (   
    int checked ) [private], [slot]  
  
17.79.2.32 checkBoxSelectionModePickDragStateChanged void checkBoxSelectionModePickDrag←  
StateChanged (   
    int checked ) [private], [slot]  
  
17.79.2.33 checkBoxSelectionModePickFirstStateChanged void checkBoxSelectionModePickFirst←  
StateChanged (   
    int checked ) [private], [slot]  
  
17.79.2.34 checkBoxShowScrollBarsStateChanged void checkBoxShowScrollBarsStateChanged (   
    int checked ) [private], [slot]  
  
17.79.2.35 checkBoxTipOfTheDayStateChanged void checkBoxTipOfTheDayStateChanged (   
    int checked ) [private], [slot]
```

17.79.2.36 `checkBoxUseOpenGLStateChanged` void checkBoxUseOpenGLStateChanged (int checked) [private], [slot]

17.79.2.37 `chooseDisplayBackgroundColor` void chooseDisplayBackgroundColor () [private], [slot]

17.79.2.38 `chooseDisplayCrossHairColor` void chooseDisplayCrossHairColor () [private], [slot]

17.79.2.39 `chooseDisplaySelectBoxLeftColor` void chooseDisplaySelectBoxLeftColor () [private], [slot]

17.79.2.40 `chooseDisplaySelectBoxLeftFill` void chooseDisplaySelectBoxLeftFill () [private], [slot]

17.79.2.41 `chooseDisplaySelectBoxRightColor` void chooseDisplaySelectBoxRightColor () [private], [slot]

17.79.2.42 `chooseDisplaySelectBoxRightFill` void chooseDisplaySelectBoxRightFill () [private], [slot]

17.79.2.43 `chooseGeneralMdiBackgroundColor` void chooseGeneralMdiBackgroundColor () [private], [slot]

17.79.2.44 `chooseGeneralMdiBackgroundLogo` void chooseGeneralMdiBackgroundLogo () [private], [slot]

17.79.2.45 `chooseGeneralMdiBackgroundTexture` void chooseGeneralMdiBackgroundTexture () [private], [slot]

17.79.2.46 `chooseGridColor` void chooseGridColor () [private], [slot]

17.79.2.47 `choosePromptBackgroundColor` void choosePromptBackgroundColor () [private], [slot]

17.79.2.48 `choosePromptTextColor` void choosePromptTextColor () [private], [slot]

17.79.2.49 `chooseRulerColor` void chooseRulerColor () [private], [slot]

17.79.2.50 `comboBoxGridTypeCurrentIndexChanged` void comboBoxGridTypeCurrentIndexChanged (const QString & type) [private], [slot]

17.79.2.51 comboBoxIconSizeCurrentIndexChanged void comboBoxIconSizeCurrentIndexChanged (int index) [private], [slot]

17.79.2.52 comboBoxIconThemeCurrentIndexChanged void comboBoxIconThemeCurrentIndexChanged (const QString & theme) [private], [slot]

17.79.2.53 comboBoxLanguageCurrentIndexChanged void comboBoxLanguageCurrentIndexChanged (const QString & lang) [private], [slot]

17.79.2.54 comboBoxPromptFontFamilyCurrentIndexChanged void comboBoxPromptFontFamilyCurrentIndexChanged (const QString & family) [private], [slot]

17.79.2.55 comboBoxPromptFontStyleCurrentIndexChanged void comboBoxPromptFontStyleCurrentIndexChanged (const QString & style) [private], [slot]

17.79.2.56 comboBoxQSnapLocatorColorCurrentIndexChanged void comboBoxQSnapLocatorColorCurrentIndexChanged (int index) [private], [slot]

Todo Figure out how to abstract the slot in a way that it can be used for comboBoxes in general Currently [comboBoxQSnapLocatorColorCurrentIndexChanged\(int index\)](#) [comboBoxSelectionCoolGripColorCurrentIndexChanged\(int index\)](#) [comboBoxSelectionHotGripColorCurrentIndexChanged\(int index\)](#) are all similar except the dialog. variable being worked on and the QVariant.

17.79.2.57 comboBoxRulerMetricCurrentIndexChanged void comboBoxRulerMetricCurrentIndexChanged (int index) [private], [slot]

17.79.2.58 comboBoxScrollBarWidgetCurrentIndexChanged void comboBoxScrollBarWidgetCurrentIndexChanged (int index) [private], [slot]

17.79.2.59 comboBoxSelectionCoolGripColorCurrentIndexChanged void comboBoxSelectionCoolGripColorCurrentIndexChanged (int index) [private], [slot]

17.79.2.60 comboBoxSelectionHotGripColorCurrentIndexChanged void comboBoxSelectionHotGripColorCurrentIndexChanged (int index) [private], [slot]

17.79.2.61 create_checkbox() QCheckBox * create_checkbox (QGroupBox * groupbox, std::string label)

```
17.79.2.62 create_float_spinbox() void create_float_spinbox (
    QGroupBox * gb,
    QGridLayout * gridLayout,
    const char * label_in,
    EmbReal single_step,
    EmbReal lower,
    EmbReal upper,
    EmbReal * ptr,
    int row )
```

17.79.2.63 **createTabDisplay()** QWidget * createTabDisplay ()

17.79.2.64 **createTabFilesPaths()** QWidget * createTabFilesPaths ()

17.79.2.65 **createTabGeneral()** QWidget * createTabGeneral ()

17.79.2.66 **createTabGridRuler()** QWidget * createTabGridRuler ()

17.79.2.67 **createTabLineWeight()** QWidget * createTabLineWeight ()

17.79.2.68 **createTabOpenSave()** QWidget * createTabOpenSave ()

17.79.2.69 **createTabOrthoPolar()** QWidget * createTabOrthoPolar ()

17.79.2.70 **createTabPrinting()** QWidget * createTabPrinting ()

17.79.2.71 **createTabPrompt()** QWidget * createTabPrompt ()

17.79.2.72 **createTabQuickSnap()** QWidget * createTabQuickSnap ()

17.79.2.73 **createTabQuickTrack()** QWidget * createTabQuickTrack ()

17.79.2.74 **createTabSelection()** QWidget * createTabSelection ()

17.79.2.75 **createTabSnap()** QWidget * createTabSnap ()

17.79.2.76 **currentDisplayBackgroundColorChanged** void currentDisplayBackgroundColorChanged (
 const QColor & color) [private], [slot]

17.79.2.77 currentDisplayCrossHairColorChanged void currentDisplayCrossHairColorChanged (const QColor & color) [private], [slot]

17.79.2.78 currentDisplaySelectBoxLeftColorChanged void currentDisplaySelectBoxLeftColorChanged (const QColor & color) [private], [slot]

17.79.2.79 currentDisplaySelectBoxLeftFillChanged void currentDisplaySelectBoxLeftFillChanged (const QColor & color) [private], [slot]

17.79.2.80 currentDisplaySelectBoxRightColorChanged void currentDisplaySelectBoxRightColorChanged (const QColor & color) [private], [slot]

17.79.2.81 currentDisplaySelectBoxRightFillChanged void currentDisplaySelectBoxRightFillChanged (const QColor & color) [private], [slot]

17.79.2.82 currentGeneralMdiBackgroundColorChanged void currentGeneralMdiBackgroundColorChanged (const QColor & color) [private], [slot]

17.79.2.83 currentGridColorChanged void currentGridColorChanged (const QColor & color) [private], [slot]

17.79.2.84 currentPromptBackgroundColorChanged void currentPromptBackgroundColorChanged (const QColor & color) [private], [slot]

17.79.2.85 currentPromptTextColorChanged void currentPromptTextColorChanged (const QColor & color) [private], [slot]

17.79.2.86 currentRulerColorChanged void currentRulerColorChanged (const QColor & color) [private], [slot]

17.79.2.87 rejectChanges void rejectChanges () [private], [slot]

17.79.2.88 set_label_visibility() void set_label_visibility (QObject * parent, const char * name, bool visibility)

Todo error reporting.

```
17.79.2.89 set_spinbox_visibility() void set_spinbox_visibility (
    QObject * parent,
    const char * name,
    bool visibility )
```

Todo error reporting.

```
17.79.2.90 sliderQSnapApertureSizeValueChanged void sliderQSnapApertureSizeValueChanged (
    int value ) [private], [slot]
```

```
17.79.2.91 sliderQSnapLocatorSizeValueChanged void sliderQSnapLocatorSizeValueChanged (
    int value ) [private], [slot]
```

```
17.79.2.92 sliderSelectionGripSizeValueChanged void sliderSelectionGripSizeValueChanged (
    int value ) [private], [slot]
```

```
17.79.2.93 sliderSelectionPickBoxSizeValueChanged void sliderSelectionPickBoxSizeValueChanged (
    int value ) [private], [slot]
```

```
17.79.2.94 spinBoxDisplaySelectBoxAlphaValueChanged void spinBoxDisplaySelectBoxAlphaValueChanged (
    int value ) [private], [slot]
```

```
17.79.2.95 spinBoxPromptFontSizeValueChanged void spinBoxPromptFontSizeValueChanged (
    int value ) [private], [slot]
```

```
17.79.2.96 spinBoxRecentMaxFilesValueChanged void spinBoxRecentMaxFilesValueChanged (
    int value ) [private], [slot]
```

```
17.79.2.97 spinBoxRulerPixelSizeValueChanged void spinBoxRulerPixelSizeValueChanged (
    double value ) [private], [slot]
```

```
17.79.2.98 spinBoxTrimDstNumJumpsValueChanged void spinBoxTrimDstNumJumpsValueChanged (
    int value ) [private], [slot]
```

```
17.79.2.99 spinBoxZoomScaleInValueChanged void spinBoxZoomScaleInValueChanged (
    double value ) [private], [slot]
```

```
17.79.2.100 spinBoxZoomScaleOutValueChanged void spinBoxZoomScaleOutValueChanged (
    double value ) [private], [slot]
```

17.79.3 Member Data Documentation

17.79.3.1 buttonBox QDialogButtonBox* buttonBox

17.79.3.2 mainWin MainWindow* mainWin

17.79.3.3 tabWidget QTabWidget* tabWidget

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

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

17.80 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](#)

17.80.1 Constructor & Destructor Documentation

17.80.1.1 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/>

17.80.2 Member Function Documentation

17.80.2.1 setMouseCoord() void setMouseCoord (
 EmbReal *x*,
 EmbReal *y*)

17.80.3 Member Data Documentation

17.80.3.1 statusBarGridButton `StatusBarButton* statusBarGridButton`

17.80.3.2 statusBarLwtButton `StatusBarButton* statusBarLwtButton`

17.80.3.3 statusBarMouseCoord `QLabel* statusBarMouseCoord`

17.80.3.4 statusBarOrthoButton `StatusBarButton* statusBarOrthoButton`

17.80.3.5 statusBarPolarButton `StatusBarButton* statusBarPolarButton`

17.80.3.6 statusBarQSnapButton `StatusBarButton* statusBarQSnapButton`

17.80.3.7 statusBarQTrackButton `StatusBarButton* statusBarQTrackButton`

17.80.3.8 statusBarRulerButton `StatusBarButton* statusBarRulerButton`

17.80.3.9 statusBarSnapButton `StatusBarButton* statusBarSnapButton`

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

- `embroidermodder2/embroidermodder.h`
- `embroidermodder2/statusbar.cpp`

17.81 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)`

17.81.1 Constructor & Destructor Documentation**17.81.1.1 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/>

17.81.2 Member Function Documentation**17.81.2.1 contextMenuEvent()** `void contextMenuEvent (`

```
    QContextMenuEvent * event = 0 ) [protected]
```

17.81.2.2 disableLwt `void disableLwt () [slot]`**17.81.2.3 disableReal** `void disableReal () [slot]`**17.81.2.4 enableLwt** `void enableLwt () [slot]`**17.81.2.5 enableReal** `void enableReal () [slot]`**17.81.2.6 settingsGrid** `void settingsGrid () [private], [slot]`

17.81.2.7 settingsLwt void settingsLwt () [private], [slot]

17.81.2.8 settingsOrtho void settingsOrtho () [private], [slot]

17.81.2.9 settingsPolar void settingsPolar () [private], [slot]

17.81.2.10 settingsQSnap void settingsQSnap () [private], [slot]

17.81.2.11 settingsQTrack void settingsQTrack () [private], [slot]

17.81.2.12 settingsRuler void settingsRuler () [private], [slot]

17.81.2.13 settingsSnap void settingsSnap () [private], [slot]

17.81.2.14 toggleGrid void toggleGrid (bool on) [private], [slot]

17.81.2.15 toggleLwt void toggleLwt (bool on) [private], [slot]

17.81.2.16 toggleOrtho void toggleOrtho (bool on) [private], [slot]

17.81.2.17 togglePolar void togglePolar (bool on) [private], [slot]

17.81.2.18 toggleQSnap void toggleQSnap (bool on) [private], [slot]

17.81.2.19 toggleQTrack void toggleQTrack (bool on) [private], [slot]

17.81.2.20 toggleRuler void toggleRuler (bool on) [private], [slot]

17.81.2.21 toggleSnap void toggleSnap (bool on) [private], [slot]

17.81.3 Member Data Documentation

17.81.3.1 mainWin `MainWindow* mainWin`**17.81.3.2 statusbar** `StatusBar* statusbar`

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

- `embroidermodder2/embroidermodder.h`
- `embroidermodder2/statusbar-button.cpp`

17.82 StxThread_ Struct Reference

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

Public Attributes

- `char * colorCode`
- `char * colorName`
- `char * sectionName`
- `SubDescriptor * subDescriptors`
- `EmbColor stxColor`

17.82.1 Member Data Documentation**17.82.1.1 colorCode** `char* colorCode`**17.82.1.2 colorName** `char* colorName`**17.82.1.3 sectionName** `char* sectionName`**17.82.1.4 stxColor** `EmbColor stxColor`**17.82.1.5 subDescriptors** `SubDescriptor* subDescriptors`

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

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

17.83 SubDescriptor_ Struct Reference

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

Public Attributes

- `int someNum`
- `int someInt`
- `int someOtherInt`
- `char * colorCode`
- `char * colorName`

17.83.1 Member Data Documentation

17.83.1.1 colorCode char* colorCode

Todo better variable naming

17.83.1.2 colorName char* colorName

17.83.1.3 someInt int someInt

Todo better variable naming

17.83.1.4 someNum int someNum

17.83.1.5 someOtherInt int someOtherInt

Todo better variable naming

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

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

17.84 SvgAttribute_ Struct Reference

#include <embroidery_internal.h>

Public Attributes

- char * [name](#)
- char * [value](#)

17.84.1 Member Data Documentation

17.84.1.1 name char* name

17.84.1.2 value char* value

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

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

17.85 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 (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 `setObjectLineWeight` (`EmbReal` lineWeight)
- void `setObjectPath` (const `QPainterPath` &p)
- void `setObjectRubberMode` (int mode)
- void `setObjectRubberPoint` (const `QString` &key, const `QPointF` &point)
- void `setObjectRubberText` (const `QString` &key, const `QString` &txt)
- virtual `QRectF boundingRect` () const
- virtual `QPainterPath shape` () const
- void `drawRubberLine` (const `QLineF` &rubLine, `QPainter` *painter=0, const char *colorFromScene=0)
- virtual void `vulcanize` ()=0
- virtual `QPointF mouseSnapPoint` (const `QPointF` &mousePoint)=0
- virtual `QList<QPointF>` `allGripPoints` ()=0
- virtual void `gripEdit` (const `QPointF` &before, const `QPointF` &after)=0

Public Attributes

- `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)

17.85.1 Member Enumeration Documentation

17.85.1.1 anonymous enum anonymous enum

Enumerator

Type	
------	--

17.85.2 Constructor & Destructor Documentation

17.85.2.1 `TextSingleObject()` [1/2]

```
TextSingleObject( const QString & str,
                  EmbReal x,
                  EmbReal y,
                  QRgb rgb,
                  QGraphicsItem * parent = 0 )
```

17.85.2.2 `TextSingleObject()` [2/2]

```
TextSingleObject( TextSingleObject * obj,
                  QGraphicsItem * parent = 0 )
```

17.85.2.3 `~TextSingleObject()`

17.85.3 Member Function Documentation

17.85.3.1 `allGripPoints()`

QList< QPointF > allGripPoints() [virtual]

Implements [BaseObject](#).

17.85.3.2 `gripEdit()`

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

Implements [BaseObject](#).

17.85.3.3 `init()`

```
void init( const QString & str,
            EmbReal x,
            EmbReal y,
            QRgb rgb,
            Qt::PenStyle lineType )
```

17.85.3.4 `mouseSnapPoint()`

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

Returns the closest snap point to the mouse point

Implements [BaseObject](#).

17.85.3.5 `objectPos()` `QPointF objectPos () const [inline]`

17.85.3.6 `objectSavePathList()` `QList< QPainterPath > objectSavePathList () const [inline]`

17.85.3.7 `objectTextJustifyList()` `QStringList objectTextJustifyList () const`

17.85.3.8 `objectX()` `EmbReal objectX () const [inline]`

17.85.3.9 `objectY()` `EmbReal objectY () const [inline]`

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

17.85.3.11 `setObjectPos()` [1/2] `void setObjectPos (`
 `const QPointF & point) [inline]`

17.85.3.12 `setObjectPos()` [2/2] `void setObjectPos (`
 `EmbReal x,`
 `EmbReal y) [inline]`

17.85.3.13 `setObjectText()` `void setObjectText (`
 `const QString & str)`

17.85.3.14 `setObjectTextBackward()` `void setObjectTextBackward (`
 `bool val)`

17.85.3.15 `setObjectTextBold()` `void setObjectTextBold (`
 `bool val)`

17.85.3.16 `setObjectTextFont()` `void setObjectTextFont (`
 `const QString & font)`

17.85.3.17 `setObjectTextItalic()` `void setObjectTextItalic (`
 `bool val)`

17.85.3.18 `setObjectTextJustify()` `void setObjectTextJustify (`
 `const QString & justify)`

17.85.3.19 `setObjectTextOverline()` void setObjectTextOverline (bool val)

17.85.3.20 `setObjectTextSize()` void setObjectTextSize (EmbReal size)

17.85.3.21 `setObjectTextStrikeOut()` void setObjectTextStrikeOut (bool val)

17.85.3.22 `setObjectTextStyle()` void setObjectTextStyle (bool bold, bool italic, bool under, bool strike, bool over)

17.85.3.23 `setObjectTextUnderline()` void setObjectTextUnderline (bool val)

17.85.3.24 `setObjectTextUpsideDown()` void setObjectTextUpsideDown (bool val)

17.85.3.25 `setObjectX()` void setObjectX (EmbReal x) [inline]

17.85.3.26 `setObjectY()` void setObjectY (EmbReal y) [inline]

17.85.3.27 `subPathList()` QList< QPainterPath > subPathList () const

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

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

17.85.3.30 `vulcanize()` void vulcanize () [virtual]
Implements [BaseObject](#).

17.85.4 Member Data Documentation

17.85.4.1 `objText` QString objText

17.85.4.2 objTextBackward bool objTextBackward

17.85.4.3 objTextBold bool objTextBold

17.85.4.4 objTextFont QString objTextFont

17.85.4.5 objTextItalic bool objTextItalic

17.85.4.6 objTextJustify QString objTextJustify

17.85.4.7 objTextOverline bool objTextOverline

17.85.4.8 objTextPath QPainterPath objTextPath

17.85.4.9 objTextSize EmbReal objTextSize

17.85.4.10 objTextStrikeOut bool objTextStrikeOut

17.85.4.11 objTextUnderline bool objTextUnderline

17.85.4.12 objTextUpsideDown bool objTextUpsideDown

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

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

17.86 thread_color_ Struct Reference

#include <embroidery.h>

Public Attributes

- char [name](#) [22]
- unsigned int [hex_code](#)
- int [manufacturer_code](#)

17.86.1 Member Data Documentation

17.86.1.1 hex_code unsigned int hex_code

17.86.1.2 manufacturer_code int manufacturer_code

17.86.1.3 name char name[22]

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

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

17.87 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]

17.87.1 Member Data Documentation

17.87.1.1 auxFormat char auxFormat**17.87.1.2 creatorName** char creatorName[50]**17.87.1.3 hoopX** float hoopX**17.87.1.4 hoopY** float hoopY**17.87.1.5 modifierName** char modifierName[50]**17.87.1.6 reserved** char reserved[31]**17.87.1.7 stitchGranularity** float stitchGranularity

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

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

17.88 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]

17.88.1 Member Data Documentation

17.88.1.1 hoopSize unsigned short hoopSize

17.88.1.2 length unsigned int length

17.88.1.3 numStiches unsigned short numStiches

17.88.1.4 reserved unsigned short reserved[7]

17.88.1.5 sigVersion unsigned int sigVersion

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

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

17.89 UiObject_ Struct Reference

This covers the inbuilt designs: Dolphin, Snowflake and Heart. Covers Rotate, Scale and Point UI events.

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

Public Attributes

- std::string [fname](#)
- std::string [command](#)
- bool [firstRun](#)
- std::vector< [EmbVector](#) > [controlPoints](#)
- std::vector< std::string > [controlPointLabels](#)
- int [numPoints](#)
- int [minPoints](#)
- int [maxPoints](#)
- [EmbVector](#) [center](#)
- [EmbVector](#) [scale](#)
- [EmbReal](#) [rotation](#)
- uint32_t [mode](#)
- std::string [path_desc](#)
- std::string [text](#)
- int [textJustify](#)
- std::string [textFont](#)
- [EmbReal](#) [textHeight](#)
- [EmbReal](#) [textRotation](#)
- std::string [id](#)
- int [pattern_index](#)
- char [type](#) [200]
- int [object_index](#)
- bool [selectable](#)
- [EmbColor](#) [color](#)

17.89.1 Detailed Description

This covers the inbuilt designs: Dolphin, Snowflake and Heart. Covers Rotate, Scale and Point UI events.

This was an idea for storing the current command state: could be combined with EmbView since you can't have more than one active command. If a command calls a sub command it will store the position in the parents.

17.89.2 Member Data Documentation**17.89.2.1 center** `EmbVector` center**17.89.2.2 color** `EmbColor` color**17.89.2.3 command** `std::string` command**17.89.2.4 controlPointLabels** `std::vector<std::string>` controlPointLabels**17.89.2.5 controlPoints** `std::vector<EmbVector>` controlPoints**17.89.2.6 firstRun** `bool` firstRun**17.89.2.7 fname** `std::string` fname**17.89.2.8 id** `std::string` id**17.89.2.9 maxPoints** `int` maxPoints**17.89.2.10 minPoints** `int` minPoints**17.89.2.11 mode** `uint32_t` mode**17.89.2.12 numPoints** `int` numPoints**17.89.2.13 object_index** `int` object_index**17.89.2.14 path_desc** `std::string` path_desc**17.89.2.15 pattern_index** `int` pattern_index**17.89.2.16 rotation** `EmbReal` rotation**17.89.2.17 scale** `EmbVector` scale

17.89.2.18 selectable bool selectable

17.89.2.19 text std::string text

17.89.2.20 textFont std::string textFont

17.89.2.21 textHeight EmbReal textHeight

17.89.2.22 textJustify int textJustify

17.89.2.23 textRotation EmbReal textRotation

17.89.2.24 type char type[200]

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

- [embroidermodder2/embroidermodder.h](#)

17.90 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

17.90.1 Constructor & Destructor Documentation

17.90.1.1 UndoableAddCommand() [UndoableAddCommand](#) (

```
    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 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`

17.91.1 Constructor & Destructor Documentation

```
17.91.1.1 UndoableDeleteCommand() UndoableDeleteCommand (  
    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 gview `View* gview`

17.91.3.2 object `BaseObject* object`

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

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

17.92 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`

17.92.1 Constructor & Destructor Documentation

```
17.92.1.1 UndoableGripEditCommand() UndoableGripEditCommand (
```

```
    const QPointF beforePoint,
```

```
    const QPointF afterPoint,
```

```
    const QString & text,
```

```
    BaseObject * obj,
```

```
    View * v,
```

```
    QUndoCommand * parent = 0 )
```

17.92.2 Member Function Documentation

```
17.92.2.1 redo() void redo ( )
```

```
17.92.2.2 undo() void undo ( )
```

17.92.3 Member Data Documentation

```
17.92.3.1 after QPointF after
```

```
17.92.3.2 before QPointF before
```

```
17.92.3.3 gview View* gview
```

```
17.92.3.4 object BaseObject* object
```

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

- `embroidermodder2/embroidermodder.h`
- `embroidermodder2/undo-commands.cpp`

17.93 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`

17.93.1 Constructor & Destructor Documentation

```
17.93.1.1 UndoableMirrorCommand() UndoableMirrorCommand (
    EmbReal x1,
    EmbReal y1,
    EmbReal x2,
    EmbReal y2,
    const QString & text,
    BaseObject * obj,
    View * v,
    QUndoCommand * parent = 0 )
```

17.93.2 Member Function Documentation

```
17.93.2.1 mirror() void mirror ( )
```

```
17.93.2.2 redo() void redo ( )
```

```
17.93.2.3 undo() void undo ( )
```

17.93.3 Member Data Documentation

```
17.93.3.1 gview View* gview
```

```
17.93.3.2 mirrorLine QLineF mirrorLine
```

```
17.93.3.3 object BaseObject* object
```

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

- `embroidermodder2/embroidermodder.h`
- `embroidermodder2/undo-commands.cpp`

17.94 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`

17.94.1 Constructor & Destructor Documentation

```
17.94.1.1 UndoableMoveCommand() UndoableMoveCommand (
```

```
    EmbReal deltaX,
```

```
    EmbReal deltaY,
```

```
    const QString & text,
```

```
    BaseObject * obj,
```

```
    View * v,
```

```
    QUndoCommand * parent = 0 )
```

17.94.2 Member Function Documentation

```
17.94.2.1 redo() void redo ( )
```

```
17.94.2.2 undo() void undo ( )
```

17.94.3 Member Data Documentation

```
17.94.3.1 dx EmbReal dx
```

```
17.94.3.2 dy EmbReal dy
```

```
17.94.3.3 gview View* gview
```

```
17.94.3.4 object BaseObject* object
```

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

- `embroidermodder2/embroidermodder.h`
- `embroidermodder2/undo-commands.cpp`

17.95 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`

17.95.1 Constructor & Destructor Documentation

```
17.95.1.1 UndoableNavCommand() UndoableNavCommand (
```

```
    const QString & type,
```

```
    View * v,
```

```
    QUndoCommand * parent = 0 )
```

17.95.2 Member Function Documentation

```
17.95.2.1 id() int id () const [inline]
```

```
17.95.2.2 mergeWith() bool mergeWith (
```

```
    const QUndoCommand * command )
```

```
17.95.2.3 redo() void redo ()
```

```
17.95.2.4 undo() void undo ()
```

17.95.3 Member Data Documentation

```
17.95.3.1 done bool done
```

```
17.95.3.2 fromCenter QPointF fromCenter
```

17.95.3.3 fromTransform `QTransform fromTransform`

17.95.3.4 gview `View* gview`

17.95.3.5 navType `QString navType`

17.95.3.6 toCenter `QPointF toCenter`

17.95.3.7 toTransform `QTransform toTransform`

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

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

17.96 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`

17.96.1 Constructor & Destructor Documentation

17.96.1.1 UndoableRotateCommand() `UndoableRotateCommand (`
`EmbReal pivotPointX,`
`EmbReal pivotPointY,`
`EmbReal rotAngle,`
`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 rotate() void rotate (
    EmbReal x,
    EmbReal y,
    EmbReal rot )
```

```
17.96.2.3 undo() void undo ( )
```

17.96.3 Member Data Documentation

```
17.96.3.1 angle EmbReal angle
```

```
17.96.3.2 gview View* gview
```

```
17.96.3.3 object BaseObject* object
```

```
17.96.3.4 pivotX EmbReal pivotX
```

```
17.96.3.5 pivotY EmbReal pivotY
```

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

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

17.97 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](#)

17.97.1 Constructor & Destructor Documentation

```
17.97.1.1 UndoableScaleCommand() UndoableScaleCommand (
    EmbReal x,
    EmbReal y,
    EmbReal scaleFactor,
    const QString & text,
    BaseObject * obj,
    View * v,
    QUndoCommand * parent = 0 )
```

17.97.2 Member Function Documentation

17.97.2.1 **redo()** void redo ()

17.97.2.2 **undo()** void undo ()

17.97.3 Member Data Documentation

17.97.3.1 **dx** EmbReal dx

17.97.3.2 **dy** EmbReal dy

17.97.3.3 **factor** EmbReal factor

17.97.3.4 **gview** View* gview

17.97.3.5 **object** BaseObject* object

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

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

17.98 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`

17.98.1 Constructor & Destructor Documentation

17.98.1.1 UndoEditor() `UndoEditor (`
 `const QString & iconDirectory = QString(),`
 `QWidget * widgetToFocus = 0,`
 `QWidget * parent = 0)`

17.98.1.2 ~UndoEditor() `~UndoEditor ()`

17.98.2 Member Function Documentation

17.98.2.1 addStack() `void addStack (`
 `QUndoStack * stack)`

17.98.2.2 canRedo() `bool canRedo () const`

17.98.2.3 canUndo() `bool canUndo () const`

17.98.2.4 redo `void redo () [slot]`

17.98.2.5 redoText() `QString redoText () const`

17.98.2.6 undo `void undo () [slot]`

17.98.2.7 undoText() `QString undoText () const`

17.98.2.8 updateCleanIcon `void updateCleanIcon (`
 `bool opened) [slot]`

17.98.3 Member Data Documentation

17.98.3.1 focusWidget QWidget* focusWidget

17.98.3.2 iconDir QString iconDir

17.98.3.3 iconSize int iconSize

17.98.3.4 undoGroup QUndoGroup* undoGroup

17.98.3.5 undoView QUndoView* undoView

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

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

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` (uint8_t 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)

Public Attributes

- EmbView `view_state`

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`
- uint8_t `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`
- `uint8_t qsnapLocatorSize`
- `uint8_t qsnapApertureSize`
- `QRgb gripColorCool`
- `QRgb gripColorHot`
- `uint8_t gripSize`
- `uint8_t 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]`

Todo turn move into an actuator call.

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 (
 uint8_t 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` uint8_t 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** uint8_t 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** uint8_t qsnapApertureSize [private]

17.99.3.36 **qsnapLocatorColor** QRgb qsnapLocatorColor [private]

17.99.3.37 **qsnapLocatorSize** uint8_t 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 `uint8_t 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 view_state `EmbView view_state`

17.99.3.56 viewMousePoint `QPoint viewMousePoint [private]`

17.99.3.57 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/data.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
- QFontComboBox * comboBoxTextSingleFont
- std::vector< std::string > group_box_list
- [GroupBoxData group_box_general \[\]](#)
- const int group_box_general_entries = 0
- [GroupBoxData group_box_arc_geometry \[\]](#)
- const int group_box_arc_geometry_entries = 13
- [GroupBoxData group_box_arc_misc \[\]](#)
- const int group_box_arc_misc_entries = 0
- [GroupBoxData group_box_ellipse_geometry \[\]](#)
- const int group_box_ellipse_geometry_entries = 6
- std::unordered_map< std::string, [GroupBoxData](#) * > group_box_data

18.3.1 Detailed Description

For all global data.

18.3.2 Variable Documentation

18.3.2.1 comboBoxes std::unordered_map<std::string, QComboBox *> comboBoxes

18.3.2.2 comboBoxTextSingleFont QFontComboBox* comboBoxTextSingleFont

18.3.2.3 group_box_arc_geometry [GroupBoxData](#) group_box_arc_geometry[]

18.3.2.4 group_box_arc_geometry_entries const int group_box_arc_geometry_entries = 13

18.3.2.5 group_box_arc_misc GroupBoxData group_box_arc_misc[]

Initial value:

```
= {  
}
```

18.3.2.6 group_box_arc_misc_entries const int group_box_arc_misc_entries = 0

18.3.2.7 group_box_data std::unordered_map<std::string, GroupBoxData*> group_box_data

18.3.2.8 group_box_ellipse_geometry GroupBoxData group_box_ellipse_geometry[]

Todo use proper icons for toolButtons

18.3.2.9 group_box_ellipse_geometry_entries const int group_box_ellipse_geometry_entries = 6

18.3.2.10 group_box_general GroupBoxData group_box_general[]

Initial value:

```
= {  
}
```

18.3.2.11 group_box_general_entries const int group_box_general_entries = 0

18.3.2.12 group_box_list std::vector<std::string> group_box_list

18.3.2.13 groupBoxes std::unordered_map<std::string, QGroupBox *> groupBoxes

18.3.2.14 lineEdits std::unordered_map<std::string, QLineEdit *> lineEdits

18.3.2.15 toolButtons std::unordered_map<std::string, QToolButton *> toolButtons

18.4 embroidermodder2/embdetails-dialog.cpp File Reference

```
#include "embroidermodder.h"
```

18.5 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

18.5.1 Function Documentation

18.5.1.1 main() int main (
 int argc,
 char * argv[])
qMain

Parameters

<i>argc</i>	
<i>argv</i>	

Returns

18.5.1.2 usage() static void usage (
 void) [static]
usage

18.5.1.3 version() static void version () [static]
version

18.5.2 Variable Documentation

18.5.2.1 _appName_ const char* **_appName_** = "Embroidermodder" [static]

18.5.2.2 _appVer_ const char* **_appVer_** = "v2.0 alpha" [static]

18.5.2.3 exitApp bool **exitApp** = false [static]

18.6 embroidermodder2/embroidermodder.h File Reference

```
#include <cstdio>
#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 [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](#)
- struct [GroupBoxData_](#)
- 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 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 [GroupBoxData_GroupBoxData](#)
- 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_MAJORDIAMETER_MINOR
    , 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 }
• 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

- 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)
- void debug_message (std::string msg)
- int read_settings (const char *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)

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
- 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
- QFontComboBox * comboBoxTextSingleFont
- std::vector< std::string > group_box_list
- GroupBoxData group_box_arc_geometry []
- const int group_box_arc_geometry_entries
- GroupBoxData group_box_ellipse_geometry []
- const int group_box_ellipse_geometry_entries
- std::unordered_map< std::string, std::string > config
- std::unordered_map< std::string, GroupBoxData * > group_box_data

18.6.1 Detailed Description

The only header for the GUI part: a good overview of this source code.

18.6.2 Typedef Documentation

18.6.2.1 Action `typedef struct Action_ Action`

18.6.2.2 EmbView `typedef struct EmbView_ EmbView`

18.6.3 EmbViews

The EmbView describes how the render is displayed.

18.6.3.1 GroupBoxData `typedef struct GroupBoxData_ GroupBoxData`

18.6.3.2 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.

18.6.3.3 UiObject `typedef struct UiObject_ UiObject`

This covers the inbuilt designs: Dolphin, Snowflake and Heart. Covers Rotate, Scale and Point UI events.

This was an idea for storing the current command state: could be combined with EmbView since you can't have more than one active command. If a command calls a sub command it will store the position in the parents.

18.6.4 Enumeration Type Documentation

18.6.4.1 OBJ_KEYS `enum OBJ_KEYS`

Custom Data used in QGraphicItems

(int, const QVariant)

I.E. object.setData(OBJ_TYPE, OBJ_TYPE_LINE); I.E. object.setData(OBJ_LAYER, "OUTLINE"); I.E. object.setData(OBJ_COLOR, 123); I.E. object.setData(OBJ_LTYPE, OBJ_LTYPE_CONT);
Keys

Enumerator

OBJ_TYPE	
OBJ_NAME	
OBJ_LAYER	
OBJ_COLOR	
OBJ_LTYPE	value type - int: 0-255 Todo Use color chart in formats/format-dxf.h for this
OBJ_LWT	
OBJ_RUBBER	

18.6.4.2 OBJ_LTYPE_VALUES enum [OBJ_LTYPE_VALUES](#)

Enumerator

OBJ_LTYPE_CONT
OBJ_LTYPE_CENTER
OBJ_LTYPE_DOT
OBJ_LTYPE_HIDDEN
OBJ_LTYPE_PHANTOM
OBJ_LTYPE_ZIGZAG
OBJ_LTYPE_RUNNING
OBJ_LTYPE_SATIN
OBJ_LTYPE_FISHBONE

18.6.4.3 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

18.6.4.4 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.
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	

18.6.4.5 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	

18.6.4.6 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

18.6.4.7 PREVIEW_CLONE_VALUES enum [PREVIEW_CLONE_VALUES](#)

Enumerator

PREVIEW_CLONE_NULL
PREVIEW_CLONE_SELECTED
PREVIEW_CLONE_RUBBER

18.6.4.8 PREVIEW_MODE_VALUES enum [PREVIEW_MODE_VALUES](#)

Enumerator

PREVIEW_MODE_NULL	
PREVIEW_MODE_MOVE	
PREVIEW_MODE_ROTATE	
PREVIEW_MODE_SCALE	

18.6.4.9 SPARE_RUBBER_VALUES enum [SPARE_RUBBER_VALUES](#)

Enumerator

SPARE_RUBBER_OFF	
SPARE_RUBBER_PATH	
SPARE_RUBBER_POLYGON	
SPARE_RUBBER_POLYLINE	

18.6.4.10 UiMode enum [UiMode](#)

Enumerator

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	

Enumerator

SNOWFLAKE_MODE_XSCALE	
SNOWFLAKE_MODE_YSCALE	

18.6.5 Function Documentation

18.6.5.1 debug_message() void debug_message (
 std::string msg)

18.6.5.2 degrees() EmbReal degrees (
 EmbReal radian) [inline]

18.6.5.3 get_action_index() int get_action_index (
 std::string cmd)

18.6.5.4 mainWin() MainWindow * mainWin ()
mainWin

Returns

18.6.5.5 operator+() EmbVector operator+ (
 EmbVector a,
 EmbVector b) [inline]

18.6.5.6 operator-() EmbVector operator- (
 EmbVector a,
 EmbVector b) [inline]

18.6.5.7 radians() EmbReal radians (
 EmbReal degrees) [inline]

18.6.5.8 read_settings() int read_settings (
 const char * file)

Read the settings from file which aren't editable by the user. These files need to be placed in the install folder.

18.6.5.9 to_EmbVector() EmbVector to_EmbVector (
 QPointF a) [inline]

18.6.5.10 to_QPointF() QPointF to_QPointF (
 EmbVector a) [inline]

```
18.6.5.11 write_settings() void write_settings (
    const char * fname )
```

18.6.6 Variable Documentation

```
18.6.6.1 action_table std::vector<Action> action_table [extern]
```

```
18.6.6.2 comboBoxes std::unordered_map<std::string, QComboBox *> comboBoxes [extern]
```

```
18.6.6.3 comboBoxTextSingleFont QFontComboBox* comboBoxTextSingleFont [extern]
```

```
18.6.6.4 config std::unordered_map<std::string, std::string> config [extern]
```

```
18.6.6.5 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.

```
18.6.6.6 edit_toolbar std::vector<std::string> edit_toolbar [extern]
```

```
18.6.6.7 emb_constant_pi const EmbReal emb_constant_pi = 3.14159265358979323846 [static]
```

```
18.6.6.8 file_toolbar std::vector<std::string> file_toolbar [extern]
```

```
18.6.6.9 group_box_arc_geometry GroupBoxData group_box_arc_geometry[] [extern]
```

```
18.6.6.10 group_box_arc_geometry_entries const int group_box_arc_geometry_entries [extern]
```

```
18.6.6.11 group_box_data std::unordered_map<std::string, GroupBoxData*> group_box_data [extern]
```

```
18.6.6.12 group_box_ellipse_geometry GroupBoxData group_box_ellipse_geometry[] [extern]
```

Todo use proper icons for toolButtons

```
18.6.6.13 group_box_ellipse_geometry_entries const int group_box_ellipse_geometry_entries [extern]
```

```
18.6.6.14 group_box_list std::vector<std::string> group_box_list [extern]
```

```
18.6.6.15 groupBoxes std::unordered_map<std::string, QGroupBox *> groupBoxes [extern]
```

18.6.6.16 lineEdits std::unordered_map<std::string, QLineEdit *> lineEdits [extern]

18.6.6.17 settings Settings settings [extern]

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/> The actuator changes the program state via these global variables.

18.6.6.18 toolButtons std::unordered_map<std::string, QToolButton *> toolButtons [extern]

18.6.6.19 view_toolbar std::vector<std::string> view_toolbar [extern]

18.6.6.20 zoom_toolbar std::vector<std::string> zoom_toolbar [extern]

18.7 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 /*
00025 * C/C++ Standard Libraries.
00026 */
00027 #include <cstdio>
00028 #include <cmath>
00029 #include <ctime>
00030 #include <cinttypes>
00031 #include <vector>
00032 #include <unordered_map>
00033 #include <string>
00034 #include <filesystem>
00035
00036 /*
00037 * Libraries included in "extern/".
00038 */
00039 #include "embroidery.h"
00040 #include "toml.h"
00041
00042 /*
00043 * Qt 6.0+ libraries.
00044 */
00045 #include <QAction>
00046 #include <QApplication>
00047 #include <QComboBox>
00048 #include <QContextMenuEvent>
00049 #include <QClipboard>
00050 #include <QDateTime>
00051 #include <QDebug>
00052 #include <QDialogButtonBox>
00053 #include <QFile>
00054 #include <QFrame>
00055 #include <QGraphicsScene>
00056 #include <QGraphicsPathItem>
00057 #include <QGridLayout>
00058 #include <QGroupBox>
00059 #include <QLabel>
00060 #include <QLineEdit>
00061 #include <QList>
```

```

00062 #include <QMainWindow>
00063 #include <QMdiArea>
00064 #include <QMenu>
00065 #include <QMetaObject>
00066 #include <QMessageBox>
00067 #include <QObject>
00068 #include <QPainter>
00069 #include <QSplitter>
00070 #include <QTextLayout>
00071 #include <QTextStream>
00072 #include <QTimer>
00073 #include <QToolBar>
00074 #include <QScrollArea>
00075 #include <QUndoStack>
00076 #include <QVBoxLayout>
00077
00078 #include <QtPrintSupport>
00079
00080 class ImageWidget;
00081 class MdiArea;
00082 class MdiWindow;
00083 class View;
00084 class StatusBar;
00085 class StatusBarButton;
00086 class CmdPrompt;
00087 class PropertyEditor;
00088 class UndoEditor;
00089 class MainWindow;
00090
00091 class BaseObject;
00092 class SelectBox;
00093 class ArcObject;
00094 class BlockObject;
00095 class CircleObject;
00096 class DimAlignedObject;
00097 class DimAngularObject;
00098 class DimArcLengthObject;
00099 class DimDiameterObject;
00100 class DimLeaderObject;
00101 class DimLinearObject;
00102 class DimOrdinateObject;
00103 class DimRadiusObject;
00104 class EllipseObject;
00105 class EllipseArcObject;
00106 class HatchObject;
00107 class ImageObject;
00108 class InfiniteLineObject;
00109 class LineObject;
00110 class PathObject;
00111 class PointObject;
00112 class PolygonObject;
00113 class PolylineObject;
00114 class RayObject;
00115 class RectObject;
00116 class SplineObject;
00117 class TextMultiObject;
00118 class TextSingleObject;
00119
00120 typedef struct UiObject_ {
00121     std::string fname;
00122     /*< \todo document this */
00123     std::string command;
00124     /*< \todo document this */
00125     bool firstRun;
00126     /*< If this UiObject has been put through the
00127      user interaction processor. */
00128     std::vector<EmbVector> controlPoints;
00129     /*< Storage for however many Rubber Points the
00130      design needs. */
00131     std::vector<std::string> controlPointLabels;
00132     /*< Storage for the labels for the Rubber Points
00133      using the same indexing. */
00134     int numPoints;
00135     /*< The number of points if we consider the object as a Polygon. */
00136     int minPoints;
00137     /*< The minimum number of points needed to make the
00138      polygon look somewhat like the design. */
00139     int maxPoints;
00140     /*< The maximum number of points before adding more will
00141      do nothing but slow down the program. */
00142     EmbVector center;
00143     /*< Where the polygon is centered. */
00144     EmbVector scale; /*< The scale of the object: note that the default
00145                      may not be (1.0, 1.0). */
00146     EmbReal rotation;
00147     /*< \todo document this */
00148     uint32_t mode;

```

```

00158     /*< The mode argument records what kind of design we are
00159         using and how to interact with it. */
00160     std::string path_desc;
00161     /*< The SVG style path spec. */
00162     std::string text;
00163     /*< The text to be rendered to the scene. */
00164     int textJustify;
00165     /*< One of the JUSTIFY_* constants representing what kind
00166         of alignment to use. */
00167     std::string textFont;
00168     /*< The file name of the font to use. */
00169     EmbReal textHeight;
00170     /*< The text height. */
00171     EmbReal textRotation;
00172     /*< The rotation of the text in the scene. */
00173     //GLuint texture_id;
00174     /*< Pointer to a texture that may be rendered to the object. */
00175     std::string id;
00176     /*< \todo document this */
00177     int pattern_index;
00178     /*< \todo document this */
00179     char type[200];
00180     /*< \todo document this */
00181     int object_index;
00182     /*< \todo document this */
00183     bool selectable;
00184     /*< \todo document this */
00185     EmbColor color;
00186     /*< \todo document this */
00187 } UiObject;
00188
00196 typedef struct EmbView_ {
00197     EmbPattern *pattern; /*< \todo document this */
00198     EmbVector origin; /*< \todo document this */
00199     EmbReal scale; /*< \todo document this */
00200     QString grid_type; /*< \todo document this */
00201     int ui_mode; /*< \todo document this */
00202     bool snap_mode; /*< \todo document this */
00203     bool grid_mode; /*< \todo document this */
00204     bool ruler_mode; /*< \todo document this */
00205     bool ortho_mode; /*< \todo document this */
00206     bool polar_mode; /*< \todo document this */
00207     bool qsnap_mode; /*< \todo document this */
00208     bool qtrack_mode; /*< \todo document this */
00209     bool lwt_mode; /*< \todo document this */
00210     bool real_render; /*< \todo document this */
00211     bool metric; /*< \todo document this */
00212     bool simulate; /*< \todo document this */
00213     clock_t simulation_start; /*< \todo document this */
00214     QString text_font; /*< \todo document this */
00215     EmbReal text_size; /*< \todo document this */
00216     EmbReal text_angle; /*< \todo document this */
00217     bool text_style_bold; /*< \todo document this */
00218     bool text_style_italic; /*< \todo document this */
00219     bool text_style_underline; /*< \todo document this */
00220     bool text_style_overline; /*< \todo document this */
00221     bool text_style_strikethrough; /*< \todo document this */
00222     QString filename; /*< \todo document this */
00223     QStringList undo_history; /*< \todo document this */
00224     int selected[100]; /*< \todo document this */
00225     int n_selected; /*< \todo document this */
00226     int rubber_mode; /*< . */
00227 } EmbView;
00228
00237 typedef struct Settings_ {
00238     QString general_language;
00239     /*< \todo document this */
00240     QString general_icon_theme;
00241     /*< \todo document this */
00242     int general_icon_size;
00243     /*< \todo document this */
00244     QString version; /*< \todo document this */
00245     bool running; /*< \todo document this */
00246     bool testing; /*< \todo document this */
00247     int debug_mode; /*< \todo document this */
00248     bool show_about_dialog; /*< \todo document this */
00249     bool show_settings_editor; /*< \todo document this */
00250     bool show_editor; /*< \todo document this */
00251     bool show_details_dialog; /*< \todo document this */
00252     bool show_open_file_dialog; /*< \todo document this */
00253     int pattern_index; /*< \todo document this */
00254     QString assets_dir; /*< \todo document this */
00255     bool use_translation; /*< \todo document this */
00256     bool general_mdi_bg_use_logo; /*< \todo document this */
00257     bool general_mdi_bg_use_texture; /*< \todo document this */
00258     bool general_mdi_bg_use_color; /*< \todo document this */
00259     QString general_mdi_bg_logo; /*< \todo document this */

```

```

00260 QString general_mdi_bg_texture; /*< \todo document this */
00261 QRgb general_mdi_bg_color; /*< \todo document this */
00262 bool general_tip_of_the_day; /*< \todo document this */
00263 uint32_t general_current_tip; /*< \todo document this */
00264 bool general_system_help_browser; /*< \todo document this */
00265 bool general_check_for_updates; /*< \todo document this */
00266 bool display_use_opengl; /*< \todo document this */
00267 bool display_renderhint_aa; /*< \todo document this */
00268 bool display_renderhint_text_aa; /*< \todo document this */
00269 bool display_renderhint_smooth_pix; /*< \todo document this */
00270 bool display_renderhint_high_aa; /*< \todo document this */
00271 bool display_renderhint_noncosmetic; /*< \todo document this */
00272 bool display_show_scrollbars; /*< \todo document this */
00273 int display_scrollbar_widget_num; /*< \todo document this */
00274 uint32_t display_crosshair_color; /*< \todo document this */
00275 uint32_t display_bg_color; /*< \todo document this */
00276 uint32_t display_selectbox_left_color; /*< \todo document this */
00277 uint32_t display_selectbox_left_fill; /*< \todo document this */
00278 uint32_t display_selectbox_right_color; /*< \todo document this */
00279 uint32_t display_selectbox_right_fill; /*< \todo document this */
00280 uint8_t display_selectbox_alpha; /*< \todo document this */
00281 EmbReal display_zoomscale_in; /*< \todo document this */
00282 EmbReal display_zoomscale_out; /*< \todo document this */
00283 uint8_t display_crosshair_percent; /*< \todo document this */
00284 QString display_units; /*< \todo document this */
00285 QString opensave_custom_filter; /*< \todo document this */
00286 QString opensave_open_format; /*< \todo document this */
00287 bool opensave_open_thumbnail; /*< \todo document this */
00288 QString opensave_save_format; /*< \todo document this */
00289 bool opensave_save_thumbnail; /*< \todo document this */
00290 uint8_t opensave_recent_max_files; /*< \todo document this */
00291 QStringList opensave_recent_list_of_files; /*< \todo document this */
00292 QString opensave_recent_directory; /*< \todo document this */
00293 uint8_t opensave_trim_dst_num_jumps; /*< \todo document this */
00294 QString printing_default_device; /*< \todo document this */
00295 bool printing_use_last_device; /*< \todo document this */
00296 bool printing_disable_bg; /*< \todo document this */
00297 bool grid_show_on_load; /*< \todo document this */
00298 bool grid_show_origin; /*< \todo document this */
00299 bool grid_color_match_crosshair; /*< \todo document this */
00300 uint32_t grid_color; /*< \todo document this */
00301 bool grid_load_from_file; /*< \todo document this */
00302 QString grid_type; /*< \todo document this */
00303 bool grid_center_on_origin; /*< \todo document this */
00304 EmbVector grid_center; /*< \todo document this */
00305 EmbVector grid_size; /*< \todo document this */
00306 EmbVector grid_spacing; /*< \todo document this */
00307 EmbReal grid_size_radius; /*< \todo document this */
00308 EmbReal grid_spacing_radius; /*< \todo document this */
00309 EmbReal grid_spacing_angle; /*< \todo document this */
00310 bool ruler_show_on_load; /*< \todo document this */
00311 bool ruler_metric; /*< \todo document this */
00312 uint32_t ruler_color; /*< \todo document this */
00313 uint8_t ruler_pixel_size; /*< \todo document this */
00314 bool qsnap_enabled; /*< \todo document this */
00315 uint32_t qsnap_locator_color; /*< \todo document this */
00316 uint8_t qsnap_locator_size; /*< \todo document this */
00317 uint8_t qsnap_aperture_size; /*< \todo document this */
00318 bool qsnap_endpoint; /*< \todo document this */
00319 bool qsnap_midpoint; /*< \todo document this */
00320 bool qsnap_center; /*< \todo document this */
00321 bool qsnap_node; /*< \todo document this */
00322 bool qsnap_quadrant; /*< \todo document this */
00323 bool qsnap_intersection; /*< \todo document this */
00324 bool qsnap_extension; /*< \todo document this */
00325 bool qsnap_insertion; /*< \todo document this */
00326 bool qsnap_perpendicular; /*< \todo document this */
00327 bool qsnap_tangent; /*< \todo document this */
00328 bool qsnap_nearest; /*< \todo document this */
00329 bool qsnap_apparent; /*< \todo document this */
00330 bool qsnap_parallel; /*< \todo document this */
00331 bool lwt_show_lwt; /*< \todo document this */
00332 bool lwt_real_render; /*< \todo document this */
00333 bool shift_hold; /*< \todo document this */
00334 EmbReal lwt_default_lwt; /*< \todo document this */
00335 bool selection_mode_pickfirst; /*< \todo document this */
00336 bool selection_mode_pickadd; /*< \todo document this */
00337 bool selection_mode_pickdrag; /*< \todo document this */
00338 uint32_t selection_coolgrip_color; /*< \todo document this */
00339 uint32_t selection_hotgrip_color; /*< \todo document this */
00340 uint8_t selection_grip_size; /*< \todo document this */
00341 uint8_t selection_pickbox_size; /*< \todo document this */
00342 QString text_font; /*< \todo document this */
00343 EmbReal text_size; /*< \todo document this */
00344 EmbReal text_angle; /*< \todo document this */
00345 bool text_style_bold; /*< \todo document this */
00346 bool text_style_italic; /*< \todo document this */

```

```

00347     bool text_style_underline; /*< \todo document this */
00348     bool text_style_overline; /*< \todo document this */
00349     bool text_style_strikethrough; /*< \todo document this */
00350     uint32_t ticks_color; /*< \todo document this */
00351     uint32_t shine_color; /*< \todo document this */
00352     QString to_open; /*< \todo document this */
00353     QString current_directory; /*< \todo document this */
00354     EmbReal zoomInLimit; /*< */
00355     EmbReal zoomOutLimit; /*< */
00356     EmbReal ruler_width; /*< */
00357     EmbReal tick_depth; /*< */
00358     EmbReal major_tick_separation;
00359     /*< \todo document this */
00360     EmbReal needle_speed;
00361     /*< \todo document this */
00362     EmbReal stitch_time;
00363     /*< \todo document this */
00364     QRgb prompt_text_color;
00365     /*< \todo document this */
00366     QRgb prompt_bg_color;
00367     /*< \todo document this */
00368     QString prompt_font_family;
00369     /*< \todo document this */
00370     QString prompt_font_style;
00371     /*< \todo document this */
00372     uint8_t prompt_font_size;
00373     /*< \todo document this */
00374     bool prompt_save_history;
00375     /*< \todo document this */
00376     bool prompt_save_history_as_html;
00377     /*< \todo document this */
00378     QString prompt_save_history_filename;
00379     /*< \todo document this */
00380 } Settings;
00381
00382 enum UiMode {
00383     DEFAULT_MODE,
00384
00385     CIRCLE_MODE_1P_RAD,
00386     CIRCLE_MODE_1P_DIA,
00387     CIRCLE_MODE_2P,
00388     CIRCLE_MODE_3P,
00389     CIRCLE_MODE_TTR,
00390
00391     ELLIPSE_MODE_MAJORDIAMETER_MINORRADIUS,
00392     ELLIPSE_MODE_MAJORRADIUS_MINORRADIUS,
00393     ELLIPSE_MODE_ELLIPSE_ROTATION,
00394
00395     DOLPHIN_MODE_NUM_POINTS,
00396     DOLPHIN_MODE_XSCALE,
00397     DOLPHIN_MODE_YSCALE,
00398
00399     HEART_MODE_NUM_POINTS,
00400     HEART_MODE_STYLE,
00401     HEART_MODE_XSCALE,
00402     HEART_MODE_YSCALE,
00403
00404     ROTATE_MODE_NORMAL,
00405     ROTATE_MODE_REFERENCE,
00406
00407     SCALE_MODE_NORMAL,
00408     SCALE_MODE_REFERENCE,
00409
00410     SINGLE_LINE_TEXT_MODE_JUSTIFY,
00411     SINGLE_LINE_TEXT_MODE_SETFONT,
00412     SINGLE_LINE_TEXT_MODE_SETGEOM,
00413     SINGLE_LINE_TEXT_MODE_RAPID,
00414
00415     STAR_MODE_NUM_POINTS,
00416     STAR_MODE_CENTER_PT,
00417     STAR_MODE_RAD_OUTER,
00418     STAR_MODE_RAD_INNER,
00419
00420     SNOWFLAKE_MODE_NUM_POINTS,
00421     SNOWFLAKE_MODE_XSCALE,
00422     SNOWFLAKE_MODE_YSCALE
00423 };
00424
00425 enum OBJ_KEYS {
00426     OBJ_TYPE = 0,
00427     /*< value type - int: See OBJ_TYPE_VALUES */
00428     OBJ_NAME = 1,
00429     /*< value type - str: See OBJ_NAME_VALUES */
00430     OBJ_LAYER = 2,
00431     /*< value type - str: "USER", "DEFINED", "STRINGS", etc... */
00432     OBJ_COLOR = 3,
00433     OBJ_LTYPE = 4,
00434
00435 }
```

```

00449     /*< value type - int: See OBJ_LTYPE_VALUES */
00450     OBJ_LWT = 5, //value type - int: 0-27
00451     OBJ_RUBBER = 6 //value type - int: See OBJ_RUBBER_VALUES
00452 };
00453
00454 //Values
00455 enum OBJ_TYPE_VALUES {
00456     OBJ_TYPE_NULL = 0,
00457     /*< NOTE: Allow this enum to evaluate false */
00458     OBJ_TYPE_BASE = 100000,
00459     /*< NOTE: Values >= 65536 ensure compatibility with qgraphicsitem_cast() */
00460     OBJ_TYPE_ARC = 100001,
00461     OBJ_TYPE_BLOCK = 100002,
00462     /*< For the block type, that has to exist for SVG. */
00463     OBJ_TYPE_CIRCLE = 100003,
00464     OBJ_TYPE_DIMALIGNED = 100004,
00465     /*< For the Aligned Dimension, that has to exist for DXF drawings. */
00466     OBJ_TYPE_DIMANGULAR = 100005,
00467     /*< For the Angular Dimension, that has to exist for DXF drawings. */
00468     OBJ_TYPE_DIMARCLENGTH = 100006,
00469     /*< For the Arc Length Dimension, that has to exist for DXF drawings. */
00470     OBJ_TYPE_DIMDIAMETER = 100007,
00471     OBJ_TYPE_DIMLEADER = 100008,
00472     OBJ_TYPE_DIMLINEAR = 100009,
00473     /*< For the Linear Dimension, that has to exist for DXF drawings. */
00474     OBJ_TYPE_DIMORDINATE = 100010,
00475     /*< For the Ordinate Dimension, that has to exist for DXF drawings. */
00476     OBJ_TYPE_DIMRADIUS = 100011,
00477     /*< For the Radial Dimension, that has to exist for DXF drawings. */
00478     OBJ_TYPE_ELLIPSE = 100012,
00479     OBJ_TYPE_ELLIPSEARC = 100013,
00480     OBJ_TYPE_RUBBER = 100014,
00481     OBJ_TYPE_GRID = 100015,
00482     OBJ_TYPE_HATCH = 100016,
00483     OBJ_TYPE_IMAGE = 100017,
00484     OBJ_TYPE_INFINITELINE = 100018,
00485     /*< For the Infinite Line object. Which should be removed from output as it exists
00486      for drafting reasons. */
00487     OBJ_TYPE_LINE = 100019,
00488     OBJ_TYPE_PATH = 100020,
00489     OBJ_TYPE_POINT = 100021,
00490     OBJ_TYPE_POLYGON = 100022,
00491     OBJ_TYPE_POLYLINE = 100023,
00492     OBJ_TYPE_RAY = 100024,
00493     /*< For the Ray object. */
00494     OBJ_TYPE_RECTANGLE = 100025,
00495     OBJ_TYPE_SLOT = 100026,
00496     OBJ_TYPE_SPLINE = 100027,
00497     OBJ_TYPE_TEXTMULTI = 100028,
00498     OBJ_TYPE_TEXTSINGLE = 100029
00499 };
00500
00501 enum OBJ_LTYPE_VALUES {
00502     //CAD Linetypes
00503     OBJ_LTYPE_CONT = 0,
00504     OBJ_LTYPE_CENTER = 1,
00505     OBJ_LTYPE_DOT = 2,
00506     OBJ_LTYPE_HIDDEN = 3,
00507     OBJ_LTYPE_PHANTOM = 4,
00508     OBJ_LTYPE_ZIGZAG = 5,
00509     //Embroidery Stitchtypes
00510     OBJ_LTYPE_RUNNING = 6, // _____
00511     OBJ_LTYPE_SATIN = 7, // vvvvvvvvvv
00512     OBJ_LTYPE_FISHBONE = 8, // >>>>
00513 };
00514
00515 enum OBJ_LWT_VALUES {
00516     OBJ_LWT_BYLAYER = -2,
00517     OBJ_LWT_BYBLOCK = -1,
00518     OBJ_LWT_DEFAULT = 0,
00519     OBJ_LWT_01 = 1,
00520     OBJ_LWT_02 = 2,
00521     OBJ_LWT_03 = 3,
00522     OBJ_LWT_04 = 4,
00523     OBJ_LWT_05 = 5,
00524     OBJ_LWT_06 = 6,
00525     OBJ_LWT_07 = 7,
00526     OBJ_LWT_08 = 8,
00527     OBJ_LWT_09 = 9,
00528     OBJ_LWT_10 = 10,
00529     OBJ_LWT_11 = 11,
00530     OBJ_LWT_12 = 12,
00531     OBJ_LWT_13 = 13,
00532     OBJ_LWT_14 = 14,
00533     OBJ_LWT_15 = 15,
00534     OBJ_LWT_16 = 16,
00535     OBJ_LWT_17 = 17,

```

```
00536     OBJ_LWT_18 = 18,
00537     OBJ_LWT_19 = 19,
00538     OBJ_LWT_20 = 20,
00539     OBJ_LWT_21 = 21,
00540     OBJ_LWT_22 = 22,
00541     OBJ_LWT_23 = 23,
00542     OBJ_LWT_24 = 24
00543 };
00544
00545 enum OBJ_SNAP_VALUES {
00546     OBJ_SNAP_NULL = 0, //NOTE: Allow this enum to evaluate false
00547     OBJ_SNAP_ENDPOINT = 1,
00548     OBJ_SNAP_MIDPOINT = 2,
00549     OBJ_SNAP_CENTER = 3,
00550     OBJ_SNAP_NODE = 4,
00551     OBJ_SNAP_QUADRANT = 5,
00552     OBJ_SNAP_INTERSECTION = 6,
00553     OBJ_SNAP_EXTENSION = 7,
00554     OBJ_SNAP_INSERTION = 8,
00555     OBJ_SNAP_PERPENDICULAR = 9,
00556     OBJ_SNAP_TANGENT = 10,
00557     OBJ_SNAP_NEAREST = 11,
00558     OBJ_SNAP_APPINTERSECTION = 12,
00559     OBJ_SNAP_PARALLEL = 13
00560 };
00561
00562 enum OBJ_RUBBER_VALUES {
00563     OBJ_RUBBER_OFF = 0, //NOTE: Allow this enum to evaluate false
00564     OBJ_RUBBER_ON = 1, //NOTE: Allow this enum to evaluate true
00565
00566     OBJ_RUBBER_CIRCLE_1P_RAD,
00570     OBJ_RUBBER_CIRCLE_1P_DIA,
00575     OBJ_RUBBER_CIRCLE_2P,
00576     OBJ_RUBBER_CIRCLE_3P,
00577     OBJ_RUBBER_CIRCLE_TTR,
00578     OBJ_RUBBER_CIRCLE_TTT,
00579
00580     OBJ_RUBBER_DIMLEADER_LINE,
00581
00582     OBJ_RUBBER_ELLIPSE_LINE,
00583     OBJ_RUBBER_ELLIPSE_MAJORDIAMETER_MINORRADIUS,
00584     OBJ_RUBBER_ELLIPSE_MAJORRADIUS_MINORRADIUS,
00585     OBJ_RUBBER_ELLIPSE_ROTATION,
00586
00587     OBJ_RUBBER_GRIP,
00588
00589     OBJ_RUBBER_LINE,
00590
00591     OBJ_RUBBER_POLYGON,
00592     OBJ_RUBBER_POLYGON_INSCRIBE,
00593     OBJ_RUBBER_POLYGON_CIRCUMSCRIBE,
00594
00595     OBJ_RUBBER_POLYLINE,
00596
00597     OBJ_RUBBER_IMAGE,
00598
00599     OBJ_RUBBER_RECTANGLE,
00600
00601     OBJ_RUBBER_TEXTSINGLE
00602 };
00603
00604 enum SPARE_RUBBER_VALUES {
00605     SPARE_RUBBER_OFF = 0, //NOTE: Allow this enum to evaluate false
00606     SPARE_RUBBER_PATH,
00607     SPARE_RUBBER_POLYGON,
00608     SPARE_RUBBER_POLYLINE
00609 };
00610
00611 enum PREVIEW_CLONE_VALUES {
00612     PREVIEW_CLONE_NULL = 0, //NOTE: Allow this enum to evaluate false
00613     PREVIEW_CLONE_SELECTED,
00614     PREVIEW_CLONE_RUBBER
00615 };
00616
00617 enum PREVIEW_MODE_VALUES {
00618     PREVIEW_MODE_NULL = 0, //NOTE: Allow this enum to evaluate false
00619     PREVIEW_MODE_MOVE,
00620     PREVIEW_MODE_ROTATE,
00621     PREVIEW_MODE_SCALE
00622 };
00623
00624 static const EmbReal emb_constant_pi = 3.14159265358979323846;
00625
00626 /*
00627     * \brief Convert \a a to a QPointF.
00628 */
00629 inline QPointF
```

```

00630     to_QPointF(EmbVector a)
00631 {
00632     QPointF result(a.x(), a.y());
00633     return result;
00634 }
00635
00636 /**
00637 * \brief Convert \a a to an EmbVector.
00638 */
00639 inline EmbVector
00640 to_EmbVector(QPointF a)
00641 {
00642     EmbVector v;
00643     v.x = a.x();
00644     v.y = a.y();
00645     return v;
00646 }
00647
00648 /**
00649 * \brief Wrapper for embVector_add to use the syntax \a a + \a b.
00650 */
00651 inline EmbVector
00652 operator+(EmbVector a, EmbVector b)
00653 {
00654     return embVector_add(a, b);
00655 }
00656
00657 /**
00658 * \brief Wrapper for embVector_subtract to use the syntax \a a - \a b.
00659 */
00660 inline EmbVector
00661 operator-(EmbVector a, EmbVector b)
00662 {
00663     return embVector_subtract(a, b);
00664 }
00665
00666 inline EmbReal
00667 radians(EmbReal degrees)
00668 {
00669     return (degrees*emb_constant_pi/180.0);
00670 }
00671
00672 inline EmbReal
00673 degrees(EmbReal radian)
00674 {
00675     return (radian*180.0/emb_constant_pi);
00676 }
00677
00678 class BaseObject : public QGraphicsPathItem
00679 {
00680     public:
00681         BaseObject(QGraphicsItem* parent = 0);
00682         virtual ~BaseObject();
00683
00684     enum { Type = OBJ_TYPE_BASE };
00685     virtual int type() const { return Type; }
00686
00687     QPen objPen;
00688     QPen lwtPen;
00689     QLineF objLine;
00690     int objRubberMode;
00691     QHash<QString, QPointF> objRubberPoints;
00692     QHash<QString, QString> objRubberTexts;
00693     qint64 objID;
00694
00695     qint64 objectID() const { return objID; }
00696     QPen objectPen() const { return objPen; }
00697     QColor objectColor() const { return objPen.color(); }
00698     QRgb objectColorRGB() const { return objPen.color().rgb(); }
00699     Qt::PenStyle objectLineType() const { return objPen.style(); }
00700     EmbReal objectLineWeight() const { return lwtPen.widthF(); }
00701     QPainterPath objectPath() const { return path(); }
00702     int objectRubberMode() const { return objRubberMode; }
00703     QPointF objectRubberPoint(const QString& key) const;
00704     QString objectRubberText(const QString& key) const;
00705
00706     QPointF objectCenter() const { return scenePos(); }
00707     EmbReal objectCenterX() const { return scenePos().x(); }
00708     EmbReal objectCenterY() const { return scenePos().y(); }
00709
00710     void setObjectCenter(EmbVector center)
00711     {
00712         setPos(center.x, center.y);
00713     }
00714     void setObjectCenterX(EmbReal centerX) { setX(centerX); }
00715     void setObjectCenterY(EmbReal centerY) { setY(centerY); }
00716
00717
00718
00719
00720
00721
00722
00723
00724
00725

```

```

00726     QRectF rect() const { return path().boundingRect(); }
00727     void setRect(const QRectF& r) { QPainterPath p; p.addRect(r); setPath(p); }
00728     void setRect(EmbReal x, EmbReal y, EmbReal w, EmbReal h) { QPainterPath p; p.addRect(x,y,w,h);
00729         setPath(p); }
00730     QLineF line() const { return objLine; }
00731     void setLine(const QLineF& li) { QPainterPath p; p.moveTo(li.p1()); p.lineTo(li.p2()); setPath(p);
00732         objLine = li; }
00733     void setLine(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2) { QPainterPath p; p.moveTo(x1,y1);
00734         p.lineTo(x2,y2); setPath(p); objLine.setLine(x1,y1,x2,y2); }
00735     void setObjectColor(const QColor& color);
00736     void setObjectColorRGB(QRgb rgb);
00737     void setObjectLineType(Qt::PenStyle lineType);
00738     void setObjectLineWeight(EmbReal lineWeight);
00739     void setObjectPath(const QPainterPath& p) { setPath(p); }
00740     void setObjectRubberMode(int mode) { objRubberMode = mode; }
00741     void setObjectRubberPoint(const QString& key, const QPointF& point) { objRubberPoints.insert(key,
00742         point); }
00743     void setObjectRubberText(const QString& key, const QString& txt) { objRubberTexts.insert(key,
00744         txt); }
00745     virtual QRectF boundingRect() const;
00746     virtual QPainterPath shape() const { return path(); }
00747     void drawRubberLine(const QLineF& rubLine, QPainter* painter = 0, const char* colorFromScene = 0);
00748     virtual void vulcanize() = 0;
00749     virtual QPointF mouseSnapPoint(const QPointF& mousePoint) = 0;
00750     virtual QList<QPointF> allGripPoints() = 0;
00751     virtual void gripEdit(const QPointF& before, const QPointF& after) = 0;
00752     protected:
00753     QPen lineWeightPen() const { return lwtPen; }
00754     void realRender(QPainter* painter, const QPainterPath& renderPath);
00755
00756
00760 class ArcObject : public BaseObject
00761 {
00762     public:
00763     QPointF arcStartPoint;
00764     QPointF arcMidPoint;
00765     QPointF arcEndPoint;
00766
00767     ArcObject(EmbArc arc, QRgb rgb, QGraphicsItem* parent = 0);
00768     ArcObject(EmbReal startX, EmbReal startY, EmbReal midX, EmbReal midY, EmbReal endX, EmbReal endY,
00769     QRgb rgb, QGraphicsItem* parent = 0);
00770     ArcObject(ArcObject* obj, QGraphicsItem* parent = 0);
00771     ~ArcObject();
00772
00773     enum { Type = OBJ_TYPE_ARC };
00774     virtual int type() const { return Type; }
00775
00776     void init(EmbReal startX, EmbReal startY, EmbReal midX, EmbReal midY, EmbReal endX, EmbReal endY,
00777     QRgb rgb, Qt::PenStyle lineType);
00778     void updatePath();
00779
00780     void calculateArcData(EmbReal startX, EmbReal startY, EmbReal midX, EmbReal midY, EmbReal endX,
00781     EmbReal endY);
00782     void updateArcRect(EmbReal radius);
00783
00784     EmbReal objectRadius() const { return rect().width()/2.0*scale(); }
00785     EmbReal objectStartAngle() const;
00786     EmbReal objectEndAngle() const;
00787     QPointF objectStartPoint() const;
00788     EmbReal objectStartX() const;
00789     EmbReal objectStartY() const;
00790     QPointF objectMidPoint() const;
00791     EmbReal objectMidX() const;
00792     EmbReal objectMidY() const;
00793     QPointF objectEndPoint() const;
00794     EmbReal objectEndX() const;
00795     EmbReal objectEndY() const;
00796     EmbReal objectArea() const;
00797     EmbReal objectArcLength() const;
00798     EmbReal objectChord() const;
00799     EmbReal objectIncludedAngle() const;
00800     bool objectClockwise() const;
00801
00802     void setObjectRadius(EmbReal radius);
00803     void setObjectStartAngle(EmbReal angle);
00804     void setObjectEndAngle(EmbReal angle);
00805     void setObjectStartPoint(const QPointF& point);
00806     void setObjectStartPoint(EmbReal pointX, EmbReal pointY);
00807     void setObjectMidPoint(const QPointF& point);
00808     void setObjectMidPoint(EmbReal pointX, EmbReal pointY);
00809     void setObjectEndPoint(const QPointF& point);
00810     void setObjectEndPoint(EmbReal pointX, EmbReal pointY);

```

```

00808
00809     void updateRubber(QPainter* painter = 0);
00810     virtual void vulcanize();
00811     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
00812     virtual QList<QPointF> allGripPoints();
00813     virtual void gripEdit(const QPointF& before, const QPointF& after);
00814 protected:
00815     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
00816 };
00817
00818
00819 class CircleObject : public BaseObject
00820 {
00821 public:
00822     CircleObject(EmbReal centerX, EmbReal centerY, EmbReal radius, QRgb rgb, QGraphicsItem* parent = 0);
00823     CircleObject(CircleObject* obj, QGraphicsItem* parent = 0);
00824     ~CircleObject();
00825
00826     void init(EmbReal centerX, EmbReal centerY, EmbReal radius, QRgb rgb, Qt::PenStyle lineType);
00827     void updatePath();
00828
00829     enum { Type = OBJ_TYPE_CIRCLE };
00830     virtual int type() const { return Type; }
00831
00832     QPainterPath objectSavePath() const;
00833
00834     EmbReal objectRadius() const { return rect().width()/2.0*scale(); }
00835     EmbReal objectDiameter() const { return rect().width()*scale(); }
00836     EmbReal objectArea() const { return emb_constant_pi*objectRadius()*objectRadius(); }
00837     EmbReal objectCircumference() const { return emb_constant_pi*objectDiameter(); }
00838     QPointF objectQuadrant0() const { return objectCenter() + QPointF(objectRadius(), 0); }
00839     QPointF objectQuadrant90() const { return objectCenter() + QPointF(0,-objectRadius()); }
00840     QPointF objectQuadrant180() const { return objectCenter() + QPointF(-objectRadius(),0); }
00841     QPointF objectQuadrant270() const { return objectCenter() + QPointF(0, objectRadius()); }
00842
00843     void setObjectRadius(EmbReal radius);
00844     void setObjectDiameter(EmbReal diameter);
00845     void setObjectArea(EmbReal area);
00846     void setObjectCircumference(EmbReal circumference);
00847
00848     void updateRubber(QPainter* painter = 0);
00849     virtual void vulcanize();
00850     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
00851     virtual QList<QPointF> allGripPoints();
00852     virtual void gripEdit(const QPointF& before, const QPointF& after);
00853 protected:
00854     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
00855 };
00856
00857
00858 class DimLeaderObject : public BaseObject
00859 {
00860 public:
00861     DimLeaderObject(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, QRgb rgb, QGraphicsItem* parent = 0);
00862     DimLeaderObject(DimLeaderObject* obj, QGraphicsItem* parent = 0);
00863     ~DimLeaderObject();
00864
00865     void init(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, QRgb rgb, Qt::PenStyle lineType);
00866
00867     bool curved;
00868     bool filled;
00869     void updateLeader();
00870     QPainterPath lineStylePath;
00871     QPainterPath arrowStylePath;
00872     EmbReal arrowStyleAngle;
00873     EmbReal arrowStyleLength;
00874     EmbReal lineStyleAngle;
00875     EmbReal lineStyleLength;
00876
00877     enum ArrowStyle {
00878         NoArrow, //NOTE: Allow this enum to evaluate false
00879         Open,
00880         Closed,
00881         Dot,
00882         Box,
00883         Tick
00884     };
00885
00886     enum lineStyle {
00887         NoLine, //NOTE: Allow this enum to evaluate false
00888         Flared,
00889         Fletching
00890     };
00891
00892     enum { Type = OBJ_TYPE_DIMLEADER };
00893     virtual int type() const { return Type; }
00894
00895

```

```

00896
00897     QPointF objectEndPoint1() const;
00898     QPointF objectEndPoint2() const;
00899     QPointF objectMidPoint() const;
00900     EmbReal objectX1() const { return objectEndPoint1().x(); }
00901     EmbReal objectY1() const { return objectEndPoint1().y(); }
00902     EmbReal objectX2() const { return objectEndPoint2().x(); }
00903     EmbReal objectY2() const { return objectEndPoint2().y(); }
00904     EmbReal objectDeltaX() const { return (objectX2() - objectX1()); }
00905     EmbReal objectDeltaY() const { return (objectY2() - objectY1()); }
00906     EmbReal objectAngle() const;
00907     EmbReal objectLength() const { return line().length(); }
00908
00909     void setObjectEndPoint1(const QPointF& endPt1);
00910     void setObjectEndPoint1(EmbReal x1, EmbReal y1);
00911     void setObjectEndPoint2(const QPointF& endPt2);
00912     void setObjectEndPoint2(EmbReal x2, EmbReal y2);
00913     void setObjectX1(EmbReal x) { setObjectEndPoint1(x, objectY1()); }
00914     void setObjectY1(EmbReal y) { setObjectEndPoint1(objectX1(), y); }
00915     void setObjectX2(EmbReal x) { setObjectEndPoint2(x, objectY2()); }
00916     void setObjectY2(EmbReal y) { setObjectEndPoint2(objectX2(), y); }
00917
00918     void updateRubber(QPainter* painter = 0);
00919     virtual void vulcanize();
00920     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
00921     virtual QList<QPointF> allGripPoints();
00922     virtual void gripEdit(const QPointF& before, const QPointF& after);
00923 protected:
00924     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*) override;
00925 };
00926
00927
00928 class EllipseObject : public BaseObject
00929 {
00930 public:
00931     EllipseObject(EmbReal centerX, EmbReal centerY, EmbReal width, EmbReal height, QRgb rgb,
00932                   QGraphicsItem* parent = 0);
00933     EllipseObject(EllipseObject* obj, QGraphicsItem* parent = 0);
00934     ~EllipseObject();
00935
00936     void init(EmbReal centerX, EmbReal centerY, EmbReal width, EmbReal height, QRgb rgb, Qt::PenStyle
00937               lineType);
00938     void updatePath();
00939
00940     enum { Type = OBJ_TYPE_ELLIPSE };
00941     virtual int type() const { return Type; }
00942
00943     QPainterPath objectSavePath() const;
00944
00945     EmbReal objectRadiusMajor() const { return qMax(rect().width(), rect().height())/2.0*scale(); }
00946     EmbReal objectRadiusMinor() const { return qMin(rect().width(), rect().height())/2.0*scale(); }
00947     EmbReal objectDiameterMajor() const { return qMax(rect().width(), rect().height())*scale(); }
00948     EmbReal objectDiameterMinor() const { return qMin(rect().width(), rect().height())*scale(); }
00949     EmbReal objectWidth() const { return rect().width()*scale(); }
00950     EmbReal objectHeight() const { return rect().height()*scale(); }
00951     QPointF objectQuadrant0() const;
00952     QPointF objectQuadrant90() const;
00953     QPointF objectQuadrant180() const;
00954     QPointF objectQuadrant270() const;
00955
00956     void setObjectSize(EmbReal width, EmbReal height);
00957     void setObjectRadiusMajor(EmbReal radius);
00958     void setObjectRadiusMinor(EmbReal radius);
00959     void setObjectDiameterMajor(EmbReal diameter);
00960     void setObjectDiameterMinor(EmbReal diameter);
00961
00962     void updateRubber(QPainter* painter = 0);
00963     virtual void vulcanize();
00964     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
00965     virtual QList<QPointF> allGripPoints();
00966     virtual void gripEdit(const QPointF& before, const QPointF& after);
00967 protected:
00968     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*) override;
00969 };
00970
00971
00972 class ImageObject : public BaseObject
00973 {
00974 public:
00975     ImageObject(EmbReal x, EmbReal y, EmbReal w, EmbReal h, QRgb rgb, QGraphicsItem* parent = 0);
00976     ImageObject(ImageObject* obj, QGraphicsItem* parent = 0);
00977     ~ImageObject();
00978
00979     void init(EmbReal x, EmbReal y, EmbReal w, EmbReal h, QRgb rgb, Qt::PenStyle lineType);
00980     void updatePath();
00981
00982     enum { Type = OBJ_TYPE_IMAGE };
00983
00984
00985
00986
00987
00988
00989
00990
00991
00992
00993
00994
00995
00996
00997
00998
00999
01000
01001
01002
01003
01004
01005
01006
01007
01008
01009
01010
01011
01012
01013
01014
01015
01016
01017
01018
01019
01020
01021
01022
01023
01024
01025
01026
01027
01028
01029
01030
01031
01032
01033
01034
01035
01036
01037
01038
01039
01040
01041
01042
01043
01044
01045
01046
01047
01048
01049
01050
01051
01052
01053
01054
01055
01056
01057
01058
01059
01060
01061
01062
01063
01064
01065
01066
01067
01068
01069
01070
01071
01072
01073
01074
01075
01076
01077
01078
01079
01080
01081
01082
01083
01084
01085
01086
01087
01088
01089
01090
01091
01092
01093
01094
01095
01096
01097
01098
01099
01100
01101
01102
01103
01104
01105
01106
01107
01108
01109
01110
01111
01112
01113
01114
01115
01116
01117
01118
01119
01120
01121
01122
01123
01124
01125
01126
01127
01128
01129
01130
01131
01132
01133
01134
01135
01136
01137
01138
01139
01140
01141
01142
01143
01144
01145
01146
01147
01148
01149
01150
01151
01152
01153
01154
01155
01156
01157
01158
01159
01160
01161
01162
01163
01164
01165
01166
01167
01168
01169
01170
01171
01172
01173
01174
01175
01176
01177
01178
01179
01180
01181
01182
01183
01184
01185
01186
01187
01188
01189
01190
01191
01192
01193
01194
01195
01196
01197
01198
01199
01200
01201
01202
01203
01204
01205
01206
01207
01208
01209
01210
01211
01212
01213
01214
01215
01216
01217
01218
01219
01220
01221
01222
01223
01224
01225
01226
01227
01228
01229
01230
01231
01232
01233
01234
01235
01236
01237
01238
01239
01240
01241
01242
01243
01244
01245
01246
01247
01248
01249
01250
01251
01252
01253
01254
01255
01256
01257
01258
01259
01260
01261
01262
01263
01264
01265
01266
01267
01268
01269
01270
01271
01272
01273
01274
01275
01276
01277
01278
01279
01280
01281
01282
01283
01284
01285
01286
01287
01288
01289
01290
01291
01292
01293
01294
01295
01296
01297
01298
01299
01300
01301
01302
01303
01304
01305
01306
01307
01308
01309
01310
01311
01312
01313
01314
01315
01316
01317
01318
01319
01320
01321
01322
01323
01324
01325
01326
01327
01328
01329
01330
01331
01332
01333
01334
01335
01336
01337
01338
01339
01340
01341
01342
01343
01344
01345
01346
01347
01348
01349
01350
01351
01352
01353
01354
01355
01356
01357
01358
01359
01360
01361
01362
01363
01364
01365
01366
01367
01368
01369
01370
01371
01372
01373
01374
01375
01376
01377
01378
01379
01380
01381
01382
01383
01384
01385
01386
01387
01388
01389
01390
01391
01392
01393
01394
01395
01396
01397
01398
01399
01400
01401
01402
01403
01404
01405
01406
01407
01408
01409
01410
01411
01412
01413
01414
01415
01416
01417
01418
01419
01420
01421
01422
01423
01424
01425
01426
01427
01428
01429
01430
01431
01432
01433
01434
01435
01436
01437
01438
01439
01440
01441
01442
01443
01444
01445
01446
01447
01448
01449
01450
01451
01452
01453
01454
01455
01456
01457
01458
01459
01460
01461
01462
01463
01464
01465
01466
01467
01468
01469
01470
01471
01472
01473
01474
01475
01476
01477
01478
01479
01480
01481
01482
01483
01484
01485
01486
01487
01488
01489
01490
01491
01492
01493
01494
01495
01496
01497
01498
01499
01500
01501
01502
01503
01504
01505
01506
01507
01508
01509
01510
01511
01512
01513
01514
01515
01516
01517
01518
01519
01520
01521
01522
01523
01524
01525
01526
01527
01528
01529
01530
01531
01532
01533
01534
01535
01536
01537
01538
01539
01540
01541
01542
01543
01544
01545
01546
01547
01548
01549
01550
01551
01552
01553
01554
01555
01556
01557
01558
01559
01560
01561
01562
01563
01564
01565
01566
01567
01568
01569
01570
01571
01572
01573
01574
01575
01576
01577
01578
01579
01580
01581
01582
01583
01584
01585
01586
01587
01588
01589
01590
01591
01592
01593
01594
01595
01596
01597
01598
01599
01600
01601
01602
01603
01604
01605
01606
01607
01608
01609
01610
01611
01612
01613
01614
01615
01616
01617
01618
01619
01620
01621
01622
01623
01624
01625
01626
01627
01628
01629
01630
01631
01632
01633
01634
01635
01636
01637
01638
01639
01640
01641
01642
01643
01644
01645
01646
01647
01648
01649
01650
01651
01652
01653
01654
01655
01656
01657
01658
01659
01660
01661
01662
01663
01664
01665
01666
01667
01668
01669
01670
01671
01672
01673
01674
01675
01676
01677
01678
01679
01680
01681
01682
01683
01684
01685
01686
01687
01688
01689
01690
01691
01692
01693
01694
01695
01696
01697
01698
01699
01700
01701
01702
01703
01704
01705
01706
01707
01708
01709
01710
01711
01712
01713
01714
01715
01716
01717
01718
01719
01720
01721
01722
01723
01724
01725
01726
01727
01728
01729
01730
01731
01732
01733
01734
01735
01736
01737
01738
01739
01740
01741
01742
01743
01744
01745
01746
01747
01748
01749
01750
01751
01752
01753
01754
01755
01756
01757
01758
01759
01760
01761
01762
01763
01764
01765
01766
01767
01768
01769
01770
01771
01772
01773
01774
01775
01776
01777
01778
01779
01780
01781
01782
01783
01784
01785
01786
01787
01788
01789
01790
01791
01792
01793
01794
01795
01796
01797
01798
01799
01800
01801
01802
01803
01804
01805
01806
01807
01808
01809
01810
01811
01812
01813
01814
01815
01816
01817
01818
01819
01820
01821
01822
01823
01824
01825
01826
01827
01828
01829
01830
01831
01832
01833
01834
01835
01836
01837
01838
01839
01840
01841
01842
01843
01844
01845
01846
01847
01848
01849
01850
01851
01852
01853
01854
01855
01856
01857
01858
01859
01860
01861
01862
01863
01864
01865
01866
01867
01868
01869
01870
01871
01872
01873
01874
01875
01876
01877
01878
01879
01880
01881
01882
01883
01884
01885
01886
01887
01888
01889
01890
01891
01892
01893
01894
01895
01896
01897
01898
01899
01900
01901
01902
01903
01904
01905
01906
01907
01908
01909
01910
01911
01912
01913
01914
01915
01916
01917
01918
01919
01920
01921
01922
01923
01924
01925
01926
01927
01928
01929
01930
01931
01932
01933
01934
01935
01936
01937
01938
01939
01940
01941
01942
01943
01944
01945
01946
01947
01948
01949
01950
01951
01952
01953
01954
01955
01956
01957
01958
01959
01960
01961
01962
01963
01964
01965
01966
01967
01968
01969
01970
01971
01972
01973
01974
01975
01976
01977
01978
01979
01980
01981
01982
01983
01984
01985
01986
01987
01988
01989
01990
01991
01992
01993
01994
01995
01996
01997
01998
01999
01999
02000
02001
02002
02003
02004
02005
02006
02007
02008
02009
02009
02010
02011
02012
02013
02014
02015
02016
02017
02018
02019
02019
02020
02021
02022
02023
02024
02025
02026
02027
02028
02029
02029
02030
02031
02032
02033
02034
02035
02036
02037
02038
02039
02039
02040
02041
02042
02043
02044
02045
02046
02047
02048
02049
02049
02050
02051
02052
02053
02054
02055
02056
02057
02058
02059
02059
02060
02061
02062
02063
02064
02065
02066
02067
02068
02069
02069
02070
02071
02072
02073
02074
02075
02076
02077
02078
02079
02079
02080
02081
02082
02083
02084
02085
02086
02087
02088
02089
02089
02090
02091
02092
02093
02094
02095
02096
02097
02098
02099
02099
02100
02101
02102
02103
02104
02105
02106
02107
02108
02109
02109
02110
02111
02112
02113
02114
02115
02116
02117
02118
02119
02119
02120
02121
02122
02123
02124
02125
02126
02127
02128
02129
02129
02130
02131
02132
02133
02134
02135
02136
02137
02138
02139
02139
02140
02141
02142
02143
02144
02145
02146
02147
02148
02149
02149
02150
02151
02152
02153
02154
02155
02156
02157
02158
02159
02159
02160
02161
02162
02163
02164
02165
02166
02167
02168
02169
02169
02170
02171
02172
02173
02174
02175
02176
02177
02178
02179
02179
02180
02181
02182
02183
02184
02185
02186
02187
02188
02189
02189
02190
02191
02192
02193
02194
02195
02196
02197
02198
02199
02199
02200
02201
02202
02203
02204
02205
02206
02207
02208
02209
02209
02210
02211
02212
02213
02214
02215
02216
02217
02218
02219
02219
02220
02221
02222
02223
02224
02225
02226
02227
02228
02229
02229
02230
02231
02232
02233
02234
02235
02236
02237
02238
02239
02239
02240
02241
02242
02243
02244
02245
02246
02247
02248
02249
02249
02250
02251
02252
02253
02254
02255
02256
02257
02258
02259
02259
02260
02261
02262
02263
02264
02265
02266
02267
02268
02269
02269
02270
02271
02272
02273
02274
02275
02276
02277
02278
02279
02279
02280
02281
02282
02283
02284
02285
02286
02287
02288
02289
02289
02290
02291
02292
02293
02294
02295
02296
02297
02298
02299
02299
02300
02301
02302
02303
02304
02305
02306
02307
02308
02309
02309
02310
02311
02312
02313
02314
02315
02316
02317
02318
02319
02319
02320
02321
02322
02323
02324
02325
02326
02327
02328
02329
02329
02330
02331
02332
02333
02334
02335
02336
02337
02338
02339
02339
02340
02341
02342
02343
02344
02345
02346
02347
02348
02349
02349
02350
02351
02352
02353
02354
02355
02356
02357
02358
02359
02359
02360
02361
02362
02363
02364
02365
02366
02367
02368
02369
02369
02370
02371
02372
02373
02374
02375
02376
02377
02378
02379
02379
02380
02381
02382
02383
02384
02385
02386
02387
02388
02389
02389
02390
02391
02392
02393
02394
02395
02396
02397
02398
02399
02399
02400
02401
02402
02403
02404
02405
02406
02407
02408
02409
02409
02410
02411
02412
02413
02414
02415
02416
02417
02418
02419
02419
02420
02421
02422
02423
02424
02425
02426
02427
02428
02429
02429
02430
02431
02432
02433
02434
02435
02436
02437
02438
02439
02439
02440
02441
02442
02443
02444
02445
02446
02447
02448
02449
02449
02450
02451
02452
02453
02454
02455
02456
02457
02458
02459
02459
02460
02461
02462
02463
02464
02465
02466
02467
02468
02469
02469
02470
02471
02472
02473
02474
02475
02476
02477
02478
02479
02479
02480
02481
02482
02483
02484
02485
02486
02487
02488
02489
02489
02490
02491
02492
02493
02494
02495
02496
02497
02498
02499
02499
02500
02501
02502
02503
02504
02505
02506
02507
02508
02509
02509
02510
02511
02512
02513
02514
02515
02516
02517
02518
02519
02519
02520
02521
02522
02523
02524
02525
02526
02527
02528
02529
02529
02530
02531
02532
02533
02534
02535
02536
02537
02538
0253
```

```

00987     virtual int type() const { return Type; }
00988
00989     QPointF objectTopLeft() const;
00990     QPointF objectTopRight() const;
00991     QPointF objectBottomLeft() const;
00992     QPointF objectBottomRight() const;
00993     EmbReal objectWidth() const { return rect().width()*scale(); }
00994     EmbReal objectHeight() const { return rect().height()*scale(); }
00995     EmbReal objectArea() const { return qAbs(objectWidth()*objectHeight()); }
00996
00997     void setObjectRect(EmbReal x, EmbReal y, EmbReal w, EmbReal h);
00998
00999     void updateRubber(QPainter* painter = 0);
01000     virtual void vulcanize();
01001     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
01002     virtual QList<QPointF> allGripPoints();
01003     virtual void gripEdit(const QPointF& before, const QPointF& after);
01004 protected:
01005     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*) override;
01006 };
01007
01008
01012 class LineObject : public BaseObject
01013 {
01014 public:
01015     LineObject(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, QRgb rgb, QGraphicsItem* parent = 0);
01016     LineObject(LineObject* obj, QGraphicsItem* parent = 0);
01017     ~LineObject();
01018
01019     void init(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, QRgb rgb, Qt::PenStyle lineType);
01020
01021     enum { Type = OBJ_TYPE_LINE };
01022     virtual int type() const { return Type; }
01023
01024     QPainterPath objectSavePath() const;
01025
01026     QPointF objectEndPoint1() const { return scenePos(); }
01027     QPointF objectEndPoint2() const;
01028     QPointF objectMidPoint() const;
01029     EmbReal objectX1() const { return objectEndPoint1().x(); }
01030     EmbReal objectY1() const { return objectEndPoint1().y(); }
01031     EmbReal objectX2() const { return objectEndPoint2().x(); }
01032     EmbReal objectY2() const { return objectEndPoint2().y(); }
01033     EmbReal objectDeltaX() const { return (objectX2() - objectX1()); }
01034     EmbReal objectDeltaY() const { return (objectY2() - objectY1()); }
01035     EmbReal objectAngle() const;
01036     EmbReal objectLength() const { return line().length()*scale(); }
01037
01038     void setObjectEndPoint1(const QPointF& endPt1);
01039     void setObjectEndPoint1(EmbReal x1, EmbReal y1);
01040     void setObjectEndPoint2(const QPointF& endPt2);
01041     void setObjectEndPoint2(EmbReal x2, EmbReal y2);
01042     void setObjectX1(EmbReal x) { setObjectEndPoint1(x, objectY1()); }
01043     void setObjectY1(EmbReal y) { setObjectEndPoint1(objectX1(), y); }
01044     void setObjectX2(EmbReal x) { setObjectEndPoint2(x, objectY2()); }
01045     void setObjectY2(EmbReal y) { setObjectEndPoint2(objectX2(), y); }
01046
01047     void updateRubber(QPainter* painter = 0);
01048     virtual void vulcanize();
01049     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
01050     virtual QList<QPointF> allGripPoints();
01051     virtual void gripEdit(const QPointF& before, const QPointF& after);
01052 protected:
01053     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*) override;
01054 };
01055
01056
01060 class PathObject : public BaseObject
01061 {
01062 public:
01063     PathObject(EmbReal x, EmbReal y, const QPainterPath p, QRgb rgb, QGraphicsItem* parent = 0);
01064     PathObject(PathObject* obj, QGraphicsItem* parent = 0);
01065     ~PathObject();
01066
01067     enum { Type = OBJ_TYPE_PATH };
01068     virtual int type() const { return Type; }
01069
01070     void init(EmbReal x, EmbReal y, const QPainterPath& p, QRgb rgb, Qt::PenStyle lineType);
01071     void updatePath(const QPainterPath& p);
01072     QPainterPath normalPath;
01073     //TODO: make paths similar to polylines. Review and implement any missing functions/members.
01074
01075     QPainterPath objectCopyPath() const;
01076     QPainterPath objectSavePath() const;
01077
01078     QPointF objectPos() const { return scenePos(); }
01079     EmbReal objectX() const { return scenePos().x(); }

```

```

01080     EmbReal objectY() const { return scenePos().y(); }
01081
01082     void setObjectPos(const QPointF& point) { setPos(point.x(), point.y()); }
01083     void setObjectPos(EmbReal x, EmbReal y) { setPos(x, y); }
01084     void setObjectX(EmbReal x) { setObjectPos(x, objectY()); }
01085     void setObjectY(EmbReal y) { setObjectPos(objectX(), y); }
01086
01087     void updateRubber(QPainter* painter = 0);
01088     virtual void vulcanize();
01089     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
01090     virtual QList<QPointF> allGripPoints();
01091     virtual void gripEdit(const QPointF& before, const QPointF& after);
01092 protected:
01093     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
01094 };
01095
01096
01100 class PointObject : public BaseObject
01101 {
01102 public:
01103     PointObject(EmbReal x, EmbReal y, QRgb rgb, QGraphicsItem* parent = 0);
01104     PointObject(PointObject* obj, QGraphicsItem* parent = 0);
01105     ~PointObject();
01106
01107     void init(EmbReal x, EmbReal y, QRgb rgb, Qt::PenStyle lineType);
01108
01109     enum { Type = OBJ_TYPE_POINT };
01110     virtual int type() const { return Type; }
01111
01112     QPainterPath objectSavePath() const;
01113
01114     QPointF objectPos() const { return scenePos(); }
01115     EmbReal objectX() const { return scenePos().x(); }
01116     EmbReal objectY() const { return scenePos().y(); }
01117
01118     void setObjectPos(const QPointF& point) { setPos(point.x(), point.y()); }
01119     void setObjectPos(EmbReal x, EmbReal y) { setPos(x, y); }
01120     void setObjectX(EmbReal x) { setObjectPos(x, objectY()); }
01121     void setObjectY(EmbReal y) { setObjectPos(objectX(), y); }
01122
01123     void updateRubber(QPainter* painter = 0);
01124     virtual void vulcanize();
01125     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
01126     virtual QList<QPointF> allGripPoints();
01127     virtual void gripEdit(const QPointF& before, const QPointF& after);
01128 protected:
01129     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
01130 };
01131
01132
01136 class PolygonObject : public BaseObject
01137 {
01138 public:
01139     PolygonObject(EmbReal x, EmbReal y, const QPainterPath& p, QRgb rgb, QGraphicsItem* parent = 0);
01140     PolygonObject(PolygonObject* obj, QGraphicsItem* parent = 0);
01141     ~PolygonObject();
01142
01143     enum { Type = OBJ_TYPE_POLYGON };
01144     virtual int type() const { return Type; }
01145
01146     void init(EmbReal x, EmbReal y, const QPainterPath& p, QRgb rgb, Qt::PenStyle lineType);
01147     void updatePath(const QPainterPath& p);
01148     QPainterPath normalPath;
01149     int findIndex(const QPointF& point);
01150     int gripIndex;
01151
01152     QPainterPath objectCopyPath() const;
01153     QPainterPath objectSavePath() const;
01154
01155     QPointF objectPos() const { return scenePos(); }
01156     EmbReal objectX() const { return scenePos().x(); }
01157     EmbReal objectY() const { return scenePos().y(); }
01158
01159     void setObjectPos(const QPointF& point) { setPos(point.x(), point.y()); }
01160     void setObjectPos(EmbReal x, EmbReal y) { setPos(x, y); }
01161     void setObjectX(EmbReal x) { setObjectPos(x, objectY()); }
01162     void setObjectY(EmbReal y) { setObjectPos(objectX(), y); }
01163
01164     void updateRubber(QPainter* painter = 0);
01165     virtual void vulcanize();
01166     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
01167     virtual QList<QPointF> allGripPoints();
01168     virtual void gripEdit(const QPointF& before, const QPointF& after);
01169 protected:
01170     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
01171 };
01172

```

```

01176 class PolylineObject : public BaseObject
01177 {
01178     public:
01179         PolylineObject(EmbReal x, EmbReal y, const QPainterPath& p, QRgb rgb, QGraphicsItem* parent = 0);
01180         PolylineObject(PolylineObject* obj, QGraphicsItem* parent = 0);
01181         ~PolylineObject();
01182
01183     enum { Type = OBJ_TYPE_POLYLINE };
01184     virtual int type() const { return Type; }
01185
01186     void init(EmbReal x, EmbReal y, const QPainterPath& p, QRgb rgb, Qt::PenStyle lineType);
01187     void updatePath(const QPainterPath& p);
01188     QPainterPath normalPath;
01189     int findIndex(const QPointF& point);
01190     int gripIndex;
01191
01192     QPainterPath objectCopyPath() const;
01193     QPainterPath objectSavePath() const;
01194
01195     QPointF objectPos() const { return scenePos(); }
01196     EmbReal objectX() const { return scenePos().x(); }
01197     EmbReal objectY() const { return scenePos().y(); }
01198
01199     void setObjectPos(const QPointF& point) { setPos(point.x(), point.y()); }
01200     void setObjectPos(EmbReal x, EmbReal y) { setPos(x, y); }
01201     void setObjectX(EmbReal x) { setObjectPos(x, objectY()); }
01202     void setObjectY(EmbReal y) { setObjectPos(objectX(), y); }
01203
01204     void updateRubber(QPainter* painter = 0);
01205     virtual void vulcanize();
01206     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
01207     virtual QList<QPointF> allGripPoints();
01208     virtual void gripEdit(const QPointF& before, const QPointF& after);
01209 protected:
01210     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
01211 };
01212
01213 class RectObject : public BaseObject
01214 {
01215     public:
01216         RectObject(EmbReal x, EmbReal y, EmbReal w, EmbReal h, QRgb rgb, QGraphicsItem* parent = 0);
01217         RectObject(RectObject* obj, QGraphicsItem* parent = 0);
01218         ~RectObject();
01219
01220     enum { Type = OBJ_TYPE_RECTANGLE };
01221     virtual int type() const { return Type; }
01222
01223     QPainterPath objectSavePath() const;
01224
01225     void init(EmbReal x, EmbReal y, EmbReal w, EmbReal h, QRgb rgb, Qt::PenStyle lineType);
01226     void updatePath();
01227
01228     QPointF objectPos() const { return scenePos(); }
01229
01230     QPointF objectTopLeft() const;
01231     QPointF objectTopRight() const;
01232     QPointF objectBottomLeft() const;
01233     QPointF objectBottomRight() const;
01234     EmbReal objectWidth() const { return rect().width()*scale(); }
01235     EmbReal objectHeight() const { return rect().height()*scale(); }
01236     EmbReal objectArea() const { return qAbs(objectWidth()*objectHeight()); }
01237
01238     void setObjectRect(EmbReal x, EmbReal y, EmbReal w, EmbReal h);
01239
01240     void updateRubber(QPainter* painter = 0);
01241     virtual void vulcanize();
01242     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
01243     virtual QList<QPointF> allGripPoints();
01244     virtual void gripEdit(const QPointF& before, const QPointF& after);
01245 protected:
01246     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
01247 };
01248
01249 class SaveObject : public QObject
01250 {
01251     Q_OBJECT
01252
01253     public:
01254         SaveObject(QGraphicsScene* theScene, QObject* parent = 0);
01255         ~SaveObject();
01256
01257     bool save(const QString &fileName);
01258
01259     void addArc          (EmbPattern* pattern, QGraphicsItem* item);
01260     void addBlock        (EmbPattern* pattern, QGraphicsItem* item);
01261     void addCircle       (EmbPattern* pattern, QGraphicsItem* item);
01262     void addDimAligned   (EmbPattern* pattern, QGraphicsItem* item);
01263
01264
01265
01266
01267
01268
01269
01270
01271
01272
01273
01274
01275
01276
01277
01278
01279
01280
01281
01282
01283
01284
01285
01286
01287
01288
01289
01290
01291
01292
01293
01294
01295
01296
01297
01298
01299
01300
01301
01302
01303
01304
01305
01306
01307
01308
01309
01310
01311
01312
01313
01314
01315
01316
01317
01318
01319
01320
01321
01322
01323
01324
01325
01326
01327
01328
01329
01330
01331
01332
01333
01334
01335
01336
01337
01338
01339
01340
01341
01342
01343
01344
01345
01346
01347
01348
01349
01350
01351
01352
01353
01354
01355
01356
01357
01358
01359
01360
01361
01362
01363
01364
01365
01366
01367
01368
01369
01370
01371
01372
01373
01374
01375
01376
01377
01378
01379
01380
01381
01382
01383
01384
01385
01386
01387
01388
01389
01390
01391
01392
01393
01394
01395
01396
01397
01398
01399
01400
01401
01402
01403
01404
01405
01406
01407
01408
01409
01410
01411
01412
01413
01414
01415
01416
01417
01418
01419
01420
01421
01422
01423
01424
01425
01426
01427
01428
01429
01430
01431
01432
01433
01434
01435
01436
01437
01438
01439
01440
01441
01442
01443
01444
01445
01446
01447
01448
01449
01450
01451
01452
01453
01454
01455
01456
01457
01458
01459
01460
01461
01462
01463
01464
01465
01466
01467
01468
01469
01470
01471
01472
01473
01474
01475
01476
01477
01478
01479
01480
01481
01482
01483
01484
01485
01486
01487
01488
01489
01490
01491
01492
01493
01494
01495
01496
01497
01498
01499
01500
01501
01502
01503
01504
01505
01506
01507
01508
01509
01510
01511
01512
01513
01514
01515
01516
01517
01518
01519
01520
01521
01522
01523
01524
01525
01526
01527
01528
01529
01530
01531
01532
01533
01534
01535
01536
01537
01538
01539
01540
01541
01542
01543
01544
01545
01546
01547
01548
01549
01550
01551
01552
01553
01554
01555
01556
01557
01558
01559
01560
01561
01562
01563
01564
01565
01566
01567
01568
01569
01570
01571
01572
01573
01574
01575
01576
01577
01578
01579
01580
01581
01582
01583
01584
01585
01586
01587
01588
01589
01590
01591
01592
01593
01594
01595
01596
01597
01598
01599
01600
01601
01602
01603
01604
01605
01606
01607
01608
01609
01610
01611
01612
01613
01614
01615
01616
01617
01618
01619
01620
01621
01622
01623
01624
01625
01626
01627
01628
01629
01630
01631
01632
01633
01634
01635
01636
01637
01638
01639
01640
01641
01642
01643
01644
01645
01646
01647
01648
01649
01650
01651
01652
01653
01654
01655
01656
01657
01658
01659
01660
01661
01662
01663
01664
01665
01666
01667
01668
01669
01670
01671
01672
01673
01674
01675
01676
01677
01678
01679
01680
01681
01682
01683
01684
01685
01686
01687
01688
01689
01690
01691
01692
01693
01694
01695
01696
01697
01698
01699
01700
01701
01702
01703
01704
01705
01706
01707
01708
01709
01710
01711
01712
01713
01714
01715
01716
01717
01718
01719
01720
01721
01722
01723
01724
01725
01726
01727
01728
01729
01730
01731
01732
01733
01734
01735
01736
01737
01738
01739
01740
01741
01742
01743
01744
01745
01746
01747
01748
01749
01750
01751
01752
01753
01754
01755
01756
01757
01758
01759
01760
01761
01762
01763
01764
01765
01766
01767
01768
01769
01770
01771
01772
01773
01774
01775
01776
01777
01778
01779
01780
01781
01782
01783
01784
01785
01786
01787
01788
01789
01790
01791
01792
01793
01794
01795
01796
01797
01798
01799
01800
01801
01802
01803
01804
01805
01806
01807
01808
01809
01810
01811
01812
01813
01814
01815
01816
01817
01818
01819
01820
01821
01822
01823
01824
01825
01826
01827
01828
01829
01830
01831
01832
01833
01834
01835
01836
01837
01838
01839
01840
01841
01842
01843
01844
01845
01846
01847
01848
01849
01850
01851
01852
01853
01854
01855
01856
01857
01858
01859
01860
01861
01862
01863
01864
01865
01866
01867
01868
01869
01870
01871
01872
01873
01874
01875
01876
01877
01878
01879
01880
01881
01882
01883
01884
01885
01886
01887
01888
01889
01890
01891
01892
01893
01894
01895
01896
01897
01898
01899
01900
01901
01902
01903
01904
01905
01906
01907
01908
01909
01910
01911
01912
01913
01914
01915
01916
01917
01918
01919
01920
01921
01922
01923
01924
01925
01926
01927
01928
01929
01930
01931
01932
01933
01934
01935
01936
01937
01938
01939
01940
01941
01942
01943
01944
01945
01946
01947
01948
01949
01950
01951
01952
01953
01954
01955
01956
01957
01958
01959
01960
01961
01962
01963
01964
01965
01966
01967
01968
01969
01970
01971
01972
01973
01974
01975
01976
01977
01978
01979
01980
01981
01982
01983
01984
01985
01986
01987
01988
01989
01990
01991
01992
01993
01994
01995
01996
01997
01998
01999
02000
02001
02002
02003
02004
02005
02006
02007
02008
02009
02010
02011
02012
02013
02014
02015
02016
02017
02018
02019
02020
02021
02022
02023
02024
02025
02026
02027
02028
02029
02030
02031
02032
02033
02034
02035
02036
02037
02038
02039
02040
02041
02042
02043
02044
02045
02046
02047
02048
02049
02050
02051
02052
02053
02054
02055
02056
02057
02058
02059
02060
02061
02062
02063
02064
02065
02066
02067
02068
02069
02070
02071
02072
02073
02074
02075
02076
02077
02078
02079
02080
02081
02082
02083
02084
02085
02086
02087
02088
02089
02090
02091
02092
02093
02094
02095
02096
02097
02098
02099
02100
02101
02102
02103
02104
02105
02106
02107
02108
02109
02110
02111
02112
02113
02114
02115
02116
02117
02118
02119
02120
02121
02122
02123
02124
02125
02126
02127
02128
02129
02130
02131
02132
02133
02134
02135
02136
02137
02138
02139
02140
02141
02142
02143
02144
02145
02146
02147
02148
02149
02150
02151
02152
02153
02154
02155
02156
02157
02158
02159
02160
02161
02162
02163
02164
02165
02166
02167
02168
02169
02170
02171
02172
02173
02174
02175
02176
02177
02178
02179
02180
02181
02182
02183
02184
02185
02186
02187
02188
02189
02190
02191
02192
02193
02194
02195
02196
02197
02198
02199
02200
02201
02202
02203
02204
02205
02206
02207
02208
02209
02210
02211
02212
02213
02214
02215
02216
02217
02218
02219
02220
02221
02222
02223
02224
02225
02226
02227
02228
02229
02230
02231
02232
02233
02234
02235
02236
02237
02238
02239
02240
02241
02242
02243
02244
02245
02246
02247
02248
02249
02250
02251
02252
02253
02254
02255
02256
02257
02258
02259
02260
02261
02262
02263
02264
02265
02266
02267
02268
02269
02270
02271
02272
02273
02274
02275
02276
02277
02278
02279
02280
02281
02282
02283
02284
02285
02286
02287
02288
02289
02290
02291
02292
02293
02294
02295
02296
02297
02298
02299
02300
02301
02302
02303
02304
02305
02306
02307
02308
02309
02310
02311
02312
02313
02314
02315
02316
02317
02318
02319
02320
02321
02322
02323
02324
02325
02326
02327
02328
02329
02330
02331
02332
02333
02334
02335
02336
02337
02338
02339
02340
02341
02342
02343
02344
02345
02346
02347
02348
02349
02350
02351
02352
02353
02354
02355
02356
02357
02358
02359
02360
02361
02362
02363
02364
02365
02366
02367
02368
02369
02370
02371
02372
02373
02374
02375
02376
02377
02378
02379
02380
02381
02382
02383
02384
02385
02386
02387
02388
02389
02390
02391
02392
02393
02394
02395
02396
02397
02398
02399
02400
02401
02402
02403
02404
02405
02406
02407
02408
02409
02410
02411
02412
02413
02414
02415
02416
02417
02418
02419
02420
02421
02422
02423
02424
02425
02426
02427
02428
02429
02430
02431
02432
02433
02434
02435
02436
02437
02438
02439
02440
02441
02442
02443
02444
02445
02446
02447
02448
02449
02450
02451
02452
02453
02454
02455
02456
02457
02458
02459
02460
02461
02462
02463
02464
02465
02466
02467
02468
02469
02470
02471
02472
02473
02474
02475
02476
02477
02478
02479
02480
02481
02482
02483
02484
02485
02486
02487
02488
02489
02490
02491
02492
02493
02494
02495
02496
02497
02498
02499
02500
02501
02502
02503
02504
02505
02506
02507
02508
02509
02510
02511
02512
02513
02514
02515
02516
02517
02518
02519
02520
02521
02522
02523
02524
02525
02526
02527
02528
02529
02530
02531
02532
02533
02534
02535
02536
02537
02538
02539
02540
02541
02542
02543
02544
02545
02546
02547
02548
02549
02550
02551
02552
02553
02554
02555
02556
02557
02558
02559
02560
02561
02562
02563
02564
02565
02566
02567
02568
02569
02570
02571
02572
02573
02574
02575
02576
02577
02578
02579
02580
02581
02582
02583
02584
02585
02586
02587
02588
02589
02590
02591
02592
02593
02594
02595
02596
02597
02598
02599
02600
02601
02602
02603
02604
02605
02606
02607
02608
02609
02610
02611
02612
02613
02614
02615
02616
02617
02618
02619
02620
02621
02622
02623
02624
02625
02626
02627
02628
02629
02630
02631
02632
02633
02634
02635
02636
02637
02638
02639
02640
02641
02642
02643
02644
02645
02646
02647
02648
02649
02650
02651
02652
02653
02654
02655
02656
02657
02658
02659
02660
02661
02662
02663
02664
02665
02666
02667
02668
02669
02670
02671
02672
02673
02674
02675
02676
02677
02678
02679
02680
02681
02682
02683
02684
02685
02686
02687
02688
02689
02690
02691
02692
02693
02694
02695
02696
02697
02698
02699
02700
02701
02702
02703
02704
02705
02706
02707
02708
02709
02710
02711
02712
02713
02714
02715
02716
02717
02718
02719
02720
02721
02722
02723
02724
02725
02726
02727
02728
02729
02730
02731
02732
02733
02734
02735
02736
02737
02738
02739
02740
02741
02742
02743
02744
02745
02746
02747
02748
02749
02750
02751
02752
02753
02754
02755
02756
02757
02758
02759
02760
02761
02762
02763
02764
02765
02766
02767
02768
02769
02770
02771
02772
02773
02774
02775
02776
02777
02778
02779
02780
02781
02782
02783
02784
02785
02786
02787
02788
02789
02790
02791
02792
02793
02794
02795
02796
02797
02798
02799
02800
02801
02802
02803
02804
02805
02806
02807
02808
02809
02810
02811
02812
02813
02814
02815
02816
02817
02818
02819
02820
02821
02822
02823
02824
02825
02826
02827
02828
02829
02830
02831
02832
02833
02834
02835
02836
02837
02838
02839
02840
02841
02842
02843
02844
02845
02846
02847
02848
02849
02850
02851
02852
02853
02854
02855
02856
02857
02858
02859
02860
02861
02862
02863
02864
02865
02866
02867
02868
02869
02870
02871
02872
02873
02874
02875
02876
02877
02878
02879
02880
02881
02882
02883
02884
02885
02886
02887
02888
02889
02890
02891
02892
02893
02894
02895
02896
02897
02898
02899
02900
02901
02902
02903
02904
02905
```

```

01269     void addDimAngular    (EmbPattern* pattern, QGraphicsItem* item);
01270     void addDimArcLength (EmbPattern* pattern, QGraphicsItem* item);
01271     void addDimDiameter  (EmbPattern* pattern, QGraphicsItem* item);
01272     void addDimLeader    (EmbPattern* pattern, QGraphicsItem* item);
01273     void addDimLinear    (EmbPattern* pattern, QGraphicsItem* item);
01274     void addDimOrdinate  (EmbPattern* pattern, QGraphicsItem* item);
01275     void addDimRadius    (EmbPattern* pattern, QGraphicsItem* item);
01276     void addEllipse       (EmbPattern* pattern, QGraphicsItem* item);
01277     void addEllipseArc   (EmbPattern* pattern, QGraphicsItem* item);
01278     void addGrid          (EmbPattern* pattern, QGraphicsItem* item);
01279     void addHatch         (EmbPattern* pattern, QGraphicsItem* item);
01280     void addImage         (EmbPattern* pattern, QGraphicsItem* item);
01281     void addInfiniteLine (EmbPattern* pattern, QGraphicsItem* item);
01282     void addLine          (EmbPattern* pattern, QGraphicsItem* item);
01283     void addPath          (EmbPattern* pattern, QGraphicsItem* item);
01284     void addPoint         (EmbPattern* pattern, QGraphicsItem* item);
01285     void addPolygon       (EmbPattern* pattern, QGraphicsItem* item);
01286     void addPolyline      (EmbPattern* pattern, QGraphicsItem* item);
01287     void addRay           (EmbPattern* pattern, QGraphicsItem* item);
01288     void addRectangle     (EmbPattern* pattern, QGraphicsItem* item);
01289     void addSlot          (EmbPattern* pattern, QGraphicsItem* item);
01290     void addSpline         (EmbPattern* pattern, QGraphicsItem* item);
01291     void addTextMulti    (EmbPattern* pattern, QGraphicsItem* item);
01292     void addTextSingle   (EmbPattern* pattern, QGraphicsItem* item);
01293
01294     QGraphicsScene* gscene;
01295     int formatType;
01296
01297     void toPolyline(EmbPattern* pattern, const QPointF& objPos, const QPainterPath& objPath, const
01298                      QString& layer, const QColor& color, const QString& lineType, const QString& lineWeight);
01299 };
01300
01301 class TextSingleObject : public BaseObject
01302 {
01303 public:
01304     TextSingleObject(const QString& str, EmbReal x, EmbReal y, QRgb rgb, QGraphicsItem* parent = 0);
01305     TextSingleObject(TextSingleObject* obj, QGraphicsItem* parent = 0);
01306     ~TextSingleObject();
01307
01308     enum { Type = OBJ_TYPE_TEXTSINGLE };
01309     virtual int type() const { return Type; }
01310
01311     void init(const QString& str, EmbReal x, EmbReal y, QRgb rgb, Qt::PenStyle lineType);
01312
01313     QString objText;
01314     QString objTextFont;
01315     QString objTextJustify;
01316     EmbReal objTextSize;
01317     bool objTextBold;
01318     bool objTextItalic;
01319     bool objTextUnderline;
01320     bool objTextStrikeOut;
01321     bool objTextOverline;
01322     bool objTextBackward;
01323     bool objTextUpsideDown;
01324     QPainterPath objTextPath;
01325
01326     QList<QPainterPath> objectSavePathList() const { return subPathList(); }
01327     QList<QPainterPath> subPathList() const;
01328
01329     QPointF objectPos() const { return scenePos(); }
01330     EmbReal objectX() const { return scenePos().x(); }
01331     EmbReal objectY() const { return scenePos().y(); }
01332
01333     QStringList objectTextJustifyList() const;
01334
01335     void setObjectText(const QString& str);
01336     void setObjectTextFont(const QString& font);
01337     void setObjectTextJustify(const QString& justify);
01338     void setObjectTextSize(EmbReal size);
01339     void setObjectTextStyle(bool bold, bool italic, bool under, bool strike, bool over);
01340     void setObjectTextBold(bool val);
01341     void setObjectTextItalic(bool val);
01342     void setObjectTextUnderline(bool val);
01343     void setObjectTextStrikeOut(bool val);
01344     void setObjectTextOverline(bool val);
01345     void setObjectTextBackward(bool val);
01346     void setObjectTextUpsideDown(bool val);
01347
01348     void setObjectPos(const QPointF& point) { setPos(point.x(), point.y()); }
01349     void setObjectPos(EmbReal x, EmbReal y) { setPos(x, y); }
01350     void setObjectX(EmbReal x) { setObjectPos(x, objectY()); }
01351     void setObjectY(EmbReal y) { setObjectPos(objectX(), y); }
01352
01353     void updateRubber(QPainter* painter = 0);
01354     virtual void vulcanize();
01355     virtual QPointF mouseSnapPoint(const QPointF& mousePoint);
01356     virtual QList<QPointF> allGripPoints();
01357

```

```

01358     virtual void gripEdit(const QPointF& before, const QPointF& after);
01359 protected:
01360     void paint(QPainter*, const QStyleOptionGraphicsItem*, QWidget*);
01361 };
01362
01363 class Application : public QApplication
01364 {
01365     Q_OBJECT
01366 public:
01367     Application(int argc, char **argv);
01368     void setMainWin(MainWindow* mainWin) { _mainWin = mainWin; }
01369     MainWindow* _mainWin;
01370 protected:
01371     virtual bool event(QEvent *e);
01372 };
01373
01374 class CmdPromptInput : public QLineEdit
01375 {
01376     Q_OBJECT
01377 public:
01378     CmdPromptInput(QWidget* parent = 0);
01379     ~CmdPromptInput();
01380
01381     QString curText;
01382     QString defaultPrefix;
01383     QString prefix;
01384
01385     QString lastCmd;
01386     QString curCmd;
01387     bool cmdActive;
01388
01389     bool rapidFireEnabled;
01390     bool isBlinking;
01391
01392     QHash<QString, QString>* aliasHash;
01393
01394     void changeFormatting(const QList<QTextLayout::FormatRange>& formats);
01395     void clearFormatting();
01396     void applyFormatting();
01397
01398 protected:
01399     void contextMenuEvent(QContextMenuEvent *event);
01400     bool eventFilter(QObject *obj, QEvent *event);
01401
01402 signals:
01403     void appendHistory(const QString& txt, int prefixLength);
01404
01405 //These connect to the CmdPrompt signals
01406 void startCommand(const QString& cmd);
01407 void runCommand(const QString& cmd, const QString& cmdtxt);
01408 void deletePressed();
01409 void tabPressed();
01410 void escapePressed();
01411 void upPressed();
01412 void downPressed();
01413 void F1Pressed();
01414 void F2Pressed();
01415 void F3Pressed();
01416 void F4Pressed();
01417 void F5Pressed();
01418 void F6Pressed();
01419 void F7Pressed();
01420 void F8Pressed();
01421 void F9Pressed();
01422 void F10Pressed();
01423 void F11Pressed();
01424 void F12Pressed();
01425 void cutPressed();
01426 void copyPressed();
01427 void pastePressed();
01428 void selectAllPressed();
01429 void undoPressed();
01430 void redoPressed();
01431
01432 void shiftPressed();
01433 void shiftReleased();
01434
01435 void showSettings();
01436
01437 void stopBlinking();
01438
01439 public slots:
01440     void addCommand(const QString& alias, const QString& cmd);
01441     void endCommand();
01442     void processInput(void);

```

```
01453     void checkSelection();
01454     void updateCurrentText(const QString& txt);
01455     void checkEditedText(const QString& txt);
01456     void checkChangedText(const QString& txt);
01457     void checkCursorPosition(int oldpos, int newpos);
01458 private slots:
01459     void copyClip();
01460     void pasteClip();
01461 };
01462
01463 class CmdPromptHistory : public QTextBrowser
01464 {
01465     Q_OBJECT
01466 public:
01467     CmdPromptHistory(QWidget* parent = 0);
01468     ~CmdPromptHistory();
01469
01470     int tmpHeight;
01471     QString applyFormatting(const QString& txt, int prefixLength);
01472
01473 protected:
01474     void contextMenuEvent(QContextMenuEvent* event);
01475
01476 public slots:
01477     void appendHistory(const QString& txt, int prefixLength);
01478     void startResizeHistory(int y);
01479     void stopResizeHistory(int y);
01480     void resizeHistory(int y);
01481
01482 signals:
01483     void historyAppended(const QString& txt);
01484 };
01485
01486 class CmdPromptSplitter : public QSplitter
01487 {
01488     Q_OBJECT
01489 public:
01490     CmdPromptSplitter(QWidget* parent = 0);
01491     ~CmdPromptSplitter();
01492
01493 protected:
01494     QSplitterHandle* createHandle();
01495
01496 signals:
01497     void pressResizeHistory(int y);
01498     void releaseResizeHistory(int y);
01499     void moveResizeHistory(int y);
01500 };
01501
01502 class CmdPromptHandle : public QSplitterHandle
01503 {
01504     Q_OBJECT
01505 public:
01506     CmdPromptHandle(Qt::Orientation orientation, QSplitter* parent);
01507     ~CmdPromptHandle();
01508
01509 signals:
01510     void handlePressed(int y);
01511     void handleReleased(int y);
01512     void handleMoved(int y);
01513 };
01514
01515 class CmdPrompt : public QWidget
01516 {
01517     Q_OBJECT
01518 public:
01519     CmdPrompt(QWidget* parent = 0);
01520     ~CmdPrompt();
01521
01522     CmdPromptInput* promptInput;
01523     CmdPromptHistory* promptHistory;
01524     QVBoxLayout* promptVBoxLayout;
01525     QFrame* promptDivider;
```

```

01552     CmdPromptSplitter* promptSplitter;
01553
01554     QHash<QString, QString>* styleHash;
01555     void updateStyle();
01556     QTimer* blinkTimer;
01557     bool blinkState;
01558
01559 protected:
01560
01561 public slots:
01562     QString getHistory() { return promptHistory->toHtml(); }
01563     QString getPrefix() { return promptInput->prefix; }
01564     QString getCurrentText() { return promptInput->curText; }
01565     void setCurrentText(const QString& txt) { promptInput->curText = promptInput->prefix + txt;
01566         promptInput->setText(promptInput->curText); }
01567     void setHistory(const QString& txt) { promptHistory->setHtml(txt);
01568         promptHistory->moveCursor(QTextCursor::End, QTextCursor::MoveAnchor); }
01569     void setPrefix(const QString& txt);
01570     void appendHistory(const QString& txt);
01571     void startResizingTheHistory(int y) { promptHistory->startResizeHistory(y); }
01572     void stopResizingTheHistory(int y) { promptHistory->stopResizeHistory(y); }
01573     void resizeTheHistory(int y) { promptHistory->resizeHistory(y); }
01574     void addCommand(const QString& alias, const QString& cmd) { promptInput->addCommand(alias, cmd); }
01575     void endCommand() { promptInput->endCommand(); }
01576     bool isCommandActive() { return promptInput->cmdActive; }
01577     QString activeCommand() { return promptInput->curCmd; }
01578     QString lastCommand() { return promptInput->lastCmd; }
01579     void processInput() { promptInput->processInput(); }
01580     void enableRapidFire() { promptInput->rapidFireEnabled = true; }
01581     void disableRapidFire() { promptInput->rapidFireEnabled = false; }
01582     bool isRapidFireEnabled() { return promptInput->rapidFireEnabled; }
01583
01584     void startBlinking();
01585     void stopBlinking();
01586     void blink();
01587
01588     void setPromptTextColor(const QColor&);
01589     void setPromptBackgroundColor(const QColor&);
01590     void setPromptFontFamily(const QString&);
01591     void setPromptFontStyle(const QString&);
01592     void setPromptFontSize(int);
01593
01594     void floatingChanged(bool);
01595
01596     void saveHistory(const QString& fileName, bool html);
01597
01598 private slots:
01599
01600 signals:
01601     void appendTheHistory(const QString& txt, int prefixLength);
01602
01603     //For connecting outside of command prompt
01604     void startCommand(const QString& cmd);
01605     void runCommand(const QString& cmd, const QString& cmdtxt);
01606     void deletePressed();
01607     void tabPressed();
01608     void escapePressed();
01609     void upPressed();
01610     void downPressed();
01611     void F1Pressed();
01612     void F2Pressed();
01613     void F3Pressed();
01614     void F4Pressed();
01615     void F5Pressed();
01616     void F6Pressed();
01617     void F7Pressed();
01618     void F8Pressed();
01619     void F9Pressed();
01620     void F10Pressed();
01621     void F11Pressed();
01622     void F12Pressed();
01623     void cutPressed();
01624     void copyPressed();
01625     void pastePressed();
01626     void selectAllPressed();
01627     void undoPressed();
01628     void redoPressed();
01629
01630     void shiftPressed();
01631     void shiftReleased();
01632
01633     void showSettings();
01634
01635     void historyAppended(const QString& txt);
01636 };

```

```
01637
01641 class EmbDetailsDialog : public QDialog
01642 {
01643     Q_OBJECT
01644
01645 public:
01646     EmbDetailsDialog(QGraphicsScene* theScene, QWidget *parent = 0);
01647     ~EmbDetailsDialog();
01648
01649     QWidget* mainWidget;
01650
01651     void getInfo();
01652     QWidget* createMainWidget();
01653     QWidget* createHistogram();
01654
01655     QDialogButtonBox* buttonBox;
01656
01657     quint32 stitchesTotal;
01658     quint32 stitchesReal;
01659     quint32 stitchesJump;
01660     quint32 stitchesTrim;
01661     quint32 colorTotal;
01662     quint32 colorChanges;
01663
01664     QRectF boundingRect;
01665 };
01666
01670 class ImageWidget : public QWidget
01671 {
01672     Q_OBJECT
01673
01674 public:
01675     QImage img;
01676     ImageWidget(const QString &filename, QWidget* parent = 0);
01677     ~ImageWidget();
01678
01679     bool load(const QString &fileName);
01680     bool save(const QString &fileName);
01681
01682 protected:
01683     void paintEvent(QPaintEvent* event);
01684 };
01685
01689 class LayerManager : public QDialog
01690 {
01691     Q_OBJECT
01692
01693 public:
01694     QStandardItemModel* layerModel;
01695     QSortFilterProxyModel* layerModelSorted;
01696     QTreeView* treeView;
01697
01698     LayerManager(MainWindow* mw, QWidget *parent = 0);
01699     ~LayerManager();
01700
01701     void addLayer(const QString& name,
01702                 const bool visible,
01703                 const bool frozen,
01704                 const EmbReal zValue,
01705                 const QRgb color,
01706                 const QString& lineType,
01707                 const QString& lineWidth,
01708                 const bool print);
01709 };
01710
01714 class MainWindow: public QMainWindow
01715 {
01716     Q_OBJECT
01717
01718 public:
01719     MainWindow();
01720     ~MainWindow();
01721
01722     MdiArea* getMdiArea();
01723     MainWindow* getApplication();
01724     MdiWindow* activeMdiWindow();
01725     View* activeView();
01726     QGraphicsScene* activeScene();
01727     QUndoStack* activeUndoStack();
01728
01729     void setUndoCleanIcon(bool opened);
01730
01731     virtual void updateMenuToolbarStatusbar();
01732
01733     MainWindow* mainWin;
01734     MdiArea* mdiArea;
01735     CmdPrompt* prompt;
```

```

01736     PropertyEditor* dockPropEdit;
01737     UndoEditor* dockUndoEdit;
01738     StatusBar* statusbar;
01739
01740     QList<QGraphicsItem*> cutCopyObjectList;
01741
01742     std::string actuator(std::string command);
01743     std::string run_script_file(std::string fname);
01744     std::string run_script(std::vector<std::string> script);
01745     void LoadCommand(QString cmdName);
01746
01747     QAction* actionHash[200];
01748     QHash<QString, QToolBar*> toolbarHash;
01749     QHash<QString, QMenu*> menuHash;
01750
01751     QString formatFilterOpen;
01752     QString formatFilterSave;
01753
01754     bool isCommandActive() { return prompt->isCommandActive(); }
01755     QString activeCommand() { return prompt->activeCommand(); }
01756     QIcon create_icon(QString stub);
01757     void create_toolbar(QToolBar* toolbar, std::string label, std::vector<std::string> entries);
01758
01759     QString platformString();
01760
01761 public slots:
01762
01763     void enablePromptRapidFire();
01764     void disablePromptRapidFire();
01765
01766     void enableMoveRapidFire();
01767     void disableMoveRapidFire();
01768
01769     void onCloseWindow();
01770     virtual void onCloseMdiWin(MdiWindow*);
01771
01772     void recentMenuAboutToShow();
01773
01774     void onWindowActivated(QMdiSubWindow* w);
01775     void windowMenuAboutToShow();
01776     void windowMenuActivated( bool checked/*int id*/ );
01777     QAction*           getAction(int actionEnum);
01778
01779     void updateAllViewScrollBars(bool val);
01780     void updateAllViewCrossHairColors(QRgb color);
01781     void updateAllViewBackgroundColors(QRgb color);
01782     void updateAllViewSelectBoxColors(QRgb colorL, QRgb fillL, QRgb colorR, QRgb fillR, int alpha);
01783     void updateAllViewGridColors(QRgb color);
01784     void updateAllViewRulerColors(QRgb color);
01785
01786     void updatePickAddMode(bool val);
01787     void pickAddModeToggled();
01788
01789     void settingsPrompt();
01790
01791     void settingsDialog(const QString& showTab = QString());
01792     void readSettings();
01793     void writeSettings();
01794
01795     static bool validFileFormat(const QString &fileName);
01796
01797 protected:
01798     virtual void resizeEvent(QResizeEvent* );
01799     void closeEvent(QCloseEvent *event);
01800     QAction* getFileSeparator();
01801     void loadFormats();
01802
01803     bool shiftKeyPressedState;
01804
01805     QByteArray layoutState;
01806
01807     int numOfDocs;
01808     int docIndex;
01809
01810     QList<MdiWindow*> listMdiWin;
01811     QMdiSubWindow* findMdiWindow(const QString &fileName);
01812     QString openFilePath;
01813
01814     QAction* myFileSeparator;
01815
01816     QWizard* wizardTipOfDay;
01817     QLabel* labelTipOfDay;
01818     QCheckBox* checkBoxTipOfDay;
01819     QStringList listTipOfDay;
01820
01821     void createAllActions();
01822

```

```
01823 // Toolbars
01824 void createAllToolbars();
01825 void createPanToolbar();
01826 void createIconToolbar();
01827 void createHelpToolbar();
01828 void createLayerToolbar();
01829 void createPropertiesToolbar();
01830 void createTextToolbar();
01831 void createPromptToolbar();
01832
01833 QToolBar* toolbarFile;
01834 QToolBar* toolbarEdit;
01835 QToolBar* toolbarView;
01836 QToolBar* toolbarZoom;
01837 QToolBar* toolbarPan;
01838 QToolBar* toolbarIcon;
01839 QToolBar* toolbarHelp;
01840 QToolBar* toolbarLayer;
01841 QToolBar* toolbarText;
01842 QToolBar* toolbarProperties;
01843 QToolBar* toolbarPrompt;
01844
01845 // Selectors
01846 QComboBox* layerSelector;
01847 QComboBox* colorSelector;
01848 QComboBox* linetypeSelector;
01849 QComboBox* linewidthSelector;
01850 QFontComboBox* textFontSelector;
01851 QComboBox* textSizeSelector;
01852
01853 // Menus
01854 void createAllMenus();
01855 void createFileMenu();
01856 void createEditMenu();
01857 void createViewMenu();
01858 void createSettingsMenu();
01859 void createWindowMenu();
01860 void createHelpMenu();
01861
01862 QMenu* fileMenu;
01863 QMenu* editMenu;
01864 QMenu* viewMenu;
01865 QMenu* settingsMenu;
01866 QMenu* windowMenu;
01867 QMenu* helpMenu;
01868
01869 // SubMenus
01870 QMenu* recentMenu;
01871 QMenu* zoomMenu;
01872 QMenu* panMenu;
01873
01874 private slots:
01875     void hideUnimplemented();
01876
01877 public slots:
01878
01879     void stub_implement(QString txt);
01880     void stub_testing();
01881
01882     void promptHistoryAppended(const QString& txt);
01883     void logPromptInput(const QString& txt);
01884     void promptInputPrevious();
01885     void promptInputNext();
01886
01887     void newFile();
01888     void openFile(bool recent = false, const QString& recentFile = "");
01889     void openFilesSelected(const QStringList&);
01890     void openrecentfile();
01891     void savefile();
01892     void saveasfile();
01893     void print();
01894     void designDetails();
01895     void exit();
01896     void quit();
01897     void checkForUpdates();
01898 // Help Menu
01899     void tipOfTheDay();
01900     void buttonTipOfTheDayClicked(int);
01901     void checkBoxTipOfTheDayStateChanged(int);
01902     void help();
01903     void changelog();
01904     void about();
01905     void whatsThisContextHelp();
01906
01907     void cut();
01908     void copy();
01909     void paste();
```

```

01910
01911 void closeToolBar(QAction* );
01912 void floatingChangedToolBar(bool);
01913
01914 void toggleGrid();
01915 void toggleRuler();
01916 void toggleLwt();
01917
01918 // Icons
01919 void iconResize(int iconSize);
01920
01921 //Selectors
01922 void layerSelectorIndexChanged(int index);
01923 void colorSelectorIndexChanged(int index);
01924 void linetypeSelectorIndexChanged(int index);
01925 void linewidthSelectorIndexChanged(int index);
01926 void textFontSelectorCurrentFontChanged(const QFont& font);
01927 void textSizeSelectorIndexChanged(int index);
01928
01929 void setTextFont(const QString& str);
01930 void setTextSize(EmbReal num);
01931 void setTextAngle(EmbReal num);
01932 void setTextBold(bool val);
01933 void setTextItalic(bool val);
01934 void setTextUnderline(bool val);
01935 void setTextStrikeOut(bool val);
01936 void setTextOverline(bool val);
01937
01938 QString getCurrentLayer();
01939 QRgb getCurrentColor();
01940 QString getCurrentLineType();
01941 QString getCurrentLineWeight();
01942
01943 // Standard Slots
01944 void undo();
01945 void redo();
01946
01947 bool isShiftPressed();
01948 void setShiftPressed();
01949 void setShiftReleased();
01950
01951 void deletePressed();
01952 void escapePressed();
01953
01954 // Layer Toolbar
01955 void makeLayerActive();
01956 void layerManager();
01957 void layerPrevious();
01958 // Zoom Toolbar
01959 void zoomRealtime();
01960 void zoomPrevious();
01961 void zoomWindow();
01962 void zoomDynamic();
01963 void zoomScale();
01964 void zoomCenter();
01965 void zoomIn();
01966 void zoomOut();
01967 void zoomSelected();
01968 void zoomAll();
01969 void zoomExtents();
01970 // Pan SubMenu
01971 void panrealtime();
01972 void panpoint();
01973 void panLeft();
01974 void panRight();
01975 void panUp();
01976 void panDown();
01977
01978 void dayVision();
01979 void nightVision();
01980
01981 void doNothing();
01982
01983 public:
01984 void nativeAlert(const QString& txt);
01985 void nativeBlinkPrompt();
01986 void nativeSetPromptPrefix(const QString& txt);
01987 void nativeAppendPromptHistory(const QString& txt);
01988 void nativeEnablePromptRapidFire();
01989 void nativeDisablePromptRapidFire();
01990 void nativeInitCommand();
01991 void nativeEndCommand();
01992
01993 void nativeEnableMoveRapidFire();
01994 void nativeDisableMoveRapidFire();
01995
01996 void nativeExit();

```

```
01997     void nativeTipOfDay();
01998
01999     void nativeMessageBox(const QString& type, const QString& title, const QString& text);
02000
02001     void nativePrintArea(EmbReal x, EmbReal y, EmbReal w, EmbReal h);
02002
02003     void nativeSetBackgroundColor(uint8_t r, uint8_t g, uint8_t b);
02004     void nativeSetCrossHairColor(uint8_t r, uint8_t g, uint8_t b);
02005     void nativeSetGridColor(uint8_t r, uint8_t g, uint8_t b);
02006
02007     void nativePreviewOn(int clone, int mode, EmbReal x, EmbReal y, EmbReal data);
02008     void nativePreviewOff();
02009
02010     void nativeVulcanize();
02011     void nativeClearRubber();
02012     bool nativeAllowRubber();
02013     void nativeSpareRubber(qint64 id);
02014     // \todo void nativeSetRubberFilter(qint64 id);
02015     // \todo This is so more than 1 rubber object can exist at one time without updating all rubber
02016     // objects at once
02017     void nativeSetRubberMode(int mode);
02018     void nativeSetRubberPoint(const QString& key, EmbReal x, EmbReal y);
02019     void nativeSetRubberText(const QString& key, const QString& txt);
02020
02021     void nativeAddTextMulti(const QString& str, EmbReal x, EmbReal y, EmbReal rot, bool fill, int
02022     rubberMode);
02023     void nativeAddTextSingle(const QString& str, EmbReal x, EmbReal y, EmbReal rot, bool fill, int
02024     rubberMode);
02025     void nativeAddInfiniteLine(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal rot);
02026     void nativeAddRay(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal rot);
02027     void nativeAddLine(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal rot, int rubberMode);
02028     void nativeAddTriangle(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal x3, EmbReal y3,
02029     EmbReal rot, bool fill);
02030     void nativeAddRectangle(EmbReal x, EmbReal y, EmbReal w, EmbReal h, EmbReal rot, bool fill, int
02031     rubberMode);
02032     void nativeAddRoundedRectangle(EmbReal x, EmbReal y, EmbReal w, EmbReal h, EmbReal rad, EmbReal
02033     rot, bool fill);
02034     void nativeAddArc(EmbReal startX, EmbReal startY, EmbReal midX, EmbReal midY, EmbReal endX,
02035     EmbReal endY, int rubberMode);
02036     void nativeAddCircle(EmbReal centerX, EmbReal centerY, EmbReal radius, bool fill, int rubberMode);
02037     void nativeAddSlot(EmbReal centerX, EmbReal centerY, EmbReal diameter, EmbReal length, EmbReal
02038     rot, bool fill, int rubberMode);
02039     void nativeAddEllipse(EmbReal centerX, EmbReal centerY, EmbReal width, EmbReal height, EmbReal
02040     rot, bool fill, int rubberMode);
02041     void nativeAddPoint(EmbReal x, EmbReal y);
02042     void nativeAddRegularPolygon(EmbReal centerX, EmbReal centerY, quint16 sides, uint8_t mode,
02043     EmbReal rad, EmbReal rot, bool fill);
02044     void nativeAddPolygon(EmbReal startX, EmbReal startY, const QPainterPath& p, int rubberMode);
02045     void nativeAddPolyline(EmbReal startX, EmbReal startY, const QPainterPath& p, int rubberMode);
02046     void nativeAddPath(EmbReal startX, EmbReal startY, const QPainterPath& p, int rubberMode);
02047     void nativeAddHorizontalDimension(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal
02048     legHeight);
02049     void nativeAddVerticalDimension(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal
02050     legHeight);
02051     void nativeAddImage(const QString& img, EmbReal x, EmbReal y, EmbReal w, EmbReal h, EmbReal rot);
02052
02053     void nativeAddDimLeader(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, EmbReal rot, int
02054     rubberMode);
02055
02056     void nativeSetCursorShape(const QString& str);
02057     EmbReal nativeCalculateAngle(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2);
02058     EmbReal nativeCalculateDistance(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2);
02059     EmbReal nativePerpendicularDistance(EmbReal px, EmbReal py, EmbReal x1, EmbReal y1, EmbReal x2,
02060     EmbReal y2);
02061     int nativeNumSelected();
02062     void nativeAddToSelection(const QPainterPath path, Qt::ItemSelectionMode mode);
02063     void nativeClearSelection();
02064     void nativeDeleteSelected();
02065     void nativeCutSelected(EmbReal x, EmbReal y);
02066     void nativeCopySelected(EmbReal x, EmbReal y);
02067     void nativePasteSelected(EmbReal x, EmbReal y);
02068     void nativeMoveSelected(EmbReal dx, EmbReal dy);
02069     void nativeScaleSelected(EmbReal x, EmbReal y, EmbReal factor);
02070     void nativeRotateSelected(EmbReal x, EmbReal y, EmbReal rot);
02071     void nativeMirrorSelected(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2);
02072
02073     EmbReal nativeQSnapX();
02074     EmbReal nativeQSnapY();
02075     EmbReal nativeMouseX();
02076     EmbReal nativeMouseY();
02077 };
02078
02079 MainWindow* mainWin();
02080
02081 }
```

```

02070 class MdiWindow: public QMdiSubWindow
02071 {
02072     Q_OBJECT
02073
02074 public:
02075     MdiWindow(const int theIndex, MainWindow* mw, QMdiArea* parent, Qt::WindowFlags wflags);
02076     ~MdiWindow();
02077
02078     virtual QSize sizeHint() const;
02079     QString getCurrentFile() { return curFile; }
02080     QString getShortCurrentFile();
02081     View* getView() { return gview; }
02082     QGraphicsScene* getScene() { return gscene; }
02083     QString getCurrentLayer() { return curLayer; }
02084     QRgb getCurrentColor() { return curColor; }
02085     QString getCurrentLineType() { return curLineType; }
02086     QString getCurrentLineWeight() { return curLineWeight; }
02087     void setCurrentLayer(const QString& layer) { curLayer = layer; }
02088     void setCurrentColor(const QRgb& color) { curColor = color; }
02089     void setCurrentLineType(const QString& lineType) { curLineType = lineType; }
02090     void setCurrentLineWeight(const QString& lineWeight) { curLineWeight = lineWeight; }
02091     void designDetails();
02092     bool loadFile(const QString &fileName);
02093     bool saveFile(const QString &fileName);
02094 signals:
02095     void sendCloseMdiWin(MdiWindow* );
02096
02097 public slots:
02098     void closeEvent(QCloseEvent* e);
02099     void onWindowActivated();
02100
02101     void currentLayerChanged(const QString& layer);
02102     void currentColorChanged(const QRgb& color);
02103     void currentLinetypeChanged(const QString& type);
02104     void currentLineweightChanged(const QString& weight);
02105
02106     void updateColorLinetypeLinewidth();
02107     void deletePressed();
02108     void escapePressed();
02109
02110     void showViewScrollBars(bool val);
02111     void setViewCrossHairColor(QRgb color);
02112     void setViewBackgroundColor(QRgb color);
02113     void setViewSelectBoxColors(QRgb colorL, QRgb fillL, QRgb colorR, QRgb fillR, int alpha);
02114     void setViewGridColor(QRgb color);
02115     void setViewRulerColor(QRgb color);
02116
02117     void print();
02118     void saveBMC();
02119
02120     void promptHistoryAppended(const QString& txt);
02121     void logPromptInput(const QString& txt);
02122     void promptInputPrevious();
02123     void promptInputNext();
02124
02125 protected:
02126
02127 private:
02128     MainWindow* mainWin;
02129     QMdiArea* mdiArea;
02130     QGraphicsScene* gscene;
02131     View* gview;
02132
02133     bool fileWasLoaded;
02134
02135     QString promptHistory;
02136     QList<QString> promptInputList;
02137     int promptInputNum;
02138
02139     QPrinter           printer;
02140
02141     QString curFile;
02142     void setCurrentFile(const QString& fileName);
02143     QString fileExtension(const QString& fileName);
02144
02145     int myIndex;
02146
02147     QString curLayer;
02148     QRgb curColor;
02149     QString curLineType;
02150     QString curLineWeight;
02151
02152     void promptInputPrevNext(bool prev);
02153 };
02154
02155 class MdiArea : public QMdiArea
02156 {

```

```
02160     Q_OBJECT
02161
02162 public:
02163     MainWindow* mainWin;
02164
02165     bool useLogo;
02166     bool useTexture;
02167     bool useColor;
02168
02169     QPixmap bgLogo;
02170     QPixmap bgTexture;
02171     QColor bgColor;
02172
02173     void zoomExtentsAllSubWindows();
02174     void forceRepaint();
02175
02176     MdiArea(MainWindow* mw, QWidget* parent = 0);
02177     ~MdiArea();
02178
02179     void useBackgroundLogo(bool use);
02180     void useBackgroundTexture(bool use);
02181     void useBackgroundColor(bool use);
02182
02183     void setBackgroundLogo(const QString& fileName);
02184     void setBackgroundTexture(const QString& fileName);
02185     void setBackgroundColor(const QColor& color);
02186
02187 public slots:
02188     void cascade();
02189     void tile();
02190 protected:
02191     virtual void mouseDoubleClickEvent(QMouseEvent* e);
02192     virtual void paintEvent(QPaintEvent* e);
02193 };
02194
02195 typedef struct GroupBoxData_ {
02196     int object;
02197     char key[200];
02198     char icon_name[200];
02199     char label[200];
02200     char type[200];
02201     char map_signal[200];
02202 } GroupBoxData;
02203
02204 class PreviewDialog : public QFileDialog
02205 {
02206     Q_OBJECT
02207
02208 public:
02209     PreviewDialog(QWidget* parent = 0,
02210         const QString& caption = QString(),
02211         const QString& directory = QString(),
02212         const QString& filter = QString());
02213     ~PreviewDialog();
02214
02215     ImageWidget* imgWidget;
02216 };
02217
02218
02219 class PropertyEditor : public QDockWidget
02220 {
02221     Q_OBJECT
02222
02223 public:
02224     PropertyEditor(const QString& iconDirectory = QString(), bool pickAddMode = true, QWidget*
02225         widgetToFocus = 0, QWidget* parent = 0); //, Qt::WindowFlags flags = 0;
02226     ~PropertyEditor();
02227
02228 protected:
02229     bool eventFilter(QObject* obj, QEvent* event);
02230
02231 signals:
02232     void pickAddModeToggled();
02233
02234 public slots:
02235     void setSelectedItems(QList<QGraphicsItem*> itemList);
02236     void updatePickAddModeButton(bool pickAddMode);
02237
02238 private slots:
02239     void fieldEdited(QObject* fieldObj);
02240     void showGroups(int objType);
02241     void showOneType(int index);
02242     void hideAllGroups();
02243     void clearAllFields();
02244     void togglePickAddMode();
```

```

02249     QWidget* focusWidget;
02250
02251     QString iconDir;
02252     int iconSize;
02253     Qt::ToolButtonStyle propertyEditorButtonStyle;
02254
02255     bool pickAdd;
02256
02257     QList<QGraphicsItem*> selectedItemList;
02258
02259     //Helper functions
02260     QToolButton* createToolButton(const QString& iconName, const QString& txt);
02261     QLineEdit* createLineEdit(const QString& validatorType = QString(), bool readOnly = false);
02262     QComboBox* createComboBox(bool disable = false);
02263     QFontComboBox* createFontComboBox(bool disable = false);
02264
02265     int precisionAngle;
02266     int precisionLength;
02267
02268     //Used when checking if fields vary
02269     QString fieldOldText;
02270     QString fieldNewText;
02271     QString fieldVariesText;
02272     QString fieldYesText;
02273     QString fieldNoText;
02274     QString fieldOnText;
02275     QString fieldOffText;
02276
02277     void updateLineEditStrIfVaries(QLineEdit* lineEdit, const QString& str);
02278     void updateLineEditNumIfVaries(QLineEdit* lineEdit, EmbReal num, bool useAnglePrecision);
02279     void updateFontComboBoxStrIfVaries(QFontComboBox* fontComboBox, const QString& str);
02280     void updateComboBoxStrIfVaries(QComboBox* comboBox, const QString& str, const QStringList&
02281         strList);
02282     void updateComboBoxBoolIfVaries(QComboBox* comboBox, bool val, bool yesOrNoText);
02283
02284     QSignalMapper* signalMapper;
02285     void mapSignal(QObject* fieldObj, const QString& name, QVariant value);
02286
02287     //=====
02288     //Selection
02289     //=====
02290     QComboBox* createComboBoxSelected();
02291     QToolButton* createToolButtonQSelect();
02292     QToolButton* createToolButtonPickAdd();
02293
02294     QComboBox* comboBoxSelected;
02295     QToolButton* toolButtonQSelect;
02296     QToolButton* toolButtonPickAdd;
02297
02298     //TODO: Alphabetic/Categorized TabWidget
02299
02300     void createGroupBox(
02301         std::string group_box_key,
02302         const char *title,
02303         GroupBoxData data[],
02304         int lines);
02305
02306     QGroupBox* createGroupBoxGeneral();
02307     QGroupBox* createGroupBoxGeometryArc();
02308     QGroupBox* createGroupBoxMiscArc();
02309     QGroupBox* createGroupBoxGeometryBlock();
02310     QGroupBox* createGroupBoxGeometryCircle();
02311     QGroupBox* createGroupBoxGeometryDimAligned();
02312     QGroupBox* createGroupBoxGeometryDimAngular();
02313     QGroupBox* createGroupBoxGeometryDimArcLength();
02314     QGroupBox* createGroupBoxGeometryDimDiameter();
02315     QGroupBox* createGroupBoxGeometryDimLeader();
02316     QGroupBox* createGroupBoxGeometryDimLinear();
02317     QGroupBox* createGroupBoxGeometryDimOrdinate();
02318     QGroupBox* createGroupBoxGeometryDimRadius();
02319     QGroupBox* createGroupBoxGeometryEllipse();
02320     QGroupBox* createGroupBoxGeometryImage();
02321     QGroupBox* createGroupBoxMiscImage();
02322     QGroupBox* createGroupBoxGeometryInfiniteLine();
02323     QGroupBox* createGroupBoxGeometryLine();
02324     QGroupBox* createGroupBoxGeometryPath();
02325     QGroupBox* createGroupBoxMiscPath();
02326     QGroupBox* createGroupBoxGeometryPoint();
02327     QGroupBox* createGroupBoxGeometryPolygon();
02328     QGroupBox* createGroupBoxGeometryPolyline();
02329     QGroupBox* createGroupBoxMiscPolyline();
02330     QGroupBox* createGroupBoxGeometryRay();
02331     QGroupBox* createGroupBoxGeometryRectangle();
02332     QGroupBox* createGroupBoxGeometryTextMulti();
02333     QGroupBox* createGroupBoxTextTextSingle();
02334     QGroupBox* createGroupBoxGeometryTextSingle();

```

```
02335     QGroupBox* createGroupBoxMiscTextSingle();
02336 };
02337
02338
02339 class SelectBox : public QRubberBand
02340 {
02341     Q_OBJECT
02342
02343 public:
02344     SelectBox(Shape s, QWidget* parent = 0);
02345
02346     QColor leftBrushColor;
02347     QColor rightBrushColor;
02348     QColor leftPenColor;
02349     QColor rightPenColor;
02350     uint8_t alpha;
02351
02352     QBrush dirBrush;
02353     QBrush leftBrush;
02354     QBrush rightBrush;
02355
02356     QPen dirPen;
02357     QPen leftPen;
02358     QPen rightPen;
02359
02360     bool boxDir;
02361
02362     void forceRepaint();
02363
02364 public slots:
02365     void setDirection(int dir);
02366     void setColors(const QColor& colorL, const QColor& fillL, const QColor& colorR, const QColor&
fillR, int newAlpha);
02367
02368 protected:
02369     void paintEvent(QPaintEvent* );
02370 };
02371
02372 class Settings_Dialog : public QDialog
02373 {
02374     Q_OBJECT
02375
02376 public:
02377     Settings_Dialog(MainWindow* mw, const QString& showTab = QString(), QWidget *parent = 0);
02378     ~Settings_Dialog();
02379
02380     MainWindow* mainWin;
02381
02382     QTabWidget* tabWidget;
02383
02384     QWidget* createTabGeneral();
02385     QWidget* createTabFilePaths();
02386     QWidget* createTabDisplay();
02387     QWidget* createTabPrompt();
02388     QWidget* createTabOpenSave();
02389     QWidget* createTabPrinting();
02390     QWidget* createTabSnap();
02391     QWidget* createTabGridRuler();
02392     QWidget* createTabOrthoPolar();
02393     QWidget* createTabQuickSnap();
02394     QWidget* createTabQuickTrack();
02395     QWidget* createTabLineWeight();
02396     QWidget* createTabSelection();
02397
02398     QDialogButtonBox* buttonBox;
02399
02400     void addColorsToComboBox(QComboBox* comboBox);
02401
02402     void create_float_spinbox(
02403         QGroupBox *gb,
02404         QGridLayout* gridLayout,
02405         const char *label_in,
02406         EmbReal single_step,
02407         EmbReal lower,
02408         EmbReal upper,
02409         EmbReal *ptr,
02410         int row);
02411     QCheckBox* create_checkbox(QGroupBox *groupbox, std::string label);
02412     void set_label_visibility(QObject *parent, const char *name, bool visibility);
02413     void set_spinbox_visibility(QObject *parent, const char *name, bool visibility);
02414
02415 private slots:
02416     void comboBoxLanguageCurrentIndexChanged(const QString& );
02417     void comboBoxIconThemeCurrentIndexChanged(const QString& );
02418     void comboBoxIconSizeCurrentIndexChanged(int );
02419     void checkBoxGeneralMdiBGUseLogoStateChanged(int );
02420     void chooseGeneralMdiBackgroundLogo();
```

```

02424 void checkBoxGeneralMdiBGUseTextureStateChanged(int);
02425 void chooseGeneralMdiBackgroundTexture();
02426 void checkBoxGeneralMdiBGUseColorStateChanged(int);
02427 void chooseGeneralMdiBackgroundColor();
02428 void currentGeneralMdiBackgroundColorChanged(const QColor&);
02429 void checkBoxTipOfTheDayStateChanged(int);
02430 void checkBoxUseOpenGLStateChanged(int);
02431 void checkBoxRenderHintAAStateChanged(int);
02432 void checkBoxRenderHintTextAAStateChanged(int);
02433 void checkBoxRenderHintSmoothPixStateChanged(int);
02434 void checkBoxRenderHintHighAAStateChanged(int);
02435 void checkBoxRenderHintNonCosmeticStateChanged(int);
02436 void checkBoxShowScrollBarsStateChanged(int);
02437 void comboBoxScrollBarWidgetCurrentIndexChanged(int);
02438 void spinBoxZoomScaleInValueChanged(double);
02439 void spinBoxZoomScaleOutValueChanged(double);
02440 void checkBoxDisableBGStateChanged(int);
02441 void chooseDisplayCrossHairColor();
02442 void currentDisplayCrossHairColorChanged(const QColor&);
02443 void chooseDisplayBackgroundColor();
02444 void currentDisplayBackgroundColorChanged(const QColor&);
02445 void chooseDisplaySelectBoxLeftColor();
02446 void currentDisplaySelectBoxLeftColorChanged(const QColor&);
02447 void chooseDisplaySelectBoxLeftFill();
02448 void currentDisplaySelectBoxLeftFillChanged(const QColor&);
02449 void chooseDisplaySelectBoxRightColor();
02450 void currentDisplaySelectBoxRightColorChanged(const QColor&);
02451 void chooseDisplaySelectBoxRightFill();
02452 void currentDisplaySelectBoxRightFillChanged(const QColor&);
02453 void spinBoxDisplaySelectBoxAlphaValueChanged(int);
02454 void choosePromptTextColor();
02455 void currentPromptTextColorChanged(const QColor&);
02456 void choosePromptBackgroundColor();
02457 void currentPromptBackgroundColorChanged(const QColor&);
02458 void comboBoxPromptFontFamilyCurrentIndexChanged(const QString&);
02459 void comboBoxPromptFontStyleCurrentIndexChanged(const QString&);
02460 void spinBoxPromptFontSizeValueChanged(int);
02461 void checkBoxPromptSaveHistoryStateChanged(int);
02462 void checkBoxPromptSaveHistoryAsHtmlStateChanged(int);
02463 void checkBoxCustomFilterStateChanged(int);
02464 void buttonCustomFilterSelectAllClicked();
02465 void buttonCustomFilterClearAllClicked();
02466 void spinBoxRecentMaxFilesValueChanged(int);
02467 void spinBoxTrimDstNumJumpsValueChanged(int);
02468 void checkBoxGridShowOnLoadStateChanged(int);
02469 void checkBoxGridShowOriginStateChanged(int);
02470 void checkBoxGridColorMatchCrossHairStateChanged(int);
02471 void chooseGridColor();
02472 void currentGridColorChanged(const QColor&);
02473 void checkBoxGridLoadFromFileStateChanged(int);
02474 void comboBoxGridTypeCurrentIndexChanged(const QString&);
02475 void checkBoxGridCenterOnOriginStateChanged(int);
02476 void checkBoxRulerShowOnLoadStateChanged(int);
02477 void comboBoxRulerMetricCurrentIndexChanged(int);
02478 void chooseRulerColor();
02479 void currentRulerColorChanged(const QColor&);
02480 void spinBoxRulerPixelSizeValueChanged(double);
02481 void buttonQSnapSelectAllClicked();
02482 void buttonQSnapClearAllClicked();
02483 void comboBoxQSnapLocatorColorCurrentIndexChanged(int);
02484 void sliderQSnapLocatorsSizeValueChanged(int);
02485 void sliderQSnapApertureSizeValueChanged(int);
02486 void checkBoxLwtShowLwtStateChanged(int);
02487 void checkBoxLwtRealRenderStateChanged(int);
02488 void checkBoxSelectionModePickFirstStateChanged(int);
02489 void checkBoxSelectionModePickAddStateChanged(int);
02490 void checkBoxSelectionModePickDragStateChanged(int);
02491 void comboBoxSelectionModeCoolGripColorCurrentIndexChanged(int);
02492 void comboBoxSelectionModeHotGripColorCurrentIndexChanged(int);
02493 void sliderSelectionGripSizeValueChanged(int);
02494 void sliderSelectionPickBoxSizeValueChanged(int);
02495
02496 void acceptChanges();
02497 void rejectChanges();
02498
02499 signals:
02500 void buttonCustomFilterSelectAll(bool);
02501 void buttonCustomFilterClearAll(bool);
02502 void buttonQSnapSelectAll(bool);
02503 void buttonQSnapClearAll(bool);
02504 };
02505
02506
02507 class StatusBar : public QStatusBar
02508 {
02509     Q_OBJECT
02510

```

```
02511 public:
02512     StatusBar(MainWindow* mw, QWidget* parent = 0);
02513
02514     StatusBarButton* statusBarSnapButton;
02515     StatusBarButton* statusBarGridButton;
02516     StatusBarButton* statusBarRulerButton;
02517     StatusBarButton* statusBarOrthoButton;
02518     StatusBarButton* statusBarPolarButton;
02519     StatusBarButton* statusBarQSnapButton;
02520     StatusBarButton* statusBarQTrackButton;
02521     StatusBarButton* statusBarLwtButton;
02522     QLabel* statusBarMouseCoord;
02523
02524     void setMouseCoord(EmbReal x, EmbReal y);
02525 };
02526
02527 class StatusBarButton : public QToolButton
02528 {
02529     Q_OBJECT
02530
02531     public:
02532         StatusBarButton(QString buttonText, MainWindow* mw, StatusBar* statbar, QWidget *parent = 0);
02533
02534     MainWindow* mainWin;
02535     StatusBar* statusbar;
02536
02537     protected:
02538         void contextMenuEvent(QContextMenuEvent *event = 0);
02539
02540     private slots:
02541         void settingsSnap();
02542         void settingsGrid();
02543         void settingsRuler();
02544         void settingsOrtho();
02545         void settingsPolar();
02546         void settingsQSnap();
02547         void settingsQTrack();
02548         void settingsLwt();
02549         void toggleSnap(bool on);
02550         void toggleGrid(bool on);
02551         void toggleRuler(bool on);
02552         void toggleOrtho(bool on);
02553         void togglePolar(bool on);
02554         void toggleQSnap(bool on);
02555         void toggleQTrack(bool on);
02556         void toggleLwt(bool on);
02557
02558     public slots:
02559         void enableLwt();
02560         void disableLwt();
02561         void enableReal();
02562         void disableReal();
02563
02564 };
02565
02566 class UndoEditor : public QDockWidget
02567 {
02568     Q_OBJECT
02569
02570     public:
02571         UndoEditor(const QString& iconDirectory = QString(), QWidget* widgetToFocus = 0, QWidget* parent =
02572             0); //, Qt::WindowFlags flags = 0);
02573         ~UndoEditor();
02574
02575         void addStack(QUndoStack* stack);
02576
02577         bool canUndo() const;
02578         bool canRedo() const;
02579
02580         QWidget* focusWidget;
02581
02582         QString iconDir;
02583         int iconSize;
02584
02585         QUndoGroup* undoGroup;
02586         QUndoView* undoView;
02587
02588         QString undoText() const;
02589         QString redoText() const;
02590
02591     protected:
02592
02593     public slots:
02594         void undo();
02595         void redo();
02596
02597         void updateCleanIcon(bool opened);
02598
02599 };
02600
02601 class UndoableAddCommand : public QUndoCommand
```

```

02606 {
02607     public:
02608         UndoableAddCommand(const QString& text, BaseObject* obj, View* v, QUndoCommand* parent = 0);
02609
02610     void undo();
02611     void redo();
02612
02613     BaseObject* object;
02614     View*      gview;
02615 };
02616
02620 class UndoableDeleteCommand : public QUndoCommand
02621 {
02622     public:
02623         UndoableDeleteCommand(const QString& text, BaseObject* obj, View* v, QUndoCommand* parent = 0);
02624
02625     void undo();
02626     void redo();
02627
02628     BaseObject* object;
02629     View*      gview;
02630 };
02631
02635 class UndoableMoveCommand : public QUndoCommand
02636 {
02637     public:
02638         UndoableMoveCommand(EmbReal deltaX, EmbReal deltaY, const QString& text, BaseObject* obj, View* v,
02639             QUndoCommand* parent = 0);
02640
02641     void undo();
02642     void redo();
02643
02644     BaseObject* object;
02645     View*      gview;
02646     EmbReal dx;
02647     EmbReal dy;
02648 };
02649
02652 class UndoableRotateCommand : public QUndoCommand
02653 {
02654     public:
02655         UndoableRotateCommand(EmbReal pivotPointX, EmbReal pivotPointY, EmbReal rotAngle, const QString&
02656             text, BaseObject* obj, View* v, QUndoCommand* parent = 0);
02657
02658     void undo();
02659     void redo();
02660
02661     void rotate(EmbReal x, EmbReal y, EmbReal rot);
02662
02663     BaseObject* object;
02664     View*      gview;
02665     EmbReal pivotX;
02666     EmbReal pivotY;
02667     EmbReal angle;
02668 };
02669
02672 class UndoableScaleCommand : public QUndoCommand
02673 {
02674     public:
02675         UndoableScaleCommand(EmbReal x, EmbReal y, EmbReal scaleFactor, const QString& text, BaseObject*  

02676             obj, View* v, QUndoCommand* parent = 0);
02677
02678     void undo();
02679     void redo();
02680
02681     BaseObject* object;
02682     View*      gview;
02683     EmbReal dx;
02684     EmbReal dy;
02685     EmbReal factor;
02686 };
02687
02690 class UndoableNavCommand : public QUndoCommand
02691 {
02692     public:
02693         UndoableNavCommand(const QString& type, View* v, QUndoCommand* parent = 0);
02694
02695         int id() const { return 1234; }
02696         bool mergeWith(const QUndoCommand* command);
02697         void undo();
02698         void redo();
02699
02700         QString navType;
02701         QTransform fromTransform;
02702         QTransform toTransform;
02703         QPointF fromCenter;
02704         QPointF toCenter;

```

```
02705     bool done;
02706     View* gview;
02707 };
02708
02712 class UndoableGripEditCommand : public QUndoCommand
02713 {
02714 public:
02715     UndoableGripEditCommand(const QPointF beforePoint, const QPointF afterPoint, const QString& text,
02716     BaseObject* obj, View* v, QUndoCommand* parent = 0);
02717     void undo();
02718     void redo();
02719
02720     BaseObject* object;
02721     View* gview;
02722     QPointF before;
02723     QPointF after;
02724 };
02725
02729 class UndoableMirrorCommand : public QUndoCommand
02730 {
02731 public:
02732     UndoableMirrorCommand(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2, const QString& text,
02733     BaseObject* obj, View* v, QUndoCommand* parent = 0);
02734     void undo();
02735     void redo();
02736     void mirror();
02737
02738     BaseObject* object;
02739     View* gview;
02740     QLineF mirrorLine;
02741 };
02742
02743
02747 class View : public QGraphicsView
02748 {
02749     Q_OBJECT
02750
02751 public:
02752     View(MainWindow* mw, QGraphicsScene* theScene, QWidget* parent);
02753     ~View();
02754
02755     EmbView view_state;
02756
02757     bool allowZoomIn();
02758     bool allowZoomOut();
02759
02760     void recalculateLimits();
02761     void zoomToPoint(const QPoint& mousePoint, int zoomDir);
02762     void centerAt(const QPointF& centerPoint);
02763     QPointF center() { return mapToScene(rect().center()); }
02764
02765     QUndoStack* getUndoStack() { return undoStack; }
02766     void addObject(BaseObject* obj);
02767     void deleteObject(BaseObject* obj);
02768     void vulcanizeObject(BaseObject* obj);
02769
02770 public slots:
02771     void zoomIn();
02772     void zoomOut();
02773     void zoomWindow();
02774     void zoomSelected();
02775     void zoomExtents();
02776     void panRealTime();
02777     void panPoint();
02778     void panLeft();
02779     void panRight();
02780     void panUp();
02781     void panDown();
02782     void selectAll();
02783     void selectionChanged();
02784     void clearSelection();
02785     void deleteSelected();
02786     void moveSelected(EmbReal dx, EmbReal dy);
02787     void cut();
02788     void copy();
02789     void paste();
02790     void repeatAction();
02791     void moveAction();
02792     void scaleAction();
02793     void scaleSelected(EmbReal x, EmbReal y, EmbReal factor);
02794     void rotateAction();
02795     void rotateSelected(EmbReal x, EmbReal y, EmbReal rot);
02796     void mirrorSelected(EmbReal x1, EmbReal y1, EmbReal x2, EmbReal y2);
02797     int numSelected();
02798
```

```

02799 void deletePressed();
02800 void escapePressed();
02801
02802 void cornerButtonClicked();
02803
02804 void showScrollBars(bool val);
02805 void setCornerButton();
02806 void setCrossHairColor(QRgb color);
02807 void setCrossHairSize(uint8_t percent);
02808 void setBackGroundColor(QRgb color);
02809 void setSelectBoxColors(QRgb colorL, QRgb fillL, QRgb colorR, QRgb fillR, int alpha);
02810 void toggleSnap(bool on);
02811 void toggleGrid(bool on);
02812 void toggleRuler(bool on);
02813 void toggleOrtho(bool on);
02814 void togglePolar(bool on);
02815 void toggleQSnap(bool on);
02816 void toggleQTrack(bool on);
02817 void toggleLwt(bool on);
02818 void toggleReal(bool on);
02819 bool isLwtEnabled();
02820 bool isRealEnabled();
02821
02822 void setGridColor(QRgb color);
02823 void createGrid(const QString& gridType);
02824 void setRulerColor(QRgb color);
02825
02826 void previewOn(int clone, int mode, EmbReal x, EmbReal y, EmbReal data);
02827 void previewOff();
02828
02829 void enableMoveRapidFire();
02830 void disableMoveRapidFire();
02831
02832 bool allowRubber();
02833 void addToRubberRoom(QGraphicsItem* item);
02834 void vulcanizeRubberRoom();
02835 void clearRubberRoom();
02836 void spareRubber(qint64 id);
02837 void setRubberMode(int mode);
02838 void setRubberPoint(const QString& key, const QPointF& point);
02839 void setRubberText(const QString& key, const QString& txt);
02840
02841 protected:
02842 void mouseDoubleClickEvent(QMouseEvent* event);
02843 void mousePressEvent(QMouseEvent* event);
02844 void mouseMoveEvent(QMouseEvent* event);
02845 void mouseReleaseEvent(QMouseEvent* event);
02846 void wheelEvent(QWheelEvent* event);
02847 void contextMenuEvent(QContextMenuEvent* event);
02848 void drawBackground(QPainter* painter, const QRectF& rect);
02849 void drawForeground(QPainter* painter, const QRectF& rect);
02850 void enterEvent(QEvent* event);
02851
02852 private:
02853 QHash<qint64, QGraphicsItem*> hashDeletedObjects;
02854
02855 QList<qint64> spareRubberList;
02856
02857 QColor gridColor;
02858 QPainterPath gridPath;
02859 void createGridRect();
02860 void createGridPolar();
02861 void createGridIso();
02862 QPainterPath originPath;
02863 void createOrigin();
02864
02865 bool rulerMetric;
02866 QColor rulerColor;
02867 uint8_t rulerPixelSize;
02868 void loadRulerSettings();
02869
02870 bool willUnderflowInt32(qint64 a, qint64 b);
02871 bool willOverflowInt32(qint64 a, qint64 b);
02872 int roundToMultiple(bool roundUp, int numToRound, int multiple);
02873 QPainterPath createRulerTextPath(float x, EmbReal y, QString str, EmbReal height);
02874
02875 QList<QGraphicsItem*> previewObjectList;
02876 QGraphicsItemGroup* previewObjectItemGroup;
02877 QPointF previewPoint;
02878 EmbReal previewData;
02879 int previewMode;
02880
02881 QList<QGraphicsItem*> createObjectList(QList<QGraphicsItem*> list);
02882 QPointF cutCopyMousePoint;
02883 QGraphicsItemGroup* pasteObjectItemGroup;
02884 QPointF pasteDelta;
02885

```

```
02886     QList<QGraphicsItem*> rubberRoomList;
02887
02888     void copySelected();
02889
02890     bool grippingActive;
02891     bool rapidMoveActive;
02892     bool previewActive;
02893     bool pastingActive;
02894     bool movingActive;
02895     bool selectingActive;
02896     bool zoomWindowActive;
02897     bool panningRealTimeActive;
02898     bool panningPointActive;
02899     bool panningActive;
02900     bool qSnapActive;
02901     bool qSnapToggle;
02902
02903     void startGripping(BaseObject* obj);
02904     void stopGripping(bool accept = false);
02905
02906     BaseObject* gripBaseObj;
02907     BaseObject* tempBaseObj;
02908
02909     MainWindow* mainWin;
02910     QGraphicsScene* gscene;
02911     QUndoStack* undoStack;
02912
02913     SelectBox* selectBox;
02914     QPointF scenePressPoint;
02915     QPoint pressPoint;
02916     QPointF sceneMovePoint;
02917     QPoint movePoint;
02918     QPointF sceneReleasePoint;
02919     QPoint releasePoint;
02920     QPointF sceneGripPoint;
02921
02922     void updateMouseCoords(int x, int y);
02923     QPoint viewMousePoint;
02924     QPointF sceneMousePoint;
02925     QRgb qsnapLocatorColor;
02926     uint8_t qsnapLocatorSize;
02927     uint8_t qsnapApertureSize;
02928     QRgb gripColorCool;
02929     QRgb gripColorHot;
02930     uint8_t gripSize;
02931     uint8_t pickBoxSize;
02932     QRgb crosshairColor;
02933     quint32 crosshairSize;
02934
02935     void panStart(const QPoint& point);
02936     int panDistance;
02937     int panStartX;
02938     int panStartY;
02939
02940     void alignScenePointWithViewPoint(const QPointF& scenePoint, const QPoint& viewPoint);
02941 };
02942
02943     typedef struct Action_ {
02944     int hash;
02945     /*< Index in the actionHash array. */
02946     std::string icon;
02947     /*< The stub used for the icon and the basic command. */
02948     std::string command;
02949     /*< . */
02950     std::string tooltip;
02951     /*< The label in the menus and the message that appears when
02952         you hover over an icon. */
02953     std::string statustip;
02954     /*< The message that appears at the bottom of the . */
02955     std::string shortcut;
02956     /*< The keyboard shortcut for this action. */
02957     std::vector<std::string> aliases;
02958     /*< A list of all alternative commands, if empty only
02959         the icon string will be . */
02960     std::vector<std::string> script;
02961     /*< If this is a compound action this will be a
02962         list of commands or it can allow for command line
02963         style command aliases. For example: icon16 would become
02964         the string list {"iconResize 16"}.
02965     std::string menu_name;
02966     /*< . */
02967     int menu_position;
02968     /*< . */
02969     std::string toolbar_name;
02970     /*< . */
02971     int toolbar_position;
02972     /*< . */
```

```
02976 } Action;
02977 /* Functions in the global namespace
02978 * -----
02980 */
02981 int get_action_index(std::string cmd);
02982 void debug_message(std::string msg);
02983 int read_settings(const char *file);
02984 void write_settings(const char *fname);
02985
02986 /* Global data
02987 * -----
02988 */
02989 extern Settings settings;
02990 extern Settings dialog;
02991 extern std::vector<Action> action_table;
02992 extern std::vector<std::string> file_toolbar;
02993 extern std::vector<std::string> edit_toolbar;
02994 extern std::vector<std::string> view_toolbar;
02995 extern std::vector<std::string> zoom_toolbar;
02996
02997 extern std::unordered_map<std::string, QGroupBox *> groupBoxes;
02998 extern std::unordered_map<std::string, QComboBox *> comboBoxes;
02999 extern std::unordered_map<std::string, QLineEdit *> lineEdits;
03000 extern std::unordered_map<std::string, QToolButton *> toolButtons;
03001 extern QFontComboBox* comboBoxTextSingleFont;
03002 extern std::vector<std::string> group_box_list;
03003 extern GroupBoxData group_box_arc_geometry[];
03004 extern const int group_box_arc_geometry_entries;
03005 extern GroupBoxData group_box_ellipse_geometry[];
03006 extern const int group_box_ellipse_geometry_entries;
03007
03008 extern std::unordered_map<std::string, std::string> config;
03009 extern std::unordered_map<std::string, GroupBoxData*> group_box_data;
03010
03011 #endif
```

18.8 embroidermodder2/imagewidget.cpp File Reference

```
#include "embroidermodder.h"
```

18.9 embroidermodder2/interface.cpp File Reference

```
#include "embroidermodder.h"
```

Functions

- void `debug_message` (std::string msg)

18.9.1 Detailed Description

For wrappers to the Qt internals.

To help reduce reliance on Qt, only the functions wrap the Qt functions have a wrapper here. Ideally we could move some of the Qt headers here.

18.9.2 Function Documentation

```
18.9.2.1 debug_message() void debug_message (
    std::string msg )
```

18.10 embroidermodder2/layer-manager.cpp File Reference

```
#include "embroidermodder.h"
```

18.10.1 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/>

18.11 embroidermodder2/mainwindow-commands.cpp File Reference

```
#include "embroidermodder.h"
```

18.12 embroidermodder2/mainwindow-menus.cpp File Reference

```
#include "embroidermodder.h"
```

18.13 embroidermodder2/mainwindow-settings.cpp File Reference

```
#include "embroidermodder.h"
#include <string>
#include <iostream>
#include <fstream>
```

Functions

- `QString SettingsDir ()`
- `QString SettingsPath ()`
- `std::vector< std::string > to_string_vector (QStringList list)`
- `void read_configuration ()`
- `std::string write_setting (std::string label, int a)`
- `std::string write_setting (std::string label, QRgb a)`
- `std::string write_setting (std::string label, QString a)`
- `std::string write_setting (std::string label, float a)`
- `std::string write_setting (std::string label, bool a)`

18.13.1 Function Documentation

18.13.1.1 `read_configuration()` void read_configuration ()

18.13.1.2 `SettingsDir()` QString SettingsDir ()

Note: on Unix we include the trailing separator. For Windows compatibility we omit it.

18.13.1.3 `SettingsPath()` QString SettingsPath ()

18.13.1.4 `to_string_vector()` std::vector< std::string > to_string_vector (QStringList list)

18.13.1.5 `write_setting()` [1/5] std::string write_setting (std::string label, bool a)

18.13.1.6 write_setting() [2/5] std::string write_setting (std::string label, float a)

18.13.1.7 write_setting() [3/5] std::string write_setting (std::string label, int a)

18.13.1.8 write_setting() [4/5] std::string write_setting (std::string label, QRgb a)

18.13.1.9 write_setting() [5/5] std::string write_setting (std::string label, QString a)

18.14 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](#)

18.14.1 Function Documentation

18.14.1.1 get_action_index() int get_action_index (std::string cmd)

18.14.2 Variable Documentation

18.14.2.1 edit_toolbar std::vector<std::string> [edit_toolbar](#)

Initial value:

```
= {  
    "cut",  
    "copy",  
    "paste"  
}
```

18.14.2.2 file_toolbar std::vector<std::string> file_toolbar**Initial value:**

```
= {  
    "new",  
    "open",  
    "save",  
    "saveas",  
    "print",  
    "designdetails",  
    "---",  
    "undo",  
    "redo",  
    "---",  
    "help"  
}
```

18.14.2.3 help_toolbar std::vector<std::string> help_toolbar**Initial value:**

```
= {  
    "help",  
    "---",  
    "changelog",  
    "---",  
    "about",  
    "---",  
    "whatsthis"  
}
```

18.14.2.4 icon_toolbar std::vector<std::string> icon_toolbar**Initial value:**

```
= {  
    "icon16",  
    "icon24",  
    "icon32",  
    "icon48",  
    "icon64",  
    "icon128"  
}
```

18.14.2.5 pan_toolbar std::vector<std::string> pan_toolbar**Initial value:**

```
= {  
    "panrealtime",  
    "panpoint",  
    "---",  
    "panleft",  
    "panright",  
    "panup",  
    "pandown"  
}
```

18.14.2.6 view_toolbar std::vector<std::string> view_toolbar**Initial value:**

```
= {  
    "day",  
    "night"  
}
```

18.14.2.7 zoom_toolbar std::vector<std::string> zoom_toolbar**Initial value:**

```
= {  
    "zoomwindow",  
    "zoomdynamic",  
    "zoomscale",  
    "---",  
    "zoomcenter",  
    "zoomin",  
    "zoomout",  
    "---",  
    "zoomselected",  
}
```

```
    "zoomall",
    "zoomextents"
}
```

18.15 embroidermodder2/mainwindow.cpp File Reference

```
#include "embroidermodder.h"
#include <cerrno>
#include <iostream>
#include <fstream>
```

Classes

- struct [Parameter_](#)

TypeDefs

- typedef struct [Parameter_](#) [Parameter](#)

Functions

- std::vector< std::string > [tokenize](#) (std::string str, const char delim)
- std::string [read_string_setting](#) (toml_table_t *table, const char *key)
- int [read_settings](#) (const char *settings_file)

Read the settings from file which aren't editable by the user. These files need to be placed in the install folder.

- bool [validRGB](#) (int r, int g, int b)
- [MainWindow](#) * [mainWin](#) ()
mainWin
- std::string [convert_args_to_type](#) (std::string label, std::vector< std::string > args, const char *args_template, Parameter result[10])
Inspired by PyArg_ParseTupleAndKeywords allowing a uniform argument parsing framework.
 - std::string [Error](#) (Parameter args[10])
 - std::string [Todo](#) (Parameter result[10])

Variables

- [MainWindow](#) * [_mainWin](#) = 0
- std::vector< [Action](#) > [action_table](#)
- std::unordered_map< std::string, std::string > [config](#)

18.15.1 Typedef Documentation

18.15.1.1 [Parameter](#) [typedef struct Parameter_ Parameter](#)

18.15.2 Function Documentation

18.15.2.1 [convert_args_to_type\(\)](#) std::string [convert_args_to_type](#) (std::string label, std::vector< std::string > args, const char * args_template, Parameter result[10])

Inspired by PyArg_ParseTupleAndKeywords allowing a uniform argument parsing framework.

Parameters

<i>label</i>	The caller's name.
<i>args</i>	The list of strings passed from the user.
<i>args_template</i>	The string of characters describing the types of the output.
<i>result</i>	The fixed length array of results.

Returns

An error message if an error occurred or an empty string if it passes.

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

```
18.15.2.3 mainWin() MainWindow * mainWin ( )
mainWin
```

Returns

```
18.15.2.4 read_settings() int read_settings (
    const char * settings_file )
```

Read the settings from file which aren't editable by the user. These files need to be placed in the install folder.

```
18.15.2.5 read_string_setting() std::string read_string_setting (
    toml_table_t * table,
    const char * key )
```

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

```
18.15.2.7 tokenize() std::vector< std::string > tokenize (
    std::string str,
    const char delim )
```

```
18.15.2.8 validRGB() bool validRGB (
    int r,
    int g,
    int b )
```

18.15.3 Variable Documentation

```
18.15.3.1 _mainWin MainWindow* _mainWin = 0
```

18.15.3.2 `action_table` `std::vector<Action> action_table`

18.15.3.3 `config` `std::unordered_map<std::string, std::string> config`

18.16 `embroidermodder2/mdiarea.cpp` File Reference

```
#include "embroidermodder.h"
```

18.17 `embroidermodder2/mdiwindow.cpp` File Reference

```
#include "embroidermodder.h"
```

18.18 `embroidermodder2/object-arc.cpp` File Reference

```
#include "embroidermodder.h"
```

Functions

- `EmbVector rotate_vector (EmbVector v, EmbReal alpha)`

18.18.1 Function Documentation

18.18.1.1 `rotate_vector()` `EmbVector rotate_vector (`
 `EmbVector v,`
 `EmbReal alpha)`

Returns

18.19 `embroidermodder2/object-base.cpp` File Reference

```
#include "embroidermodder.h"
```

18.20 `embroidermodder2/object-circle.cpp` File Reference

```
#include "embroidermodder.h"
```

18.21 `embroidermodder2/object-dimleader.cpp` File Reference

```
#include "embroidermodder.h"
```

18.22 `embroidermodder2/object-ellipse.cpp` File Reference

```
#include "embroidermodder.h"
```

18.23 `embroidermodder2/object-image.cpp` File Reference

```
#include "embroidermodder.h"
```

18.24 **embroidermodder2/object-line.cpp File Reference**

```
#include "embroidermodder.h"
```

18.25 **embroidermodder2/object-path.cpp File Reference**

```
#include "embroidermodder.h"
```

18.26 **embroidermodder2/object-point.cpp File Reference**

```
#include "embroidermodder.h"
```

18.27 **embroidermodder2/object-polygon.cpp File Reference**

```
#include "embroidermodder.h"
```

18.28 **embroidermodder2/object-polyline.cpp File Reference**

```
#include "embroidermodder.h"
```

18.29 **embroidermodder2/object-rect.cpp File Reference**

```
#include "embroidermodder.h"
```

18.29.1 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/>

18.30 **embroidermodder2/object-save.cpp File Reference**

```
#include "embroidermodder.h"
```

18.31 **embroidermodder2/object-textsingle.cpp File Reference**

```
#include "embroidermodder.h"
```

18.32 **embroidermodder2/preview-dialog.cpp File Reference**

```
#include "embroidermodder.h"
```

18.33 **embroidermodder2/property-editor.cpp File Reference**

```
#include "embroidermodder.h"
```

Variables

- `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`
- `QToolButton * toolButtonArcClockwise`
- `QComboBox * comboBoxArcClockwise`
- `QToolButton * toolButtonBlockX`
- `QToolButton * toolButtonBlockY`
- `QLineEdit * lineEditBlockX`
- `QLineEdit * lineEditBlockY`
- `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 * 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 * `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`
- QComboBox * `comboBoxPathVertexNum`
- QLineEdit * `lineEditPathVertexX`
- QLineEdit * `lineEditPathVertexY`
- QLineEdit * `lineEditPathArea`
- QLineEdit * `lineEditPathLength`
- QComboBox * `comboBoxPathClosed`
- 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`
- 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`
- QComboBox * `comboBoxTextSingleJustify`
- QLineEdit * `lineEditTextSingleHeight`
- QLineEdit * `lineEditTextSingleRotation`
- QLineEdit * `lineEditTextSingleY`
- QComboBox * `comboBoxTextSingleBackward`
- QComboBox * `comboBoxTextSingleUpsideDown`

18.33.1 Variable Documentation

18.33.1.1 comboBoxArcClockwise `QComboBox* comboBoxArcClockwise`

18.33.1.2 comboBoxPathClosed `QComboBox* comboBoxPathClosed`

18.33.1.3 comboBoxPathVertexNum `QComboBox* comboBoxPathVertexNum`

18.33.1.4 comboBoxPolylineClosed `QComboBox* comboBoxPolylineClosed`

18.33.1.5 comboBoxPolylineVertexNum QComboBox* comboBoxPolylineVertexNum

18.33.1.6 comboBoxTextSingleBackward QComboBox* comboBoxTextSingleBackward

18.33.1.7 comboBoxTextSingleJustify QComboBox* comboBoxTextSingleJustify

18.33.1.8 comboBoxTextSingleUpsideDown QComboBox* comboBoxTextSingleUpsideDown

18.33.1.9 lineEditBlockX QLineEdit* lineEditBlockX

18.33.1.10 lineEditBlockY QLineEdit* lineEditBlockY

18.33.1.11 lineEditEllipseCenterX QLineEdit* lineEditEllipseCenterX

18.33.1.12 lineEditEllipseCenterY QLineEdit* lineEditEllipseCenterY

18.33.1.13 lineEditEllipseDiameterMajor QLineEdit* lineEditEllipseDiameterMajor

18.33.1.14 lineEditEllipseDiameterMinor QLineEdit* lineEditEllipseDiameterMinor

18.33.1.15 lineEditEllipseRadiusMajor QLineEdit* lineEditEllipseRadiusMajor

18.33.1.16 lineEditEllipseRadiusMinor QLineEdit* lineEditEllipseRadiusMinor

18.33.1.17 lineEditImageHeight QLineEdit* lineEditImageHeight

18.33.1.18 lineEditImageName QLineEdit* lineEditImageName

18.33.1.19 lineEditImagePath QLineEdit* lineEditImagePath

18.33.1.20 lineEditImageWidth QLineEdit* lineEditImageWidth

18.33.1.21 lineEditImageX QLineEdit* lineEditImageX

18.33.1.22 lineEditImageY QLineEdit* lineEditImageY

18.33.1.23 `lineEditInfiniteLineVectorX` `QLineEdit* lineEditInfiniteLineVectorX`

18.33.1.24 `lineEditInfiniteLineVectorY` `QLineEdit* lineEditInfiniteLineVectorY`

18.33.1.25 `lineEditInfiniteLineX2` `QLineEdit* lineEditInfiniteLineX2`

18.33.1.26 `lineEditInfiniteLineY1` `QLineEdit* lineEditInfiniteLineY1`

18.33.1.27 `lineEditInfiniteLineY2` `QLineEdit* lineEditInfiniteLineY2`

18.33.1.28 `lineEditLineAngle` `QLineEdit* lineEditLineAngle`

18.33.1.29 `lineEditLineDeltaX` `QLineEdit* lineEditLineDeltaX`

18.33.1.30 `lineEditLineDeltaY` `QLineEdit* lineEditLineDeltaY`

18.33.1.31 `lineEditLineEndX` `QLineEdit* lineEditLineEndX`

18.33.1.32 `lineEditLineEndY` `QLineEdit* lineEditLineEndY`

18.33.1.33 `lineEditLineLength` `QLineEdit* lineEditLineLength`

18.33.1.34 `lineEditLineStartX` `QLineEdit* lineEditLineStartX`

18.33.1.35 `lineEditLineStartY` `QLineEdit* lineEditLineStartY`

18.33.1.36 `lineEditPathArea` `QLineEdit* lineEditPathArea`

18.33.1.37 `lineEditPathLength` `QLineEdit* lineEditPathLength`

18.33.1.38 `lineEditPathVertexX` `QLineEdit* lineEditPathVertexX`

18.33.1.39 `lineEditPathVertexY` `QLineEdit* lineEditPathVertexY`

18.33.1.40 `lineEditPointX` `QLineEdit* lineEditPointX`

18.33.1.41 `lineEditPointY` `QLineEdit* lineEditPointY`

18.33.1.42 `lineEditPolygonCenterX` `QLineEdit* lineEditPolygonCenterX`

18.33.1.43 `lineEditPolygonCenterY` `QLineEdit* lineEditPolygonCenterY`

18.33.1.44 `lineEditPolygonDiameterSide` `QLineEdit* lineEditPolygonDiameterSide`

18.33.1.45 `lineEditPolygonDiameterVertex` `QLineEdit* lineEditPolygonDiameterVertex`

18.33.1.46 `lineEditPolygonInteriorAngle` `QLineEdit* lineEditPolygonInteriorAngle`

18.33.1.47 `lineEditPolygonRadiusSide` `QLineEdit* lineEditPolygonRadiusSide`

18.33.1.48 `lineEditPolygonRadiusVertex` `QLineEdit* lineEditPolygonRadiusVertex`

18.33.1.49 `lineEditPolylineArea` `QLineEdit* lineEditPolylineArea`

18.33.1.50 `lineEditPolylineLength` `QLineEdit* lineEditPolylineLength`

18.33.1.51 `lineEditPolylineVertexX` `QLineEdit* lineEditPolylineVertexX`

18.33.1.52 `lineEditPolylineVertexY` `QLineEdit* lineEditPolylineVertexY`

18.33.1.53 `lineEditRayVectorX` `QLineEdit* lineEditRayVectorX`

18.33.1.54 `lineEditRayVectorY` `QLineEdit* lineEditRayVectorY`

18.33.1.55 `lineEditRayX1` `QLineEdit* lineEditRayX1`

18.33.1.56 `lineEditRayX2` `QLineEdit* lineEditRayX2`

18.33.1.57 `lineEditRayY1` `QLineEdit* lineEditRayY1`

18.33.1.58 `lineEditRayY2` `QLineEdit* lineEditRayY2`

18.33.1.59 `lineEditRectangleArea` `QLineEdit* lineEditRectangleArea`

18.33.1.60 `lineEditRectangleCorner1X` `QLineEdit* lineEditRectangleCorner1X`

18.33.1.61 `lineEditRectangleCorner1Y` `QLineEdit* lineEditRectangleCorner1Y`

18.33.1.62 `lineEditRectangleCorner2X` `QLineEdit* lineEditRectangleCorner2X`

18.33.1.63 `lineEditRectangleCorner2Y` `QLineEdit* lineEditRectangleCorner2Y`

18.33.1.64 `lineEditRectangleCorner3X` `QLineEdit* lineEditRectangleCorner3X`

18.33.1.65 `lineEditRectangleCorner3Y` `QLineEdit* lineEditRectangleCorner3Y`

18.33.1.66 `lineEditRectangleCorner4X` `QLineEdit* lineEditRectangleCorner4X`

18.33.1.67 `lineEditRectangleCorner4Y` `QLineEdit* lineEditRectangleCorner4Y`

18.33.1.68 `lineEditRectangleHeight` `QLineEdit* lineEditRectangleHeight`

18.33.1.69 `lineEditRectangleWidth` `QLineEdit* lineEditRectangleWidth`

18.33.1.70 `lineEditTextMultiX` `QLineEdit* lineEditTextMultiX`

18.33.1.71 `lineEditTextMultiY` `QLineEdit* lineEditTextMultiY`

18.33.1.72 `lineEditTextSingleContents` `QLineEdit* lineEditTextSingleContents`

18.33.1.73 `lineEditTextSingleHeight` `QLineEdit* lineEditTextSingleHeight`

18.33.1.74 `lineEditTextSingleRotation` `QLineEdit* lineEditTextSingleRotation`

18.33.1.75 `lineEditTextSingleY` `QLineEdit* lineEditTextSingleY`

18.33.1.76 `tempArcObj` `ArcObject*` `tempArcObj`

18.33.1.77 tempBlockObj BlockObject* tempBlockObj

18.33.1.78 tempCircleObj CircleObject* tempCircleObj

18.33.1.79 tempDimAlignedObj DimAlignedObject* tempDimAlignedObj

18.33.1.80 tempDimAngularObj DimAngularObject* tempDimAngularObj

18.33.1.81 tempDimArcLenObj DimArcLengthObject* tempDimArcLenObj

18.33.1.82 tempDimDiamObj DimDiameterObject* tempDimDiamObj

18.33.1.83 tempDimLeaderObj DimLeaderObject* tempDimLeaderObj

18.33.1.84 tempDimLinearObj DimLinearObject* tempDimLinearObj

18.33.1.85 tempDimOrdObj DimOrdinateObject* tempDimOrdObj

18.33.1.86 tempDimRadiusObj DimRadiusObject* tempDimRadiusObj

18.33.1.87 tempEllipseArcObj EllipseArcObject* tempEllipseArcObj

18.33.1.88 tempEllipseObj EllipseObject* tempEllipseObj

18.33.1.89 tempHatchObj HatchObject* tempHatchObj

18.33.1.90 templImageObj ImageObject* tempImageObj

18.33.1.91 templInfLineObj InfiniteLineObject* tempInfLineObj

18.33.1.92 tempLineObj LineObject* tempLineObj

18.33.1.93 tempPathObj PathObject* tempPathObj

18.33.1.94 tempPointObj PointObject* tempPointObj

18.33.1.95 tempPolygonObj `PolygonObject*` `tempPolygonObj`

18.33.1.96 tempPolylineObj `PolylineObject*` `tempPolylineObj`

18.33.1.97 tempRayObj `RayObject*` `tempRayObj`

18.33.1.98 tempRectObj `RectObject*` `tempRectObj`

18.33.1.99 tempSplineObj `SplineObject*` `tempSplineObj`

18.33.1.100 tempTextMultiObj `TextMultiObject*` `tempTextMultiObj`

18.33.1.101 tempTextSingleObj `TextSingleObject*` `tempTextSingleObj`

18.33.1.102 toolButtonArcClockwise `QToolButton*` `toolButtonArcClockwise`

18.33.1.103 toolButtonBlockX `QToolButton*` `toolButtonBlockX`

18.33.1.104 toolButtonBlockY `QToolButton*` `toolButtonBlockY`

18.33.1.105 toolButtonImageHeight `QToolButton*` `toolButtonImageHeight`

18.33.1.106 toolButtonImageName `QToolButton*` `toolButtonImageName`

18.33.1.107 toolButtonImagePath `QToolButton*` `toolButtonImagePath`

18.33.1.108 toolButtonImageWidth `QToolButton*` `toolButtonImageWidth`

18.33.1.109 toolButtonImageX `QToolButton*` `toolButtonImageX`

18.33.1.110 toolButtonImageY `QToolButton*` `toolButtonImageY`

18.33.1.111 toolButtonInfiniteLineVectorX `QToolButton*` `toolButtonInfiniteLineVectorX`

18.33.1.112 toolButtonInfiniteLineVectorY `QToolButton*` `toolButtonInfiniteLineVectorY`

18.33.1.113 toolButtonInfiniteLineX2 QToolButton* toolButtonInfiniteLineX2

18.33.1.114 toolButtonInfiniteLineY2 QToolButton* toolButtonInfiniteLineY2

18.33.1.115 toolButtonLineAngle QToolButton* toolButtonLineAngle

18.33.1.116 toolButtonLineDeltaX QToolButton* toolButtonLineDeltaX

18.33.1.117 toolButtonLineDeltaY QToolButton* toolButtonLineDeltaY

18.33.1.118 toolButtonLineEndX QToolButton* toolButtonLineEndX

18.33.1.119 toolButtonLineEndY QToolButton* toolButtonLineEndY

18.33.1.120 toolButtonLineLength QToolButton* toolButtonLineLength

18.33.1.121 toolButtonLineStartX QToolButton* toolButtonLineStartX

18.33.1.122 toolButtonLineStartY QToolButton* toolButtonLineStartY

18.33.1.123 toolButtonPolygonDiameterSide QToolButton* toolButtonPolygonDiameterSide

18.33.1.124 toolButtonPolygonDiameterVertex QToolButton* toolButtonPolygonDiameterVertex

18.33.1.125 toolButtonPolygonInteriorAngle QToolButton* toolButtonPolygonInteriorAngle

18.33.1.126 toolButtonPolygonRadiusSide QToolButton* toolButtonPolygonRadiusSide

18.33.1.127 toolButtonPolygonRadiusVertex QToolButton* toolButtonPolygonRadiusVertex

18.33.1.128 toolButtonPolylineArea QToolButton* toolButtonPolylineArea

18.33.1.129 toolButtonPolylineClosed QToolButton* toolButtonPolylineClosed

18.33.1.130 toolButtonPolylineLength QToolButton* toolButtonPolylineLength

18.33.1.131 toolButtonPolylineVertexNum QToolButton* toolButtonPolylineVertexNum

18.33.1.132 toolButtonPolylineVertexX QToolButton* toolButtonPolylineVertexX

18.33.1.133 toolButtonPolylineVertexY QToolButton* toolButtonPolylineVertexY

18.33.1.134 toolButtonRayVectorX QToolButton* toolButtonRayVectorX

18.33.1.135 toolButtonRayVectorY QToolButton* toolButtonRayVectorY

18.33.1.136 toolButtonRayX1 QToolButton* toolButtonRayX1

18.33.1.137 toolButtonRayX2 QToolButton* toolButtonRayX2

18.33.1.138 toolButtonRayY1 QToolButton* toolButtonRayY1

18.33.1.139 toolButtonRayY2 QToolButton* toolButtonRayY2

18.33.1.140 toolButtonTextMultiX QToolButton* toolButtonTextMultiX

18.33.1.141 toolButtonTextMultiY QToolButton* toolButtonTextMultiY

18.33.1.142 toolButtonTextSingleContents QToolButton* toolButtonTextSingleContents

18.33.1.143 toolButtonTextSingleFont QToolButton* toolButtonTextSingleFont

18.33.1.144 toolButtonTextSingleHeight QToolButton* toolButtonTextSingleHeight

18.33.1.145 toolButtonTextSingleJustify QToolButton* toolButtonTextSingleJustify

18.33.1.146 toolButtonTextSingleRotation QToolButton* toolButtonTextSingleRotation

18.34 embroidermodder2/README.md File Reference

18.35 embroidermodder2/selectbox.cpp File Reference

#include "embroidermodder.h"

18.36 embroidermodder2/settings-dialog.cpp File Reference

```
#include "embroidermodder.h"
```

Variables

- `Settings settings`
- `Settings dialog`
- `Settings preview`
- `Settings accept_`
- `std::vector< std::string > extensions`

18.36.1 Variable Documentation

18.36.1.1 `accept_ Settings accept_`

18.36.1.2 `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.

18.36.1.3 `extensions std::vector<std::string> extensions`

Initial value:

```
= {
    "100", "10o", "ART", "BMC", "BRO",
    "CND", "COL", "CSD", "CSV", "DAT",
    "DEM", "DSB", "DST", "DSZ", "DXF",
    "EDR", "EMD", "EXP", "EXY", "EYS",
    "FXY", "GNC", "GT", "HUS", "INB",
    "JEF", "KSM", "PCD", "PCM", "PCO",
    "PCS", "PEC", "PEL", "PEM", "PES",
    "PHB", "PHC", "RGB", "SEW", "SHV",
    "SST", "STX", "SVG", "T09", "TAP",
    "THR", "TXT", "U00", "U01", "VIP",
    "VP3", "XXX", "ZSK"
}
```

18.36.1.4 `preview Settings preview`

18.36.1.5 `settings Settings settings`

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/> The actuator changes the program state via these global variables.

18.37 embroidermodder2/statusbar-button.cpp File Reference

```
#include "embroidermodder.h"
```

18.38 embroidermodder2/statusbar.cpp File Reference

```
#include "embroidermodder.h"
```

18.39 `embroidermodder2/undo-commands.cpp` File Reference

```
#include "embroidermodder.h"
```

18.40 `embroidermodder2/undo-editor.cpp` File Reference

```
#include "embroidermodder.h"
```

18.40.1 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/>

18.41 `embroidermodder2/view.cpp` File Reference

```
#include "embroidermodder.h"
```

18.41.1 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/>

18.42 `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)`

18.42.1 Function Documentation

```
18.42.1.1 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.

```
18.42.1.2 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.

```
18.42.1.3 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.

```
18.42.1.4 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.

```
18.42.1.5 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.

```
18.42.1.6 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.

```
18.42.1.7 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.

```
18.42.1.8 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.

```
18.42.1.9 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.

```
18.42.1.10 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.

```
18.42.1.11 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.

```
18.42.1.12 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.

```
18.42.1.13 embArray_copy() void embArray_copy (
    EmbArray * dst,
    EmbArray * src )
```

Copies all entries in the EmbArray struct from *src* to *dst*.

```
18.42.1.14 embArray_create() EmbArray * embArray_create (
    int type )
```

Allocates memory for an EmbArray of the type determined by the argument *type*.

```
18.42.1.15 embArray_free() void embArray_free (
    EmbArray * a )
```

Free the memory of EmbArray *a*, recursively if necessary.

```
18.42.1.16 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.

18.43 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]

18.43.1 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.

18.43.2 Function Documentation

18.43.2.1 compress_get_bits() `int compress_get_bits (`
 `compress * c,`
 `int length)`

c length Returns .

18.43.2.2 compress_get_position() `int compress_get_position (`
 `compress * c)`

c. Returns the position as an int.

18.43.2.3 compress_get_token() `int compress_get_token (`
 `compress * c)`

c. Returns the token as an int.

18.43.2.4 compress_init() `void compress_init ()`

18.43.2.5 compress_load_block() `void compress_load_block (`
 `compress * c)`

c. Returns nothing.

18.43.2.6 compress_load_character_huffman() `void compress_load_character_huffman (`
 `compress * c)`

Load character table to compress struct *c*. Returns nothing.

18.43.2.7 compress_load_character_length_huffman() `void compress_load_character_length_huffman (`
 `compress * c)`

c. Returns.

18.43.2.8 compress_load_distance_huffman() `void compress_load_distance_huffman (`
 `compress * c)`

c. Returns nothing.

18.43.2.9 compress_peek() `int compress_peek (`
 `compress * c,`
 `int bit_count)`

c bit_count. Returns.

18.43.2.10 compress_pop() `int compress_pop (`
 `compress * c,`
 `int bit_count)`

c bit_count. Returns.

```
18.43.2.11 compress_read_variable_length() int compress_read_variable_length (
    compress * c )
```

c. Returns.

```
18.43.2.12 huffman_build_table() void huffman_build_table (
    huffman * h )
```

These next 2 functions represent the [Huffman](#) class in tartarize's code. *h*

```
18.43.2.13 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*.

```
18.43.2.14 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).

```
18.43.2.15 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.

18.43.3 Variable Documentation

```
18.43.3.1 huffman_lookup_data int huffman_lookup_data[2]
```

18.44 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.*

18.44.1 Macro Definition Documentation

18.44.1.1 Arc_Polyester #define Arc_Polyester 0

18.44.1.2 Arc_Rayon #define Arc_Rayon 1

18.44.1.3 CHUNK_SIZE #define CHUNK_SIZE 128

18.44.1.4 CoatsAndClark_Rayon #define CoatsAndClark_Rayon 2

18.44.1.5 dxf_color #define dxf_color 29

18.44.1.6 EMB_ARC #define EMB_ARC 1

18.44.1.7 EMB_ARRAY #define EMB_ARRAY 0

18.44.1.8 EMB_CIRCLE #define EMB_CIRCLE 2

18.44.1.9 EMB_DIM_DIAMETER #define EMB_DIM_DIAMETER 3

18.44.1.10 EMB_DIM_LEADER #define EMB_DIM_LEADER 4

18.44.1.11 EMB_ELLIPSE #define EMB_ELLIPSE 5

18.44.1.12 EMB_FLAG #define EMB_FLAG 6

18.44.1.13 EMB_FORMAT_100 #define EMB_FORMAT_100 0
Format identifiers

18.44.1.14 EMB_FORMAT_10O #define EMB_FORMAT_10O 1

18.44.1.15 EMB_FORMAT_ART #define EMB_FORMAT_ART 2

18.44.1.16 EMB_FORMAT_BMC #define EMB_FORMAT_BMC 3

18.44.1.17 EMB_FORMAT_BRO #define EMB_FORMAT_BRO 4

18.44.1.18 EMB_FORMAT_CND #define EMB_FORMAT_CND 5

18.44.1.19 EMB_FORMAT_COL #define EMB_FORMAT_COL 6

18.44.1.20 EMB_FORMAT_CSD #define EMB_FORMAT_CSD 7

18.44.1.21 EMB_FORMAT_CSV #define EMB_FORMAT_CSV 8

18.44.1.22 EMB_FORMAT_DAT #define EMB_FORMAT_DAT 9

18.44.1.23 EMB_FORMAT_DEM #define EMB_FORMAT_DEM 10

18.44.1.24 EMB_FORMAT_DSB #define EMB_FORMAT_DSB 11

18.44.1.25 EMB_FORMAT_DST #define EMB_FORMAT_DST 12

18.44.1.26 EMB_FORMAT_DSZ #define EMB_FORMAT_DSZ 13

18.44.1.27 EMB_FORMAT_DXF #define EMB_FORMAT_DXF 14

18.44.1.28 EMB_FORMAT_EDR #define EMB_FORMAT_EDR 15

18.44.1.29 EMB_FORMAT_EMD #define EMB_FORMAT_EMD 16

18.44.1.30 EMB_FORMAT_EXP #define EMB_FORMAT_EXP 17

18.44.1.31 EMB_FORMAT_EXY #define EMB_FORMAT_EXY 18

18.44.1.32 EMB_FORMAT_EYS #define EMB_FORMAT_EYS 19

18.44.1.33 EMB_FORMAT_FXY #define EMB_FORMAT_FXY 20

18.44.1.34 EMB_FORMAT_GC #define EMB_FORMAT_GC 21

18.44.1.35 EMB_FORMAT_GNC #define EMB_FORMAT_GNC 22

18.44.1.36 EMB_FORMAT_GT #define EMB_FORMAT_GT 23

18.44.1.37 EMB_FORMAT_HUS #define EMB_FORMAT_HUS 24

18.44.1.38 EMB_FORMAT_INB #define EMB_FORMAT_INB 25

18.44.1.39 EMB_FORMAT_INF #define EMB_FORMAT_INF 26

18.44.1.40 EMB_FORMAT_JEF #define EMB_FORMAT_JEF 27

18.44.1.41 EMB_FORMAT_KSM #define EMB_FORMAT_KSM 28

18.44.1.42 EMB_FORMAT_MAX #define EMB_FORMAT_MAX 29

18.44.1.43 EMB_FORMAT_MIT #define EMB_FORMAT_MIT 30

18.44.1.44 EMB_FORMAT_NEW #define EMB_FORMAT_NEW 31

18.44.1.45 EMB_FORMAT_OFM #define EMB_FORMAT_OFM 32

18.44.1.46 EMB_FORMAT_PCD #define EMB_FORMAT_PCD 33

18.44.1.47 EMB_FORMAT_PCM #define EMB_FORMAT_PCM 34

18.44.1.48 EMB_FORMAT_PCQ #define EMB_FORMAT_PCQ 35

18.44.1.49 EMB_FORMAT_PCS #define EMB_FORMAT_PCS 36

18.44.1.50 EMB_FORMAT_PEC #define EMB_FORMAT_PEC 37

18.44.1.51 EMB_FORMAT_PEL #define EMB_FORMAT_PEL 38

18.44.1.52 EMB_FORMAT_PEM #define EMB_FORMAT_PEM 39

18.44.1.53 EMB_FORMAT_PES #define EMB_FORMAT_PES 40

18.44.1.54 EMB_FORMAT_PHB #define EMB_FORMAT_PHB 41

18.44.1.55 EMB_FORMAT_PHC #define EMB_FORMAT_PHC 42

18.44.1.56 EMB_FORMAT_PLT #define EMB_FORMAT_PLT 43

18.44.1.57 EMB_FORMAT_RGB #define EMB_FORMAT_RGB 44

18.44.1.58 EMB_FORMAT_SEW #define EMB_FORMAT_SEW 45

18.44.1.59 EMB_FORMAT_SHV #define EMB_FORMAT_SHV 46

18.44.1.60 EMB_FORMAT_SST #define EMB_FORMAT_SST 47

18.44.1.61 EMB_FORMAT_STX #define EMB_FORMAT_STX 48

18.44.1.62 EMB_FORMAT_SVG #define EMB_FORMAT_SVG 49

18.44.1.63 EMB_FORMAT_T01 #define EMB_FORMAT_T01 50

18.44.1.64 EMB_FORMAT_T09 #define EMB_FORMAT_T09 51

18.44.1.65 EMB_FORMAT_TAP #define EMB_FORMAT_TAP 52

18.44.1.66 EMB_FORMAT_THR #define EMB_FORMAT_THR 53

18.44.1.67 EMB_FORMAT_TXT #define EMB_FORMAT_TXT 54

18.44.1.68 EMB_FORMAT_U00 #define EMB_FORMAT_U00 55

18.44.1.69 EMB_FORMAT_U01 #define EMB_FORMAT_U01 56

18.44.1.70 EMB_FORMAT_VIP #define EMB_FORMAT_VIP 57

18.44.1.71 EMB_FORMAT_VP3 #define EMB_FORMAT_VP3 58

18.44.1.72 EMB_FORMAT_XXX #define EMB_FORMAT_XXX 59

18.44.1.73 EMB_FORMAT_ZSK #define EMB_FORMAT_ZSK 60

18.44.1.74 EMB_IMAGE #define EMB_IMAGE 8

18.44.1.75 EMB_LINE #define EMB_LINE 7

18.44.1.76 EMB_MAX_LAYERS #define EMB_MAX_LAYERS 10

18.44.1.77 EMB_PATH #define EMB_PATH 9

18.44.1.79 EMB_POLYGON #define EMB_POLYGON 11

18.44.1.80 EMB_POLYLINE #define EMB_POLYLINE 12

18.44.1.81 EMB_PUBLIC #define EMB_PUBLIC

18.44.1.82 EMB_RECT #define EMB_RECT 13

18.44.1.83 EMB_SPLINE #define EMB_SPLINE 14

18.44.1.84 EMB_STITCH #define EMB_STITCH 15

18.44.1.85 EMB_TEXT_MULTI #define EMB_TEXT_MULTI 17

18.44.1.86 EMB_TEXT_SINGLE #define EMB_TEXT_SINGLE 16

18.44.1.87 EMB_THREAD #define EMB_THREAD 19

18.44.1.88 EMB_VECTOR #define EMB_VECTOR 18

18.44.1.89 EMBFORMAT_MAXDESC #define EMBFORMAT_MAXDESC 50

18.44.1.90 EMBFORMAT_MAXEXT #define EMBFORMAT_MAXEXT 3

18.44.1.91 EMBFORMAT_OBJECTONLY #define EMBFORMAT_OBJECTONLY 2

18.44.1.92 EMBFORMAT_STCHANDOBJ #define EMBFORMAT_STCHANDOBJ 3 /* binary operation←
: 1+2=3 */

18.44.1.93 EMBFORMAT_STITCHONLY #define EMBFORMAT_STITCHONLY 1

18.44.1.94 EMBFORMAT_UNSUPPORTED #define EMBFORMAT_UNSUPPORTED 0

18.44.1.95 END #define END 16 /*! end of program */

18.44.1.96 Exquisite_Polyester #define Exquisite_Polyester 3

18.44.1.97 Fufu_Polyester #define Fufu_Polyester 4

18.44.1.98 Fufu_Rayon #define Fufu_Rayon 5

18.44.1.99 Hemingworth_Polyester #define Hemingworth_Polyester 6

18.44.1.100 hus_thread #define hus_thread 24

18.44.1.101 Isacord_Polyester #define Isacord_Polyester 7

18.44.1.102 Isafil_Rayon #define Isafil_Rayon 8

18.44.1.103 jef_thread #define jef_thread 25

18.44.1.104 JUMP #define JUMP 1 /*! move to (x, y) */

18.44.1.105 LIBEMBROIDERY_EMBEDDED_VERSION #define LIBEMBROIDERY_EMBEDDED_VERSION 0

18.44.1.106 Madeira_Polyester #define Madeira_Polyester 11

18.44.1.107 Madeira_Rayon #define Madeira_Rayon 12

18.44.1.108 Marathon_Polyester #define Marathon_Polyester 9

18.44.1.109 Marathon_Rayon #define Marathon_Rayon 10

18.44.1.110 MAX_STITCHES #define MAX_STITCHES 1000000

18.44.1.111 MAX_THREADS #define MAX_THREADS 256

18.44.1.112 Metro_Polyester #define Metro_Polyester 13

18.44.1.113 NORMAL #define NORMAL 0 /*! stitch to (x, y) */
Machine codes for stitch flags

18.44.1.114 numberOfFormats #define numberOfFormats 61

18.44.1.115 Pantone #define Pantone 14

18.44.1.116 pcm_thread #define pcm_thread 26

18.44.1.117 pec_thread #define pec_thread 27

18.44.1.118 RobisonAnton_Polyester #define RobisonAnton_Polyester 15

18.44.1.119 RobisonAnton_Rayon #define RobisonAnton_Rayon 16

18.44.1.120 SEQUIN #define SEQUIN 8 /*! sequin */

18.44.1.121 shv_thread #define shv_thread 28

18.44.1.122 Sigma_Polyester #define Sigma_Polyester 17

18.44.1.123 STOP #define STOP 4 /*! pause machine for thread change */

18.44.1.124 Sulky_Rayon #define Sulky_Rayon 18

18.44.1.125 SVG_Colors #define SVG_Colors 23

18.44.1.126 ThreadArt_Polyester #define ThreadArt_Polyester 20

18.44.1.127 ThreadArt_Rayon #define ThreadArt_Rayon 19

18.44.1.128 ThreaDelight_Polyester #define ThreaDelight_Polyester 21

18.44.1.129 TRIM #define TRIM 2 /*! trim + move to (x, y) */

18.44.1.130 Z102_Isacord_Polyester #define Z102_Isacord_Polyester 22

18.44.2 Typedef Documentation

18.44.2.1 EmbAlignedDim typedef struct EmbAlignedDim_ EmbAlignedDim

18.44.2.2 EmbAngularDim typedef struct EmbAngularDim_ EmbAngularDim

18.44.2.3 EmbArc typedef struct EmbArc_ EmbArc
absolute position (not relative)

18.44.2.4 EmbArcLengthDim typedef struct EmbArcLengthDim_ EmbArcLengthDim

18.44.2.5 EmbArray typedef struct EmbArray_ EmbArray
The basic array type.

18.44.2.6 EmbBezier typedef struct EmbBezier_ EmbBezier

18.44.2.7 EmbBlock typedef struct EmbBlock_ EmbBlock

18.44.2.8 EmbCircle typedef struct EmbCircle_ EmbCircle

18.44.2.9 EmbColor typedef struct EmbColor_ EmbColor
EmbColor uses the light primaries: red, green, blue in that order.

18.44.2.10 EmbDiameterDim typedef struct EmbDiameterDim_ EmbDiameterDim

18.44.2.11 EmbEllipse typedef struct EmbEllipse_ EmbEllipse

18.44.2.12 EmbFlag typedef int EmbFlag

18.44.2.13 EmbFormatList typedef struct EmbFormatList_ EmbFormatList

18.44.2.14 EmbGeometry `typedef struct EmbGeometry_ EmbGeometry`

18.44.2.15 EmbImage `typedef struct EmbImage_ EmbImage`

18.44.2.16 EmbInfiniteLine `typedef struct EmbInfiniteLine_ EmbInfiniteLine`

18.44.2.17 EmbLayer `typedef struct EmbLayer_ EmbLayer`

18.44.2.18 EmbLeaderDim `typedef struct EmbLeaderDim_ EmbLeaderDim`

18.44.2.19 EmbLine `typedef struct EmbLine_ EmbLine`

18.44.2.20 EmbLinearDim `typedef struct EmbLinearDim_ EmbLinearDim`

18.44.2.21 EmbOrdinateDim `typedef struct EmbOrdinateDim_ EmbOrdinateDim`

18.44.2.22 EmbPath `typedef struct EmbPath_ EmbPath`

18.44.2.23 EmbPattern `typedef struct EmbPattern_ EmbPattern`

18.44.2.24 EmbPoint `typedef struct EmbPoint_ EmbPoint`

18.44.2.25 EmbPolygon `typedef EmbPath EmbPolygon`

18.44.2.26 EmbPolyline `typedef EmbPath EmbPolyline`

18.44.2.27 EmbRadiusDim `typedef struct EmbRadiusDim_ EmbRadiusDim`

18.44.2.28 EmbRay `typedef struct EmbRay_ EmbRay`

18.44.2.29 EmbReal `typedef float EmbReal`

18.44.2.30 EmbRect `typedef struct EmbRect_ EmbRect`

18.44.2.31 EmbSatinOutline `typedef struct EmbSatinOutline_ EmbSatinOutline`

18.44.2.32 EmbSpline `typedef struct EmbSpline_ EmbSpline`

18.44.2.33 EmbStitch `typedef struct EmbStitch_ EmbStitch`

18.44.2.34 EmbTextMulti `typedef struct EmbTextMulti_ EmbTextMulti`

18.44.2.35 EmbTextSingle `typedef struct EmbTextSingle_ EmbTextSingle`

18.44.2.36 EmbThread `typedef struct EmbThread_ EmbThread`

18.44.2.37 EmbTime `typedef struct EmbTime_ EmbTime`

18.44.2.38 EmbVector `typedef struct EmbVector_ EmbVector`

The basic type to represent points absolutely or represent directions.

Positive y is up, units are in mm.

18.44.2.39 L_system `typedef struct LSYSTEM L_system`

18.44.2.40 thread_color `typedef struct thread_color_ thread_color`

18.44.3 Function Documentation

18.44.3.1 convert() `EMB_PUBLIC int convert (`
 `const char * inf,`
 `const char * outf)`

18.44.3.2 degrees() `EMB_PUBLIC EmbReal degrees (`
 `EmbReal radian) [inline]`

18.44.3.3 emb_identify_format() `EMB_PUBLIC int emb_identify_format (`
 `const char * fileName)`

Returns

`int`

18.44.3.4 emb_round() `EMB_PUBLIC int emb_round (`
 `EmbReal x)`

18.44.3.5 embArc_clockwise() `EMB_PUBLIC char embArc_clockwise (`
 `EmbArc arc)`

18.44.3.6 `embArc_init()` `EMB_PUBLIC EmbArc embArc_init (`
 `void)`

18.44.3.7 `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.

18.44.3.8 `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.

18.44.3.9 `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.

18.44.3.10 `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.

18.44.3.11 `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.

18.44.3.12 `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.

18.44.3.13 `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.

18.44.3.14 `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.

18.44.3.15 `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.

18.44.3.16 `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.

18.44.3.17 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.

18.44.3.18 embArray_addThread() `EMB_PUBLIC int embArray_addThread (EmbArray * g, EmbThread p)`

18.44.3.19 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.

18.44.3.20 embArray_copy() `EMB_PUBLIC void embArray_copy (EmbArray * dst, EmbArray * src)`

Copies all entries in the EmbArray struct from *src* to *dst*.

18.44.3.21 embArray_create() `EMB_PUBLIC EmbArray * embArray_create (int type)`

Allocates memory for an EmbArray of the type determined by the argument *type*.

18.44.3.22 embArray_free() `EMB_PUBLIC void embArray_free (EmbArray * a)`

Free the memory of EmbArray *a*, recursively if necessary.

18.44.3.23 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.

18.44.3.24 embCircle_init() `EMB_PUBLIC EmbCircle embCircle_init (void)`

18.44.3.25 embColor_create() `EMB_PUBLIC EmbColor * embColor_create (unsigned char r, unsigned char g, unsigned char b)`

18.44.3.26 embColor_distance() `EMB_PUBLIC int embColor_distance (EmbColor a, EmbColor b)`

a b

Returns

`int`

18.44.3.27 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.

18.44.3.28 embColor_make() `EMB_PUBLIC EmbColor embColor_make (`
 `unsigned char r,`
 `unsigned char g,`
 `unsigned char b)`

18.44.3.29 embEllipse_area() `EMB_PUBLIC EmbReal embEllipse_area (`
 `EmbEllipse ellipse)`

18.44.3.30 embEllipse_diameterX() `EMB_PUBLIC EmbReal embEllipse_diameterX (`
 `EmbEllipse ellipse)`

18.44.3.31 embEllipse_diameterY() `EMB_PUBLIC EmbReal embEllipse_diameterY (`
 `EmbEllipse ellipse)`

18.44.3.32 embEllipse_height() `EMB_PUBLIC EmbReal embEllipse_height (`
 `EmbEllipse ellipse)`

18.44.3.33 embEllipse_init() `EMB_PUBLIC EmbEllipse embEllipse_init (`
 `void)`

18.44.3.34 embEllipse_make() `EMB_PUBLIC EmbEllipse embEllipse_make (`
 `EmbReal cx,`
 `EmbReal cy,`
 `EmbReal rx,`
 `EmbReal ry)`

18.44.3.35 embEllipse_perimeter() `EMB_PUBLIC EmbReal embEllipse_perimeter (`
 `EmbEllipse ellipse)`

18.44.3.36 embEllipse_width() `EMB_PUBLIC EmbReal embEllipse_width (`
 `EmbEllipse ellipse)`

18.44.3.37 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).

18.44.3.38 embGeometry_free() `EMB_PUBLIC void embGeometry_free (EmbGeometry * obj)`

Free the memory occupied by a non-stitch geometry object.
obj Pointer to geometry memory.

18.44.3.39 embGeometry_init() `EMB_PUBLIC EmbGeometry * embGeometry_init (int type_in)`

Our generic object interface backends to each individual type.
type_in

Returns

`EmbGeometry*`

18.44.3.40 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.

18.44.3.41 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.

18.44.3.42 embImage_create() `EMB_PUBLIC EmbImage embImage_create (int , int)`

18.44.3.43 embImage_free() `EMB_PUBLIC void embImage_free (EmbImage * image)`

18.44.3.44 embImage_read() `EMB_PUBLIC void embImage_read (EmbImage * image, char * fname)`

18.44.3.45 embImage_write() `EMB_PUBLIC int embImage_write (EmbImage * image, char * fname)`

18.44.3.46 embLine_intersectionPoint() `EMB_PUBLIC EmbVector embLine_intersectionPoint (EmbLine line1, EmbLine line2)`

18.44.3.47 `embLine_make()` `EMB_PUBLIC EmbLine embLine_make (`
 `EmbReal x1,`
 `EmbReal y1,`
 `EmbReal x2,`
 `EmbReal y2)`

18.44.3.48 `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)

18.44.3.49 `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.

18.44.3.50 `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.

18.44.3.51 `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.

18.44.3.52 `embPattern_addPathAbs()` `EMB_PUBLIC void embPattern_addPathAbs (`
 `EmbPattern * p,`
 `EmbPath obj)`

18.44.3.53 `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.

18.44.3.54 `embPattern_addPolygonAbs()` `EMB_PUBLIC void embPattern_addPolygonAbs (`
 `EmbPattern * p,`
 `EmbPolygon obj)`

18.44.3.55 `embPattern_addPolylineAbs()` `EMB_PUBLIC void embPattern_addPolylineAbs (`
 `EmbPattern * p,`
 `EmbPolyline obj)`

18.44.3.56 `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.

```
18.44.3.57 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.

```
18.44.3.58 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.

```
18.44.3.59 embPattern_addThread() EMB_PUBLIC int embPattern_addThread (
    EmbPattern * pattern,
    EmbThread thread )
```

pattern thread

Returns

int

```
18.44.3.60 embPattern_calcBoundingBox() EMB_PUBLIC EmbRect embPattern_calcBoundingBox (
    EmbPattern * p )
```

Returns an EmbRect that encapsulates all stitches and objects in the pattern (*p*).

```
18.44.3.61 embPattern_center() EMB_PUBLIC void embPattern_center (
    EmbPattern * p )
```

Center the pattern *p*.

```
18.44.3.62 embPattern_changeColor() EMB_PUBLIC void embPattern_changeColor (
    EmbPattern * p,
    int index )
```

Change the currentColorIndex of pattern *p* to *index*.

```
18.44.3.63 embPattern_color_count() EMB_PUBLIC int embPattern_color_count (
    EmbPattern * pattern,
    EmbColor startColor )
```

```
18.44.3.64 embPattern_combine() EMB_PUBLIC EmbPattern * embPattern_combine (
    EmbPattern * p1,
    EmbPattern * p2 )
```

p1 p2

Returns

EmbPattern*

```
18.44.3.65 embPattern_combineJumpStitches() EMB_PUBLIC void embPattern_combineJumpStitches (
    EmbPattern * p )
```

p

```
18.44.3.66 embPattern_convertGeometry() EMB_PUBLIC void embPattern_convertGeometry (
    EmbPattern * p )
p

18.44.3.67 embPattern_copyPolylinesToStitchList() EMB_PUBLIC void embPattern_copyPolylinesTo←
StitchList (
    EmbPattern * pattern )

18.44.3.68 embPattern_copyStitchListToPolylines() EMB_PUBLIC void embPattern_copyStitchListTo←
Polylines (
    EmbPattern * pattern )

18.44.3.69 embPattern_correctForMaxStitchLength() EMB_PUBLIC void embPattern_correctForMax←
StitchLength (
    EmbPattern * p,
    EmbReal maxStitchLength,
    EmbReal maxJumpLength )

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



---

18.44.3.70 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*


```
18.44.3.71 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.
```

18.44.3.72 embPattern_designDetails() EMB_PUBLIC void embPattern_designDetails (
 EmbPattern * p)


```
18.44.3.73 embPattern_end() EMB_PUBLIC void embPattern_end (
    EmbPattern * p )

```



```
18.44.3.74 embPattern_fixColorCount() EMB_PUBLIC void embPattern_fixColorCount (
    EmbPattern * p )
p
```

```
18.44.3.75 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.

```
18.44.3.76 embPattern_flipHorizontal() EMB_PUBLIC void embPattern_flipHorizontal (
    EmbPattern * p )
```

Flips the entire pattern (*p*) horizontally about the y-axis.

```
18.44.3.77 embPattern_flipVertical() EMB_PUBLIC void embPattern_flipVertical (
    EmbPattern * p )
```

Flips the entire pattern (*p*) vertically about the x-axis.

```
18.44.3.78 embPattern_free() EMB_PUBLIC void embPattern_free (
    EmbPattern * p )
```

Frees all memory allocated in the pattern (*p*).

```
18.44.3.79 embPattern_hideStitchesOverLength() EMB_PUBLIC void embPattern_hideStitchesOver←
Length (
    EmbPattern * p,
    int length )
```

p length

```
18.44.3.80 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) x (-30, 30).

Use render then image difference to see how well it passes.

```
18.44.3.81 embPattern_jumpStitches() EMB_PUBLIC int embPattern_jumpStitches (
    EmbPattern * pattern )
```

```
18.44.3.82 embPattern_lengthHistogram() EMB_PUBLIC void embPattern_lengthHistogram (
    EmbPattern * pattern,
    int * bin,
    int NUMBINS )
```

```
18.44.3.83 embPattern_loadExternalColorFile() EMB_PUBLIC void embPattern_loadExternalColorFile (
    EmbPattern * p,
    const char * fileName )
```

TODO: Description needed.

```
18.44.3.84 embPattern_maximumStitchLength() EMB_PUBLIC EmbReal embPattern_maximumStitchLength
(
    EmbPattern * pattern )
```

18.44.3.85 embPattern_minimumStitchLength() `EMB_PUBLIC EmbReal embPattern_minimumStitchLength (EmbPattern * pattern)`

18.44.3.86 embPattern_movePolylinesToStitchList() `EMB_PUBLIC void embPattern_movePolylinesToStitchList (EmbPattern * pattern)`

18.44.3.87 embPattern_moveStitchListToPolylines() `EMB_PUBLIC void embPattern_moveStitchListToPolylines (EmbPattern * pattern)`

18.44.3.88 embPattern_read() `EMB_PUBLIC char embPattern_read (EmbPattern * pattern, const char * fileName, int format)`

pattern fileName format

Returns

`char`

18.44.3.89 embPattern_readAuto() `EMB_PUBLIC char embPattern_readAuto (EmbPattern * pattern, const char * fileName)`

pattern fileName

Returns

`char`

18.44.3.90 embPattern_realStitches() `EMB_PUBLIC int embPattern_realStitches (EmbPattern * pattern)`

18.44.3.91 embPattern_render() `EMB_PUBLIC int embPattern_render (EmbPattern * pattern, char * fname)`

18.44.3.92 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.

18.44.3.93 embPattern_simulate() `EMB_PUBLIC int embPattern_simulate (EmbPattern * pattern, char * fname)`

18.44.3.94 embPattern_totalStitchLength() `EMB_PUBLIC EmbReal embPattern_totalStitchLength (`
`EmbPattern * pattern)`

pattern

Returns

`float`

18.44.3.95 embPattern_trimStitches() `EMB_PUBLIC int embPattern_trimStitches (`
`EmbPattern * pattern)`

18.44.3.96 embPattern_write() `EMB_PUBLIC char embPattern_write (`
`EmbPattern * pattern,`
`const char * fileName,`
`int format)`

pattern fileName format

Returns

`char`

18.44.3.97 embPattern_writeAuto() `EMB_PUBLIC char embPattern_writeAuto (`
`EmbPattern * pattern,`
`const char * fileName)`

pattern fileName

Returns

`char`

18.44.3.98 embRect_area() `EMB_PUBLIC EmbReal embRect_area (`
`EmbRect rect)`

18.44.3.99 embRect_init() `EMB_PUBLIC EmbRect embRect_init (`
`void)`

18.44.3.100 embSatinOutline_generateSatinOutline() `EMB_PUBLIC void embSatinOutline_generate←`
`SatinOutline (`
`EmbArray * lines,`
`EmbReal thickness,`
`EmbSatinOutline * result)`

lines thickness result

18.44.3.101 embSatinOutline_renderStitches() `EMB_PUBLIC EmbArray * embSatinOutline_render←`
`Stitches (`
`EmbSatinOutline * result,`
`EmbReal density)`

result density

Returns

`EmbArray*`

```
18.44.3.102 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.

```
18.44.3.103 embThread_findNearestThread() EMB_PUBLIC int embThread_findNearestThread (
    EmbColor color,
    EmbThread * thread_list,
    int n_threads )
```

color *thread_list* *n_threads*

Returns

int

```
18.44.3.104 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.

```
18.44.3.105 embTime_initNow() EMB_PUBLIC void embTime_initNow (
    EmbTime * t )
```

t

```
18.44.3.106 embTime_time() EMB_PUBLIC EmbTime embTime_time (
    EmbTime * t )
```

t

Returns

EmbTime

```
18.44.3.107 embVector_add() EMB_PUBLIC EmbVector embVector_add (
    EmbVector a,
    EmbVector b )
```

The sum of vectors *a* and *b* returned as a vector.

Equivalent to:

$$\mathbf{c} = \mathbf{a} + \mathbf{b} = \begin{pmatrix} a_x + b_x \\ a_y + b_y \end{pmatrix}$$

18.44.3.108 embVector_angle() `EMB_PUBLIC EmbReal embVector_angle (EmbVector v)`

The angle, measured anti-clockwise from the x-axis, of a vector *v*.

18.44.3.109 embVector_average() `EMB_PUBLIC EmbVector embVector_average (EmbVector a, EmbVector b)`

The average of vectors *v1* and *v2* returned as a vector.

Equivalent to:

$$\mathbf{c} = \frac{\mathbf{a} + \mathbf{b}}{2} = \begin{pmatrix} \frac{a_x+b_x}{2} \\ \frac{a_y+b_y}{2} \end{pmatrix}$$

18.44.3.110 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:

$$|c| = \left| \begin{pmatrix} a_x \\ a_y \\ 0 \end{pmatrix} \times \begin{pmatrix} b_x \\ b_y \\ 0 \end{pmatrix} \right| = \left| \begin{pmatrix} 0 \\ 0 \\ a_x b_y - a_y b_x \end{pmatrix} \right| = a_x b_y - a_y b_x$$

18.44.3.111 embVector_distance() `EMB_PUBLIC EmbReal embVector_distance (EmbVector a, EmbVector b)`

The distance between *a* and *b* returned as a real value.

$$d = |\mathbf{a} - \mathbf{b}| = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$$

18.44.3.112 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:

$$c = \mathbf{a} \cdot \mathbf{b} = a_x b_x + a_y b_y$$

18.44.3.113 embVector_length() `EMB_PUBLIC EmbReal embVector_length (EmbVector vector)`

The length or absolute value of the vector *vector*.

Equivalent to:

$$|v| = \sqrt{v_x^2 + v_y^2}$$

18.44.3.114 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.

18.44.3.115 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.

18.44.3.116 embVector_relativeX() `EMB_PUBLIC EmbReal embVector_relativeX (EmbVector a1, EmbVector a2, EmbVector a3)`

The x-component of the vector

18.44.3.117 embVector_relativeY() `EMB_PUBLIC EmbReal embVector_relativeY (EmbVector a1, EmbVector a2, EmbVector a3)`

The y-component of the vector

18.44.3.118 embVector_subtract() `EMB_PUBLIC EmbVector embVector_subtract (EmbVector v1, EmbVector v2)`

The difference between vectors *v1* and *v2* returned as *result*.

Equivalent to:

$$\mathbf{c} = \mathbf{a} - \mathbf{b} = \begin{pmatrix} a_x - b_x \\ a_y - b_y \end{pmatrix}$$

18.44.3.119 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 $(1 \ 0) \cdot (a) = (x_a \ y_a) \cdot (0 \ 1) \cdot (b) = (y_b)$

18.44.3.120 embVector_unit() `EMB_PUBLIC EmbVector embVector_unit (EmbReal alpha)`

The unit vector in the direction *angle*.

$$\mathbf{a}_\alpha = \begin{pmatrix} \cos(\alpha) \\ \sin(\alpha) \end{pmatrix}$$

18.44.3.121 full_test_matrix() `EMB_PUBLIC int full_test_matrix (char * fname)`

18.44.3.122 getArcCenter() `EMB_PUBLIC void getArcCenter (`
`EmbArc arc,`
`EmbVector * arcCenter)`

18.44.3.123 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)`

18.44.3.124 getCircleCircleIntersections() `EMB_PUBLIC int getCircleCircleIntersections (`
`EmbCircle c0,`
`EmbCircle c1,`
`EmbVector * v0,`
`EmbVector * v1)`

18.44.3.125 getCircleTangentPoints() `EMB_PUBLIC int getCircleTangentPoints (`
`EmbCircle c,`
`EmbVector p,`
`EmbVector * v0,`
`EmbVector * v1)`

18.44.3.126 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.

18.44.3.127 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.

18.44.3.128 radians() `EMB_PUBLIC EmbReal radians (`
`EmbReal degrees) [inline]`

18.44.3.129 report() `EMB_PUBLIC void report (`
 `int result,`
 `char * label)`

18.44.3.130 testMain() `EMB_PUBLIC void testMain (`
 `int level)`

18.44.3.131 threadColor() `EMB_PUBLIC int threadColor (`
 `const char * name,`
 `int brand)`

18.44.3.132 threadColorName() `EMB_PUBLIC const char * threadColorName (`
 `unsigned int color,`
 `int brand)`

18.44.3.133 threadColorNum() `EMB_PUBLIC int threadColorNum (`
 `unsigned int color,`
 `int brand)`

18.44.4 Variable Documentation

18.44.4.1 _dxfColorTable `const unsigned char _dxfColorTable[][3] [extern]`

18.44.4.2 black_thread `EmbThread black_thread [extern]`

18.44.4.3 emb_error `int emb_error [extern]`
Error code storage for optional control flow blocking.

18.44.4.4 emb_verbose `int emb_verbose [extern]`
Verbosity level.

18.44.4.5 embConstantPi `const EmbReal embConstantPi [extern]`

18.44.4.6 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). *

18.44.4.7 husThreads `const EmbThread husThreads[] [extern]`

18.44.4.8 jefThreads const EmbThread jefThreads[] [extern]

18.44.4.9 pcmThreads const EmbThread pcmThreads[] [extern]

18.44.4.10 pecThreadCount const int pecThreadCount [extern]

18.44.4.11 pecThreads const EmbThread pecThreads[] [extern]

18.44.4.12 shvThreadCount const int shvThreadCount [extern]

18.44.4.13 shvThreads const EmbThread shvThreads[] [extern]

18.44.4.14 vipDecodingTable const unsigned char vipDecodingTable[] [extern]

18.44.4.15 Embroidery Format (.pcq) The Pfaff vip format is stitch-only.

18.45 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;
00392 } EmbStitch;
00393
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;
```

```

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[EML_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 */
00804
00805 #endif /* LIBEMBROIDERY_HEADER */
00806
00807 #endif /* LIBEMBROIDERY_HEADER */
00808

```

18.46 `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]`

18.46.1 Macro Definition Documentation

18.46.1.1 BULGETOCONTROL #define BULGETOCONTROL 2

18.46.1.2 BULGETOEND #define BULGETOEND 4

18.46.1.3 CompoundFileSector_DIFAT_Sector #define CompoundFileSector_DIFAT_Sector 0xFFFFFFFFFC

18.46.1.4 CompoundFileSector_EndOfChain #define CompoundFileSector_EndOfChain 0xFFFFFFFFFE

18.46.1.5 CompoundFileSector_FAT_Sector #define CompoundFileSector_FAT_Sector 0xFFFFFFFFFD

18.46.1.6 CompoundFileSector_FreeSector #define CompoundFileSector_FreeSector 0xFFFFFFFFFF

18.46.1.7 CompoundFileSector_MaxRegSector #define CompoundFileSector_MaxRegSector 0xFFFFFFFFFA
Type of sector

18.46.1.8 CompoundFileStreamId_MaxRegularStreamId #define CompoundFileStreamId_MaxRegular←
StreamId 0xFFFFFFFFFA
Special values for Stream Identifiers All real stream Ids are less than this

18.46.1.9 CompoundFileStreamId_NoStream #define CompoundFileStreamId_NoStream 0xFFFFFFFFFF
There is no valid stream Id

18.46.1.10 CUBICTOCONTROL1 #define CUBICTOCONTROL1 32

18.46.1.11 CUBICTOCONTROL2 #define CUBICTOCONTROL2 64

18.46.1.12 CUBICTOEND #define CUBICTOEND 128

18.46.1.13 DXF_VERSION_2000 #define DXF_VERSION_2000 "AC1015"

18.46.1.14 DXF_VERSION_2002 #define DXF_VERSION_2002 "AC1015"

18.46.1.15 DXF_VERSION_2004 #define DXF_VERSION_2004 "AC1018"

18.46.1.16 DXF_VERSION_2006 #define DXF_VERSION_2006 "AC1018"

18.46.1.17 DXF_VERSION_2007 #define DXF_VERSION_2007 "AC1021"

18.46.1.18 DXF_VERSION_2009 #define DXF_VERSION_2009 "AC1021"

18.46.1.19 DXF_VERSION_2010 #define DXF_VERSION_2010 "AC1024"

18.46.1.20 DXF_VERSION_2013 #define DXF_VERSION_2013 "AC1027"

18.46.1.21 DXF_VERSION_R10 #define DXF_VERSION_R10 "AC1006"

18.46.1.22 DXF_VERSION_R11 #define DXF_VERSION_R11 "AC1009"

18.46.1.23 DXF_VERSION_R12 #define DXF_VERSION_R12 "AC1009"

18.46.1.24 DXF_VERSION_R13 #define DXF_VERSION_R13 "AC1012"

18.46.1.25 DXF_VERSION_R14 #define DXF_VERSION_R14 "AC1014"

18.46.1.26 DXF_VERSION_R15 #define DXF_VERSION_R15 "AC1015"

18.46.1.27 DXF_VERSION_R18 #define DXF_VERSION_R18 "AC1018"

18.46.1.28 DXF_VERSION_R21 #define DXF_VERSION_R21 "AC1021"

18.46.1.29 DXF_VERSION_R24 #define DXF_VERSION_R24 "AC1024"

18.46.1.30 DXF_VERSION_R27 #define DXF_VERSION_R27 "AC1027"

18.46.1.31 ELEMENT_A #define ELEMENT_A 1

18.46.1.32 ELEMENT_ANIMATE #define ELEMENT_ANIMATE 2

18.46.1.33 ELEMENT_ANIMATECOLOR #define ELEMENT_ANIMATECOLOR 3

18.46.1.34 ELEMENT_ANIMATEMOTION #define ELEMENT_ANIMATEMOTION 4

18.46.1.35 ELEMENT_ANIMATETRANSFORM #define ELEMENT_ANIMATETRANSFORM 5

18.46.1.36 ELEMENT_ANIMATION #define ELEMENT_ANIMATION 6

18.46.1.37 ELEMENT_AUDIO #define ELEMENT_AUDIO 7

18.46.1.38 ELEMENT_CIRCLE #define ELEMENT_CIRCLE 8

18.46.1.39 ELEMENT_DEFS #define ELEMENT_DEFS 9

18.46.1.40 ELEMENT_DESC #define ELEMENT_DESC 10

18.46.1.41 ELEMENT_DISCARD #define ELEMENT_DISCARD 11

18.46.1.42 ELEMENT_ELLIPSE #define ELEMENT_ELLIPSE 12

18.46.1.43 ELEMENT_FONT #define ELEMENT_FONT 13

18.46.1.44 ELEMENT_FONT_FACE #define ELEMENT_FONT_FACE 14

18.46.1.45 ELEMENT_FONT_FACE_SRC #define ELEMENT_FONT_FACE_SRC 15

18.46.1.46 ELEMENT_FONT_FACE_URI #define ELEMENT_FONT_FACE_URI 16

18.46.1.47 ELEMENT_FOREIGN_OBJECT #define ELEMENT_FOREIGN_OBJECT 17

18.46.1.48 ELEMENT_G #define ELEMENT_G 18

18.46.1.49 ELEMENT_GLYPH #define ELEMENT_GLYPH 19

18.46.1.50 ELEMENT_HANDLER #define ELEMENT_HANDLER 20

18.46.1.51 ELEMENT_HKERN #define ELEMENT_HKERN 21

18.46.1.52 ELEMENT_IMAGE #define ELEMENT_IMAGE 22

18.46.1.53 ELEMENT_LINE #define ELEMENT_LINE 23

18.46.1.54 ELEMENT_LINEAR_GRADIENT #define ELEMENT_LINEAR_GRADIENT 24

18.46.1.55 ELEMENT_LISTENER #define ELEMENT_LISTENER 25

18.46.1.56 ELEMENT_METADATA #define ELEMENT_METADATA 26

18.46.1.57 ELEMENT_MISSING_GLYPH #define ELEMENT_MISSING_GLYPH 27

18.46.1.58 ELEMENT_MPATH #define ELEMENT_MPATH 28

18.46.1.59 ELEMENT_PATH #define ELEMENT_PATH 29

18.46.1.60 ELEMENT_POLYGON #define ELEMENT_POLYGON 30

18.46.1.61 ELEMENT_POLYLINE #define ELEMENT_POLYLINE 31

18.46.1.62 ELEMENT_PREFETCH #define ELEMENT_PREFETCH 32

18.46.1.63 ELEMENT_RADIAL_GRADIENT #define ELEMENT_RADIAL_GRADIENT 33

18.46.1.64 ELEMENT_RECT #define ELEMENT_RECT 34

18.46.1.65 ELEMENT_SCRIPT #define ELEMENT_SCRIPT 35

18.46.1.66 ELEMENT_SET #define ELEMENT_SET 36

18.46.1.67 ELEMENT_SOLID_COLOR #define ELEMENT_SOLID_COLOR 37

18.46.1.68 ELEMENT_STOP #define ELEMENT_STOP 38

18.46.1.69 ELEMENT_SVG #define ELEMENT_SVG 39

18.46.1.70 ELEMENT_SWITCH #define ELEMENT_SWITCH 40

18.46.1.71 ELEMENT_TBREAK #define ELEMENT_TBREAK 41

18.46.1.72 ELEMENT_TEXT #define ELEMENT_TEXT 42

18.46.1.73 ELEMENT_TEXT_AREA #define ELEMENT_TEXT_AREA 43

18.46.1.74 ELEMENT_TITLE #define ELEMENT_TITLE 44

18.46.1.75 ELEMENT_TSPAN #define ELEMENT_TSPAN 45

18.46.1.76 ELEMENT_USE #define ELEMENT_USE 46

18.46.1.77 ELEMENT_VIDEO #define ELEMENT_VIDEO 47

18.46.1.78 ELEMENT_XML #define ELEMENT_XML 0

18.46.1.79 ELLIPSETOEND #define ELLIPSETOEND 16

18.46.1.80 ELLIPSETORAD #define ELLIPSETORAD 8

18.46.1.81 EMB_BIG_ENDIAN #define EMB_BIG_ENDIAN 0

18.46.1.82 EMB_INT16_BIG #define EMB_INT16_BIG 2

18.46.1.83 EMB_INT16_LITTLE #define EMB_INT16_LITTLE 3

18.46.1.84 EMB_INT32_BIG #define EMB_INT32_BIG 4

18.46.1.85 EMB_INT32_LITTLE #define EMB_INT32_LITTLE 5

18.46.1.86 EMB_LITTLE_ENDIAN #define EMB_LITTLE_ENDIAN 1

18.46.1.87 EMB_MAX #define EMB_MAX(
 A,
 B) (((A) > (B)) ? (A) : (B))

18.46.1.88 EMB_MIN #define EMB_MIN(
 A,
 B) (((A) < (B)) ? (A) : (B))

18.46.1.89 ENDIAN_HOST #define ENDIAN_HOST EMB_LITTLE_ENDIAN

18.46.1.90 GREEN_TERM_COLOR #define GREEN_TERM_COLOR "\x1B[0;32m"

18.46.1.91 HOOP_110X110 #define HOOP_110X110 1

18.46.1.92 HOOP_126X110 #define HOOP_126X110 0

18.46.1.93 HOOP_140X200 #define HOOP_140X200 3

18.46.1.94 HOOP_230X200 #define HOOP_230X200 4

18.46.1.95 HOOP_50X50 #define HOOP_50X50 2

18.46.1.96 LINETO #define LINETO 0

18.46.1.97 MOVETO #define MOVETO 1

18.46.1.98 N_PES VERSIONS #define N_PES_VERSIONS 13

18.46.1.99 ObjectTypeRootEntry #define ObjectTypeRootEntry 0x05
the root entry

18.46.1.100 ObjectTypeStorage #define ObjectTypeStorage 0x01
a directory type object

18.46.1.101 ObjectTypeStream #define ObjectTypeStream 0x02
a file type object

18.46.1.102 ObjectTypeUnknown #define ObjectTypeUnknown 0x00
Type of directory object Probably unallocated

18.46.1.103 PES0001 #define PES0001 0

18.46.1.104 PES0020 #define PES0020 1

18.46.1.105 PES0022 #define PES0022 2

18.46.1.106 PES0030 #define PES0030 3

18.46.1.107 PES0040 #define PES0040 4

18.46.1.108 PES0050 #define PES0050 5

18.46.1.109 PES0055 #define PES0055 6

18.46.1.110 PES0056 #define PES0056 7

18.46.1.111 PES0060 #define PES0060 8

18.46.1.112 PES0070 #define PES0070 9

18.46.1.113 PES0080 #define PES0080 10

18.46.1.114 PES0090 #define PES0090 11

18.46.1.115 PES0100 #define PES0100 12

18.46.1.116 QUADTOCONTROL #define QUADTOCONTROL 256

18.46.1.117 QUADTOEND #define QUADTOEND 512

18.46.1.118 RED_TERM_COLOR #define RED_TERM_COLOR "\x1B[0;31m"

18.46.1.119 RESET_TERM_COLOR #define RESET_TERM_COLOR "\033[0m"

18.46.1.120 SVG_ATTRIBUTE #define SVG_ATTRIBUTE 4

18.46.1.121 SVG_CATCH_ALL #define SVG_CATCH_ALL 5

18.46.1.122 SVG_CREATOR_EMBROIDERMODDER #define SVG_CREATOR_EMBROIDERMODDER 1

18.46.1.123 SVG_CREATOR_ILLUSTRATOR #define SVG_CREATOR_ILLUSTRATOR 2

18.46.1.124 SVG_CREATOR_INKSCAPE #define SVG_CREATOR_INKSCAPE 3

18.46.1.125 SVG_CREATOR_NULL #define SVG_CREATOR_NULL 0

18.46.1.126 SVG_ELEMENT #define SVG_ELEMENT 1

18.46.1.127 SVG_EXPECT_ATTRIBUTE #define SVG_EXPECT_ATTRIBUTE 2

18.46.1.128 SVG_EXPECT_ELEMENT #define SVG_EXPECT_ELEMENT 1

18.46.1.129 SVG_EXPECT_NULL #define SVG_EXPECT_NULL 0

18.46.1.130 SVG_EXPECT_VALUE #define SVG_EXPECT_VALUE 3

18.46.1.131 SVG_MEDIA_PROPERTY #define SVG_MEDIA_PROPERTY 3

18.46.1.132 SVG_NULL #define SVG_NULL 0

18.46.1.133 SVG_PROPERTY #define SVG_PROPERTY 2

18.46.1.134 YELLOW_TERM_COLOR #define YELLOW_TERM_COLOR "\x1B[1;33m"

18.46.2 Typedef Documentation

18.46.2.1 bcf_directory typedef struct [_bcf_directory](#) bcf_directory

Todo possibly add a directory tree in the future.

18.46.2.2 bcf_directory_entry typedef struct [_bcf_directory_entry](#) bcf_directory_entry

18.46.2.3 bcf_file typedef struct [_bcf_file](#) bcf_file

18.46.2.4 bcf_file_difat typedef struct [_bcf_file_difat](#) bcf_file_difat

18.46.2.5 bcf_file_fat typedef struct [_bcf_file_fat](#) bcf_file_fat

18.46.2.6 bcf_file_header `typedef struct _bcf_file_header bcf_file_header`

Todo CLSID should be a separate type.

18.46.2.7 compress `typedef struct Compress compress`

18.46.2.8 huffman `typedef struct Huffman huffman`

18.46.2.9 StxThread `typedef struct StxThread_ StxThread`

18.46.2.10 SubDescriptor `typedef struct SubDescriptor_ SubDescriptor`

18.46.2.11 SvgAttribute `typedef struct SvgAttribute_ SvgAttribute`

18.46.2.12 ThredExtension `typedef struct ThredExtension_ ThredExtension`

18.46.2.13 ThredHeader `typedef struct ThredHeader_ ThredHeader`

18.46.2.14 VipHeader `typedef struct VipHeader_ VipHeader`

18.46.2.15 vp3Hoop `typedef struct _vp3Hoop vp3Hoop`

18.46.3 Enumeration Type Documentation

18.46.3.1 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>	

18.46.3.2 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	
-----------------	--

18.46.4 Function Documentation

18.46.4.1 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*`

18.46.4.2 bcf_directory_free() `void bcf_directory_free (bcf_directory ** dir)`

dir

18.46.4.3 bcf_file_difat_free() `void bcf_file_difat_free (bcf_file_difat * difat)`

18.46.4.4 bcf_file_fat_free() `void bcf_file_fat_free (bcf_file_fat ** fat)`

18.46.4.5 bcf_file_free() `void bcf_file_free (bcf_file * bcfFile)`

18.46.4.6 bcfFile_read() `int bcfFile_read (FILE * file, bcf_file * bcfFile)`

file *bcfFile*

Returns

`int`

18.46.4.7 bcfFileFat_create() `bcf_file_fat * bcfFileFat_create (const unsigned int sectorSize)`

sectorSize

Returns

`bcf_file_fat*`

18.46.4.8 bcfFileHeader_isValid() `int bcfFileHeader_isValid (`
`bcf_file_header header)`

18.46.4.9 bcfFileHeader_read() `bcf_file_header bcfFileHeader_read (`
`FILE * file)`

file

Returns

`bcf_file_header`

18.46.4.10 binaryReadString() `void binaryReadString (`

`FILE * file,`
`char * buffer,`
`int maxLength)`

file buffer maxLength

18.46.4.11 binaryReadUnicodeString() `void binaryReadUnicodeString (`

`FILE * file,`
`char * buffer,`
`const int stringLength)`

file buffer stringLength

18.46.4.12 binaryWriteInt() `void binaryWriteInt (`

`FILE * f,`
`int data)`

f data

Todo replace with emblnt_read

18.46.4.13 binaryWriteIntBE() `void binaryWriteIntBE (`

`FILE * f,`
`int data)`

f data

Todo replace with emblnt_read

18.46.4.14 binaryWriteShort() `void binaryWriteShort (`

`FILE * f,`
`short data)`

f data

Todo replace with emblnt_read

18.46.4.15 binaryWriteUInt() `void binaryWriteUInt (`

`FILE * f,`
`unsigned int data)`

f data

Todo replace with emblnt_read

18.46.4.16 binaryWriteUIntBE() void binaryWriteUIntBE (

```
FILE * f,
unsigned int data )
```

f data

Todo replace with emblnt_read

18.46.4.17 binaryWriteUShort() void binaryWriteUShort (

```
FILE * f,
unsigned short data )
```

f data

Todo replace with emblnt_read

18.46.4.18 binaryWriteUShortBE() void binaryWriteUShortBE (

```
FILE * f,
unsigned short data )
```

f data

Todo replace with emblnt_read

18.46.4.19 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.

18.46.4.20 CompoundFileDirectory() bcf_directory * CompoundFileDirectory (

```
const unsigned int maxNumberOfDirectoryEntries )
```

maxNumberOfDirectoryEntries

Returns

bcf_directory*

18.46.4.21 CompoundFileDirectoryEntry() bcf_directory_entry * CompoundFileDirectoryEntry (

```
FILE * file )
```

file

Returns

bcf_directory_entry*

18.46.4.22 compress_get_bits() int compress_get_bits (

```
compress * c,
int length )
```

c length Returns .

18.46.4.23 compress_get_position() `int compress_get_position (`
 `compress * c)`
c. Returns the position as an int.

18.46.4.24 compress_get_token() `int compress_get_token (`
 `compress * c)`
c. Returns the token as an int.

18.46.4.25 compress_load_block() `void compress_load_block (`
 `compress * c)`
c. Returns nothing.

18.46.4.26 compress_load_character_huffman() `void compress_load_character_huffman (`
 `compress * c)`
Load character table to compress struct *c*. Returns nothing.

18.46.4.27 compress_load_character_length_huffman() `void compress_load_character_length_`←
 `huffman (`
 `compress * c)`
c. Returns.

18.46.4.28 compress_load_distance_huffman() `void compress_load_distance_huffman (`
 `compress * c)`
c. Returns nothing.

18.46.4.29 compress_pop() `int compress_pop (`
 `compress * c,`
 `int bit_count)`
c bit_count. Returns.

18.46.4.30 compress_read_variable_length() `int compress_read_variable_length (`
 `compress * c)`
c. Returns.

18.46.4.31 copy_trim() `char * copy_trim (`
 `char const * s)`
s

Returns

`char*`

Todo description

18.46.4.32 create_test_file_1() `int create_test_file_1 (`
 `const char * outf)`

18.46.4.33 create_test_file_2() `int create_test_file_2 (`
 `const char * outf)`

18.46.4.34 create_test_file_3() `int create_test_file_3 (`
 `const char * outf)`

18.46.4.35 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.

18.46.4.36 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.

18.46.4.37 decodeNewStitch() `int decodeNewStitch (`
 `unsigned char value)`

Returns

`int`

18.46.4.38 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*`

18.46.4.39 emb_readline() `int emb_readline (`
 `FILE * file,`
 `char * line,`
 `int maxLength)`

file line maxLength

Returns

`int`

18.46.4.40 embColor_read() `void embColor_read (`
 `FILE * f,`
 `EmbColor * c,`
 `int toRead)`

f c toRead

18.46.4.41 embColor_write() void embColor_write (

```
FILE * f,
EmbColor c,
int toWrite )
```

f c toWrite

18.46.4.42 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.

18.46.4.43 embInt_write() void embInt_write (

```
FILE * f,
char * label,
void * b,
int mode )
```

f label b mode

18.46.4.44 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*.

18.46.4.45 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.

18.46.4.46 entriesInDifatSector() unsigned int entriesInDifatSector (

```
bcf_file_difat * fat )
```

fat

Returns

unsigned int

18.46.4.47 fpad() void fpad (

```
FILE * file,
char c,
int n )
```

f

Returns

int

18.46.4.48 `fread_int16()` short fread_int16 (

```
    FILE * f )
```

f

Returns

short

18.46.4.49 `fread_int32_be()` int fread_int32_be (

```
    FILE * f )
```

f

Returns

int

Todo replace with emblnt_read

18.46.4.50 `fread_uint16()` unsigned short fread_uint16 (

```
    FILE * f )
```

f

Returns

unsigned short

Todo replace with emblnt_read

18.46.4.51 `GetFile()` FILE * GetFile (

```
    bcf_file * bcfFile,
```

```
    FILE * file,
```

```
    char * fileToFind )
```

Get the File object.

bcfFile file fileToFind

Returns

FILE*

18.46.4.52 `huffman_build_table()` void huffman_build_table (

```
    huffman * h )
```

These next 2 functions represent the [Huffman](#) class in tartarize's code. *h*

18.46.4.53 `huffman_table_lookup()` int * huffman_table_lookup (

```
    huffman * h,
```

```
    int byte_lookup,
```

```
    int * lengths )
```

18.46.4.54 `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).

```
18.46.4.55 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.
```

```
18.46.4.56 loadFatFromSector() void loadFatFromSector (
    bcf_file_fat * fat,
    FILE * file )
fat file
```

18.46.4.57 mitDecodeStitch() int mitDecodeStitch (

unsigned char value)

value

Returns

int

```
18.46.4.58 mitEncodeStitch() unsigned char mitEncodeStitch (
    EmbReal value )
```

value

Returns

unsigned char

```
18.46.4.59 numberOfEntriesInDifatSector() unsigned int numberOfEntriesInDifatSector (
    bcf_file_difat * fat )
```

```
18.46.4.60 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.

```
18.46.4.61 pfaffEncode() void pfaffEncode (
    FILE * file,
    int dx,
    int dy,
    int flags )
```

file dx dy flags

```
18.46.4.62 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 )
```

18.46.4.63 `read100()` `char read100 (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.64 `read10o()` `char read10o (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.65 `readArt()` `char readArt (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.66 `readBmc()` `char readBmc (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.67 `readBro()` `char readBro (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.68 `readCnd()` `char readCnd (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.69 `readCol()` `char readCol (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.70 `readCsd()` `char readCsd (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.71 `readCsv()` `char readCsv (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.72 `readDat()` `char readDat (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.73 readDem() `char readDem (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.74 readDescriptions() `void readDescriptions (`
 `FILE * file,`
 `EmbPattern * pattern)`

18.46.4.75 readDsb() `char readDsb (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.76 readDst() `char readDst (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.77 readDsz() `char readDsz (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.78 ZSK USA Embroidery Format (.dsz) The ZSK USA dsz format is stitch-only.

18.46.4.79 readDxf() `char readDxf (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.80 readEdr() `char readEdr (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.81 Embird Embroidery Format (.edr) Stitch Only Format

18.46.4.82 readEmd() `char readEmd (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.83 readExp() `char readExp (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.84 readExy() `char readExy (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.85 readEys() `char readEys (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.86 Sierra Expanded Embroidery Format (.eys) Stitch Only Format.
Smoothie G-Code Embroidery Format (.fxy)?

18.46.4.87 `readFeatherPatterns()` void readFeatherPatterns (FILE * file, EmbPattern * pattern)

18.46.4.88 `readFullSector()` unsigned int readFullSector (FILE * file, bcf_file_difat * bcfFile, unsigned int * difatEntriesToRead)
file bcfFile difatEntriesToRead

Returns

unsigned int

18.46.4.89 `readFxy()` char readFxy (EmbPattern * pattern, FILE * file)

18.46.4.90 Embroidery Format (.fxy) Stitch Only Format.

18.46.4.91 `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

18.46.4.92 `readGnc()` char readGnc (EmbPattern * pattern, FILE * file)

18.46.4.93 Great Notions Embroidery Format (.gnc) Stitch Only Format.

18.46.4.94 `readGt()` char readGt (EmbPattern * pattern, FILE * file)

18.46.4.95 Gold Thread Embroidery Format (.gt) Stitch Only Format.

18.46.4.96 `readHoopName()` void readHoopName (FILE * file, EmbPattern * pattern)

18.46.4.97 `readHus()` char readHus (EmbPattern * pattern, FILE * file)

18.46.4.98 readImageString() void readImageString (FILE * file, EmbPattern * pattern)

18.46.4.99 readInb() char readInb (EmbPattern * pattern, FILE * file)

18.46.4.100 Inbro Embroidery Format (.inb) Stitch Only Format.

18.46.4.101 readInf() char readInf (EmbPattern * pattern, FILE * file)

18.46.4.102 Embroidery Color Format (.inf) Stitch Only Format.

18.46.4.103 readJef() char readJef (EmbPattern * pattern, FILE * file)

18.46.4.104 readKsm() char readKsm (EmbPattern * pattern, FILE * file)

18.46.4.105 readMax() char readMax (EmbPattern * pattern, FILE * file)

18.46.4.106 readMit() char readMit (EmbPattern * pattern, FILE * file)

18.46.4.107 Mitsubishi Embroidery Format (.mit) Stitch Only Format.

18.46.4.108 readMotifPatterns() void readMotifPatterns (FILE * file, EmbPattern * pattern)

18.46.4.109 readNew() char readNew (EmbPattern * pattern, FILE * file)

18.46.4.110 Ameco Embroidery Format (.new) Stitch Only Format.

18.46.4.111 readNextSector() void readNextSector (FILE * file, bcf_directory * dir)
file dir

```
18.46.4.112 readOfm() char readOfm (
    EmbPattern * pattern,
    FILE * file )
```

```
18.46.4.113 readPcd() char readPcd (
    EmbPattern * pattern,
    const char * fileName,
    FILE * file )
```

18.46.4.114 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).

```
18.46.4.115 readPcm() char readPcm (
    EmbPattern * pattern,
    FILE * file )
```

18.46.4.116 Pfaff Embroidery Format (.pcm) The Pfaff pcm format is stitch-only.

```
18.46.4.117 readPcq() char readPcq (
    EmbPattern * pattern,
    const char * fileName,
    FILE * file )
```

18.46.4.118 Embroidery Format (.pcq) The Pfaff pcq format is stitch-only.

```
18.46.4.119 readPcs() char readPcs (
    EmbPattern * pattern,
    const char * fileName,
    FILE * file )
```

18.46.4.120 Embroidery Format (.pcq) The Pfaff pcs format is stitch-only.

```
18.46.4.121 readPec() char readPec (
    EmbPattern * pattern,
    const char * fileName,
    FILE * file )
```

```
18.46.4.122 readPecStitches() void readPecStitches (
    EmbPattern * pattern,
    FILE * file )
```

18.46.4.123 Embroidery Format (.pec) The Brother pec format is stitch-only.

```
18.46.4.124 readPel() char readPel (
    EmbPattern * pattern,
    FILE * file )
```

18.46.4.125 Embroidery Format (.pec) The Brother pel format is stitch-only.

18.46.4.126 `readPem()` `char readPem (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.127 Embroidery Format (.pec) The Brother pem format is stitch-only.

18.46.4.128 `readPes()` `char readPes (`
 `EmbPattern * pattern,`
 `const char * fileName,`
 `FILE * file)`

18.46.4.129 `readPESHeaderV10()` `void readPESHeaderV10 (`
 `FILE * file,`
 `EmbPattern * pattern)`

18.46.4.130 `readPESHeaderV5()` `void readPESHeaderV5 (`
 `FILE * file,`
 `EmbPattern * pattern)`

18.46.4.131 `readPESHeaderV6()` `void readPESHeaderV6 (`
 `FILE * file,`
 `EmbPattern * pattern)`

18.46.4.132 `readPESHeaderV7()` `void readPESHeaderV7 (`
 `FILE * file,`
 `EmbPattern * pattern)`

18.46.4.133 `readPESHeaderV8()` `void readPESHeaderV8 (`
 `FILE * file,`
 `EmbPattern * pattern)`

18.46.4.134 `readPESHeaderV9()` `void readPESHeaderV9 (`
 `FILE * file,`
 `EmbPattern * pattern)`

18.46.4.135 `readPhb()` `char readPhb (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.136 Embroidery Format (.pec) The Brother phb format is stitch-only.

18.46.4.137 `readPhc()` `char readPhc (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.138 Embroidery Format (.pec) The Brother phc format is stitch-only.

18.46.4.139 `readPlt()` `char readPlt (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.140 Embroidery Format (.plt) The AutoCAD plt format is stitch-only.

18.46.4.141 `readProgrammableFills()` `void readProgrammableFills (`
 `FILE * file,`
 `EmbPattern * pattern)`

18.46.4.142 `readRgb()` `char readRgb (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.143 Color File (.rgb) The RGB format is a color-only format to act as an external color file for other formats.

18.46.4.144 `readSew()` `char readSew (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.145 `readShv()` `char readShv (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.146 `readSst()` `char readSst (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.147 Embroidery Format (.sst) The Sunstar sst format is stitch-only.

18.46.4.148 `readStx()` `char readStx (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.149 `readSvg()` `char readSvg (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.150 `readT01()` `char readT01 (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.151 Embroidery Format (.pcq) The Pfaff t01 format is stitch-only.

18.46.4.152 `readT09()` `char readT09 (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.152.1 Embroidery Format (.pcq) The Pfaff t09 format is stitch-only.

18.46.4.153 `readTap()` `char readTap (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.154 `readThr()` `char readThr (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.155 Embroidery Format (.thr) The ThreadWorks thr format is stitch-only.

18.46.4.156 `readThreads()` `void readThreads (`
 `FILE * file,`
 `EmbPattern * pattern)`

18.46.4.157 `readTxt()` `char readTxt (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.158 File (.txt) The txt format is stitch-only and isn't associated with a specific company.

18.46.4.159 `readU00()` `char readU00 (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.160 Embroidery Format (.u00) The Barudan u00 format is stitch-only.

18.46.4.161 `readU01()` `char readU01 (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.162 Embroidery Format (.u00) The Barudan u01 format is stitch-only.

18.46.4.163 `readVip()` `char readVip (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.164 `readVp3()` `char readVp3 (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.165 `readXxx()` `char readXxx (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.166 `readZsk()` `char readZsk (`
 `EmbPattern * pattern,`
 `FILE * file)`

```
18.46.4.167 safe_free() void safe_free (
    void * data )
```

```
18.46.4.168 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.

```
18.46.4.169 testEmbCircle() int testEmbCircle (
    void )
```

```
18.46.4.170 testEmbCircle_2() int testEmbCircle_2 (
    void )
```

```
18.46.4.171 testEmbFormat() int testEmbFormat (
    void )
```

```
18.46.4.172 testGeomArc() int testGeomArc (
    void )
```

```
18.46.4.173 testTangentPoints() void testTangentPoints (
    EmbCircle c,
    EmbVector p,
    EmbVector * t0,
    EmbVector * t1 )
```

```
18.46.4.174 testThreadColor() int testThreadColor (
    void )
```

```
18.46.4.175 write100() char write100 (
    EmbPattern * pattern,
    FILE * file )
```

```
18.46.4.176 write10o() char write10o (
    EmbPattern * pattern,
    FILE * file )
```

18.46.4.177 write_24bit() void write_24bit (FILE * file, int x)
file x

18.46.4.178 writeArt() char writeArt (EmbPattern * pattern, FILE * file)

18.46.4.179 writeBmc() char writeBmc (EmbPattern * pattern, FILE * file)

18.46.4.180 writeBro() char writeBro (EmbPattern * pattern, FILE * file)

18.46.4.181 writeCnd() char writeCnd (EmbPattern * pattern, FILE * file)

18.46.4.182 writeCol() char writeCol (EmbPattern * pattern, FILE * file)

18.46.4.183 writeCsd() char writeCsd (EmbPattern * pattern, FILE * file)

18.46.4.184 writeCsv() char writeCsv (EmbPattern * pattern, FILE * file)

18.46.4.185 writeDat() char writeDat (EmbPattern * pattern, FILE * file)

18.46.4.186 writeDem() char writeDem (EmbPattern * pattern, FILE * file)

18.46.4.187 writeDsb() char writeDsb (EmbPattern * pattern, FILE * file)

18.46.4.188 `writeDst()` `char writeDst (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.189 `writeDsz()` `char writeDsz (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.190 `writeDxf()` `char writeDxf (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.191 `writeEdr()` `char writeEdr (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.192 `writeEmd()` `char writeEmd (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.193 `writeExp()` `char writeExp (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.194 `writeExy()` `char writeExy (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.195 `writeEys()` `char writeEys (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.196 `writeFxy()` `char writeFxy (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.197 `writeGc()` `char writeGc (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.198 `writeGnc()` `char writeGnc (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.46.4.199 writeGt() char writeGt (

```
EmbPattern * pattern,
FILE * file )
```

18.46.4.200 writeHus() char writeHus (

```
EmbPattern * pattern,
FILE * file )
```

18.46.4.201 writelnb() char writelnb (

```
EmbPattern * pattern,
FILE * file )
```

18.46.4.202 writelnf() char writelnf (

```
EmbPattern * pattern,
FILE * file )
```

18.46.4.203 writeJef() char writeJef (

```
EmbPattern * pattern,
FILE * file )
```

18.46.4.204 writeKsm() char writeKsm (

```
EmbPattern * pattern,
FILE * file )
```

18.46.4.205 writeMax() char writeMax (

```
EmbPattern * pattern,
FILE * file )
```

18.46.4.206 writeMit() char writeMit (

```
EmbPattern * pattern,
FILE * file )
```

18.46.4.207 writeNew() char writeNew (

```
EmbPattern * pattern,
FILE * file )
```

18.46.4.208 writeOfm() char writeOfm (

```
EmbPattern * pattern,
FILE * file )
```

18.46.4.209 writePcd() char writePcd (

```
EmbPattern * pattern,
FILE * file )
```

18.46.4.210 `writePcm()` `char writePcm (`
`EmbPattern * pattern,`
`FILE * file)`

18.46.4.211 `writePcq()` `char writePcq (`
`EmbPattern * pattern,`
`FILE * file)`

18.46.4.212 `writePcs()` `char writePcs (`
`EmbPattern * pattern,`
`FILE * file)`

18.46.4.213 `writePec()` `char writePec (`
`EmbPattern * pattern,`
`const char * fileName,`
`FILE * file)`

18.46.4.214 `writePecStitches()` `void writePecStitches (`
`EmbPattern * pattern,`
`FILE * file,`
`const char * filename)`

18.46.4.215 `writePel()` `char writePel (`
`EmbPattern * pattern,`
`FILE * file)`

18.46.4.216 `writePem()` `char writePem (`
`EmbPattern * pattern,`
`FILE * file)`

18.46.4.217 `writePes()` `char writePes (`
`EmbPattern * pattern,`
`const char * fileName,`
`FILE * file)`

18.46.4.218 `writePhb()` `char writePhb (`
`EmbPattern * pattern,`
`FILE * file)`

18.46.4.219 `writePhc()` `char writePhc (`
`EmbPattern * pattern,`
`FILE * file)`

18.46.4.220 writePlt() char writePlt (

```
EmbPattern * pattern,
FILE * file )
```

18.46.4.221 writeRgb() char writeRgb (

```
EmbPattern * pattern,
FILE * file )
```

18.46.4.222 writeSew() char writeSew (

```
EmbPattern * pattern,
FILE * file )
```

18.46.4.223 writeShv() char writeShv (

```
EmbPattern * pattern,
FILE * file )
```

18.46.4.224 writeSst() char writeSst (

```
EmbPattern * pattern,
FILE * file )
```

18.46.4.225 writeStx() char writeStx (

```
EmbPattern * pattern,
FILE * file )
```

18.46.4.226 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.

18.46.4.227 writeT01() char writeT01 (

```
EmbPattern * pattern,
FILE * file )
```

18.46.4.228 writeT09() char writeT09 (

```
EmbPattern * pattern,
FILE * file )
```

18.46.4.229 writeTap() char writeTap (

```
EmbPattern * pattern,
FILE * file )
```

18.46.4.230 writeThr() char writeThr (

```
EmbPattern * pattern,
FILE * file )
```

```
18.46.4.231 writeTxt() char writeTxt (
    EmbPattern * pattern,
    FILE * file )
```

```
18.46.4.232 writeU00() char writeU00 (
    EmbPattern * pattern,
    FILE * file )
```

```
18.46.4.233 writeU01() char writeU01 (
    EmbPattern * pattern,
    FILE * file )
```

```
18.46.4.234 writeVip() char writeVip (
    EmbPattern * pattern,
    FILE * file )
```

```
18.46.4.235 writeVp3() char writeVp3 (
    EmbPattern * pattern,
    FILE * file )
```

```
18.46.4.236 writeXxx() char writeXxx (
    EmbPattern * pattern,
    FILE * file )
```

```
18.46.4.237 writeZsk() char writeZsk (
    EmbPattern * pattern,
    FILE * file )
```

18.46.5 Variable Documentation

```
18.46.5.1 imageWithFrame const char imageWithFrame[38][48] [extern]
```

18.47 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 "\x033[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;
00211
00212     unsigned int
00213 }
```

```

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 *****
```

```
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

```

18.48 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)

18.48.1 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.

18.48.2 Function Documentation

18.48.2.1 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.

18.48.2.2 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.

18.48.2.3 decodeNewStitch() int `decodeNewStitch` (
 unsigned char value)

value

Returns

int

18.48.2.4 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.

18.48.2.5 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.

18.48.2.6 embInt_write() `void embInt_write (`
 `FILE * f,`
 `char * label,`
 `void * b,`
 `int mode)`

f label b mode

18.48.2.7 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*.

18.48.2.8 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.

18.48.2.9 mitDecodeStitch() `int mitDecodeStitch (`
 `unsigned char value)`

value

Returns

`int`

18.48.2.10 mitEncodeStitch() `unsigned char mitEncodeStitch (`
 `EmbReal value)`

value

Returns

`unsigned char`

```
18.48.2.11 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.

```
18.48.2.12 pfaffEncode() void pfaffEncode (
    FILE * file,
    int dx,
    int dy,
    int flags )
```

file dx dy flags

```
18.48.2.13 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.

```
18.48.2.14 write_24bit() void write_24bit (
    FILE * file,
    int x )
```

file x

18.49 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 threshold)
- void `embPattern_crossstitch` (`EmbPattern` *pattern, `EmblImage` *image, int threshold)
- 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**

18.49.1 Function Documentation

18.49.1.1 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.

18.49.1.2 embPattern_combine() **EmbPattern** * embPattern_combine (

EmbPattern * *p1*,

EmbPattern * *p2*)

p1 p2

Returns

EmbPattern*

18.49.1.3 embPattern_convertGeometry() void embPattern_convertGeometry (

EmbPattern * *p*)

p

18.49.1.4 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.

18.49.1.5 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.

18.49.1.6 embPattern_stitchArc() void embPattern_stitchArc (

EmbPattern * *p*,

EmbArc *arc*,

int *thread_index*,

int *style*)

p arc thread_index style

18.49.1.7 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:

```
$(c - r(d/|d|), c + r(d/|d|))$
```

Lines that are above and below this with an even separation \$s\$ can be found by taking the point on the line to be $c + sn$ where the n is the unit normal vector to d and the vector to be 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:

```
$(c + sn - \sqrt{r^2 - s^2}(d/|d|), c + sn + \sqrt{r^2 - s^2}(d/|d|))$
```

repeating this process gives us all the end points and the fill only alters these lines by splitting the ones longer than some tolerance.

18.49.1.8 embPattern_stitchEllipse() void embPattern_stitchEllipse (

```
    EmbPattern * p,
    EmbEllipse ellipse,
    int thread_index,
    int style )
```

p ellipse thread_index style

Todo finish stitchEllipse

18.49.1.9 embPattern_stitchPath() void embPattern_stitchPath (

```
    EmbPattern * p,
    EmbPath path,
    int thread_index,
    int style )
```

p rect thread_index style

Todo finish stitch path

18.49.1.10 embPattern_stitchPolygon() void embPattern_stitchPolygon (

```
    EmbPattern * p,
    EmbPolygon polygon,
    int thread_index,
    int style )
```

p rect thread_index style

Todo finish stitch polygon

18.49.1.11 embPattern_stitchPolyline() void embPattern_stitchPolyline (

```
    EmbPattern * p,
    EmbPolyline polyline,
    int thread_index,
    int style )
```

p rect thread_index style

Todo finish stitch polyline

18.49.1.12 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.

18.49.1.13 embPattern_stitchText() void embPattern_stitchText (

```
    EmbPattern * p,
    EmbRect rect,
    int thread_index,
    int style )
```

p rect thread_index style

18.49.1.14 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".

18.49.1.15 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.

18.49.1.16 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

18.49.1.17 greedy_algorithm() static void greedy_algorithm (

```
    int * points,
    int n_points,
    int width,
    EmbReal bias ) [static]
```

points n_points width bias

18.49.1.18 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.

18.49.1.19 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.

18.49.1.20 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.

18.49.1.21 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.

18.49.1.22 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

18.49.1.23 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.

18.49.2 Variable Documentation

18.49.2.1 hilbert_curve_l_system L_system hilbert_curve_l_system

Initial value:

```
= {
    'A', "AB", "F+-", (char**)rules
}
```

18.49.2.2 rules const char* rules[] = {"+BF-AFA-FB+", "-AF+BFB+FA-"}

18.50 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]`

18.50.1 Function Documentation**18.50.1.1 binaryWriteInt()** `void binaryWriteInt (`

```
FILE * f,  
     int data )
```

f data

Todo replace with emblnt_read

18.50.1.2 binaryWriteIntBE() `void binaryWriteIntBE (`

```
FILE * f,  
     int data )
```

f data

Todo replace with emblnt_read

18.50.1.3 binaryWriteShort() `void binaryWriteShort (`

```
FILE * f,  
     short data )
```

f data

Todo replace with emblnt_read

18.50.1.4 binaryWriteUInt() `void binaryWriteUInt (`

```
FILE * f,  
     unsigned int data )
```

f data

Todo replace with emblnt_read

18.50.1.5 binaryWriteUIntBE() `void binaryWriteUIntBE (`

```
FILE * f,  
     unsigned int data )
```

f data

Todo replace with emblnt_read

18.50.1.6 binaryWriteUShort() `void binaryWriteUShort (`

```
FILE * f,  
     unsigned short data )
```

f data

Todo replace with emblnt_read

18.50.1.7 binaryWriteUShortBE() void binaryWriteUShortBE (FILE * *f*, unsigned short *data*)
f data

Todo replace with emblnt_read

18.50.1.8 emb_identify_format() int emb_identify_format (const char * *fileName*)
fileName

Returns

int

18.50.1.9 embFormat_getExtension() int embFormat_getExtension (const char * *fileName*, char * *ending*)
fileName ending

Returns

int

18.50.1.10 embPattern_read() char embPattern_read (EmbPattern * *pattern*, const char * *fileName*, int *format*)
pattern fileName format

Returns

char

18.50.1.11 embPattern_readAuto() char embPattern_readAuto (EmbPattern * *pattern*, const char * *fileName*)
pattern fileName

Returns

char

18.50.1.12 embPattern_write() char embPattern_write (EmbPattern * *pattern*, const char * *fileName*, int *format*)
pattern fileName format

Returns

char

18.50.1.13 embPattern_writeAuto() `char embPattern_writeAuto (`
 `EmbPattern * pattern,`
 `const char * fileName)`
pattern fileName

Returns

`char`

18.50.1.14 fpad() `void fpad (`
 `FILE * file,`
 `char c,`
 `int n)`
f

Returns

`int`

18.50.1.15 fread_int16() `short fread_int16 (`
 `FILE * f)`
f

Returns

`short`

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

Returns

`int`

Todo replace with emblnt_read

18.50.1.17 fread_uint16() `unsigned short fread_uint16 (`
 `FILE * f)`
f

Returns

`unsigned short`

Todo replace with emblnt_read

18.50.1.18 safe_free() `void safe_free (`
 `void * data)`
data

18.50.2 Variable Documentation

18.50.2.1 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). *

18.50.2.2 imageWithFrame `const char imageWithFrame[38][48]`**18.51 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`)

18.51.1 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.

18.51.2 Function Documentation**18.51.2.1 read100()** `char read100 (`
`EmbPattern * pattern,`
`FILE * file)`**18.51.2.2 write100()** `char write100 (`
`EmbPattern * pattern,`
`FILE * file)`**18.52 extern/libembroidery/src/formats/format_10o.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`)

18.52.1 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.

18.52.2 Function Documentation

18.52.2.1 `read10o()` char `read10o (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.52.2.2 `write10o()` char `write10o (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.53 `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)`

18.53.1 Detailed Description

The Bernina Embroidery Format (.art)
We don't know much about this format.

Todo Find a source.

18.53.2 Function Documentation

18.53.2.1 `readArt()` char `readArt (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.53.2.2 `writeArt()` char `writeArt (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.54 `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)`

18.54.1 Detailed Description

The Bitmap Cache Embroidery Format (.bmc)
We don't know much about this format.

Todo Find a source.

18.54.2 Function Documentation

18.54.2.1 readBmc() char readBmc (

```
    EmbPattern * pattern,
    FILE * file )
```

18.54.2.2 writeBmc() char writeBmc (

```
    EmbPattern * pattern,
    FILE * file )
```

18.55 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)

18.55.1 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.

18.55.2 Function Documentation

18.55.2.1 readBro() char readBro (

```
    EmbPattern * pattern,
    FILE * file )
```

18.55.2.2 writeBro() char writeBro (

```
    EmbPattern * pattern,
    FILE * file )
```

18.56 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)`

18.56.1 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.

18.56.2 Function Documentation

18.56.2.1 `readCnd()` char `readCnd (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.56.2.2 `writeCnd()` char `writeCnd (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.57 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)`

18.57.1 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.

18.57.1.0.1 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
```

18.57.2 Function Documentation

18.57.2.1 readCol() char readCol (
 EmbPattern * pattern,
 FILE * file)

18.57.2.2 writeCol() char writeCol (
 EmbPattern * pattern,
 FILE * file)

18.58 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** []

18.58.1 Detailed Description

The Singer Embroidery Format (.csd)
Stitch Only Format.

18.58.2 Macro Definition Documentation

18.58.2.1 CsdSubMaskSize #define CsdSubMaskSize 479

18.58.2.2 CsdXorMaskSize #define CsdXorMaskSize 501

18.58.3 Function Documentation

18.58.3.1 BuildDecryptionTable() void BuildDecryptionTable (int seed)

18.58.3.2 DecodeCsdByte() unsigned char DecodeCsdByte (long fileOffset, unsigned char val, int type)

18.58.3.3 readCsd() char readCsd (EmbPattern * pattern, FILE * file)

18.58.3.4 writeCsd() char writeCsd (EmbPattern * pattern, FILE * file)

18.58.4 Variable Documentation

18.58.4.1 _subMask char _subMask[CsdSubMaskSize]

18.58.4.2 _xorMask char _xorMask[CsdXorMaskSize]

18.58.4.3 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
}
```

18.59 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)`

18.59.1 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.

18.59.1.0.1 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.

18.59.1.0.2 EmBird CSV Dialect

18.59.2 Function Documentation

18.59.2.1 csvStitchFlagToStr() char * csvStitchFlagToStr (int flags)

18.59.2.2 csvStrToStitchFlag() int csvStrToStitchFlag (const char * str)

18.59.2.3 readCsv() char readCsv (EmbPattern * pattern, FILE * file)

18.59.2.4 writeCsv() char writeCsv (EmbPattern * pattern, FILE * file)

18.60 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)`

18.60.1 Function Documentation

18.60.1.1 readDat() char readDat (EmbPattern * pattern, FILE * file)

```
18.60.1.2 writeDat() char writeDat (
    EmbPattern * pattern,
    FILE * file )
```

18.61 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)

18.61.1 Detailed Description

The Melco Embroidery Format (.dem)
Stitch Only Format

18.61.2 Function Documentation

```
18.61.2.1 readDem() char readDem (
    EmbPattern * pattern,
    FILE * file )
```

```
18.61.2.2 writeDem() char writeDem (
    EmbPattern * pattern,
    FILE * file )
```

18.62 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)

18.62.1 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

18.62.2 Function Documentation

```
18.62.2.1 readDsb() char readDsb (
    EmbPattern * pattern,
    FILE * file )
```

```
18.62.2.2 writeDsb() char writeDsb (
    EmbPattern * pattern,
    FILE * file )
```

18.63 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)

18.63.1 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.

18.63.1.1 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].

18.63.1.1.1 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.	

18.63.1.1.2 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.

18.63.2 Macro Definition Documentation

```
18.63.2.1 cci #define cci(
    c1,
    c2 ) (c1*256+c2)
```

18.63.3 Function Documentation

```
18.63.3.1 decode_record_flags() int decode_record_flags (
    unsigned char b2 )
```

```
18.63.3.2 encode_record() void encode_record (
    FILE * file,
    int x,
    int y,
    int flags )
```

```
18.63.3.3 readDst() char readDst (
    EmbPattern * pattern,
    FILE * file )
```

```
18.63.3.4 set_dst_variable() void set_dst_variable (
    EmbPattern * pattern,
    char * var,
    char * val )
```

```
18.63.3.5 writeDst() char writeDst (
    EmbPattern * pattern,
    FILE * file )
```

18.64 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)

18.64.1 Function Documentation

```
18.64.1.1 readDsz() char readDsz (
    EmbPattern * pattern,
    FILE * file )
```

18.64.1.2 ZSK USA Embroidery Format (.dsz) The ZSK USA dsz format is stitch-only.

```
18.64.1.3 writeDsz() char writeDsz (
    EmbPattern * pattern,
    FILE * file )
```

18.65 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)

18.65.1 Function Documentation

18.65.1.1 `readDxf()` char `readDxf` (
 EmbPattern * *pattern*,
 FILE * *file*)

18.65.1.2 `readLine()` void `readLine` (
 FILE * *file*,
 char * *str*)

18.65.1.3 Drawing Exchange Format (.dxf) Graphics format for drawing files designed and used by AutoDesk for their AutoCAD program. [\[dxf_reference\]](#)

18.65.1.4 `writeDxf()` char `writeDxf` (
 EmbPattern * *pattern*,
 FILE * *file*)

18.66 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)

18.66.1 Function Documentation

18.66.1.1 `readEdr()` char `readEdr` (
 EmbPattern * *pattern*,
 FILE * *file*)

18.66.1.2 Embird Embroidery Format (.edr) Stitch Only Format

18.66.1.3 `writeEdr()` char `writeEdr` (
 EmbPattern * *pattern*,
 FILE * *file*)

18.67 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*)

18.67.1 Detailed Description

The Elna Embroidery Format (.emd)
Stitch Only Format.

18.67.2 Function Documentation

18.67.2.1 `emdDecode()` char `emdDecode` (

```
    unsigned char inputByte )
```

18.67.2.2 `readEmd()` char `readEmd` (

```
    EmbPattern * pattern,
    FILE * file )
```

18.67.2.3 `writeEmd()` char `writeEmd` (

```
    EmbPattern * pattern,
    FILE * file )
```

18.68 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*)

18.68.1 Function Documentation

18.68.1.1 `expDecode()` char `expDecode` (

```
    unsigned char a1 )
```

18.68.1.2 Melco Embroidery Format (.exp) Stitch Only Format.

```
18.68.1.3 readExp() char readExp (
    EmbPattern * pattern,
    FILE * file )
```

```
18.68.1.4 writeExp() char writeExp (
    EmbPattern * pattern,
    FILE * file )
```

18.69 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)

18.69.1 Function Documentation

```
18.69.1.1 decode_exy_flags() int decode_exy_flags (
    unsigned char b2 )
```

18.69.1.2 Eltac Embroidery Format (.exy) Stitch Only Format.

```
18.69.1.3 readExy() char readExy (
    EmbPattern * pattern,
    FILE * file )
```

```
18.69.1.4 writeExy() char writeExy (
    EmbPattern * pattern,
    FILE * file )
```

18.70 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)

18.70.1 Function Documentation

```
18.70.1.1 readEys() char readEys (
    EmbPattern * pattern,
    FILE * file )
```

18.70.1.2 **Sierra Expanded Embroidery Format (.eys)** Stitch Only Format.
Smoothie G-Code Embroidery Format (.fxy)?

```
18.70.1.3 writeEys() char writeEys (
    EmbPattern * pattern,
    FILE * file )
```

18.71 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)

18.71.1 Function Documentation

```
18.71.1.1 readFxy() char readFxy (
    EmbPattern * pattern,
    FILE * file )
```

18.71.1.2 **Embroidery Format (.fxy)** Stitch Only Format.

```
18.71.1.3 writeFxy() char writeFxy (
    EmbPattern * pattern,
    FILE * file )
```

18.72 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)

18.72.1 Function Documentation

```
18.72.1.1 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

```
18.72.1.2 writeGc() char writeGc (
    EmbPattern * pattern,
    FILE * file )
```

18.73 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)

18.73.1 Function Documentation

```
18.73.1.1 readGnc() char readGnc (
    EmbPattern * pattern,
    FILE * file )
```

18.73.1.2 Great Notions Embroidery Format (.gnc) Stitch Only Format.

```
18.73.1.3 writeGnc() char writeGnc (
    EmbPattern * pattern,
    FILE * file )
```

18.74 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)

18.74.1 Function Documentation

```
18.74.1.1 readGt() char readGt (
    EmbPattern * pattern,
    FILE * file )
```

18.74.1.2 Gold Thread Embroidery Format (.gt) Stitch Only Format.

```
18.74.1.3 writeGt() char writeGt (
    EmbPattern * pattern,
    FILE * file )
```

18.75 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)

18.75.1 Function Documentation

18.75.1.1 husCompressData() unsigned char * husCompressData (

```
    unsigned char * input,
    int decompressedInputSize,
    int * compressedSize )
```

18.75.1.2 husDecodeByte() int husDecodeByte (

```
    unsigned char b )
```

18.75.1.3 husDecodeStitchType() int husDecodeStitchType (

```
    unsigned char b )
```

18.75.1.4 Husqvarna Viking Embroidery Format (.hus) Stitch Only Format.

18.75.1.5 husDecompressData() unsigned char * husDecompressData (

```
    unsigned char * input,
    int compressedInputLength,
    int decompressedContentLength )
```

18.75.1.6 husEncodeByte() unsigned char husEncodeByte (

```
    EmbReal f )
```

```
18.75.1.7 husEncodeStitchType() unsigned char husEncodeStitchType (
    int st )
```

```
18.75.1.8 readHus() char readHus (
    EmbPattern * pattern,
    FILE * file )
```

```
18.75.1.9 writeHus() char writeHus (
    EmbPattern * pattern,
    FILE * file )
```

18.76 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 **writelnb** (EmbPattern *pattern, FILE *file)

18.76.1 Function Documentation

```
18.76.1.1 readInb() char readInb (
    EmbPattern * pattern,
    FILE * file )
```

18.76.1.2 Inbro Embroidery Format (.inb) Stitch Only Format.

```
18.76.1.3 writelnb() char writeInb (
    EmbPattern * pattern,
    FILE * file )
```

18.77 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 **writelinf** (EmbPattern *pattern, FILE *file)

18.77.1 Function Documentation

```
18.77.1.1 readInf() char readInf (
    EmbPattern * pattern,
    FILE * file )
```

18.77.1.2 **Embroidery Color Format (.inf)** Stitch Only Format.

```
18.77.1.3 writelInf() char writeInf (
    EmbPattern * pattern,
    FILE * file )
```

18.78 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)

18.78.1 Function Documentation

```
18.78.1.1 jefDecode() char jefDecode (
    unsigned char inputByte )
```

```
18.78.1.2 jefEncode() void jefEncode (
    unsigned char * b,
    char dx,
    char dy,
    int flags )
```

```
18.78.1.3 jefGetHoopSize() int jefGetHoopSize (
    int width,
    int height )
```

18.78.1.4 **Janome Embroidery Format (.jef)** Stitch Only Format.

```
18.78.1.5 jefSetHoopFromId() void jefSetHoopFromId (
    EmbPattern * pattern,
    int hoopCode )
```

```
18.78.1.6 read_hoop() void read_hoop (
    FILE * file,
    struct hoop_padding * hoop,
    char * label )
```

```
18.78.1.7 readJef() char readJef (
    EmbPattern * pattern,
    FILE * file )
```

```
18.78.1.8 writeJef() char writeJef (
    EmbPattern * pattern,
    FILE * file )
```

18.79 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)

18.79.1 Function Documentation

```
18.79.1.1 ksmEncode() void ksmEncode (
    unsigned char * b,
    char dx,
    char dy,
    int flags )
```

18.79.1.2 Pfaff professional Design format (.ksm) Stitch Only Format.

```
18.79.1.3 readKsm() char readKsm (
    EmbPattern * pattern,
    FILE * file )
```

```
18.79.1.4 writeKsm() char writeKsm (
    EmbPattern * pattern,
    FILE * file )
```

18.80 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 []

18.80.1 Function Documentation

18.80.1.1 **readMax()** char readMax (

```
EmbPattern * pattern,  
FILE * file )
```

18.80.1.2 writeMax() char writeMax (

```
EmbPattern * pattern,  
FILE * file )
```

18.80.2 Variable Documentation

18.80.2.1 max_header const unsigned char max_header[]

Initial value:

18.80.2.2 Pfaff Embroidery Format (.max) Stitch Only Format.

18.81 [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)

18.81.1 Function Documentation

```
18.81.1.1 readMit() char readMit (
    EmbPattern * pattern,
    FILE * file )
```

18.81.1.2 **Mitsubishi Embroidery Format (.mit)** Stitch Only Format.

```
18.81.1.3 writeMit() char writeMit (
    EmbPattern * pattern,
    FILE * file )
```

18.82 **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)

18.82.1 Function Documentation

```
18.82.1.1 readNew() char readNew (
    EmbPattern * pattern,
    FILE * file )
```

18.82.1.2 **Ameco Embroidery Format (.new)** Stitch Only Format.

```
18.82.1.3 writeNew() char writeNew (
    EmbPattern * pattern,
    FILE * file )
```

18.83 **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)

18.83.1 Function Documentation

18.83.1.1 ofmDecode() `EmbReal ofmDecode (`
 `unsigned char b1,`
 `unsigned char b2)`

18.83.1.2 ofmReadBlockHeader() `void ofmReadBlockHeader (`
 `FILE * file)`

18.83.1.3 ofmReadClass() `static int ofmReadClass (`
 `FILE * file) [static]`

18.83.1.4 ofmReadColorChange() `void ofmReadColorChange (`
 `FILE * file,`
 `EmbPattern * pattern)`

18.83.1.5 ofmReadExpanded() `void ofmReadExpanded (`
 `FILE * file,`
 `EmbPattern * p)`

18.83.1.6 ofmReadLibrary() `char * ofmReadLibrary (`
 `FILE * file)`

18.83.1.7 Melco Embroidery Format (.ofm) Stitch Only Format.

18.83.1.8 ofmReadThreads() `void ofmReadThreads (`
 `FILE * file,`
 `EmbPattern * p)`

18.83.1.9 readOfm() `char readOfm (`
 `EmbPattern * pattern,`
 `FILE * fileCompound)`

18.83.1.10 writeOfm() `char writeOfm (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.84 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)

18.84.1 Function Documentation

18.84.1.1 `readPcd()` char `readPcd` (

```
    EmbPattern * pattern,
    const char * fileName,
    FILE * file )
```

18.84.1.2 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).

18.84.1.3 `writePcd()` char `writePcd` (

```
    EmbPattern * pattern,
    FILE * file )
```

18.85 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)

18.85.1 Function Documentation

18.85.1.1 `readPcm()` char `readPcm` (

```
    EmbPattern * pattern,
    FILE * file )
```

18.85.1.2 Pfaff Embroidery Format (.pcm) The Pfaff pcm format is stitch-only.

18.85.1.3 `writePcm()` char `writePcm` (

```
    EmbPattern * pattern,
    FILE * file )
```

18.86 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)`

18.86.1 Function Documentation

18.86.1.1 `readPcq()` char `readPcq (`
`EmbPattern * pattern,`
`const char * fileName,`
`FILE * file)`

18.86.1.2 Embroidery Format (.pcq) The Pfaff pcq format is stitch-only.

18.86.1.3 `writePcq()` char `writePcq (`
`EmbPattern * pattern,`
`FILE * file)`

18.87 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)`

18.87.1 Function Documentation

18.87.1.1 `readPcs()` char `readPcs (`
`EmbPattern * pattern,`
`const char * fileName,`
`FILE * file)`

18.87.1.2 Embroidery Format (.pcq) The Pfaff pcs format is stitch-only.

18.87.1.3 `writePcs()` char `writePcs (`
`EmbPattern * pattern,`
`FILE * file)`

18.88 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)

18.88.1 Function Documentation

18.88.1.1 `pecEncode()` void `pecEncode` (
 `FILE` * *file*,
 `EmbPattern` * *p*)

18.88.1.2 `pecEncodeJump()` void `pecEncodeJump` (
 `FILE` * *file*,
 int *x*,
 int *types*)

18.88.1.3 `pecEncodeStop()` void `pecEncodeStop` (
 `FILE` * *file*,
 unsigned char *val*)

18.88.1.4 `readPec()` char `readPec` (
 `EmbPattern` * *pattern*,
 const char * *fileName*,
 `FILE` * *file*)

18.88.1.5 `readPecStitches()` void `readPecStitches` (
 `EmbPattern` * *pattern*,
 `FILE` * *file*)

18.88.1.6 Embroidery Format (.pec) The Brother pec format is stitch-only.

18.88.1.7 `writelImage()` void `writelImage` (
 `FILE` * *file*,
 unsigned char *image*[][48])

Write a PES embedded *image* to the given *file* pointer.

18.88.1.8 `writePec()` char `writePec` (
 `EmbPattern` * *pattern*,
 const char * *fileName*,
 `FILE` * *file*)

```
18.88.1.9 writePecStitches() void writePecStitches (
    EmbPattern * pattern,
    FILE * file,
    const char * fileName )
```

18.89 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)

18.89.1 Function Documentation

```
18.89.1.1 readPel() char readPel (
    EmbPattern * pattern,
    FILE * file )
```

18.89.1.2 Embroidery Format (.pec) The Brother pel format is stitch-only.

```
18.89.1.3 writePel() char writePel (
    EmbPattern * pattern,
    FILE * file )
```

18.90 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)

18.90.1 Function Documentation

```
18.90.1.1 readPem() char readPem (
    EmbPattern * pattern,
    FILE * file )
```

18.90.1.2 Embroidery Format (.pec) The Brother pem format is stitch-only.

```
18.90.1.3 writePem() char writePem (
    EmbPattern * pattern,
    FILE * file )
```

18.91 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

18.91.1 Function Documentation

18.91.1.1 pesWriteEmbOneSection() void pesWriteEmbOneSection (

```
    EmbPattern * pattern,
    FILE * file )
```

18.91.1.2 pesWriteSewSegSection() void pesWriteSewSegSection (

```
    EmbPattern * pattern,
    FILE * file )
```

18.91.1.3 readDescriptions() void readDescriptions (

```
    FILE * file,
    EmbPattern * pattern )
```

18.91.1.4 readFeatherPatterns() void readFeatherPatterns (

```
    FILE * file,
    EmbPattern * pattern )
```

18.91.1.5 `readHoopName()` void readHoopName (FILE * file, EmbPattern * pattern)

18.91.1.6 `readImageString()` void readImageString (FILE * file, EmbPattern * pattern)

18.91.1.7 `readMotifPatterns()` void readMotifPatterns (FILE * file, EmbPattern * pattern)

18.91.1.8 `readPes()` char readPes (EmbPattern * pattern, const char * fileName, FILE * file)

18.91.1.9 `readPESHeaderV10()` void readPESHeaderV10 (FILE * file, EmbPattern * pattern)

18.91.1.10 `readPESHeaderV5()` void readPESHeaderV5 (FILE * file, EmbPattern * pattern)

18.91.1.11 `readPESHeaderV6()` void readPESHeaderV6 (FILE * file, EmbPattern * pattern)

18.91.1.12 `readPESHeaderV7()` void readPESHeaderV7 (FILE * file, EmbPattern * pattern)

18.91.1.13 `readPESHeaderV8()` void readPESHeaderV8 (FILE * file, EmbPattern * pattern)

18.91.1.14 `readPESHeaderV9()` void readPESHeaderV9 (FILE * file, EmbPattern * pattern)

18.91.1.15 `readProgrammableFills()` void readProgrammableFills (FILE * file, EmbPattern * pattern)

```
18.91.1.16 readThreads() void readThreads (
    FILE * file,
    EmbPattern * pattern )
```

```
18.91.1.17 writePes() char writePes (
    EmbPattern * pattern,
    const char * fileName,
    FILE * file )
```

18.91.2 Variable Documentation

```
18.91.2.1 pes_version int pes_version = PES0001
```

```
18.91.2.2 pes_version_strings const char* pes_version_strings[]
```

Initial value:

```
= {
    "#PES0001",
    "#PES0020",
    "#PES0022",
    "#PES0030",
    "#PES0040",
    "#PES0050",
    "#PES0055",
    "#PES0056",
    "#PES0060",
    "#PES0070",
    "#PES0080",
    "#PES0090",
    "#PES0100",
}
```

18.91.2.3 Embroidery Format (.pec) The Brother pes format is stitch-only.

18.92 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)

18.92.1 Function Documentation

```
18.92.1.1 readPhb() char readPhb (
    EmbPattern * pattern,
    FILE * file )
```

18.92.1.2 Embroidery Format (.pec) The Brother phb format is stitch-only.

```
18.92.1.3 writePhb() char writePhb (
    EmbPattern * pattern,
    FILE * file )
```

18.93 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)

18.93.1 Function Documentation

18.93.1.1 `readPhc()` char `readPhc` (
 `EmbPattern` * pattern,
 `FILE` * file)

18.93.1.2 Embroidery Format (.pec) The Brother phc format is stitch-only.

18.93.1.3 `writePhc()` char `writePhc` (
 `EmbPattern` * pattern,
 `FILE` * file)

18.94 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)

18.94.1 Function Documentation

18.94.1.1 `readPlt()` char `readPlt` (
 `EmbPattern` * pattern,
 `FILE` * file)

18.94.1.2 Embroidery Format (.plt) The AutoCAD plt format is stitch-only.

18.94.1.3 `writePlt()` char `writePlt` (
 `EmbPattern` * pattern,
 `FILE` * file)

18.95 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*)

18.95.1 Function Documentation

18.95.1.1 `readRgb()` `char readRgb (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.95.1.2 Color File (.rgb) The RGB format is a color-only format to act as an external color file for other formats.

18.95.1.3 `writeRgb()` `char writeRgb (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.96 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*)

18.96.1 Function Documentation

18.96.1.1 `readSew()` `char readSew (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.96.1.2 `sewDecode()` `char sewDecode (`
 `unsigned char inputByte)`

18.96.1.3 Embroidery Format (.sew) The Janome sew format is stitch-only.

18.96.1.4 `writeSew()` `char writeSew (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.97 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*)

18.97.1 Function Documentation

18.97.1.1 `readShv()` char `readShv` (
 `EmbPattern` * *pattern*,
 `FILE` * *file*)

18.97.1.2 `shvDecode()` char `shvDecode` (
 unsigned char *inputByte*)

18.97.1.3 Viking Embroidery Format (.shv) The Husqvarna Viking shv format is stitch-only.

18.97.1.4 `shvDecodeShort()` short `shvDecodeShort` (
 unsigned short *inputByte*)

18.97.1.5 `writeShv()` char `writeShv` (
 `EmbPattern` * *pattern*,
 `FILE` * *file*)

18.98 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*)

18.98.1 Function Documentation

18.98.1.1 `readSst()` char `readSst` (
 `EmbPattern` * *pattern*,
 `FILE` * *file*)

18.98.1.2 Embroidery Format (.sst) The Sunstar sst format is stitch-only.

18.98.1.3 writeSst() char writeSst (

```
EmbPattern * pattern,
FILE * file )
```

18.99 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)`

18.99.1 Function Documentation

18.99.1.1 readStx() char readStx (

```
EmbPattern * pattern,
FILE * file )
```

18.99.1.2 stxReadThread() int stxReadThread (

```
StxThread * thread,
FILE * file )
```

18.99.1.3 Stitch Embroidery Format (.stx) The Data Stitch stx format is stitch-only.

18.99.1.4 writeStx() char writeStx (

```
EmbPattern * pattern,
FILE * file )
```

18.100 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]`

18.100.1 Function Documentation

18.100.1.1 `readSvg()` char `readSvg (`
 `EmbPattern * pattern,`
 `FILE * file)`

18.100.1.2 `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`.

18.100.2 Variable Documentation

18.100.2.1 `attributeList` `SvgAttribute attributeList [1000]`

18.100.2.2 `current_element_id` int `current_element_id`

18.100.2.3 `currentAttribute` char `currentAttribute[1000]`

18.100.2.4 `currentValue` char `currentValue[1000]`

18.100.2.5 `n_attributes` int `n_attributes = 0`

18.100.2.6 `svgCreator` int `svgCreator`

18.100.2.7 Vector Graphics (.svg) The scalable vector graphics (SVG) format is a graphics format maintained by ...

18.100.2.8 `svgExpect` int `svgExpect`

18.100.2.9 `svgMultiValue` int `svgMultiValue`

18.101 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*)

18.101.1 Function Documentation

18.101.1.1 `readT01()` char `readT01` (
 `EmbPattern` * *pattern*,
 `FILE` * *file*)

18.101.1.2 Embroidery Format (.pcq) The Pfaff t01 format is stitch-only.

18.101.1.3 `writeT01()` char `writeT01` (
 `EmbPattern` * *pattern*,
 `FILE` * *file*)

18.102 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*)

18.102.1 Function Documentation

18.102.1.1 `readT09()` char `readT09` (
 `EmbPattern` * *pattern*,
 `FILE` * *file*)

18.102.1.1.1 Embroidery Format (.pcq) The Pfaff t09 format is stitch-only.

18.102.1.2 `writeT09()` char `writeT09` (
 `EmbPattern` * *pattern*,
 `FILE` * *file*)

18.103 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)

18.103.1 Function Documentation

18.103.1.1 decode_tap_record_flags() int decode_tap_record_flags (unsigned char b2)

18.103.1.2 encode_tap_record() void encode_tap_record (FILE * file, int x, int y, int flags)

18.103.1.3 Embroidery Format (.tap) The Happy tap format is stitch-only.

18.103.1.4 readTap() char readTap (EmbPattern * pattern, FILE * file)

18.103.1.5 writeTap() char writeTap (EmbPattern * pattern, FILE * file)

18.104 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)

18.104.1 Function Documentation

```
18.104.1.1 readThr() char readThr (
    EmbPattern * pattern,
    FILE * file )
```

18.104.1.2 **Embroidery Format (.thr)** The ThreadWorks thr format is stitch-only.

```
18.104.1.3 writeThr() char writeThr (
    EmbPattern * pattern,
    FILE * file )
```

18.105 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)

18.105.1 Function Documentation

```
18.105.1.1 readTxt() char readTxt (
    EmbPattern * pattern,
    FILE * file )
```

18.105.1.2 **File (.txt)** The txt format is stitch-only and isn't associated with a specific company.

```
18.105.1.3 writeTxt() char writeTxt (
    EmbPattern * pattern,
    FILE * file )
```

18.106 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)

18.106.1 Function Documentation

```
18.106.1.1 readU00() char readU00 (
    EmbPattern * pattern,
    FILE * file )
```

18.106.1.2 Embroidery Format (.u00) The Barudan u00 format is stitch-only.

```
18.106.1.3 writeU00() char writeU00 (
    EmbPattern * pattern,
    FILE * file )
```

18.107 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`)

18.107.1 Function Documentation

```
18.107.1.1 readU01() char readU01 (
    EmbPattern * pattern,
    FILE * file )
```

18.107.1.2 Embroidery Format (.u00) The Barudan u01 format is stitch-only.

```
18.107.1.3 writeU01() char writeU01 (
    EmbPattern * pattern,
    FILE * file )
```

18.108 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` []

18.108.1 Function Documentation

```
18.108.1.1 readVip() char readVip (
    EmbPattern * pattern,
    FILE * file )
```

```
18.108.1.2 vipCompressData() unsigned char * vipCompressData (
    unsigned char * input,
    int decompressedInputSize,
    int * compressedSize )
```

```
18.108.1.3 vipDecodeByte() int vipDecodeByte (
    unsigned char b )
```

```
18.108.1.4 vipDecodeStitchType() int vipDecodeStitchType (
    unsigned char b )
```

```
18.108.1.5 vipDecompressData() unsigned char * vipDecompressData (
    unsigned char * input,
    int compressedInputLength,
    int decompressedContentLength )
```

```
18.108.1.6 vipEncodeByte() unsigned char vipEncodeByte (
    EmbReal f )
```

```
18.108.1.7 vipEncodeStitchType() unsigned char vipEncodeStitchType (
    int st )
```

```
18.108.1.8 writeVip() char writeVip (
    EmbPattern * pattern,
    FILE * file )
```

18.108.2 Variable Documentation

18.108.2.1 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, 0x4, 0x95, 0xF0, 0x3B, 0xEF,
    0xF7, 0x40, 0x24, 0x18, 0x39, 0x31, 0xBB, 0xE1, 0x53, 0xA8, 0x1F, 0xB1, 0x3A, 0x07, 0xFB, 0xCB,
    0xE6, 0x0, 0x81, 0x50, 0x0E, 0x40, 0xE1, 0x2C, 0x73, 0x50, 0xD, 0x91, 0xD6, 0x0A, 0x5D, 0xD6,
    0x8B, 0xB8, 0x62, 0xAE, 0x47, 0x0, 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, 0x0,
    0x0E, 0x68, 0x6A, 0x1E, 0x09, 0x85, 0xC0, 0x53, 0x81, 0xD1, 0x98, 0x89, 0xAF, 0xE8, 0x85, 0x4F,
    0xE3, 0x69, 0x89, 0x03, 0x1, 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, 0x05, 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, 0x28, 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, 0x0A, 0x14, 0x0A, 0x6B,
0x40, 0x26, 0xA0, 0x88, 0xD1, 0x62, 0x6A, 0xB3, 0x50, 0x12, 0x89, 0x9B, 0xB5, 0x83, 0x9B, 0x37
}

```

18.108.2.2 Embroidery Format (.pcq)

The Pfaff vip format is stitch-only.

18.109 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)`

18.109.1 Function Documentation

18.109.1.1 `readVp3()` `char readVp3 (`

```

    EmbPattern * pattern,
    FILE * file )

```

18.109.1.2 `vp3Decode()` `int vp3Decode (`

```

    unsigned char inputByte )

```

18.109.1.3 `vp3DecodeInt16()` `short vp3DecodeInt16 (`

```

    unsigned short inputByte )

```

18.109.1.4 `vp3PatchByteCount()` `void vp3PatchByteCount (`

```

    FILE * file,
    int offset,
    int adjustment )

```

18.109.1.5 vp3ReadHoopSection() `vp3Hoop vp3ReadHoopSection (`
`FILE * file)`

18.109.1.6 vp3ReadString() `unsigned char * vp3ReadString (`
`FILE * file)`

18.109.1.7 Embroidery Format (.pcq) The Pfaff vp3 format is stitch-only.

18.109.1.8 vp3WriteString() `void vp3WriteString (`
`FILE * file,`
`const char * str)`

18.109.1.9 vp3WriteStringLen() `void vp3WriteStringLen (`
`FILE * file,`
`const char * str,`
`int len)`

18.109.1.10 writeVp3() `char writeVp3 (`
`EmbPattern * pattern,`
`FILE * file)`

18.110 `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)`

18.110.1 Function Documentation

18.110.1.1 readXxx() `char readXxx (`
`EmbPattern * pattern,`
`FILE * file)`

18.110.1.2 writeXxx() `char writeXxx (`
`EmbPattern * pattern,`
`FILE * file)`

18.110.1.3 xxxDecodeByte() `char xxxDecodeByte (`
`unsigned char inputByte)`

18.110.1.4 Embroidery Format (.xxx) The Singer xxx format is stitch-only.

```
18.110.1.5 xxxEncodeDesign() void xxxEncodeDesign (
    FILE * file,
    EmbPattern * p )
```

```
18.110.1.6 xxxEncodeStitch() void xxxEncodeStitch (
    FILE * file,
    EmbReal deltaX,
    EmbReal deltaY,
    int flags )
```

```
18.110.1.7 xxxEncodeStop() void xxxEncodeStop (
    FILE * file,
    EmbStitch s )
```

18.111 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)`

18.111.1 Detailed Description

The ZSK USA Embroidery Format (.zsk)

The ZSK USA zsk format is stitch-only.

18.111.2 Function Documentation

```
18.111.2.1 readZsk() char readZsk (
    EmbPattern * pattern,
    FILE * file )
```

```
18.111.2.2 writeZsk() char writeZsk (
    EmbPattern * pattern,
    FILE * file )
```

18.112 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.

18.112.1 Function Documentation

18.112.1.1 **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).

18.112.1.2 **embGeometry_free()** `void embGeometry_free (` `EmbGeometry * obj)`

Free the memory occupied by a non-stitch geometry object.

obj Pointer to geometry memory.

18.112.1.3 **embGeometry_init()** `EmbGeometry * embGeometry_init (` `int type_in)`

Our generic object interface backends to each individual type.

type_in

Returns

`EmbGeometry*`

18.112.1.4 **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.

18.112.1.5 **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.

18.113 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)`

18.113.1 Function Documentation

18.113.1.1 Arc_clockwise() `char Arc_clockwise ()`

18.113.1.2 **Base_objectRubberPoint()** `EmbVector` `Base_objectRubberPoint (`
 `EmbGeometry * obj,`
 `const char * key)`

18.113.1.3 **Base_objectRubberText()** `const char *` `Base_objectRubberText (`
 `EmbGeometry * obj,`
 `const char * key)`

18.113.1.4 **Base_setLineType()** `void` `Base_setLineType (`
 `EmbGeometry * obj,`
 `int lineType)`

18.113.1.5 **Base_setLineWeight()** `void` `Base_setLineWeight (`
 `EmbGeometry * obj,`
 `float lineWeight)`

18.113.1.6 **clockwise()** `char` `clockwise (`
 `EmbGeometry * obj)`

18.113.1.7 **embArc_arcLength()** `float` `embArc_arcLength (`
 `EmbArc arc)`

18.113.1.8 **embArc_area()** `float` `embArc_area (`
 `EmbArc arc)`

18.113.1.9 **embArc_chord()** `float` `embArc_chord (`
 `EmbArc arc)`

18.113.1.10 **embArc_clockwise()** `char` `embArc_clockwise (`
 `EmbArc arc)`

18.113.1.11 **embArc_endAngle()** `float` `embArc_endAngle (`
 `EmbArc arc)`

18.113.1.12 **embArc_gripEdit()** `void` `embArc_gripEdit (`
 `EmbArc * arc,`
 `EmbVector before,`
 `EmbVector after)`

18.113.1.13 **embArc_includedAngle()** `float` `embArc_includedAngle (`
 `EmbArc arc)`

18.113.1.14 embArc_init() `EmbArc` `embArc_init` (
 `void`)

18.113.1.15 embArc_mouseSnapPoint() `EmbVector` `embArc_mouseSnapPoint` (
 `EmbArc` `arc`,
 `EmbVector` `mousePoint`)

18.113.1.16 embArc_paint() `void` `embArc_paint` (
 `void`)

18.113.1.17 embArc_setCenter() `void` `embArc_setCenter` (
 `EmbArc` * `arc`,
 `EmbVector` `point`)

18.113.1.18 embArc_setEndAngle() `void` `embArc_setEndAngle` (
 `EmbArc` * `arc`,
 `float` `angle`)

18.113.1.19 embArc_setRadius() `void` `embArc_setRadius` (
 `EmbArc` * `arc`,
 `float` `radius`)

18.113.1.20 embArc_setStartAngle() `void` `embArc_setStartAngle` (
 `EmbArc` * `arc`,
 `float` `angle`)

18.113.1.21 embArc_startAngle() `float` `embArc_startAngle` (
 `EmbArc` `arc`)

18.113.1.22 embArc_updatePath() `void` `embArc_updatePath` (
 `EmbArc` `arc`)

18.113.1.23 embArc_updateRubber() `void` `embArc_updateRubber` (
 `EmbArc` `arc`,
 `int` `pattern`,
 `int` `layer`,
 `int` `index`)

18.113.1.24 embBaseSetColorRGB() `void` `embBaseSetColorRGB` (
 `EmbGeometry` * `obj`,
 `unsigned int` `rgb`)

18.113.1.25 embCircle_prompt() `void` `embCircle_prompt` (
 `const char` * `str`)

18.113.1.26 `embCircle_setArea()` `void embCircle_setArea (`
`EmbCircle * circle,`
`float area)`

18.113.1.27 `embCircle_setCircumference()` `void embCircle_setCircumference (`
`EmbCircle * circle,`
`float circumference)`

18.113.1.28 `embEllipse_click()` `void embEllipse_click (`
`float x,`
`float y)`

18.113.1.29 `embEllipse_main()` `void embEllipse_main ()`

18.113.1.30 `embRect_bottomLeft()` `EmbVector embRect_bottomLeft (`
`EmbRect rect)`

18.113.1.31 `embRect_bottomRight()` `EmbVector embRect_bottomRight (`
`EmbRect rect)`

18.113.1.32 `getArcCenter()` `void getArcCenter (`
`EmbArc arc,`
`EmbVector * arcCenter)`

18.113.1.33 `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)`

18.113.1.34 `set_object_color()` `void set_object_color (`
`EmbGeometry * obj,`
`EmbColor color)`

18.114 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)

18.114.1 Function Documentation

18.114.1.1 embCircle_area() EmbReal embCircle_area (

```
EmbCircle circle )
```

18.114.1.2 embCircle_circumference() EmbReal embCircle_circumference (

```
EmbCircle circle )
```

18.114.1.3 embCircle_init() EmbCircle embCircle_init (

```
void )
```

18.114.1.4 getCircleCircleIntersections() int getCircleCircleIntersections (

```
EmbCircle c0,
EmbCircle c1,
EmbVector * p0,
EmbVector * p1 )
```

18.114.1.5 getCircleTangentPoints() int getCircleTangentPoints (

```
EmbCircle c,
EmbVector point,
EmbVector * t0,
EmbVector * t1 )
```

18.115 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` ()

18.115.1 Function Documentation

18.115.1.1 `ellipse_objectQuadrant0()` `EmbVector` `ellipse_objectQuadrant0` (
`EmbEllipse` * `ellipse`)

18.115.1.2 `ellipse_objectQuadrant180()` `EmbVector` `ellipse_objectQuadrant180` (
`EmbEllipse` * `ellipse`)

18.115.1.3 `ellipse_objectQuadrant270()` `EmbVector` `ellipse_objectQuadrant270` (
`EmbEllipse` * `ellipse`)

18.115.1.4 `ellipse_objectQuadrant90()` `EmbVector` `ellipse_objectQuadrant90` (
`EmbEllipse` * `ellipse`)

18.115.1.5 `embEllipse_area()` `EmbReal` `embEllipse_area` (
`EmbEllipse` `ellipse`)

18.115.1.6 `embEllipse_diameterX()` `EmbReal` `embEllipse_diameterX` (
`EmbEllipse` `ellipse`)

18.115.1.7 `embEllipse_diameterY()` `EmbReal` `embEllipse_diameterY` (
`EmbEllipse` `ellipse`)

18.115.1.8 `embEllipse_height()` `EmbReal` `embEllipse_height` (
`EmbEllipse` `ellipse`)

18.115.1.9 `embEllipse_init()` `EmbEllipse` `embEllipse_init` (
`void`)

18.115.1.10 `embEllipse_perimeter()` `EmbReal` `embEllipse_perimeter` (
`EmbEllipse` `ellipse`)

18.115.1.11 embEllipse_setDiameterMajor() void embEllipse_setDiameterMajor (EmbEllipse * ellipse,
float diameter)

18.115.1.12 embEllipse_setDiameterMinor() void embEllipse_setDiameterMinor (EmbEllipse * ellipse,
float diameter)

18.115.1.13 embEllipse_setRadiusMajor() void embEllipse_setRadiusMajor (float radius)

18.115.1.14 embEllipse_setRadiusMinor() void embEllipse_setRadiusMinor (float radius)

18.115.1.15 embEllipse_setSize() void embEllipse_setSize (float width,
float height)

18.115.1.16 embEllipse_updatePath() void embEllipse_updatePath ()

18.115.1.17 embEllipse_width() EmbReal embEllipse_width (EmbEllipse ellipse)

18.116 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)

18.116.1 Function Documentation

18.116.1.1 degrees() EmbReal degrees (EmbReal radian) [inline]

18.116.1.2 emb_round() int emb_round (EmbReal x)

```
18.116.1.3 radians() EmbReal radians (
    EmbReal degree ) [inline]
```

18.117 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)

18.117.1 Function Documentation

```
18.117.1.1 embLine_intersectionPoint() EmbVector embLine_intersectionPoint (
    EmbLine line1,
    EmbLine line2 )
```

```
18.117.1.2 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)

```
18.117.1.3 embLine_toVector() EmbVector embLine_toVector (
    EmbLine line )
```

18.118 extern/libembroidery/src/geometry/path.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "../embroidery.h"
```

18.119 extern/libembroidery/src/geometry/polygon.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "../embroidery.h"
```

18.120 extern/libembroidery/src/geometry/polyline.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "../embroidery.h"
```

18.121 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)`

18.121.1 Function Documentation

18.121.1.1 `embRect_area()` `EmbReal embRect_area (`
`EmbRect rect)`

18.121.1.2 `embRect_init()` `EmbRect embRect_init (`
`void)`

18.122 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)`

18.122.1 Function Documentation

18.122.1.1 `textSingle_gripEdit()` `void textSingle_gripEdit (`
`EmbVector before,`
`EmbVector after)`

18.122.1.2 `textSingle_mouseSnapPoint()` `EmbVector` `textSingle_mouseSnapPoint (`
`EmbVector mousePoint)`

18.122.1.3 `textSingle_paint()` `void` `textSingle_paint ()`

18.122.1.4 `textSingle_setJustify()` `void` `textSingle_setJustify (`
`const char * justify)`

18.122.1.5 `textSingle_setTextBackward()` `void` `textSingle_setTextBackward (`
`char val)`

18.122.1.6 `textSingle_setTextBold()` `void` `textSingle_setTextBold (`
`char val)`

18.122.1.7 `textSingle_setTextFont()` `void` `textSingle_setTextFont (`
`const char * font)`

18.122.1.8 `textSingle_setTextItalic()` `void` `textSingle_setTextItalic (`
`char val)`

18.122.1.9 `textSingle_setTextOverline()` `void` `textSingle_setTextOverline (`
`char val)`

18.122.1.10 `textSingle_setTextSize()` `void` `textSingle_setTextSize (`
`float size)`

18.122.1.11 `textSingle_setTextStrikeOut()` `void` `textSingle_setTextStrikeOut (`
`char val)`

18.122.1.12 `textSingle_setTextStyle()` `void` `textSingle_setTextStyle (`
`char bold,`
`char italic,`
`char under,`
`char strike,`
`char over)`

18.122.1.13 `textSingle_setTextUnderline()` `void` `textSingle_setTextUnderline (`
`char val)`

18.122.1.14 `textSingle_setTextUpsideDown()` `void` `textSingle_setTextUpsideDown (`
`char val)`

18.122.1.15 textSingle_updateRubber() void textSingle_updateRubber ()

18.123 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)

18.123.1 Function Documentation

18.123.1.1 embVector_add() EmbVector **embVector_add** (

```
    EmbVector a,
    EmbVector b )
```

The sum of vectors *a* and *b* returned as a vector.

Equivalent to:

$$\mathbf{c} = \mathbf{a} + \mathbf{b} = \begin{pmatrix} a_x + b_x \\ a_y + b_y \end{pmatrix}$$

18.123.1.2 embVector_angle() EmbReal **embVector_angle** (

```
    EmbVector v )
```

The angle, measured anti-clockwise from the x-axis, of a vector *v*.

18.123.1.3 embVector_average() EmbVector **embVector_average** (

```
    EmbVector a,
    EmbVector b )
```

The average of vectors *v1* and *v2* returned as a vector.

Equivalent to:

$$\mathbf{c} = \frac{\mathbf{a} + \mathbf{b}}{2} = \begin{pmatrix} \frac{a_x + b_x}{2} \\ \frac{a_y + b_y}{2} \end{pmatrix}$$

```
18.123.1.4 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:

$$|c| = \left| \begin{pmatrix} a_x \\ a_y \\ 0 \end{pmatrix} \times \begin{pmatrix} b_x \\ b_y \\ 0 \end{pmatrix} \right| = \left| \begin{pmatrix} 0 \\ 0 \\ a_x b_y - a_y b_x \end{pmatrix} \right| = a_x b_y - a_y b_x$$

```
18.123.1.5 embVector_distance() EmbReal embVector_distance (
    EmbVector a,
    EmbVector b )
```

The distance between *a* and *b* returned as a real value.

$$d = |\mathbf{a} - \mathbf{b}| = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$$

```
18.123.1.6 embVector_dot() EmbReal embVector_dot (
    EmbVector a,
    EmbVector b )
```

The dot product as vectors *v1* and *v2* returned as a EmbReal.

Equivalent to:

$$c = \mathbf{a} \cdot \mathbf{b} = a_x b_x + a_y b_y$$

```
18.123.1.7 embVector_length() EmbReal embVector_length (
    EmbVector vector )
```

The length or absolute value of the vector *vector*.

Equivalent to:

$$|v| = \sqrt{v_x^2 + v_y^2}$$

```
18.123.1.8 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.

```
18.123.1.9 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.

```
18.123.1.10 embVector_relativeX() EmbReal embVector_relativeX (
    EmbVector a1,
    EmbVector a2,
    EmbVector a3 )
```

The x-component of the vector

```
18.123.1.11 embVector_relativeY() EmbReal embVector_relativeY (
    EmbVector a1,
    EmbVector a2,
    EmbVector a3 )
```

The y-component of the vector

```
18.123.1.12 embVector_subtract() EmbVector embVector_subtract (
    EmbVector v1,
    EmbVector v2 )
```

The difference between vectors *v1* and *v2* returned as *result*.

Equivalent to:

$$\mathbf{c} = \mathbf{a} - \mathbf{b} = \begin{pmatrix} a_x - b_x \\ a_y - b_y \end{pmatrix}$$

```
18.123.1.13 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 $v1^T \cdot v2$ for our vectors *v1* and *v2* so a "component-wise product". The result is stored at the pointer *result*.

That is $(1 \ 0) \cdot (a) = (xa \ (x \ y)(0 \ 1) \cdot (yb)$

```
18.123.1.14 embVector_unit() EmbVector embVector_unit (
    EmbReal alpha )
```

The unit vector in the direction *angle*.

$$\mathbf{a}_\alpha = \begin{pmatrix} \cos(\alpha) \\ \sin(\alpha) \end{pmatrix}$$

18.124 extern/libembroidery/src/image.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "embroidery_internal.h"
```

Functions

- void `writelImage` (FILE *file, unsigned char image[][48])
- float `image_diff` (unsigned char *a, unsigned char *b, int size)

18.124.1 Detailed Description

This backends to the stb libraries and nanosvg library.

Use Python PEP7 for coding style.

18.124.2 Function Documentation

```
18.124.2.1 image_diff() float image_diff (
    unsigned char * a,
    unsigned char * b,
    int size )
```

The distance between the arrays *a* and *b* of length *size*.

```
18.124.2.2 writelImage() void writeImage (
    FILE * file,
    unsigned char image[ ][48] )
```

Write a PES embedded *image* to the given *file* pointer.

18.125 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_SIERPINSKI_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` * `CompoundFileDialog` (const unsigned int maxNumberOfDirectoryEntries)
maxNumberOfDirectoryEntries
- `EmbTime parseTime` (FILE *file)
file
- `bcf_directory_entry` * `CompoundFileDialogEntry` (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 `WHITE SPACE` [] = " \t\n\r"

18.125.1 Macro Definition Documentation

18.125.1.1 FLAG_CIRCLE #define FLAG_CIRCLE 12

18.125.1.2 FLAG_CIRCLE_SHORT #define FLAG_CIRCLE_SHORT 13

18.125.1.3 FLAG_COMBINE #define FLAG_COMBINE 35

18.125.1.4 FLAG_CROSS_STITCH #define FLAG_CROSS_STITCH 36

18.125.1.5 FLAG_ELLIPSE #define FLAG_ELLIPSE 14

18.125.1.6 FLAG_ELLIPSE_SHORT #define FLAG_ELLIPSE_SHORT 15

18.125.1.7 FLAG_FILL #define FLAG_FILL 32

18.125.1.8 FLAG_FILL_SHORT #define FLAG_FILL_SHORT 33

18.125.1.9 FLAG_FORMATS #define FLAG_FORMATS 4

18.125.1.10 FLAG_FORMATS_SHORT #define FLAG_FORMATS_SHORT 5

18.125.1.11 FLAG_FULL_TEST_SUITE #define FLAG_FULL_TEST_SUITE 29

18.125.1.12 FLAG_HELP #define FLAG_HELP 2

18.125.1.13 FLAG_HELP_SHORT #define FLAG_HELP_SHORT 3

18.125.1.14 FLAG_HILBERT_CURVE #define FLAG_HILBERT_CURVE 30

18.125.1.15 FLAG_LINE #define FLAG_LINE 16

18.125.1.16 FLAG_LINE_SHORT #define FLAG_LINE_SHORT 17

18.125.1.17 FLAG_POLYGON #define FLAG_POLYGON 18

18.125.1.18 FLAG_POLYGON_SHORT #define FLAG_POLYGON_SHORT 19

18.125.1.19 FLAG_POLYLINE #define FLAG_POLYLINE 20

18.125.1.20 FLAG_POLYLINE_SHORT #define FLAG_POLYLINE_SHORT 21

18.125.1.21 FLAG QUIET #define FLAG QUIET 6

18.125.1.22 FLAG QUIET_SHORT #define FLAG QUIET_SHORT 7

18.125.1.23 FLAG_RENDER #define FLAG_RENDER 22

18.125.1.24 FLAG_RENDER_SHORT #define FLAG_RENDER_SHORT 23

18.125.1.25 FLAG_SATIN #define FLAG_SATIN 24

18.125.1.26 FLAG_SATIN_SHORT #define FLAG_SATIN_SHORT 25

18.125.1.27 FLAG_SIERPINSKI_TRIANGLE #define FLAG_SIERPINSKI_TRIANGLE 31

18.125.1.28 FLAG_SIMULATE #define FLAG_SIMULATE 34

18.125.1.29 FLAG_STITCH #define FLAG_STITCH 26

18.125.1.30 FLAG_STITCH_SHORT #define FLAG_STITCH_SHORT 27

18.125.1.31 FLAG_TEST #define FLAG_TEST 28

18.125.1.32 FLAG_TO #define FLAG_TO 0

18.125.1.33 FLAG_TO_SHORT #define FLAG_TO_SHORT 1

18.125.1.34 FLAG_VERBOSE #define FLAG_VERBOSE 8

18.125.1.35 FLAG_VERBOSE_SHORT #define FLAG_VERBOSE_SHORT 9

18.125.1.36 FLAG_VERSION #define FLAG_VERSION 10

18.125.1.37 FLAG_VERSION_SHORT #define FLAG_VERSION_SHORT 11

18.125.1.38 NUM_FLAGS #define NUM_FLAGS 37

18.125.2 Function Documentation

18.125.2.1 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*`

18.125.2.2 bcf_directory_free() `void bcf_directory_free (bcf_directory ** dir)`
dir

18.125.2.3 bcf_file_free() `void bcf_file_free (bcf_file * bcfFile)`
bcfFile

18.125.2.4 bcfFile_read() `int bcfFile_read (FILE * file, bcf_file * bcfFile)`
file *bcfFile*

Returns

`int`

18.125.2.5 bcfFileFat_create() `bcf_file_fat * bcfFileFat_create (const unsigned int sectorSize)`
sectorSize

Returns

`bcf_file_fat*`

18.125.2.6 bcfFileHeader_read() `bcf_file_header bcfFileHeader_read (FILE * file)`
file

Returns

`bcf_file_header`

18.125.2.7 binaryReadString() void binaryReadString (

```
FILE * file,
char * buffer,
int maxLength )
```

file buffer maxLength

18.125.2.8 binaryReadUnicodeString() void binaryReadUnicodeString (

```
FILE * file,
char * buffer,
const int stringLength )
```

file buffer stringLength

18.125.2.9 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.

18.125.2.10 CompoundFileDirectory() [bcf_directory](#) * CompoundFileDirectory (

```
const unsigned int maxNumberOfDirectoryEntries )
```

maxNumberOfDirectoryEntries

Returns

[bcf_directory](#)*

18.125.2.11 CompoundFileDirectoryEntry() [bcf_directory_entry](#) * CompoundFileDirectoryEntry (

```
FILE * file )
```

file

Returns

[bcf_directory_entry](#)*

18.125.2.12 copy_trim() char * copy_trim (

```
char const * s )
```

s

Returns

char*

Todo decription

18.125.2.13 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*`

18.125.2.14 emb_readline() `int emb_readline (`
 `FILE * file,`
 `char * line,`
 `int maxLength)`

file line maxLength

Returns

`int`

18.125.2.15 embArc_print() `void embArc_print (`
 `EmbArc arc)`

arc

Todo move to `arc.c`

18.125.2.16 embColor_distance() `int embColor_distance (`
 `EmbColor a,`
 `EmbColor b)`

a b

Returns

`int`

18.125.2.17 embColor_read() `void embColor_read (`
 `FILE * f,`
 `EmbColor * c,`
 `int toRead)`

f c toRead

18.125.2.18 embColor_write() `void embColor_write (`
 `FILE * f,`
 `EmbColor c,`
 `int toWrite)`

f c toWrite

18.125.2.19 embSatinOutline_generateSatinOutline() `void embSatinOutline_generateSatinOutline (`
 `EmbArray * lines,`
 `EmbReal thickness,`
 `EmbSatinOutline * result)`

lines thickness result

18.125.2.20 embSatinOutline_renderStitches() `EmbArray * embSatinOutline_renderStitches (`
 `EmbSatinOutline * result,`
 `EmbReal density)`

result density

Returns

`EmbArray*`

18.125.2.21 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.

18.125.2.22 embThread_findNearestThread() `int embThread_findNearestThread (`
 `EmbColor color,`
 `EmbThread * thread_list,`
 `int n_threads)`

color *thread_list* *n_threads*

Returns

`int`

18.125.2.23 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.

18.125.2.24 embTime_initNow() `void embTime_initNow (`
 `EmbTime * t)`

t

18.125.2.25 embTime_time() `EmbTime embTime_time (`
 `EmbTime * t)`

t

Returns

`EmbTime`

18.125.2.26 embVector_print() void embVector_print (

```
EmbVector v,
char * label )
```

v label
move to [vector.c](#)

18.125.2.27 entriesInDifatSector() unsigned int entriesInDifatSector (

```
bcf_file_difat * fat )
```

fat
Returns

unsigned int

18.125.2.28 get_trim_bounds() void get_trim_bounds (

```
char const * s,
char const ** firstWord,
char const ** trailingSpace )
```

Get the trim bounds object.

s firstWord trailingSpace

18.125.2.29 GetFile() FILE * GetFile (

```
bcf_file * bcfFile,
FILE * file,
char * fileToFind )
```

Get the File object.

bcfFile file fileToFind

Returns

FILE*

18.125.2.30 haveExtraDIFATSectors() int haveExtraDIFATSectors (

```
bcf_file * file )
```

file

Returns

int

18.125.2.31 loadFatFromSector() void loadFatFromSector (

```
bcf_file_fat * fat,
FILE * file )
```

fat file

18.125.2.32 parseDIFATSectors() void parseDIFATSectors (

```
FILE * file,
bcf_file * bcfFile )
```

file bcfFile

18.125.2.33 parseDirectoryEntryName() void parseDirectoryEntryName (

```
FILE * file,
bcf_directory_entry * dir )
```

file dir

18.125.2.34 parseTime() `EmbTime` `parseTime (`
 `FILE * file)`
file

Returns

`EmbTime`

18.125.2.35 readFullSector() `unsigned int` `readFullSector (`
 `FILE * file,`
 `bcf_file_difat * bcfFile,`
 `unsigned int * difatEntriesToRead)`
file bcfFile difatEntriesToRead

Returns

`unsigned int`

18.125.2.36 readNextSector() `void` `readNextSector (`
 `FILE * file,`
 `bcf_directory * dir)`
file dir

18.125.2.37 sectorSize() `unsigned int` `sectorSize (`
 `bcf_file * bcfFile)`

bcfFile

Returns

`unsigned int`

18.125.2.38 seekToSector() `int` `seekToSector (`
 `bcf_file * bcfFile,`
 `FILE * file,`
 `const unsigned int sector)`
bcfFile file sector

Returns

`int`

18.125.2.39 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.

```
18.125.2.40 write_24bit() void write_24bit (
    FILE * file,
    int x )
file x
```

18.125.3 Variable Documentation

18.125.3.1 black_thread `EmbThread` `black_thread = { { 0, 0, 0 }, "Black", "Black" }`

18.125.3.2 difatEntriesInHeader `const unsigned int difatEntriesInHeader = 109`

18.125.3.3 emb_error `int emb_error = 0`
Error code storage for optional control flow blocking.

18.125.3.4 emb_verbose `int emb_verbose = 0`
Verbosity level.

18.125.3.5 embConstantPi `const EmbReal embConstantPi = 3.1415926535`

18.125.3.6 sizeOfChainingEntryAtEndOfDifatSector `const unsigned int sizeOfChainingEntryAtEndOfDifatSector = 4`

18.125.3.7 sizeOfDifatEntry `const unsigned int sizeOfDifatEntry = 4`

18.125.3.8 sizeOfDirectoryEntry `const unsigned int sizeOfDirectoryEntry = 128`

18.125.3.9 sizeOfFatEntry `const unsigned int sizeOfFatEntry = sizeof(unsigned int)`

18.125.3.10 WHITESPACE `char const WHITESPACE[] = " \t\n\r"`

18.126 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*)

18.126.1 Detailed Description

The file is for the management of the main struct: `EmbPattern`.

18.126.2 Function Documentation

```
18.126.2.1 convert() int convert (
    const char * inf,
    const char * outf )
```

```
18.126.2.2 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.

```
18.126.2.3 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.

```
18.126.2.4 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.

```
18.126.2.5 embPattern_addPathAbs() void embPattern_addPathAbs (
    EmbPattern * p,
    EmbPath obj )
```

```
18.126.2.6 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.

```
18.126.2.7 embPattern_addPolygonAbs() void embPattern_addPolygonAbs (
    EmbPattern * p,
    EmbPolygon obj )
```

```
18.126.2.8 embPattern_addPolylineObjectAbs() void embPattern_addPolylineObjectAbs (
    EmbPattern * p,
    EmbPolyline obj )
```

```
18.126.2.9 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.

```
18.126.2.10 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.

```
18.126.2.11 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.

18.126.2.12 embPattern_addThread() `int embPattern_addThread (`
 `EmbPattern * pattern,`
 `EmbThread thread)`

pattern thread

Returns

int

18.126.2.13 embPattern_calcBoundingBox() `EmbRect embPattern_calcBoundingBox (`
 `EmbPattern * p)`

Returns an EmbRect that encapsulates all stitches and objects in the pattern (*p*).

18.126.2.14 embPattern_center() `void embPattern_center (`
 `EmbPattern * p)`

Center the pattern *p*.

18.126.2.15 embPattern_changeColor() `void embPattern_changeColor (`
 `EmbPattern * p,`
 `int index)`

Change the currentColorIndex of pattern *p* to *index*.

18.126.2.16 embPattern_color_count() `int embPattern_color_count (`
 `EmbPattern * pattern,`
 `EmbColor startColor)`

18.126.2.17 embPattern_combineJumpStitches() `void embPattern_combineJumpStitches (`
 `EmbPattern * p)`

p

18.126.2.18 embPattern_copyPolylinesToStitch_list() `void embPattern_copyPolylinesToStitch_list (`
 `EmbPattern * p)`

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

18.126.2.19 embPattern_copystitch_listToPolylines() `void embPattern_copystitch_listToPolylines (`
 `EmbPattern * p)`

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

18.126.2.20 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.

18.126.2.21 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*

18.126.2.22 embPattern_designDetails() void embPattern_designDetails (EmbPattern * pattern)

18.126.2.23 embPattern_end() void embPattern_end (EmbPattern * p)

18.126.2.24 embPattern_fixColorCount() void embPattern_fixColorCount (EmbPattern * p)
p

18.126.2.25 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.

18.126.2.26 embPattern_flipHorizontal() void embPattern_flipHorizontal (EmbPattern * p)

Flips the entire pattern (*p*) horizontally about the y-axis.

18.126.2.27 embPattern_flipVertical() void embPattern_flipVertical (EmbPattern * p)

Flips the entire pattern (*p*) vertically about the x-axis.

18.126.2.28 embPattern_free() void embPattern_free (EmbPattern * p)

Frees all memory allocated in the pattern (*p*).

18.126.2.29 embPattern_hideStitchesOverLength() void embPattern_hideStitchesOverLength (EmbPattern * p,
int length)

p length

18.126.2.30 embPattern_jumpStitches() int embPattern_jumpStitches (EmbPattern * pattern)

18.126.2.31 embPattern_lengthHistogram() void embPattern_lengthHistogram (EmbPattern * pattern,
int * bin,
int NUMBINS)

18.126.2.32 embPattern_loadExternalColorFile() void embPattern_loadExternalColorFile (EmbPattern * p,
const char * fileName)

TODO: Description needed.

18.126.2.33 embPattern_maximumStitchLength() float embPattern_maximumStitchLength (EmbPattern * pattern)

18.126.2.34 embPattern_minimumStitchLength() float embPattern_minimumStitchLength (EmbPattern * pattern)

18.126.2.35 embPattern_movePolylineObjectListToStitchList() void embPattern_movePolylineObjectListToStitchList (EmbPattern * p)

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

18.126.2.36 embPattern_movingStitchListToPolylineObjectList() void embPattern_movingStitchListToPolylineObjectList (EmbPattern * p)

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

18.126.2.37 embPattern_realStitches() int embPattern_realStitches (EmbPattern * pattern)

18.126.2.38 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.

18.126.2.39 embPattern_totalStitchLength() float embPattern_totalStitchLength (EmbPattern * pattern)

pattern

Returns

float

18.126.2.40 embPattern_trimStitches() int embPattern_trimStitches (EmbPattern * pattern)

18.127 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** []

18.127.1 Function Documentation

18.127.1.1 threadColor() int threadColor (

```
    const char * name,
    int brand )
```

18.127.1.2 threadColorName() const char * threadColorName (

```
    unsigned int color,
    int brand )
```

18.127.1.3 threadColorNum() int threadColorNum (

```
    unsigned int color,
    int brand )
```

18.127.2 Variable Documentation

18.127.2.1 _dxfColorTable const unsigned char _dxfColorTable[][3] = {{ 0, 0, 0 }}

18.127.2.2 brand_codes [thread_color*](#) brand_codes[]

18.127.2.3 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"
}
```

18.127.2.4 husThreads const [EmbThread](#) husThreads[] = {{{ 0, 0, 0 }, "END", "END" }}}

18.127.2.5 jefThreads const [EmbThread](#) jefThreads[] = {{{ 0, 0, 0 }, "END", "END" }}}

18.127.2.6 pcmThreads const [EmbThread](#) pcmThreads[] = {{{ 0, 0, 0 }, "END", "END" }}}

18.127.2.7 pecThreadCount const int pecThreadCount = 65

18.127.2.8 pecThreads const EmbThread pecThreads[] = {{{ 0, 0, 0 }, "END", "END"}}

18.127.2.9 shvThreadCount const int shvThreadCount = 42

18.127.2.10 shvThreads const EmbThread shvThreads[] = {{{ 0, 0, 0 }, "END", "END"}}

18.128 **privacy_policy.md** File Reference

References

- [1] acatina. Technical info. [511](#)
- [2] KDE Community. Projects/liberty/file formats/tajima ternary - kde community wiki. [511](#)
- [3] G. van Rossum and B. Warsaw. Python pep 7. [20](#)

Index

_appName
 embroidermodder.cpp, 334
_appVer
 embroidermodder.cpp, 334
_bcf_directory, 50
 dirEntries, 50
 maxNumberOfDirectoryEntries, 50
_bcf_directory_entry, 50
 childId, 51
 CLSID, 51
 colorFlag, 51
 creationTime, 51
 directoryEntryName, 51
 directoryEntryNameLength, 51
 leftSiblingId, 51
 modifiedTime, 52
 next, 52
 objectType, 52
 rightSiblingId, 52
 startingSectorLocation, 52
 stateBits, 52
 streamSize, 52
 streamSizeHigh, 52
_bcf_file, 52
 difat, 53
 directory, 53
 fat, 53
 header, 53
_bcf_file_difat, 53
 fatSectorCount, 53
 fatSectorEntries, 54
 sectorSize, 54
_bcf_file_fat, 54
 fatEntries, 54
 fatEntryCount, 54
 numberOfEntriesInFatSector, 54
_bcf_file_header, 54
 byteOrder, 55
 CLSID, 55
 firstDifatSectorLocation, 55
 firstDirectorySectorLocation, 55
 firstMiniFATSectorLocation, 55
 majorVersion, 55
 miniSectorShift, 56
 miniStreamCutoffSize, 56
 minorVersion, 56
 numberOfDifatSectors, 56
 numberOfDirectorySectors, 56
 numberOfFATSectors, 56
 numberOfMiniFatSectors, 56
 reserved1, 56
 reserved2, 56
 sectorShift, 56
 signature, 56
 transactionSignatureNumber, 57
_dxIColorTable
 embroidery.h, 435
 thread-color.c, 576
_mainWin
 Application, 62
 mainwindow.cpp, 384
_subMask
 format_csd.c, 507
_vp3Hoop, 57
 bottom, 57
 bottom2, 57
 byte1, 57
 byte2, 58
 byte3, 58
 height, 58
 left, 58
 left2, 58
 numberOfBytesRemaining, 58
 numberOfColors, 58
 right, 58
 right2, 58
 threadLength, 58
 top, 58
 top2, 59
 unknown2, 59
 unknown3, 59
 unknown4, 59
 width, 59
 xOffset, 59
 yOffset, 59
_xorMask
 format_csd.c, 507
~ArcObject
 ArcObject, 67
~BaseObject
 BaseObject, 76
~CircleObject
 CircleObject, 84
~CmdPrompt
 CmdPrompt, 89
~CmdPromptHandle
 CmdPromptHandle, 96
~CmdPromptHistory
 CmdPromptHistory, 98
~CmdPromptInput
 CmdPromptInput, 101
~CmdPromptSplitter
 CmdPromptSplitter, 107
~DimLeaderObject
 DimLeaderObject, 113
~EllipseObject
 EllipseObject, 119
~EmbDetailsDialog
 EmbDetailsDialog, 129
~ImageObject

ImageObject, 157
 ~ImageWidget
 ImageWidget, 161
 ~LayerManager
 LayerManager, 163
 ~LineObject
 LineObject, 166
 ~MainWindow
 MainWindow, 177
 ~MdiArea
 MdiArea, 210
 ~MdiWindow
 MdiWindow, 215
 ~PathObject
 PathObject, 226
 ~PointObject
 PointObject, 231
 ~PolygonObject
 PolygonObject, 236
 ~PolylineObject
 PolylineObject, 241
 ~PreviewDialog
 PreviewDialog, 245
 ~PropertyEditor
 PropertyEditor, 247
 ~RectObject
 RectObject, 260
 ~SaveObject
 SaveObject, 264
 ~Settings_Dialog
 Settings_Dialog, 284
 ~TextSingleObject
 TextSingleObject, 300
 ~UndoEditor
 UndoEditor, 316
 ~View
 View, 320
 10o, 16, 502
 100, 16, 502
 about
 MainWindow, 178
 accept_
 settings-dialog.cpp, 398
 acceptChanges
 Settings_Dialog, 284
 Action
 embroidermodder.h, 339
 Action___, 59
 aliases, 60
 command, 60
 hash, 60
 icon, 60
 menu_name, 60
 menu_position, 60
 script, 60
 shortcut, 60
 statustip, 60
 toolbar_name, 60
 toolbar_position, 60
 tooltip, 61
 action_table
 embroidermodder.h, 345
 mainwindow.cpp, 384
 actionHash
 MainWindow, 204
 activeCommand
 CmdPrompt, 89
 MainWindow, 178
 activeMdiWindow
 MainWindow, 178
 activeScene
 MainWindow, 178
 activeUndoStack
 MainWindow, 178
 activeView
 MainWindow, 178
 actuator
 MainWindow, 178
 addArc
 SaveObject, 264
 addBlock
 SaveObject, 264
 addCircle
 SaveObject, 264
 addColorsToComboBox
 Settings_Dialog, 284
 addCommand
 CmdPrompt, 89
 CmdPromptInput, 101
 addDimAligned
 SaveObject, 265
 addDimAngular
 SaveObject, 265
 addDimArcLength
 SaveObject, 265
 addDimDiameter
 SaveObject, 266
 addDimLeader
 SaveObject, 266
 addDimLinear
 SaveObject, 266
 addDimOrdinate
 SaveObject, 266
 addDimRadius
 SaveObject, 266
 addEllipse
 SaveObject, 266
 addEllipseArc
 SaveObject, 266
 addGrid
 SaveObject, 267
 addHatch
 SaveObject, 267
 addImage
 SaveObject, 267

addInfiniteLine
 SaveObject, 267
addLayer
 LayerManager, 163
addLine
 SaveObject, 267
addObject
 View, 320
addPath
 SaveObject, 267
addPoint
 SaveObject, 267
addPolygon
 SaveObject, 267
addPolyline
 SaveObject, 268
addRay
 SaveObject, 268
addRectangle
 SaveObject, 268
addSlot
 SaveObject, 268
addSpline
 SaveObject, 268
addStack
 UndoEditor, 316
addTextMulti
 SaveObject, 268
addTextSingle
 SaveObject, 268
addToRubberRoom
 View, 320
after
 UndoableGripEditCommand, 309
alert
 CmdPrompt, 89
aliases
 Action__, 60
aliasHash
 CmdPromptInput, 106
alignScenePointWithViewPoint
 View, 321
allGripPoints
 ArcObject, 67
 BaseObject, 76
 CircleObject, 84
 DimLeaderObject, 113
 EllipseObject, 120
 ImageObject, 158
 LineObject, 167
 PathObject, 226
 PointObject, 231
 PolygonObject, 236
 PolylineObject, 242
 RectObject, 260
 TextSingleObject, 300
allowRubber
 View, 321
allowZoomIn
 View, 321
allowZoomOut
 View, 321
alpha
 SelectBox, 271
alphabet
 LSYSTEM, 170
Ameco, 472, 523
angle
 UndoableRotateCommand, 314
appendHistory
 CmdPrompt, 89
 CmdPromptHistory, 98
 CmdPromptInput, 102
appendTheHistory
 CmdPrompt, 89
Application, 61
 _mainWin, 62
 Application, 61
 event, 62
 setMainWin, 62
applyFormatting
 CmdPromptHistory, 99
 CmdPromptInput, 102
arc
 EmbGeometry_, 133
arc.c
 Arc_clockwise, 546
 Base_objectRubberPoint, 546
 Base_objectRubberText, 547
 Base_setLineType, 547
 Base_setLineWidth, 547
 clockwise, 547
 embArc_arcLength, 547
 embArc_area, 547
 embArc_chord, 547
 embArc_clockwise, 547
 embArc_endAngle, 547
 embArc_gripEdit, 547
 embArc_includedAngle, 547
 embArc_init, 547
 embArc_mouseSnapPoint, 548
 embArc_paint, 548
 embArc_setCenter, 548
 embArc_setEndAngle, 548
 embArc_setRadius, 548
 embArc_setStartAngle, 548
 embArc_startAngle, 548
 embArc_updatePath, 548
 embArc_updateRubber, 548
 embBaseSetColorRGB, 548
 embCircle_prompt, 548
 embCircle_setArea, 548
 embCircle_setCircumference, 549
 embEllipse_click, 549
 embEllipse_main, 549
 embRect_bottomLeft, 549

embRect_bottomRight, 549
 getArcCenter, 549
 getArcDataFromBulge, 549
 set_object_color, 549
Arc_clockwise
 arc.c, 546
Arc_Polyester
 embroidery.h, 410
Arc_Rayon
 embroidery.h, 411
arcEndPoint
 ArcObject, 74
arcMidPoint
 ArcObject, 74
ArcObject, 62
 ~ArcObject, 67
 allGripPoints, 67
 arcEndPoint, 74
 arcMidPoint, 74
 ArcObject, 66
 arcStartPoint, 74
 calculateArcData, 67
 gripEdit, 68
 init, 68
 mouseSnapPoint, 68
 objectArcLength, 69
 objectArea, 69
 objectChord, 69
 objectClockwise, 69
 objectEndAngle, 69
 objectEndPoint, 70
 objectEndX, 70
 objectEndY, 70
 objectIncludedAngle, 70
 objectMidPoint, 70
 objectMidX, 71
 objectMidY, 71
 objectRadius, 71
 objectStartAngle, 71
 object startPoint, 71
 objectStartX, 71
 objectStartY, 72
 paint, 72
 setObjectEndAngle, 72
 setObjectEndPoint, 72, 73
 setObjectMidPoint, 73
 setObjectRadius, 73
 setObjectStartAngle, 73
 setObjectStartPoint, 73
 Type, 66
 type, 73
 updateArcRect, 73
 updatePath, 74
 updateRubber, 74
 vulcanize, 74
arcEndPoint
 ArcObject, 74
array.c

embArray_addArc, 399
 embArray_addCircle, 400
 embArray_addEllipse, 400
 embArray_addFlag, 400
 embArray_addLine, 400
 embArray_addPath, 400
 embArray_addPoint, 400
 embArray_addPolygon, 400
 embArray_addPolyline, 400
 embArray_addRect, 400
 embArray_addStitch, 400
 embArray_addVector, 400
 embArray_copy, 401
 embArray_create, 401
 embArray_free, 401
 embArray_resize, 401
ArrowStyle
 DimLeaderObject, 112
arrowStyleAngle
 DimLeaderObject, 116
arrowStyleLength
 DimLeaderObject, 116
arrowStylePath
 DimLeaderObject, 116
art, 16, 503
assets_dir
 Settings_, 274
attributeList
 format_svg.c, 536
attributeOffset
 VipHeader_, 331
AutoCAD, 475, 513, 532
AutoDesk, 513
auxFormat
 ThredExtension_, 304
axiom
 LSYSTEM, 170
b
 EmbColor_, 128
 Barudan, 476, 509, 540
Base_objectRubberPoint
 arc.c, 546
Base_objectRubberText
 arc.c, 547
Base_setLineType
 arc.c, 547
Base_setLineWidth
 arc.c, 547
BaseObject, 75
 ~BaseObject, 76
 allGripPoints, 76
BaseObject, 76
 boundingRect, 76
 drawRubberLine, 77
 gripEdit, 77
 line, 77
 lineWeightPen, 77
 lwtPen, 81

mouseSnapPoint, 77
objectCenter, 77
objectCenterX, 77
objectCenterY, 77
objectColor, 77
objectColorRGB, 78
objectID, 78
objectLineType, 78
objectLineWidth, 78
objectPath, 78
objectPen, 78
objectRubberMode, 78
objectRubberPoint, 78
objectRubberText, 78
objID, 81
objLine, 81
objPen, 81
objRubberMode, 81
objRubberPoints, 81
objRubberTexts, 81
realRender, 78
rect, 78
setLine, 79
setObjectCenter, 79
setObjectCenterX, 79
setObjectCenterY, 79
setObjectColor, 79
setObjectColorRGB, 79
setObjectLineType, 79
setObjectLineWidth, 79
setObjectPath, 80
setObjectRubberMode, 80
setObjectRubberPoint, 80
setObjectRubberText, 80
setRect, 80
shape, 80
Type, 76
type, 80
vulcanize, 80
bcf_difat_create
 embroidery_internal.h, 461
 main.c, 564
bcf_directory
 embroidery_internal.h, 459
bcf_directory_entry
 embroidery_internal.h, 459
bcf_directory_free
 embroidery_internal.h, 461
 main.c, 564
bcf_file
 embroidery_internal.h, 459
bcf_file_difat
 embroidery_internal.h, 459
bcf_file_difat_free
 embroidery_internal.h, 461
bcf_file_fat
 embroidery_internal.h, 459
bcf_file_fat_free
 embroidery_internal.h, 461
 main.c, 564
bcf_file_header
 embroidery_internal.h, 459
bcfFile_read
 embroidery_internal.h, 461
 main.c, 564
bcfFileFat_create
 embroidery_internal.h, 461
 main.c, 564
bcfFileHeader_isValid
 embroidery_internal.h, 461
bcfFileHeader_read
 embroidery_internal.h, 462
 main.c, 564
before
 UndoableGripEditCommand, 309
Bernina, 503
beziers
 EmbSpline_, 144
bgColor
 MdiArea, 213
bgLogo
 MdiArea, 213
bgTexture
 MdiArea, 213
binaryReadString
 embroidery_internal.h, 462
 main.c, 564
binaryReadUnicodeString
 embroidery_internal.h, 462
 main.c, 565
binaryWriteInt
 embroidery_internal.h, 462
 formats.c, 499
binaryWriteIntBE
 embroidery_internal.h, 462
 formats.c, 499
binaryWriteShort
 embroidery_internal.h, 462
 formats.c, 499
binaryWriteUInt
 embroidery_internal.h, 462
 formats.c, 499
binaryWriteUIntBE
 embroidery_internal.h, 462
 formats.c, 499
binaryWriteUShort
 embroidery_internal.h, 463
 formats.c, 499
binaryWriteUShortBE
 embroidery_internal.h, 463
 formats.c, 499
bit_position
 Compress, 108
Bitmap Cache, 504

Bits and Volts, 504
bits_total
 Compress, 108
black_thread
 embroidery.h, 435
 main.c, 570
blink
 CmdPrompt, 89
blinkState
 CmdPrompt, 95
blinkTimer
 CmdPrompt, 95
block_elements
 Compress, 109
bmc, 504
bottom
 _vp3Hoop, 57
 EmbRect_, 142
 hoop_padding, 153
bottom2
 _vp3Hoop, 57
boundingRect
 BaseObject, 76
 EmbDetailsDialog, 129
Box
 DimLeaderObject, 112
boxDir
 SelectBox, 271
brand_codes
 thread-color.c, 576
brand_codes_files
 thread-color.c, 576
bro, 16, 504
Brother, 473, 474, 527, 528, 531, 532
BuildDecryptionTable
 format_csd.c, 506
BULGETOCONTROL
 embroidery_internal.h, 451
BULGETOEND
 embroidery_internal.h, 452
buttonBox
 EmbDetailsDialog, 129
 Settings_Dialog, 291
buttonCustomFilterClearAll
 Settings_Dialog, 284
buttonCustomFilterClearAllClicked
 Settings_Dialog, 284
buttonCustomFilterSelectAll
 Settings_Dialog, 284
buttonCustomFilterSelectAllClicked
 Settings_Dialog, 284
buttonQSnapClearAll
 Settings_Dialog, 284
buttonQSnapClearAllClicked
 Settings_Dialog, 284
buttonQSnapSelectAll
 Settings_Dialog, 284
buttonQSnapSelectAllClicked
 Settings_Dialog, 284
Settings_Dialog, 284
buttonTipOfTheDayClicked
 MainWindow, 179
byte1
 _vp3Hoop, 57
byte2
 _vp3Hoop, 58
byte3
 _vp3Hoop, 58
byteOrder
 _bcf_file_header, 55
calculateArcData
 ArcObject, 67
canRedo
 UndoEditor, 316
canUndo
 UndoEditor, 316
cascade
 MdiArea, 210
catalogNumber
 EmbThread_, 146
cci
 format_dst.c, 511
center
 EmbCircle_, 127
 EmbEllipse_, 131
 UiObject_, 306
 View, 321
centerAt
 View, 321
changeFormatting
 CmdPromptInput, 102
changelog
 MainWindow, 179
character_huffman
 Compress, 109
character_length_huffman
 Compress, 109
check_for_color_file
 EmbFormatList_, 131
check_header_present
 embroidery_internal.h, 463
 main.c, 565
checkBoxCustomFilterStateChanged
 Settings_Dialog, 284
checkBoxDisableBGStateChanged
 Settings_Dialog, 284
checkBoxGeneralMdiBGUseColorStateChanged
 Settings_Dialog, 285
checkBoxGeneralMdiBGUseLogoStateChanged
 Settings_Dialog, 285
checkBoxGeneralMdiBGUseTextureStateChanged
 Settings_Dialog, 285
checkBoxGridCenterOnOriginStateChanged
 Settings_Dialog, 285
checkBoxGridColorMatchCrossHairStateChanged
 Settings_Dialog, 285
checkBoxGridLoadFromFileStateChanged
 Settings_Dialog, 285

Settings_Dialog, 285
checkBoxGridShowOnLoadStateChanged
 Settings_Dialog, 285
checkBoxGridShowOriginStateChanged
 Settings_Dialog, 285
checkBoxLwtRealRenderStateChanged
 Settings_Dialog, 285
checkBoxLwtShowLwtStateChanged
 Settings_Dialog, 285
checkBoxPromptSaveHistoryAsHtmlStateChanged
 Settings_Dialog, 285
checkBoxPromptSaveHistoryStateChanged
 Settings_Dialog, 285
checkBoxRenderHintAAStateChanged
 Settings_Dialog, 286
checkBoxRenderHintHighAAStateChanged
 Settings_Dialog, 286
checkBoxRenderHintNonCosmeticStateChanged
 Settings_Dialog, 286
checkBoxRenderHintSmoothPixStateChanged
 Settings_Dialog, 286
checkBoxRenderHintTextAAStateChanged
 Settings_Dialog, 286
checkBoxRulerShowOnLoadStateChanged
 Settings_Dialog, 286
checkBoxSelectionModePickAddStateChanged
 Settings_Dialog, 286
checkBoxSelectionModePickDragStateChanged
 Settings_Dialog, 286
checkBoxSelectionModePickFirstStateChanged
 Settings_Dialog, 286
checkBoxShowScrollBarsStateChanged
 Settings_Dialog, 286
checkBoxTipOfTheDay
 MainWindow, 204
checkBoxTipOfTheDayStateChanged
 MainWindow, 179
 Settings_Dialog, 286
checkBoxUseOpenGLStateChanged
 Settings_Dialog, 286
checkChangedText
 CmdPromptInput, 102
checkCursorPosition
 CmdPromptInput, 102
checkEditedText
 CmdPromptInput, 102
checkForUpdates
 MainWindow, 179
checkSelection
 CmdPromptInput, 102
childId
 _bcf_directory_entry, 51
chooseDisplayBackgroundColor
 Settings_Dialog, 287
chooseDisplayCrossHairColor
 Settings_Dialog, 287
chooseDisplaySelectBoxLeftColor
 Settings_Dialog, 287
chooseDisplaySelectBoxLeftFill
 Settings_Dialog, 287
chooseDisplaySelectBoxRightColor
 Settings_Dialog, 287
chooseDisplaySelectBoxRightFill
 Settings_Dialog, 287
chooseGeneralMdiBackgroundColor
 Settings_Dialog, 287
chooseGeneralMdiBackgroundLogo
 Settings_Dialog, 287
chooseGeneralMdiBackgroundTexture
 Settings_Dialog, 287
chooseGridColor
 Settings_Dialog, 287
choosePromptBackgroundColor
 Settings_Dialog, 287
choosePromptTextColor
 Settings_Dialog, 287
chooseRulerColor
 Settings_Dialog, 287
CHUNK_SIZE
 embroidery.h, 411
circle
 EmbGeometry_, 133
circle.c
 embCircle_area, 550
 embCircle_circumference, 550
 embCircle_init, 550
 getCircleCircleIntersections, 550
 getCircleTangentPoints, 550
CIRCLE_MODE_1P_DIA
 embroidermodder.h, 343
CIRCLE_MODE_1P_RAD
 embroidermodder.h, 343
CIRCLE_MODE_2P
 embroidermodder.h, 343
CIRCLE_MODE_3P
 embroidermodder.h, 343
CIRCLE_MODE_TTR
 embroidermodder.h, 343
CircleObject, 81
 ~CircleObject, 84
 allGripPoints, 84
 CircleObject, 84
 gripEdit, 84
 init, 84
 mouseSnapPoint, 85
 objectArea, 85
 objectCircumference, 85
 objectDiameter, 85
 objectQuadrant0, 85
 objectQuadrant180, 85
 objectQuadrant270, 85
 objectQuadrant90, 85
 objectRadius, 85
 objectSavePath, 85
 paint, 86
 setObjectArea, 86

setObjectCircumference, 86
 setObjectDiameter, 86
 setObjectRadius, 86
 Type, 84
 type, 86
 updatePath, 86
 updateRubber, 86
 vulcanize, 86
 clearAllFields
 PropertyEditor, 247
 clearFormatting
 CmdPromptInput, 102
 clearRubberRoom
 View, 321
 clearSelection
 View, 321
 clockwise
 arc.c, 547
 Closed
 DimLeaderObject, 112
 closeEvent
 MainWindow, 179
 MdiWindow, 216
 closeToolBar
 MainWindow, 179
 CLSID
 _bcf_directory_entry, 51
 _bcf_file_header, 55
 cmdActive
 CmdPromptInput, 106
 CmdPrompt, 87
 ~CmdPrompt, 89
 activeCommand, 89
 addCommand, 89
 alert, 89
 appendHistory, 89
 appendTheHistory, 89
 blink, 89
 blinkState, 95
 blinkTimer, 95
 CmdPrompt, 89
 copyPressed, 89
 cutPressed, 90
 deletePressed, 90
 disableRapidFire, 90
 downPressed, 90
 enableRapidFire, 90
 endCommand, 90
 escapePressed, 90
 F10Pressed, 90
 F11Pressed, 90
 F12Pressed, 90
 F1Pressed, 90
 F2Pressed, 91
 F3Pressed, 91
 F4Pressed, 91
 F5Pressed, 91
 F6Pressed, 91
 F7Pressed, 91
 F8Pressed, 91
 F9Pressed, 91
 floatingChanged, 91
 getCurrentText, 91
 getHistory, 91
 getPrefix, 92
 historyAppended, 92
 isCommandActive, 92
 isRapidFireEnabled, 92
 lastCommand, 92
 pastePressed, 92
 processInput, 92
 promptDivider, 95
 promptHistory, 95
 promptInput, 95
 promptSplitter, 95
 promptVBoxLayout, 95
 redoPressed, 92
 resizeTheHistory, 92
 runCommand, 92
 saveHistory, 92
 selectAllPressed, 93
 setCurrentText, 93
 setHistory, 93
 setPrefix, 93
 setPromptBackgroundColor, 93
 setPromptFontFamily, 93
 setPromptFontSize, 93
 setPromptFontStyle, 93
 setPromptTextColor, 93
 shiftPressed, 93
 shiftReleased, 94
 showSettings, 94
 startBlinking, 94
 startCommand, 94
 startResizingTheHistory, 94
 stopBlinking, 94
 stopResizingTheHistory, 94
 styleHash, 95
 tabPressed, 94
 undoPressed, 94
 updateStyle, 94
 upPressed, 94
 CmdPromptHandle, 95
 ~CmdPromptHandle, 96
 CmdPromptHandle, 96
 handleMoved, 96
 handlePressed, 96
 handleReleased, 97
 mouseMoveEvent, 97
 mousePressEvent, 97
 mouseReleaseEvent, 97
 moveY, 97
 pressY, 97
 releaseY, 97
 CmdPromptHistory, 97
 ~CmdPromptHistory, 98

appendHistory, 98
applyFormatting, 99
CmdPromptHistory, 98
contextMenuEvent, 99
historyAppended, 99
resizeHistory, 99
startResizeHistory, 99
stopResizeHistory, 99
tmpHeight, 99
CmdPromptInput, 100
 ~CmdPromptInput, 101
 addCommand, 101
 aliasHash, 106
 appendHistory, 102
 applyFormatting, 102
 changeFormatting, 102
 checkChangedText, 102
 checkCursorPosition, 102
 checkEditedText, 102
 checkSelection, 102
 clearFormatting, 102
 cmdActive, 106
 CmdPromptInput, 101
 contextMenuEvent, 102
 copyClip, 102
 copyPressed, 103
 curCmd, 106
 curText, 106
 cutPressed, 103
 defaultPrefix, 106
 deletePressed, 103
 downPressed, 103
 endCommand, 103
 escapePressed, 103
 eventFilter, 103
 F10Pressed, 103
 F11Pressed, 103
 F12Pressed, 103
 F1Pressed, 103
 F2Pressed, 104
 F3Pressed, 104
 F4Pressed, 104
 F5Pressed, 104
 F6Pressed, 104
 F7Pressed, 104
 F8Pressed, 104
 F9Pressed, 104
 isBlinking, 106
 lastCmd, 106
 pasteClip, 104
 pastePressed, 104
 prefix, 106
 processInput, 104
 rapidFireEnabled, 106
 redoPressed, 105
 runCommand, 105
 selectAllPressed, 105
 shiftPressed, 105
shiftReleased, 105
showSettings, 105
startCommand, 105
stopBlinking, 105
tabPressed, 105
undoPressed, 105
updateCurrentText, 105
upPressed, 106
CmdPromptSplitter, 107
 ~CmdPromptSplitter, 107
 CmdPromptSplitter, 107
 createHandle, 107
 moveResizeHistory, 108
 pressResizeHistory, 108
 releaseResizeHistory, 108
cnd, 16, 505
CoatsAndClark_Rayon
 embroidery.h, 411
CODE_OF_CONDUCT.md, 332
col, 16, 505
color
 EmbGeometry_, 133
 EmbLine_, 137
 EmbPath_, 139
 EmbPoint_, 141
 EmbStitch_, 144
 EmbThread_, 146
 UiObject_, 306
color_only
 EmbFormatList_, 131
colorChanges
 EmbDetailsDialog, 129
colorCode
 StxThread_, 296
 SubDescriptor_, 296
colorFlag
 _bcf_directory_entry, 51
colorLength
 VipHeader_, 331
colorName
 StxThread_, 296
 SubDescriptor_, 297
colorSelector
 MainWindow, 204
colorSelectorIndexChanged
 MainWindow, 179
colorTotal
 EmbDetailsDialog, 129
comboBoxArcClockwise
 property-editor.cpp, 389
comboBoxes
 data.cpp, 332
 embroidermodder.h, 345
comboBoxGridTypeCurrentIndexChanged
 Settings_Dialog, 287
comboBoxIconSizeCurrentIndexChanged
 Settings_Dialog, 287
comboBoxIconThemeCurrentIndexChanged

Settings_Dialog, 288
comboBoxLanguageCurrentIndexChanged
 Settings_Dialog, 288
comboBoxPathClosed
 property-editor.cpp, 389
comboBoxPathVertexNum
 property-editor.cpp, 389
comboBoxPolylineClosed
 property-editor.cpp, 389
comboBoxPolylineVertexNum
 property-editor.cpp, 389
comboBoxPromptFontFamilyCurrentIndexChanged
 Settings_Dialog, 288
comboBoxPromptFontStyleCurrentIndexChanged
 Settings_Dialog, 288
comboBoxQSnapLocatorColorCurrentIndexChanged
 Settings_Dialog, 288
comboBoxRulerMetricCurrentIndexChanged
 Settings_Dialog, 288
comboBoxScrollBarWidgetCurrentIndexChanged
 Settings_Dialog, 288
comboBoxSelected
 PropertyEditor, 255
comboBoxSelectionCoolGripColorCurrentIndexChanged
 Settings_Dialog, 288
comboBoxSelectionHotGripColorCurrentIndexChanged
 Settings_Dialog, 288
comboBoxTextSingleBackward
 property-editor.cpp, 390
comboBoxTextSingleFont
 data.cpp, 332
 embroidermodder.h, 345
comboBoxTextSingleJustify
 property-editor.cpp, 390
comboBoxTextSingleUpsideDown
 property-editor.cpp, 390
command
 Action_, 60
 UiObject_, 306
CompoundFileDirectory
 embroidery_internal.h, 463
 main.c, 565
CompoundFileDirectoryEntry
 embroidery_internal.h, 463
 main.c, 565
CompoundFileSector_DIFAT_Sector
 embroidery_internal.h, 452
CompoundFileSector_EndOfChain
 embroidery_internal.h, 452
CompoundFileSector_FAT_Sector
 embroidery_internal.h, 452
CompoundFileSector_FreeSector
 embroidery_internal.h, 452
CompoundFileSector_MaxRegSector
 embroidery_internal.h, 452
CompoundFileStreamId_MaxRegularStreamId
 embroidery_internal.h, 452
CompoundFileStreamId_NoStream
 embroidery_internal.h, 452
embroidery_internal.h, 452
Compress, 108
 bit_position, 108
 bits_total, 108
 block_elements, 109
 character_huffman, 109
 character_length_huffman, 109
 distance_huffman, 109
 input_data, 109
 input_length, 109
compress
 embroidery_internal.h, 460
compress.c
 compress_get_bits, 402
 compress_get_position, 402
 compress_get_token, 402
 compress_init, 402
 compress_load_block, 402
 compress_load_character_huffman, 402
 compress_load_character_length_huffman, 402
 compress_load_distance_huffman, 402
 compress_peek, 402
 compress_pop, 402
 compress_read_variable_length, 402
 huffman_build_table, 403
 huffman_lookup, 403
 huffman_lookup_data, 403
 hus_compress, 403
 hus_decompress, 403
compress_get_bits
 compress.c, 402
 embroidery_internal.h, 463
compress_get_position
 compress.c, 402
 embroidery_internal.h, 463
compress_get_token
 compress.c, 402
 embroidery_internal.h, 464
compress_init
 compress.c, 402
compress_load_block
 compress.c, 402
 embroidery_internal.h, 464
compress_load_character_huffman
 compress.c, 402
 embroidery_internal.h, 464
compress_load_character_length_huffman
 compress.c, 402
 embroidery_internal.h, 464
compress_load_distance_huffman
 compress.c, 402
 embroidery_internal.h, 464
compress_peek
 compress.c, 402
compress_pop
 compress.c, 402
 embroidery_internal.h, 464
compress_read_variable_length

compress.c, 402
embroidery_internal.h, 464
config
 embroidermodder.h, 345
 mainwindow.cpp, 385
constants
 LSYSTEM, 170
contextMenuEvent
 CmdPromptHistory, 99
 CmdPromptInput, 102
 StatusBarButton, 294
 View, 321
control1
 EmbBezier_, 126
control2
 EmbBezier_, 126
controlPointLabels
 UiObject_, 306
controlPoints
 UiObject_, 306
convert
 embroidery.h, 420
 pattern.c, 571
convert_args_to_type
 mainwindow.cpp, 383
copy
 MainWindow, 180
 View, 321
copy_trim
 embroidery_internal.h, 464
 main.c, 565
copyClip
 CmdPromptInput, 102
copyPressed
 CmdPrompt, 89
 CmdPromptInput, 103
copySelected
 View, 321
cornerButtonClicked
 View, 321
count
 EmbArray_, 125
create_checkbox
 Settings_Dialog, 288
create_float_spinbox
 Settings_Dialog, 288
create_icon
 MainWindow, 180
create_test_file_1
 embroidery_internal.h, 464
create_test_file_2
 embroidery_internal.h, 464
create_test_file_3
 embroidery_internal.h, 464
create_toolbar
 MainWindow, 180
createAllActions
 MainWindow, 180
createAllMenus
 MainWindow, 180
createAllToolbars
 MainWindow, 180
createComboBox
 PropertyEditor, 248
createComboBoxSelected
 PropertyEditor, 248
createEditMenu
 MainWindow, 180
createFileMenu
 MainWindow, 180
createFontComboBox
 PropertyEditor, 248
createGrid
 View, 321
createGridIso
 View, 321
createGridPolar
 View, 321
createGridRect
 View, 321
createGroupBox
 PropertyEditor, 248
createGroupBoxGeneral
 PropertyEditor, 248
createGroupBoxGeometryArc
 PropertyEditor, 248
createGroupBoxGeometryBlock
 PropertyEditor, 249
createGroupBoxGeometryCircle
 PropertyEditor, 249
createGroupBoxGeometryDimAligned
 PropertyEditor, 249
createGroupBoxGeometryDimAngular
 PropertyEditor, 249
createGroupBoxGeometryDimArcLength
 PropertyEditor, 249
createGroupBoxGeometryDimDiameter
 PropertyEditor, 249
createGroupBoxGeometryDimLeader
 PropertyEditor, 250
createGroupBoxGeometryDimLinear
 PropertyEditor, 250
createGroupBoxGeometryDimOrdinate
 PropertyEditor, 250
createGroupBoxGeometryDimRadius
 PropertyEditor, 250
createGroupBoxGeometryEllipse
 PropertyEditor, 250
createGroupBoxGeometryImage
 PropertyEditor, 250
createGroupBoxGeometryInfiniteLine
 PropertyEditor, 250
createGroupBoxGeometryLine
 PropertyEditor, 251
createGroupBoxGeometryPath
 PropertyEditor, 251

createGroupBoxGeometryPoint
 PropertyEditor, 251
 createGroupBoxGeometryPolygon
 PropertyEditor, 251
 createGroupBoxGeometryPolyline
 PropertyEditor, 251
 createGroupBoxGeometryRay
 PropertyEditor, 251
 createGroupBoxGeometryRectangle
 PropertyEditor, 252
 createGroupBoxGeometryTextMulti
 PropertyEditor, 252
 createGroupBoxGeometryTextSingle
 PropertyEditor, 252
 createGroupBoxMiscArc
 PropertyEditor, 252
 createGroupBoxMisclImage
 PropertyEditor, 252
 createGroupBoxMiscPath
 PropertyEditor, 252
 createGroupBoxMiscPolyline
 PropertyEditor, 253
 createGroupBoxMiscTextSingle
 PropertyEditor, 253
 createGroupBoxTextTextSingle
 PropertyEditor, 253
 createHandle
 CmdPromptSplitter, 107
 createHelpMenu
 MainWindow, 181
 createHelpToolbar
 MainWindow, 181
 createHistogram
 EmbDetailsDialog, 129
 createIconToolbar
 MainWindow, 181
 createLayerToolbar
 MainWindow, 181
 createLineEdit
 PropertyEditor, 253
 createMainWidget
 EmbDetailsDialog, 129
 createObjectList
 View, 322
 createOrigin
 View, 322
 createPanToolbar
 MainWindow, 181
 createPromptToolbar
 MainWindow, 181
 createPropertiesToolbar
 MainWindow, 181
 createRulerTextPath
 View, 322
 createSettingsMenu
 MainWindow, 181
 createTabDisplay
 Settings_Dialog, 289
 createTabFilesPaths
 Settings_Dialog, 289
 createTabGeneral
 Settings_Dialog, 289
 createTabGridRuler
 Settings_Dialog, 289
 createTabLineWeight
 Settings_Dialog, 289
 createTabOpenSave
 Settings_Dialog, 289
 createTabOrthoPolar
 Settings_Dialog, 289
 createTabPrinting
 Settings_Dialog, 289
 createTabPrompt
 Settings_Dialog, 289
 createTabQuickSnap
 Settings_Dialog, 289
 createTabQuickTrack
 Settings_Dialog, 289
 createTabSelection
 Settings_Dialog, 289
 createTabSnap
 Settings_Dialog, 289
 createTextToolbar
 MainWindow, 181
 createToolButton
 PropertyEditor, 253
 createToolButtonPickAdd
 PropertyEditor, 253
 createToolButtonQSelect
 PropertyEditor, 253
 createViewMenu
 MainWindow, 181
 createWindowMenu
 MainWindow, 182
 creationTime
 _bcf_directory_entry, 51
 creatorName
 ThredExtension_, 304
 crosshairColor
 View, 327
 crosshairSize
 View, 327
 csd, 16, 506
 csd_decryptArray
 format_csd.c, 507
 CsdSubMaskSize
 format_csd.c, 506
 CsdXorMaskSize
 format_csd.c, 506
 csv, 508
 CSV_EXPECT
 embroidery_internal.h, 460
 CSV_EXPECT_COMMA
 embroidery_internal.h, 460
 CSV_EXPECT_NULL
 embroidery_internal.h, 460

CSV_EXPECT_QUOTE1
embroidery_internal.h, 460

CSV_EXPECT_QUOTE2
embroidery_internal.h, 460

CSV_MODE
embroidery_internal.h, 460

CSV_MODE_COMMENT
embroidery_internal.h, 460

CSV_MODE_NULL
embroidery_internal.h, 460

CSV_MODE_STITCH
embroidery_internal.h, 461

CSV_MODE_THREAD
embroidery_internal.h, 460

CSV_MODE_VARIABLE
embroidery_internal.h, 460

csvStitchFlagToStr
format_csv.c, 508

csvStrToStitchFlag
format_csv.c, 508

CUBICTOCONTROL1
embroidery_internal.h, 452

CUBICTOCONTROL2
embroidery_internal.h, 452

CUBICTOEND
embroidery_internal.h, 452

curCmd
CmdPromptInput, 106

curColor
MdiWindow, 222

curFile
MdiWindow, 222

curlayer
MdiWindow, 222

curlineType
MdiWindow, 222

curlineWeight
MdiWindow, 222

current_directory
Settings_, 274

current_element_id
format_svg.c, 536

currentAttribute
format_svg.c, 536

currentColorChanged
MdiWindow, 216

currentColorIndex
EmbPattern_, 140

currentDisplayBackgroundColorChanged
Settings_Dialog, 289

currentDisplayCrossHairColorChanged
Settings_Dialog, 289

currentDisplaySelectBoxLeftColorChanged
Settings_Dialog, 290

currentDisplaySelectBoxLeftFillChanged
Settings_Dialog, 290

currentDisplaySelectBoxRightColorChanged
Settings_Dialog, 290

currentDisplaySelectBoxRightFillChanged
Settings_Dialog, 290

currentGeneralMdiBackgroundColorChanged
Settings_Dialog, 290

currentGridColorChanged
Settings_Dialog, 290

currentLayerChanged
MdiWindow, 216

currentLinetypeChanged
MdiWindow, 216

currentLineweightChanged
MdiWindow, 216

currentPromptBackgroundColorChanged
Settings_Dialog, 290

currentPromptTextColorChanged
Settings_Dialog, 290

currentRulerColorChanged
Settings_Dialog, 290

currentValue
format_svg.c, 536

curText
CmdPromptInput, 106

curved
DimLeaderObject, 116

cut
MainWindow, 182
View, 322

cutCopyMousePoint
View, 327

cutCopyObjectList
MainWindow, 205

cutPressed
CmdPrompt, 90
CmdPromptInput, 103

dat, 16

data
EmblImage_, 135

data.cpp
comboBoxes, 332
comboBoxTextSingleFont, 332
group_box_arc_geometry, 332
group_box_arc_geometry_entries, 332
group_box_arc_misc, 333
group_box_arc_misc_entries, 333
group_box_data, 333
group_box_ellipse_geometry, 333
group_box_ellipse_geometry_entries, 333
group_box_general, 333
group_box_general_entries, 333
group_box_list, 333
groupBoxes, 333
lineEdits, 333
toolButtons, 333

day
EmbTime_, 147

dayVision
MainWindow, 182

debug_message

embroidermodder.h, 344
 interface.cpp, 379
debug_mode
 Settings_, 274
decode_exy_flags
 format_exy.c, 515
decode_record_flags
 format_dst.c, 511
decode_t01_record
 embroidery_internal.h, 464
 encoding.c, 491
decode_tajima_ternary
 embroidery_internal.h, 465
 encoding.c, 491
decode_tap_record_flags
 format_tap.c, 538
DecodeCsdByte
 format_csd.c, 507
decodeNewStitch
 embroidery_internal.h, 465
 encoding.c, 491
DEFAULT_MODE
 embroidermodder.h, 343
default_value
 Huffman, 154
defaultPrefix
 CmdPromptInput, 106
degrees
 embroidermodder.h, 344
 embroidery.h, 420
 functions.c, 552
deleteObject
 View, 322
deletePressed
 CmdPrompt, 90
 CmdPromptInput, 103
 MainWindow, 182
 MdiWindow, 217
 View, 322
deleteSelected
 View, 322
dem, 16, 509
description
 EmbFormatList_, 132
 EmbThread_, 146
designDetails
 MainWindow, 182
 MdiWindow, 217
dialog
 embroidermodder.h, 345
 settings-dialog.cpp, 398
difat
 _bcf_file, 53
difatEntriesInHeader
 main.c, 570
dimensions
 EmlImage_, 135
DimLeaderObject, 109
 ~DimLeaderObject, 113
 allGripPoints, 113
 ArrowStyle, 112
 arrowStyleAngle, 116
 arrowStyleLength, 116
 arrowStylePath, 116
 Box, 112
 Closed, 112
 curved, 116
 DimLeaderObject, 112, 113
 Dot, 112
 filled, 116
 Flared, 112
 Fletching, 112
 gripEdit, 113
 init, 113
 lineStyle, 112
 lineStyleAngle, 116
 lineStyleLength, 116
 lineStylePath, 117
 mouseSnapPoint, 113
 NoArrow, 112
 NoLine, 112
 objectAngle, 113
 objectDeltaX, 114
 objectDeltaY, 114
 objectEndPoint1, 114
 objectEndPoint2, 114
 objectLength, 114
 objectMidPoint, 114
 objectX1, 114
 objectX2, 114
 objectY1, 114
 objectY2, 114
 Open, 112
 paint, 114
 setObjectEndPoint1, 115
 setObjectEndPoint2, 115
 setObjectX1, 115
 setObjectX2, 115
 setObjectY1, 115
 setObjectY2, 115
 Tick, 112
 Type, 112
 type, 115
 updateLeader, 116
 updateRubber, 116
 vulcanize, 116
dirBrush
 SelectBox, 271
directory
 _bcf_file, 53
directoryEntryName
 _bcf_directory_entry, 51
directoryEntryNameLength
 _bcf_directory_entry, 51
dirEntries
 _bcf_directory, 50

dirPen
 SelectBox, 271
disableLwt
 StatusBarButton, 294
disableMoveRapidFire
 MainWindow, 182
 View, 322
disablePromptRapidFire
 MainWindow, 182
disableRapidFire
 CmdPrompt, 90
disableReal
 StatusBarButton, 294
display_bg_color
 Settings, 274
display_crosshair_color
 Settings, 274
display_crosshair_percent
 Settings, 274
display_renderhint_aa
 Settings, 274
display_renderhint_high_aa
 Settings, 275
display_renderhint_noncosmetic
 Settings, 275
display_renderhint_smooth_pix
 Settings, 275
display_renderhint_text_aa
 Settings, 275
display_scrollbar_widget_num
 Settings, 275
display_selectbox_alpha
 Settings, 275
display_selectbox_left_color
 Settings, 275
display_selectbox_left_fill
 Settings, 275
display_selectbox_right_color
 Settings, 275
display_selectbox_right_fill
 Settings, 275
display_show_scrollbars
 Settings, 275
display_units
 Settings, 275
display_use_opengl
 Settings, 275
display_zoomscale_in
 Settings, 275
display_zoomscale_out
 Settings, 275
distance_huffman
 Compress, 109
docIndex
 MainWindow, 205
dockPropEdit
 MainWindow, 205
dockUndoEdit
 MainWindow, 205
 MainWindow, 205
DOLPHIN_MODE_NUM_POINTS
 embroidermodder.h, 343
DOLPHIN_MODE_XSCALE
 embroidermodder.h, 343
DOLPHIN_MODE_YSCALE
 embroidermodder.h, 343
done
 UndoableNavCommand, 312
doNothing
 MainWindow, 182
Dot
 DimLeaderObject, 112
downPressed
 CmdPrompt, 90
 CmdPromptInput, 103
dragon_curve
 fill.c, 494
drawBackground
 View, 322
drawForeground
 View, 322
drawRubberLine
 BaseObject, 77
dsb, 16, 509
dst, 16, 510
dstJumpsPerTrim
 EmbPattern, 140
dsz, 16, 470, 512
dx
 UndoableMoveCommand, 311
 UndoableScaleCommand, 315
dxf, 16, 513
dxf_color
 embroidery.h, 411
DXF_VERSION_2000
 embroidery_internal.h, 452
DXF_VERSION_2002
 embroidery_internal.h, 452
DXF_VERSION_2004
 embroidery_internal.h, 452
DXF_VERSION_2006
 embroidery_internal.h, 452
DXF_VERSION_2007
 embroidery_internal.h, 452
DXF_VERSION_2009
 embroidery_internal.h, 452
DXF_VERSION_2010
 embroidery_internal.h, 453
DXF_VERSION_2013
 embroidery_internal.h, 453
DXF_VERSION_R10
 embroidery_internal.h, 453
DXF_VERSION_R11
 embroidery_internal.h, 453
DXF_VERSION_R12
 embroidery_internal.h, 453
DXF_VERSION_R13

embroidery_internal.h, 453
DXF_VERSION_R14
embroidery_internal.h, 453
DXF_VERSION_R15
embroidery_internal.h, 453
DXF_VERSION_R18
embroidery_internal.h, 453
DXF_VERSION_R21
embroidery_internal.h, 453
DXF_VERSION_R24
embroidery_internal.h, 453
DXF_VERSION_R27
embroidery_internal.h, 453
dy
UndoableMoveCommand, 311
UndoableScaleCommand, 315

edit_toolbar
embroidermodder.h, 345
mainwindow-toolbars.cpp, 381
editMenu
MainWindow, 205
edr, 16, 470, 513
ELEMENT_A
embroidery_internal.h, 453
ELEMENT_ANIMATE
embroidery_internal.h, 453
ELEMENT_ANIMATECOLOR
embroidery_internal.h, 453
ELEMENT_ANIMATEMOTION
embroidery_internal.h, 453
ELEMENT_ANIMATETRANSFORM
embroidery_internal.h, 453
ELEMENT_ANIMATION
embroidery_internal.h, 453
ELEMENT_AUDIO
embroidery_internal.h, 454
ELEMENT_CIRCLE
embroidery_internal.h, 454
ELEMENT_DEFS
embroidery_internal.h, 454
ELEMENT_DESC
embroidery_internal.h, 454
ELEMENT_DISCARD
embroidery_internal.h, 454
ELEMENT_ELLIPSE
embroidery_internal.h, 454
ELEMENT_FONT
embroidery_internal.h, 454
ELEMENT_FONT_FACE
embroidery_internal.h, 454
ELEMENT_FONT_FACE_SRC
embroidery_internal.h, 454
ELEMENT_FONT_FACE_URI
embroidery_internal.h, 454
ELEMENT_FOREIGN_OBJECT
embroidery_internal.h, 454
ELEMENT_G
embroidery_internal.h, 454
ELEMENT_GLYPH
embroidery_internal.h, 454
ELEMENT_HANDLER
embroidery_internal.h, 454
ELEMENT_HKERN
embroidery_internal.h, 454
ELEMENT_IMAGE
embroidery_internal.h, 454
ELEMENT_LINE
embroidery_internal.h, 454
ELEMENT_LINEAR_GRADIENT
embroidery_internal.h, 454
ELEMENT_LISTENER
embroidery_internal.h, 455
ELEMENT_METADATA
embroidery_internal.h, 455
ELEMENT_MISSING_GLYPH
embroidery_internal.h, 455
ELEMENT_MPATH
embroidery_internal.h, 455
ELEMENT_PATH
embroidery_internal.h, 455
ELEMENT_POLYGON
embroidery_internal.h, 455
ELEMENT_POLYLINE
embroidery_internal.h, 455
ELEMENT_PREFETCH
embroidery_internal.h, 455
ELEMENT_RADIAL_GRADIENT
embroidery_internal.h, 455
ELEMENT_RECT
embroidery_internal.h, 455
ELEMENT_SCRIPT
embroidery_internal.h, 455
ELEMENT_SET
embroidery_internal.h, 455
ELEMENT_SOLID_COLOR
embroidery_internal.h, 455
ELEMENT_STOP
embroidery_internal.h, 455
ELEMENT_SVG
embroidery_internal.h, 455
ELEMENT_SWITCH
embroidery_internal.h, 455
ELEMENT_TBREAK
embroidery_internal.h, 455
ELEMENT_TEXT
embroidery_internal.h, 455
ELEMENT_TEXT_AREA
embroidery_internal.h, 456
ELEMENT_TITLE
embroidery_internal.h, 456
ELEMENT_TSPAN
embroidery_internal.h, 456
ELEMENT_USE
embroidery_internal.h, 456
ELEMENT_VIDEO
embroidery_internal.h, 456

ELEMENT_XML
embroidery_internal.h, 456

ellipse
EmbGeometry_, 133

ellipse.c
ellipse_objectQuadrant0, 551
ellipse_objectQuadrant180, 551
ellipse_objectQuadrant270, 551
ellipse_objectQuadrant90, 551
embEllipse_area, 551
embEllipse_diameterX, 551
embEllipse_diameterY, 551
embEllipse_height, 551
embEllipse_init, 551
embEllipse_perimeter, 551
embEllipse_setDiameterMajor, 551
embEllipse_setDiameterMinor, 552
embEllipse_setRadiusMajor, 552
embEllipse_setRadiusMinor, 552
embEllipse_setSize, 552
embEllipse_updatePath, 552
embEllipse_width, 552

ELLIPSE_MODE_ELLIPSE_ROTATION
embroidermodder.h, 343

ELLIPSE_MODE_MAJORDIAMETER_MINORRADIUS
embroidermodder.h, 343

ELLIPSE_MODE_MAJORRADIUS_MINORRADIUS
embroidermodder.h, 343

ellipse_objectQuadrant0
ellipse.c, 551
ellipse_objectQuadrant180
ellipse.c, 551
ellipse_objectQuadrant270
ellipse.c, 551
ellipse_objectQuadrant90
ellipse.c, 551

EllipseObject, 117
~EllipseObject, 119
allGripPoints, 120
EllipseObject, 119
gripEdit, 120
init, 120
mouseSnapPoint, 120
objectDiameterMajor, 120
objectDiameterMinor, 120
objectHeight, 121
objectQuadrant0, 121
objectQuadrant180, 121
objectQuadrant270, 121
objectQuadrant90, 121
objectRadiusMajor, 121
objectRadiusMinor, 121
objectSavePath, 121
objectWidth, 121
paint, 121
setObjectDiameterMajor, 121
setObjectDiameterMinor, 122
setObjectRadiusMajor, 122

setObjectRadiusMinor, 122
setObjectSize, 122
Type, 119
type, 122
updatePath, 122
updateRubber, 122
vulcanize, 122

ELLIPSETOEND
embroidery_internal.h, 456

ELLIPSETORAD
embroidery_internal.h, 456

Elna, 514

Eltac, 515

EMB_ARC
embroidery.h, 411

EMB_ARRAY
embroidery.h, 411

EMB_BIG_ENDIAN
embroidery_internal.h, 456

EMB_CIRCLE
embroidery.h, 411

emb_constant_pi
embroidermodder.h, 345

EMB_DIM_DIAMETER
embroidery.h, 411

EMB_DIM_LEADER
embroidery.h, 411

EMB_ELLIPSE
embroidery.h, 411

emb_error
embroidery.h, 435
main.c, 570

EMB_FLAG
embroidery.h, 411

EMB_FORMAT_100
embroidery.h, 411

EMB_FORMAT_10O
embroidery.h, 411

EMB_FORMAT_ART
embroidery.h, 411

EMB_FORMAT_BMC
embroidery.h, 411

EMB_FORMAT_BRO
embroidery.h, 411

EMB_FORMAT_CND
embroidery.h, 411

EMB_FORMAT_COL
embroidery.h, 411

EMB_FORMAT_CSD
embroidery.h, 412

EMB_FORMAT_CSV
embroidery.h, 412

EMB_FORMAT_DAT
embroidery.h, 412

EMB_FORMAT_DEM
embroidery.h, 412

EMB_FORMAT_DSB
embroidery.h, 412

EMB_FORMAT_DST
embroidery.h, 412
EMB_FORMAT_DSZ
embroidery.h, 412
EMB_FORMAT_DXF
embroidery.h, 412
EMB_FORMAT_EDR
embroidery.h, 412
EMB_FORMAT_EMD
embroidery.h, 412
EMB_FORMAT_EXP
embroidery.h, 412
EMB_FORMAT_EXY
embroidery.h, 412
EMB_FORMAT_EYS
embroidery.h, 412
EMB_FORMAT_FXY
embroidery.h, 412
EMB_FORMAT_GC
embroidery.h, 412
EMB_FORMAT_GNC
embroidery.h, 412
EMB_FORMAT_GT
embroidery.h, 412
EMB_FORMAT_HUS
embroidery.h, 412
EMB_FORMAT_INB
embroidery.h, 413
EMB_FORMAT_INF
embroidery.h, 413
EMB_FORMAT_JEF
embroidery.h, 413
EMB_FORMAT_KSM
embroidery.h, 413
EMB_FORMAT_MAX
embroidery.h, 413
EMB_FORMAT_MIT
embroidery.h, 413
EMB_FORMAT_NEW
embroidery.h, 413
EMB_FORMAT_OFM
embroidery.h, 413
EMB_FORMAT_PCD
embroidery.h, 413
EMB_FORMAT_PCM
embroidery.h, 413
EMB_FORMAT_PCQ
embroidery.h, 413
EMB_FORMAT_PCS
embroidery.h, 413
EMB_FORMAT_PEC
embroidery.h, 413
EMB_FORMAT_PEL
embroidery.h, 413
EMB_FORMAT_PEM
embroidery.h, 413
EMB_FORMAT_PES
embroidery.h, 413
EMB_FORMAT_PHB
embroidery.h, 413
EMB_FORMAT_PHC
embroidery.h, 413
EMB_FORMAT_PLT
embroidery.h, 414
EMB_FORMAT_RGB
embroidery.h, 414
EMB_FORMAT_SEW
embroidery.h, 414
EMB_FORMAT_SHV
embroidery.h, 414
EMB_FORMAT_SST
embroidery.h, 414
EMB_FORMAT_STX
embroidery.h, 414
EMB_FORMAT_SVG
embroidery.h, 414
EMB_FORMAT_T01
embroidery.h, 414
EMB_FORMAT_T09
embroidery.h, 414
EMB_FORMAT_TAP
embroidery.h, 414
EMB_FORMAT_THR
embroidery.h, 414
EMB_FORMAT_TXT
embroidery.h, 414
EMB_FORMAT_U00
embroidery.h, 414
EMB_FORMAT_U01
embroidery.h, 414
EMB_FORMAT_VIP
embroidery.h, 414
EMB_FORMAT_VP3
embroidery.h, 414
EMB_FORMAT_XXX
embroidery.h, 414
EMB_FORMAT_ZSK
embroidery.h, 414
emb_identify_format
embroidery.h, 420
formats.c, 500
EMB_IMAGE
embroidery.h, 415
EMB_INT16_BIG
embroidery_internal.h, 456
EMB_INT16_LITTLE
embroidery_internal.h, 456
EMB_INT32_BIG
embroidery_internal.h, 456
EMB_INT32_LITTLE
embroidery_internal.h, 456
EMB_LINE
embroidery.h, 415
EMB_LITTLE_ENDIAN
embroidery_internal.h, 456
EMB_MAX

embroidery_internal.h, 456
EMB_MAX_LAYERS
embroidery.h, 415
EMB_MIN
embroidery_internal.h, 456
emb_optOut
embroidery_internal.h, 465
main.c, 565
EMB_PATH
embroidery.h, 415
EMB_POINT
embroidery.h, 415
EMB_POLYGON
embroidery.h, 415
EMB_POLYLINE
embroidery.h, 415
EMB_PUBLIC
embroidery.h, 415
emb_readline
embroidery_internal.h, 465
main.c, 566
EMB_RECT
embroidery.h, 415
emb_round
embroidery.h, 420
functions.c, 552
EMB_SPLINE
embroidery.h, 415
EMB_STITCH
embroidery.h, 415
EMB_TEXT_MULTI
embroidery.h, 415
EMB_TEXT_SINGLE
embroidery.h, 415
EMB_THREAD
embroidery.h, 415
EMB_VECTOR
embroidery.h, 415
emb_verbose
embroidery.h, 435
main.c, 570
EmbAlignedDim
embroidery.h, 418
EmbAlignedDim_, 123
position, 123
EmbAngularDim
embroidery.h, 418
EmbAngularDim_, 123
position, 123
EmbArc
embroidery.h, 418
EmbArc_, 123
end, 124
mid, 124
start, 124
embArc_arcLength
arc.c, 547
embArc_area
arc.c, 547
embArc_chord
arc.c, 547
embArc_clockwise
arc.c, 547
embroidery.h, 420
embArc_endAngle
arc.c, 547
embArc_gripEdit
arc.c, 547
embArc_includedAngle
arc.c, 547
embArc_init
arc.c, 547
embroidery.h, 420
embArc_mouseSnapPoint
arc.c, 548
embArc_paint
arc.c, 548
embArc_print
main.c, 566
embArc_setCenter
arc.c, 548
embArc_setEndAngle
arc.c, 548
embArc_setRadius
arc.c, 548
embArc_setStartAngle
arc.c, 548
embArc_startAngle
arc.c, 548
embArc_updatePath
arc.c, 548
embArc_updateRubber
arc.c, 548
EmbArcLengthDim
embroidery.h, 418
EmbArcLengthDim_, 124
position, 124
EmbArray
embroidery.h, 418
EmbArray_, 125
count, 125
geometry, 125
length, 125
stitch, 125
thread, 125
type, 125
embArray_addArc
array.c, 399
embroidery.h, 421
embArray_addCircle
array.c, 400
embroidery.h, 421
embArray_addEllipse
array.c, 400
embroidery.h, 421
embArray_addFlag

array.c, 400
embroidery.h, 421
embArray_addLine
 array.c, 400
 embroidery.h, 421
embArray_addPath
 array.c, 400
 embroidery.h, 421
embArray_addPoint
 array.c, 400
 embroidery.h, 421
embArray_addPolygon
 array.c, 400
 embroidery.h, 421
embArray_addPolyline
 array.c, 400
 embroidery.h, 421
embArray_addRect
 array.c, 400
 embroidery.h, 421
embArray_addStitch
 array.c, 400
 embroidery.h, 421
embArray_addThread
 embroidery.h, 422
embArray_addVector
 array.c, 400
 embroidery.h, 422
embArray_copy
 array.c, 401
 embroidery.h, 422
embArray_create
 array.c, 401
 embroidery.h, 422
embArray_free
 array.c, 401
 embroidery.h, 422
embArray_resize
 array.c, 401
 embroidery.h, 422
embBaseSetColorRGB
 arc.c, 548
EmbBezier
 embroidery.h, 418
EmbBezier_, 126
 control1, 126
 control2, 126
 end, 126
 start, 126
EmbBlock
 embroidery.h, 418
EmbBlock_, 126
 position, 126
EmbCircle
 embroidery.h, 418
EmbCircle_, 127
 center, 127
 radius, 127
embCircle_area
 circle.c, 550
embCircle_circumference
 circle.c, 550
embCircle_init
 circle.c, 550
 embroidery.h, 422
embCircle_prompt
 arc.c, 548
embCircle_setArea
 arc.c, 548
embCircle_setCircumference
 arc.c, 549
EmbColor
 embroidery.h, 418
EmbColor_, 127
 b, 128
 g, 128
 r, 128
embColor_create
 embroidery.h, 422
embColor_distance
 embroidery.h, 422
 main.c, 566
embColor_fromHexStr
 embroidery.h, 422
 encoding.c, 491
embColor_make
 embroidery.h, 423
embColor_read
 embroidery_internal.h, 465
 main.c, 566
embColor_write
 embroidery_internal.h, 465
 main.c, 566
embConstantPi
 embroidery.h, 435
 main.c, 570
EmbDetailsDialog, 128
 ~EmbDetailsDialog, 129
 boundingRect, 129
 buttonBox, 129
 colorChanges, 129
 colorTotal, 129
 createHistogram, 129
 createMainWidget, 129
 EmbDetailsDialog, 128
 getInfo, 129
 mainWidget, 129
 stitchesJump, 130
 stitchesReal, 130
 stitchesTotal, 130
 stitchesTrim, 130
EmbDiameterDim
 embroidery.h, 418
EmbDiameterDim_, 130
 position, 130
EmbEllipse

embroidery.h, 418
EmbEllipse_, 130
 center, 131
 radius, 131
 rotation, 131
embEllipse_area
 ellipse.c, 551
 embroidery.h, 423
embEllipse_click
 arc.c, 549
embEllipse_diameterX
 ellipse.c, 551
 embroidery.h, 423
embEllipse_diameterY
 ellipse.c, 551
 embroidery.h, 423
embEllipse_height
 ellipse.c, 551
 embroidery.h, 423
embEllipse_init
 ellipse.c, 551
 embroidery.h, 423
embEllipse_main
 arc.c, 549
embEllipse_make
 embroidery.h, 423
embEllipse_perimeter
 ellipse.c, 551
 embroidery.h, 423
embEllipse_setDiameterMajor
 ellipse.c, 551
embEllipse_setDiameterMinor
 ellipse.c, 552
embEllipse_setRadiusMajor
 ellipse.c, 552
embEllipse_setRadiusMinor
 ellipse.c, 552
embEllipse_setSize
 ellipse.c, 552
embEllipse_updatePath
 ellipse.c, 552
embEllipse_width
 ellipse.c, 552
 embroidery.h, 423
EmbFlag
 embroidery.h, 418
embFormat_getExtension
 formats.c, 500
EMBFORMAT_MAXDESC
 embroidery.h, 415
EMBFORMAT_MAXEXT
 embroidery.h, 415
EMBFORMAT_OBJECTONLY
 embroidery.h, 415
EMBFORMAT_STCHANDOBJ
 embroidery.h, 416
EMBFORMAT_STITCHONLY
 embroidery.h, 416
EMBFORMAT_UNSUPPORTED
 embroidery.h, 416
EmbFormatList
 embroidery.h, 418
EmbFormatList_
 check_for_color_file, 131
 color_only, 131
 description, 132
 extension, 132
 reader_state, 132
 type, 132
 write_external_color_file, 132
 writer_state, 132
EmbGeometry
 embroidery.h, 418
EmbGeometry_
 arc, 133
 circle, 133
 color, 133
 ellipse, 133
 flag, 133
 line, 133
 lineType, 133
 object, 133
 path, 133
 point, 133
 polygon, 133
 polyline, 134
 rect, 134
 spline, 134
 stitch, 134
 thread, 134
 type, 134
 vector, 134
embGeometry_boundingRect
 embroidery.h, 423
 geometry.c, 545
embGeometry_free
 embroidery.h, 423
 geometry.c, 545
embGeometry_init
 embroidery.h, 424
 geometry.c, 545
embGeometry_move
 embroidery.h, 424
 geometry.c, 545
embGeometry_vulcanize
 embroidery.h, 424
 geometry.c, 545
EmblImage
 embroidery.h, 419
EmblImage_
 data, 135
 dimensions, 135
 height, 135
 name, 135
 path, 135
 position, 135

```

        width, 135
emblImage_create
    embroidery.h, 424
emblImage_free
    embroidery.h, 424
emblImage_read
    embroidery.h, 424
emblImage_write
    embroidery.h, 424
EmbInfiniteLine
    embroidery.h, 419
EmbInfiniteLine_
    135
    position, 136
embInt_read
    embroidery_internal.h, 466
    encoding.c, 492
embInt_write
    embroidery_internal.h, 466
    encoding.c, 492
Embroid, 470, 508, 513
EmbLayer
    embroidery.h, 419
EmbLayer_
    136
    geometry, 136
    name, 136
EmbLeaderDim
    embroidery.h, 419
EmbLeaderDim_
    136
    position, 137
EmbLine
    embroidery.h, 419
EmbLine_
    137
    color, 137
    end, 137
    lineType, 137
    start, 137
embLine_intersectionPoint
    embroidery.h, 424
    line.c, 553
embLine_make
    embroidery.h, 424
embLine_normalVector
    embroidery.h, 425
    line.c, 553
embLine_toVector
    line.c, 553
EmbLinearDim
    embroidery.h, 419
EmbLinearDim_
    138
    position, 138
EmbOrdinateDim
    embroidery.h, 419
EmbOrdinateDim_
    138
    position, 138
EmbPath
    embroidery.h, 419
EmbPath_
    138
    color, 139
flagList, 139
lineType, 139
pointList, 139
EmbPattern
    embroidery.h, 419
EmbPattern_
    139
    currentColorIndex, 140
    dstJumpsPerTrim, 140
    geometry, 140
    home, 140
    hoop_height, 140
    hoop_width, 140
    layer, 140
    stitch_list, 140
    thread_list, 140
embPattern_addCircleAbs
    embroidery.h, 425
    pattern.c, 571
embPattern_addEllipseAbs
    embroidery.h, 425
    pattern.c, 571
embPattern_addLineAbs
    embroidery.h, 425
    pattern.c, 572
embPattern_addPathAbs
    embroidery.h, 425
    pattern.c, 572
embPattern_addPointAbs
    embroidery.h, 425
    pattern.c, 572
embPattern_addPolygonAbs
    embroidery.h, 425
    pattern.c, 572
embPattern_addPolylineAbs
    embroidery.h, 425
embPattern_addPolylineObjectAbs
    pattern.c, 572
embPattern_addRectAbs
    embroidery.h, 425
    pattern.c, 572
embPattern_addStitchAbs
    embroidery.h, 425
    pattern.c, 572
embPattern_addStitchRel
    embroidery.h, 426
    pattern.c, 572
embPattern_addThread
    embroidery.h, 426
    pattern.c, 572
embPattern_calcBoundingBox
    embroidery.h, 426
    pattern.c, 573
embPattern_center
    embroidery.h, 426
    pattern.c, 573
embPattern_changeColor
    embroidery.h, 426
    pattern.c, 573

```

embPattern_color_count
embroidery.h, 426
pattern.c, 573
embPattern_combine
embroidery.h, 426
fill.c, 494
embPattern_combineJumpStitches
embroidery.h, 426
pattern.c, 573
embPattern_convertGeometry
embroidery.h, 426
fill.c, 494
embPattern_copyPolylinesTostitch_list
pattern.c, 573
embPattern_copyPolylinesToStitchList
embroidery.h, 427
embPattern_copystitch_listToPolylines
pattern.c, 573
embPattern_copyStitchListToPolylines
embroidery.h, 427
embPattern_correctForMaxStitchLength
embroidery.h, 427
pattern.c, 573
embPattern_create
embroidery.h, 427
pattern.c, 573
embPattern_crossstitch
embroidery.h, 427
fill.c, 494
embPattern_designDetails
embroidery.h, 427
pattern.c, 573
embPattern_end
embroidery.h, 427
pattern.c, 574
embPattern_fixColorCount
embroidery.h, 427
pattern.c, 574
embPattern_flip
embroidery.h, 427
pattern.c, 574
embPattern_flipHorizontal
embroidery.h, 428
pattern.c, 574
embPattern_flipVertical
embroidery.h, 428
pattern.c, 574
embPattern_free
embroidery.h, 428
pattern.c, 574
embPattern_hideStitchesOverLength
embroidery.h, 428
pattern.c, 574
embPattern_horizontal_fill
embroidery.h, 428
fill.c, 494
embPattern_jumpStitches
embroidery.h, 428
pattern.c, 574
embPattern_lengthHistogram
embroidery.h, 428
pattern.c, 574
embPattern_loadExternalColorFile
embroidery.h, 428
pattern.c, 574
embPattern_maximumStitchLength
embroidery.h, 428
pattern.c, 574
embPattern_minimumStitchLength
embroidery.h, 428
pattern.c, 574
embPattern_movePolylinesTostitch_list
pattern.c, 575
embPattern_movePolylinesToStitchList
embroidery.h, 429
embPattern_movestitch_listToPolylines
pattern.c, 575
embPattern_moveStitchListToPolylines
embroidery.h, 429
embPattern_read
embroidery.h, 429
formats.c, 500
embPattern_readAuto
embroidery.h, 429
formats.c, 500
embPattern_realStitches
embroidery.h, 429
pattern.c, 575
embPattern_render
embroidery.h, 429
embPattern_scale
embroidery.h, 429
pattern.c, 575
embPattern_simulate
embroidery.h, 429
embPattern_stitchArc
fill.c, 494
embPattern_stitchCircle
fill.c, 494
embPattern_stitchEllipse
fill.c, 495
embPattern_stitchPath
fill.c, 495
embPattern_stitchPolygon
fill.c, 495
embPattern_stitchPolyline
fill.c, 495
embPattern_stitchRect
fill.c, 496
embPattern_stitchText
fill.c, 496
embPattern_totalStitchLength
embroidery.h, 429
pattern.c, 575
embPattern_trimStitches
embroidery.h, 430

pattern.c, 575
embPattern_write
 embroidery.h, 430
 formats.c, 500
embPattern_writeAuto
 embroidery.h, 430
 formats.c, 500
EmbPoint
 embroidery.h, 419
EmbPoint_, 140
 color, 141
 lineType, 141
 position, 141
EmbPolygon
 embroidery.h, 419
embPolygon_reduceByDistance
 fill.c, 496
embPolygon_reduceByNth
 fill.c, 496
EmbPolyline
 embroidery.h, 419
EmbRadiusDim
 embroidery.h, 419
EmbRadiusDim_, 141
 position, 141
EmbRay
 embroidery.h, 419
EmbRay_, 142
 position, 142
EmbReal
 embroidery.h, 419
EmbRect
 embroidery.h, 419
EmbRect_, 142
 bottom, 142
 left, 142
 radius, 142
 right, 143
 rotation, 143
 top, 143
embRect_area
 embroidery.h, 430
 rect.c, 554
embRect_bottomLeft
 arc.c, 549
embRect_bottomRight
 arc.c, 549
embRect_init
 embroidery.h, 430
 rect.c, 554
embroidermodder.cpp
 appName, 334
 appVer, 334
 exitApp, 334
 main, 334
 usage, 334
 version, 334
embroidermodder.h
 Action, 339
 action_table, 345
 CIRCLE_MODE_1P_DIA, 343
 CIRCLE_MODE_1P_RAD, 343
 CIRCLE_MODE_2P, 343
 CIRCLE_MODE_3P, 343
 CIRCLE_MODE_TTR, 343
 comboBoxes, 345
 comboBoxTextSingleFont, 345
 config, 345
 debug_message, 344
 DEFAULT_MODE, 343
 degrees, 344
 dialog, 345
 DOLPHIN_MODE_NUM_POINTS, 343
 DOLPHIN_MODE_XSCALE, 343
 DOLPHIN_MODE_YSCALE, 343
 edit_toolbar, 345
 ELLIPSE_MODE_ELLIPSE_ROTATION, 343
 ELLIPSE_MODE_MAJORDIAMETER_MINORRADIUS, 343
 ELLIPSE_MODE_MAJORRADIUS_MINORRADIUS, 343
 emb_constant_pi, 345
EmbView, 339
 file_toolbar, 345
 get_action_index, 344
 group_box_arc_geometry, 345
 group_box_arc_geometry_entries, 345
 group_box_data, 345
 group_box_ellipse_geometry, 345
 group_box_ellipse_geometry_entries, 345
 group_box_list, 345
 GroupBoxData, 339
 groupBoxes, 345
 HEART_MODE_NUM_POINTS, 343
 HEART_MODE_STYLE, 343
 HEART_MODE_XSCALE, 343
 HEART_MODE_YSCALE, 343
 lineEdits, 345
 mainWin, 344
 OBJ_COLOR, 339
 OBJ_KEYS, 339
 OBJ_LAYER, 339
 OBJ_LTYPE, 339
 OBJ_LTYPE_CENTER, 340
 OBJ_LTYPE_CONT, 340
 OBJ_LTYPE_DOT, 340
 OBJ_LTYPE_FISHBONE, 340
 OBJ_LTYPE_HIDDEN, 340
 OBJ_LTYPE_PHANTOM, 340
 OBJ_LTYPE_RUNNING, 340
 OBJ_LTYPE_SATIN, 340
 OBJ_LTYPE_VALUES, 339
 OBJ_LTYPE_ZIGZAG, 340
 OBJ_LWT, 339
 OBJ_LWT_01, 340
 OBJ_LWT_02, 340

OBJ_LWT_03, 340
OBJ_LWT_04, 340
OBJ_LWT_05, 340
OBJ_LWT_06, 340
OBJ_LWT_07, 340
OBJ_LWT_08, 340
OBJ_LWT_09, 340
OBJ_LWT_10, 340
OBJ_LWT_11, 340
OBJ_LWT_12, 340
OBJ_LWT_13, 340
OBJ_LWT_14, 340
OBJ_LWT_15, 340
OBJ_LWT_16, 340
OBJ_LWT_17, 340
OBJ_LWT_18, 340
OBJ_LWT_19, 340
OBJ_LWT_20, 340
OBJ_LWT_21, 340
OBJ_LWT_22, 340
OBJ_LWT_23, 340
OBJ_LWT_24, 340
OBJ_LWT_BYBLOCK, 340
OBJ_LWT_BYLAYER, 340
OBJ_LWT_DEFAULT, 340
OBJ_LWT_VALUES, 340
OBJ_NAME, 339
OBJ_RUBBER, 339
OBJ_RUBBER_CIRCLE_1P_DIA, 341
OBJ_RUBBER_CIRCLE_1P_RAD, 341
OBJ_RUBBER_CIRCLE_2P, 341
OBJ_RUBBER_CIRCLE_3P, 341
OBJ_RUBBER_CIRCLE_TTR, 341
OBJ_RUBBER_CIRCLE_TTT, 341
OBJ_RUBBER_DIMLEADER_LINE, 341
OBJ_RUBBER_ELLIPSE_LINE, 341
OBJ_RUBBER_ELLIPSE_MAJORDIAMETER_MINORRADIUS, 341
OBJ_RUBBER_ELLIPSE_MAJORMINORRADII, 341
OBJ_RUBBER_ELLIPSE_ROTATION, 341
OBJ_RUBBER_GRIP, 341
OBJ_RUBBER_IMAGE, 341
OBJ_RUBBER_LINE, 341
OBJ_RUBBER_OFF, 341
OBJ_RUBBER_ON, 341
OBJ_RUBBER_POLYGON, 341
OBJ_RUBBER_POLYGON_CIRCUMSCRIBE, 341
OBJ_RUBBER_POLYGON_INSCRIBE, 341
OBJ_RUBBER_POLYLINE, 341
OBJ_RUBBER_RECTANGLE, 341
OBJ_RUBBER_TEXTSINGLE, 341
OBJ_RUBBER_VALUES, 340
OBJ_SNAP_APPINTERSECTION, 341
OBJ_SNAP_CENTER, 341
OBJ_SNAP_ENDPOINT, 341
OBJ_SNAP_EXTENSION, 341
OBJ_SNAP_INSERTION, 341
OBJ_SNAP_INTERSECTION, 341
OBJ_SNAP_MIDPOINT, 341
OBJ_SNAP_NEAREST, 341
OBJ_SNAP_NODE, 341
OBJ_SNAP_NULL, 341
OBJ_SNAP_PARALLEL, 341
OBJ_SNAP_PERPENDICULAR, 341
OBJ_SNAP_QUADRANT, 341
OBJ_SNAP_TANGENT, 341
OBJ_SNAP_VALUES, 341
OBJ_TYPE, 339
OBJ_TYPE_ARC, 342
OBJ_TYPE_BASE, 342
OBJ_TYPE_BLOCK, 342
OBJ_TYPE_CIRCLE, 342
OBJ_TYPE_DIMALIGNED, 342
OBJ_TYPE_DIMANGULAR, 342
OBJ_TYPE_DIMARCLENGTH, 342
OBJ_TYPE_DIMDIAMETER, 342
OBJ_TYPE_DIMLEADER, 342
OBJ_TYPE_DIMLINEAR, 342
OBJ_TYPE_DIMORDINATE, 342
OBJ_TYPE_DIMRADIUS, 342
OBJ_TYPE_ELLIPSE, 342
OBJ_TYPE_ELLIPSEARC, 342
OBJ_TYPE_GRID, 342
OBJ_TYPE_HATCH, 342
OBJ_TYPE_IMAGE, 342
OBJ_TYPE_INFINITELINE, 342
OBJ_TYPE_LINE, 342
OBJ_TYPE_NULL, 342
OBJ_TYPE_PATH, 342
OBJ_TYPE_POINT, 342
OBJ_TYPE_POLYGON, 342
OBJ_TYPE_POLYLINE, 342
OBJ_TYPE_RAY, 342
OBJ_TYPE_RECTANGLE, 342
OBJ_TYPE_RUBBER, 342
OBJ_TYPE_SLOT, 342
OBJ_TYPE_SPLINE, 342
OBJ_TYPE_TEXTMULTI, 342
OBJ_TYPE_TEXTSINGLE, 342
OBJ_TYPE_VALUES, 342
operator+, 344
operator-, 344
PREVIEW_CLONE_NULL, 342
PREVIEW_CLONE_RUBBER, 342
PREVIEW_CLONE_SELECTED, 342
PREVIEW_CLONE_VALUES, 342
PREVIEW_MODE_MOVE, 343
PREVIEW_MODE_NULL, 343
PREVIEW_MODE_ROTATE, 343
PREVIEW_MODE_SCALE, 343
PREVIEW_MODE_VALUES, 342
radians, 344
read_settings, 344
ROTATE_MODE_NORMAL, 343
ROTATE_MODE_REFERENCE, 343

SCALE_MODE_NORMAL, 343
SCALE_MODE_REFERENCE, 343
Settings, 339
settings, 346
SINGLE_LINE_TEXT_MODE_JUSTIFY, 343
SINGLE_LINE_TEXT_MODE_RAPID, 343
SINGLE_LINE_TEXT_MODE_SETFONT, 343
SINGLE_LINE_TEXT_MODE_SETGEOM, 343
SNOWFLAKE_MODE_NUM_POINTS, 343
SNOWFLAKE_MODE_XSCALE, 344
SNOWFLAKE_MODE_YSCALE, 344
SPARE_RUBBER_OFF, 343
SPARE_RUBBER_PATH, 343
SPARE_RUBBER_POLYGON, 343
SPARE_RUBBER_POLYLINE, 343
SPARE_RUBBER_VALUES, 343
STAR_MODE_CENTER_PT, 343
STAR_MODE_NUM_POINTS, 343
STAR_MODE_RAD_INNER, 343
STAR_MODE_RAD_OUTER, 343
to_EmbVector, 344
to_QPointF, 344
toolButtons, 346
UiMode, 343
UiObject, 339
view_toolbar, 346
write_settings, 344
zoom_toolbar, 346
embroidermodder2/cmdprompt.cpp, 332
embroidermodder2/data.cpp, 332
embroidermodder2/embeddetails-dialog.cpp, 333
embroidermodder2/embroidermodder.cpp, 333
embroidermodder2/embroidermodder.h, 335, 346
embroidermodder2/imagewidget.cpp, 379
embroidermodder2/interface.cpp, 379
embroidermodder2/layer-manager.cpp, 379
embroidermodder2/mainwindow-commands.cpp, 380
embroidermodder2/mainwindow-menus.cpp, 380
embroidermodder2/mainwindow-settings.cpp, 380
embroidermodder2/mainwindow-toolbars.cpp, 381
embroidermodder2/mainwindow.cpp, 383
embroidermodder2/mdiarea.cpp, 385
embroidermodder2/mdiwindow.cpp, 385
embroidermodder2/object-arc.cpp, 385
embroidermodder2/object-base.cpp, 385
embroidermodder2/object-circle.cpp, 385
embroidermodder2/object-dimleader.cpp, 385
embroidermodder2/object-ellipse.cpp, 385
embroidermodder2/object-image.cpp, 385
embroidermodder2/object-line.cpp, 386
embroidermodder2/object-path.cpp, 386
embroidermodder2/object-point.cpp, 386
embroidermodder2/object-polygon.cpp, 386
embroidermodder2/object-polyline.cpp, 386
embroidermodder2/object-rect.cpp, 386
embroidermodder2/object-save.cpp, 386
embroidermodder2/object-textsingle.cpp, 386
embroidermodder2/preview-dialog.cpp, 386
embroidermodder2/property-editor.cpp, 386
embroidermodder2/README.md, 397
embroidermodder2/selectbox.cpp, 397
embroidermodder2/settings-dialog.cpp, 398
embroidermodder2/statusbar-button.cpp, 398
embroidermodder2/statusbar.cpp, 398
embroidermodder2/undo-commands.cpp, 399
embroidermodder2/undo-editor.cpp, 399
embroidermodder2/view.cpp, 399
embroidery.h
 _dxsetColorTable, 435
 Arc_Polyester, 410
 Arc_Rayon, 411
 black_thread, 435
 CHUNK_SIZE, 411
 CoatsAndClark_Rayon, 411
 convert, 420
 degrees, 420
 dxf_color, 411
 EMB_ARC, 411
 EMB_ARRAY, 411
 EMB_CIRCLE, 411
 EMB_DIM_DIAMETER, 411
 EMB_DIM_LEADER, 411
 EMB_ELLIPSE, 411
 emb_error, 435
 EMB_FLAG, 411
 EMB_FORMAT_100, 411
 EMB_FORMAT_10O, 411
 EMB_FORMAT_ART, 411
 EMB_FORMAT_BMC, 411
 EMB_FORMAT_BRO, 411
 EMB_FORMAT_CND, 411
 EMB_FORMAT_COL, 411
 EMB_FORMAT_CSD, 412
 EMB_FORMAT_CSV, 412
 EMB_FORMAT_DAT, 412
 EMB_FORMAT_DEM, 412
 EMB_FORMAT_DSB, 412
 EMB_FORMAT_DST, 412
 EMB_FORMAT_DSZ, 412
 EMB_FORMAT_DXF, 412
 EMB_FORMAT_EDR, 412
 EMB_FORMAT_EMD, 412
 EMB_FORMAT_EXP, 412
 EMB_FORMAT_EXY, 412
 EMB_FORMAT_EYS, 412
 EMB_FORMAT_FXY, 412
 EMB_FORMAT_GC, 412
 EMB_FORMAT_GNC, 412
 EMB_FORMAT_GT, 412
 EMB_FORMAT_HUS, 412
 EMB_FORMAT_INB, 413
 EMB_FORMAT_INF, 413
 EMB_FORMAT_JEF, 413
 EMB_FORMAT_KSM, 413
 EMB_FORMAT_MAX, 413
 EMB_FORMAT_MIT, 413

EMB_FORMAT_NEW, 413
EMB_FORMAT_OFM, 413
EMB_FORMAT_PCD, 413
EMB_FORMAT_PCM, 413
EMB_FORMAT_PCQ, 413
EMB_FORMAT_PCS, 413
EMB_FORMAT_PEC, 413
EMB_FORMAT_PEL, 413
EMB_FORMAT_PEM, 413
EMB_FORMAT_PES, 413
EMB_FORMAT_PHB, 413
EMB_FORMAT_PHC, 413
EMB_FORMAT_PLT, 414
EMB_FORMAT_RGB, 414
EMB_FORMAT_SEW, 414
EMB_FORMAT_SHV, 414
EMB_FORMAT_SST, 414
EMB_FORMAT_STX, 414
EMB_FORMAT_SVG, 414
EMB_FORMAT_T01, 414
EMB_FORMAT_T09, 414
EMB_FORMAT_TAP, 414
EMB_FORMAT_THR, 414
EMB_FORMAT_TXT, 414
EMB_FORMAT_U00, 414
EMB_FORMAT_U01, 414
EMB_FORMAT_VIP, 414
EMB_FORMAT_VP3, 414
EMB_FORMAT_XXX, 414
EMB_FORMAT_ZSK, 414
emb_identify_format, 420
EMB_IMAGE, 415
EMB_LINE, 415
EMB_MAX_LAYERS, 415
EMB_PATH, 415
EMB_POINT, 415
EMB_POLYGON, 415
EMB_POLYLINE, 415
EMB_PUBLIC, 415
EMB_RECT, 415
emb_round, 420
EMB_SPLINE, 415
EMB_STITCH, 415
EMB_TEXT_MULTI, 415
EMB_TEXT_SINGLE, 415
EMB_THREAD, 415
EMB_VECTOR, 415
emb_verbose, 435
EmbAlignedDim, 418
EmbAngularDim, 418
EmbArc, 418
embArc_clockwise, 420
embArc_init, 420
EmbArcLengthDim, 418
EmbArray, 418
embArray_addArc, 421
embArray_addCircle, 421
embArray_addEllipse, 421
embArray_addFlag, 421
embArray_addLine, 421
embArray_addPath, 421
embArray_addPoint, 421
embArray_addPolygon, 421
embArray_addPolyline, 421
embArray_addRect, 421
embArray_addStitch, 421
embArray_addThread, 422
embArray_addVector, 422
embArray_copy, 422
embArray_create, 422
embArray_free, 422
embArray_resize, 422
EmbBezier, 418
EmbBlock, 418
EmbCircle, 418
embCircle_init, 422
EmbColor, 418
embColor_create, 422
embColor_distance, 422
embColor_fromHexStr, 422
embColor_make, 423
embConstantPi, 435
EmbDiameterDim, 418
EmbEllipse, 418
embEllipse_area, 423
embEllipse_diameterX, 423
embEllipse_diameterY, 423
embEllipse_height, 423
embEllipse_init, 423
embEllipse_make, 423
embEllipse_perimeter, 423
embEllipse_width, 423
EmbFlag, 418
EMBFORMAT_MAXDESC, 415
EMBFORMAT_MAXEXT, 415
EMBFORMAT_OBJECTONLY, 415
EMBFORMAT_STCHANDOBJ, 416
EMBFORMAT_STITCHONLY, 416
EMBFORMAT_UNSUPPORTED, 416
EmbFormatList, 418
EmbGeometry, 418
embGeometry_boundingRect, 423
embGeometry_free, 423
embGeometry_init, 424
embGeometry_move, 424
embGeometry_vulcanize, 424
EmblImage, 419
emblImage_create, 424
emblImage_free, 424
emblImage_read, 424
emblImage_write, 424
EmblInfiniteLine, 419
EmbLayer, 419
EmbLeaderDim, 419
EmbLine, 419
embLine_intersectionPoint, 424

embLine_make, 424
 embLine_normalVector, 425
 EmbLinearDim, 419
 EmbOrdinateDim, 419
 EmbPath, 419
 EmbPattern, 419
 embPattern_addCircleAbs, 425
 embPattern_addEllipseAbs, 425
 embPattern_addLineAbs, 425
 embPattern_addPathAbs, 425
 embPattern_addPointAbs, 425
 embPattern_addPolygonAbs, 425
 embPattern_addPolylineAbs, 425
 embPattern_addRectAbs, 425
 embPattern_addStitchAbs, 425
 embPattern_addStitchRel, 426
 embPattern_addThread, 426
 embPattern_calcBoundingBox, 426
 embPattern_center, 426
 embPattern_changeColor, 426
 embPattern_color_count, 426
 embPattern_combine, 426
 embPattern_combineJumpStitches, 426
 embPattern_convertGeometry, 426
 embPattern_copyPolylinesToStitchList, 427
 embPattern_copyStitchListToPolylines, 427
 embPattern_correctForMaxStitchLength, 427
 embPattern_create, 427
 embPattern_crossstitch, 427
 embPattern_designDetails, 427
 embPattern_end, 427
 embPattern_fixColorCount, 427
 embPattern_flip, 427
 embPattern_flipHorizontal, 428
 embPattern_flipVertical, 428
 embPattern_free, 428
 embPattern_hideStitchesOverLength, 428
 embPattern_horizontal_fill, 428
 embPattern_jumpStitches, 428
 embPattern_lengthHistogram, 428
 embPattern_loadExternalColorFile, 428
 embPattern_maximumStitchLength, 428
 embPattern_minimumStitchLength, 428
 embPattern_movePolylinesToStitchList, 429
 embPattern_moveStitchListToPolylines, 429
 embPattern_read, 429
 embPattern_readAuto, 429
 embPattern_realStitches, 429
 embPattern_render, 429
 embPattern_scale, 429
 embPattern_simulate, 429
 embPattern_totalStitchLength, 429
 embPattern_trimStitches, 430
 embPattern_write, 430
 embPattern_writeAuto, 430
 EmbPoint, 419
 EmbPolygon, 419
 EmbPolyline, 419
 EmbRadiusDim, 419
 EmbRay, 419
 EmbReal, 419
 EmbRect, 419
 embRect_area, 430
 embRect_init, 430
 EmbSatinOutline, 419
 embSatinOutline_generateSatinOutline, 430
 embSatinOutline_renderStitches, 430
 EmbSpline, 419
 EmbStitch, 420
 EmbTextMulti, 420
 EmbTextSingle, 420
 EmbThread, 420
 embThread_findNearestColor, 430
 embThread_findNearestThread, 431
 embThread_getRandom, 431
 EmbTime, 420
 embTime_initNow, 431
 embTime_time, 431
 EmbVector, 420
 embVector_add, 431
 embVector_angle, 431
 embVector_average, 432
 embVector_cross, 432
 embVector_distance, 432
 embVector_dot, 432
 embVector_length, 432
 embVector_multiply, 432
 embVector_normalize, 433
 embVector_relativeX, 433
 embVector_relativeY, 433
 embVector_subtract, 433
 embVector_transpose_product, 433
 embVector_unit, 433
 END, 416
 Exquisite_Polyester, 416
 formatTable, 435
 Fufu_Polyester, 416
 Fufu_Rayon, 416
 full_test_matrix, 433
 getArcCenter, 433
 getArcDataFromBulge, 434
 getCircleCircleIntersections, 434
 getCircleTangentPoints, 434
 Hemingworth_Polyester, 416
 hilbert_curve, 434
 hus_thread, 416
 husThreads, 435
 Isacord_Polyester, 416
 Isafil_Rayon, 416
 jef_thread, 416
 jefThreads, 435
 JUMP, 416
 L_system, 420
 LIBEMBROIDERY_EMBEDDED_VERSION, 416
 lindenmayer_system, 434
 Madeira_Polyester, 416

Madeira_Rayon, 416
Marathon_Polyester, 416
Marathon_Rayon, 416
MAX_STITCHES, 417
MAX_THREADS, 417
Metro_Polyester, 417
NORMAL, 417
numberOfFormats, 417
Pantone, 417
pcm_thread, 417
pcmThreads, 436
pec_thread, 417
pecThreadCount, 436
pecThreads, 436
radians, 434
report, 434
RobisonAnton_Polyester, 417
RobisonAnton_Rayon, 417
SEQUIN, 417
shv_thread, 417
shvThreadCount, 436
shvThreads, 436
Sigma_Polyester, 417
STOP, 417
Sulky_Rayon, 417
SVG_Colors, 417
testMain, 435
thread_color, 420
ThreadArt_Polyester, 417
ThreadArt_Rayon, 417
threadColor, 435
threadColorName, 435
threadColorNum, 435
ThreaDelight_Polyester, 418
TRIM, 418
vipDecodingTable, 436
Z102_Isacord_Polyester, 418
embroidery_internal.h
 bcf_difat_create, 461
 bcf_directory, 459
 bcf_directory_entry, 459
 bcf_directory_free, 461
 bcf_file, 459
 bcf_file_difat, 459
 bcf_file_difat_free, 461
 bcf_file_fat, 459
 bcf_file_fat_free, 461
 bcf_file_free, 461
 bcf_file_header, 459
 bcfFile_read, 461
 bcfFileFat_create, 461
 bcfFileHeader_isValid, 461
 bcfFileHeader_read, 462
 binaryReadString, 462
 binaryReadUnicodeString, 462
 binaryWriteInt, 462
 binaryWriteIntBE, 462
 binaryWriteShort, 462
 binaryWriteUInt, 462
 binaryWriteUIntBE, 462
 binaryWriteUShort, 463
 binaryWriteUShortBE, 463
 BULGETOCONTROL, 451
 BULGETOEND, 452
 check_header_present, 463
 CompoundFileDirectory, 463
 CompoundFileDirectoryEntry, 463
 CompoundFileSector_DIFAT_Sector, 452
 CompoundFileSector_EndOfChain, 452
 CompoundFileSector_FAT_Sector, 452
 CompoundFileSector_FreeSector, 452
 CompoundFileSector_MaxRegSector, 452
 CompoundFileStreamId_MaxRegularStreamId, 452
 CompoundFileStreamId_NoStream, 452
 compress, 460
 compress_get_bits, 463
 compress_get_position, 463
 compress_get_token, 464
 compress_load_block, 464
 compress_load_character_huffman, 464
 compress_load_character_length_huffman, 464
 compress_load_distance_huffman, 464
 compress_pop, 464
 compress_read_variable_length, 464
 copy_trim, 464
 create_test_file_1, 464
 create_test_file_2, 464
 create_test_file_3, 464
 CSV_EXPECT, 460
 CSV_EXPECT_COMMA, 460
 CSV_EXPECT_NULL, 460
 CSV_EXPECT_QUOTE1, 460
 CSV_EXPECT_QUOTE2, 460
 CSV_MODE, 460
 CSV_MODE_COMMENT, 460
 CSV_MODE_NULL, 460
 CSV_MODE_STITCH, 461
 CSV_MODE_THREAD, 460
 CSV_MODE_VARIABLE, 460
 CUBICTOCONTROL1, 452
 CUBICTOCONTROL2, 452
 CUBICTOEND, 452
 decode_t01_record, 464
 decode_tajima_ternary, 465
 decodeNewStitch, 465
 DXF_VERSION_2000, 452
 DXF_VERSION_2002, 452
 DXF_VERSION_2004, 452
 DXF_VERSION_2006, 452
 DXF_VERSION_2007, 452
 DXF_VERSION_2009, 452
 DXF_VERSION_2010, 453
 DXF_VERSION_2013, 453
 DXF_VERSION_R10, 453
 DXF_VERSION_R11, 453

DXF_VERSION_R12, 453
DXF_VERSION_R13, 453
DXF_VERSION_R14, 453
DXF_VERSION_R15, 453
DXF_VERSION_R18, 453
DXF_VERSION_R21, 453
DXF_VERSION_R24, 453
DXF_VERSION_R27, 453
ELEMENT_A, 453
ELEMENT_ANIMATE, 453
ELEMENT_ANIMATECOLOR, 453
ELEMENT_ANIMATEMOTION, 453
ELEMENT_ANIMATETRANSFORM, 453
ELEMENT_ANIMATION, 453
ELEMENT_AUDIO, 454
ELEMENT_CIRCLE, 454
ELEMENT_DEFS, 454
ELEMENT_DESC, 454
ELEMENT_DISCARD, 454
ELEMENT_ELLIPSE, 454
ELEMENT_FONT, 454
ELEMENT_FONT_FACE, 454
ELEMENT_FONT_FACE_SRC, 454
ELEMENT_FONT_FACE_URI, 454
ELEMENT_FOREIGN_OBJECT, 454
ELEMENT_G, 454
ELEMENT_GLYPH, 454
ELEMENT_HANDLER, 454
ELEMENT_HKERN, 454
ELEMENT_IMAGE, 454
ELEMENT_LINE, 454
ELEMENT_LINEAR_GRADIENT, 454
ELEMENT_LISTENER, 455
ELEMENT_METADATA, 455
ELEMENT_MISSING_GLYPH, 455
ELEMENT_MPATH, 455
ELEMENT_PATH, 455
ELEMENT_POLYGON, 455
ELEMENT_POLYLINE, 455
ELEMENT_PREFETCH, 455
ELEMENT_RADIAL_GRADIENT, 455
ELEMENT_RECT, 455
ELEMENT_SCRIPT, 455
ELEMENT_SET, 455
ELEMENT_SOLID_COLOR, 455
ELEMENT_STOP, 455
ELEMENT_SVG, 455
ELEMENT_SWITCH, 455
ELEMENT_TBREAK, 455
ELEMENT_TEXT, 455
ELEMENT_TEXT_AREA, 456
ELEMENT_TITLE, 456
ELEMENT_TSPAN, 456
ELEMENT_USE, 456
ELEMENT_VIDEO, 456
ELEMENT_XML, 456
ELLIPSETOEND, 456
ELLIPSETORAD, 456
EMB_BIG_ENDIAN, 456
EMB_INT16_BIG, 456
EMB_INT16_LITTLE, 456
EMB_INT32_BIG, 456
EMB_INT32_LITTLE, 456
EMB_LITTLE_ENDIAN, 456
EMB_MAX, 456
EMB_MIN, 456
emb_optOut, 465
emb_readline, 465
embColor_read, 465
embColor_write, 465
emblnt_read, 466
emblnt_write, 466
encode_t01_record, 466
encode_tajima_ternary, 466
ENDIAN_HOST, 456
entriesInDifatSector, 466
fpad, 466
fread_int16, 466
fread_int32_be, 467
fread_uint16, 467
GetFile, 467
GREEN_TERM_COLOR, 457
HOOP_110X110, 457
HOOP_126X110, 457
HOOP_140X200, 457
HOOP_230X200, 457
HOOP_50X50, 457
huffman, 460
huffman_build_table, 467
huffman_table_lookup, 467
hus_compress, 467
hus_decompress, 467
imageWithFrame, 483
LINETO, 457
loadFatFromSector, 468
mitDecodeStitch, 468
mitEncodeStitch, 468
MOVETO, 457
N_PES VERSIONS, 457
numberOfEntriesInDifatSector, 468
ObjectTypeRootEntry, 457
ObjectTypeStorage, 457
ObjectTypeStream, 457
ObjectTypeUnknown, 457
PES0001, 457
PES0020, 457
PES0022, 457
PES0030, 457
PES0040, 458
PES0050, 458
PES0055, 458
PES0056, 458
PES0060, 458
PES0070, 458
PES0080, 458
PES0090, 458

PES0100, 458
pfaffDecode, 468
pfaffEncode, 468
printArcResults, 468
QUADTOCONTROL, 458
QUADTOEND, 458
read100, 469
read10o, 469
readArt, 469
readBmc, 469
readBro, 469
readCnd, 469
readCol, 469
readCsd, 469
readCsv, 469
readDat, 469
readDem, 469
readDescriptions, 470
readDsb, 470
readDst, 470
readDsz, 470
readDxf, 470
readEdr, 470
readEmd, 470
readExp, 470
readExy, 470
readEys, 470
readFeatherPatterns, 471
readFullSector, 471
readFxy, 471
readGc, 471
readGnc, 471
readGt, 471
readHoopName, 471
readHus, 471
readImageString, 471
readInb, 472
readInf, 472
readJef, 472
readKsm, 472
readMax, 472
readMit, 472
readMotifPatterns, 472
readNew, 472
readNextSector, 472
readOfm, 472
readPcd, 473
readPcm, 473
readPcq, 473
readPcs, 473
readPec, 473
readPecStitches, 473
readPel, 473
readPem, 473
readPes, 474
readPESHeaderV10, 474
readPESHeaderV5, 474
readPESHeaderV6, 474
readPESHeaderV7, 474
readPESHeaderV8, 474
readPESHeaderV9, 474
readPhb, 474
readPhc, 474
readPlt, 474
readProgrammableFills, 475
readRgb, 475
readSew, 475
readShv, 475
readSst, 475
readStx, 475
readSvg, 475
readT01, 475
readT09, 475
readTap, 475
readThr, 476
readThreads, 476
readTxt, 476
readU00, 476
readU01, 476
readVip, 476
readVp3, 476
readXxx, 476
readZsk, 476
RED_TERM_COLOR, 458
RESET_TERM_COLOR, 458
safe_free, 476
stringInArray, 477
StxThread, 460
SubDescriptor, 460
SVG_ATTRIBUTE, 458
SVG_CATCH_ALL, 458
SVG_CREATOR_EMBROIDERMODDER, 458
SVG_CREATOR_ILLUSTRATOR, 458
SVG_CREATOR_INKSCAPE, 458
SVG_CREATOR_NULL, 459
SVG_ELEMENT, 459
SVG_EXPECT_ATTRIBUTE, 459
SVG_EXPECT_ELEMENT, 459
SVG_EXPECT_NULL, 459
SVG_EXPECT_VALUE, 459
SVG_MEDIA_PROPERTY, 459
SVG_NULL, 459
SVG_PROPERTY, 459
SvgAttribute, 460
testEmbCircle, 477
testEmbCircle_2, 477
testEmbFormat, 477
testGeomArc, 477
testTangentPoints, 477
testThreadColor, 477
ThredExtension, 460
ThredHeader, 460
VipHeader, 460
vp3Hoop, 460
write100, 477
write10o, 477

write_24bit, 477
writeArt, 478
writeBmc, 478
writeBro, 478
writeCnd, 478
writeCol, 478
writeCsd, 478
writeCsv, 478
writeDat, 478
writeDem, 478
writeDsb, 478
writeDst, 478
writeDsz, 479
writeDxf, 479
writeEdr, 479
writeEmd, 479
writeExp, 479
writeExy, 479
writeEys, 479
writeFxy, 479
writeGc, 479
writeGnc, 479
writeGt, 479
writeHus, 480
writeInb, 480
writeInf, 480
writeJef, 480
writeKsm, 480
writeMax, 480
writeMit, 480
writeNew, 480
writeOfm, 480
writePcd, 480
writePcm, 480
writePcq, 481
writePcs, 481
writePec, 481
writePecStitches, 481
writePel, 481
writePem, 481
writePes, 481
writePhb, 481
writePhc, 481
writePlt, 481
writeRgb, 482
writeSew, 482
writeShv, 482
writeSst, 482
writeStx, 482
writeSvg, 482
writeT01, 482
writeT09, 482
writeTap, 482
writeThr, 482
writeTxt, 482
writeU00, 483
writeU01, 483
writeVip, 483
writeVp3, 483
writeXxx, 483
writeZsk, 483
YELLOW_TERM_COLOR, 459
EmbSatinOutline
 embroidery.h, 419
EmbSatinOutline__{_}, 143
 length, 143
 side1, 143
 side2, 143
embSatinOutline_generateSatinOutline
 embroidery.h, 430
 main.c, 566
embSatinOutline_renderStitches
 embroidery.h, 430
 main.c, 566
EmbSpline
 embroidery.h, 419
EmbSpline__{_}, 144
 beziers, 144
EmbStitch
 embroidery.h, 420
EmbStitch__{_}, 144
 color, 144
 flags, 144
 x, 144
 y, 145
EmbTextMulti
 embroidery.h, 420
EmbTextMulti__{_}, 145
 position, 145
 text, 145
EmbTextSingle
 embroidery.h, 420
EmbTextSingle__{_}, 145
 position, 146
 text, 146
EmbThread
 embroidery.h, 420
EmbThread__{_}, 146
 catalogNumber, 146
 color, 146
 description, 146
embThread_findNearestColor
 embroidery.h, 430
 main.c, 567
embThread_findNearestThread
 embroidery.h, 431
 main.c, 567
embThread_getRandom
 embroidery.h, 431
 main.c, 567
EmbTime
 embroidery.h, 420
EmbTime__{_}, 147
 day, 147
 hour, 147
 minute, 147

month, 147
second, 147
year, 147
embTime_initNow
 embroidery.h, 431
 main.c, 567
embTime_time
 embroidery.h, 431
 main.c, 567
EmbVector
 embroidery.h, 420
EmbVector_, 148
 x, 148
 y, 148
embVector_add
 embroidery.h, 431
 vector.c, 556
embVector_angle
 embroidery.h, 431
 vector.c, 556
embVector_average
 embroidery.h, 432
 vector.c, 556
embVector_cross
 embroidery.h, 432
 vector.c, 556
embVector_distance
 embroidery.h, 432
 vector.c, 557
embVector_dot
 embroidery.h, 432
 vector.c, 557
embVector_length
 embroidery.h, 432
 vector.c, 557
embVector_multiply
 embroidery.h, 432
 vector.c, 557
embVector_normalize
 embroidery.h, 433
 vector.c, 557
embVector_print
 main.c, 567
embVector_relativeX
 embroidery.h, 433
 vector.c, 557
embVector_relativeY
 embroidery.h, 433
 vector.c, 558
embVector_subtract
 embroidery.h, 433
 vector.c, 558
embVector_transpose_product
 embroidery.h, 433
 vector.c, 558
embVector_unit
 embroidery.h, 433
 vector.c, 558

EmbView
 embroidermodder.h, 339
EmbView_-, 148
 filename, 149
 grid_mode, 149
 grid_type, 149
 lwt_mode, 150
 metric, 150
 n_selected, 150
 origin, 150
 ortho_mode, 150
 pattern, 150
 polar_mode, 150
 qsnap_mode, 150
 qtrack_mode, 150
 real_render, 150
 rubber_mode, 150
 ruler_mode, 151
 scale, 151
 selected, 151
 simulate, 151
 simulation_start, 151
 snap_mode, 151
 text_angle, 151
 text_font, 151
 text_size, 151
 text_style_bold, 151
 text_style_italic, 151
 text_style_overline, 152
 text_style_strikeout, 152
 text_style_underline, 152
 ui_mode, 152
 undo_history, 152
emd, 16, 514
emdDecode
 format_emd.c, 514
enableLwt
 StatusBarButton, 294
enableMoveRapidFire
 MainWindow, 182
 View, 322
enablePromptRapidFire
 MainWindow, 182
enableRapidFire
 CmdPrompt, 90
enableReal
 StatusBarButton, 294
encode_record
 format_dst.c, 512
encode_t01_record
 embroidery_internal.h, 466
 encoding.c, 492
encode_tajima_ternary
 embroidery_internal.h, 466
 encoding.c, 492
encode_tap_record
 format_tap.c, 538
encoding.c

decode_t01_record, 491
 decode_tajima_ternary, 491
 decodeNewStitch, 491
 embColor_fromHexStr, 491
 emblnt_read, 492
 emblnt_write, 492
 encode_t01_record, 492
 encode_tajima_ternary, 492
 mitDecodeStitch, 492
 mitEncodeStitch, 492
 pfaffDecode, 492
 pfaffEncode, 493
 reverse_byte_order, 493
 write_24bit, 493
END
 embroidery.h, 416
end
 EmbArc_, 124
 EmbBezier_, 126
 EmbLine_, 137
endCommand
 CmdPrompt, 90
 CmdPromptInput, 103
ENDIAN_HOST
 embroidery_internal.h, 456
enterEvent
 View, 322
entriesInDifatSector
 embroidery_internal.h, 466
 main.c, 568
Error
 mainwindow.cpp, 384
escapePressed
 CmdPrompt, 90
 CmdPromptInput, 103
 MainWindow, 182
 MdiWindow, 217
 View, 322
event
 Application, 62
eventFilter
 CmdPromptInput, 103
 PropertyEditor, 254
exit
 MainWindow, 182
exitApp
 embroidermodder.cpp, 334
exp, 16, 514
expDecode
 format_exp.c, 514
Exquisite_Polyester
 embroidery.h, 416
extension
 EmbFormatList_, 132
extensions
 settings-dialog.cpp, 398
extern/libembroidery/src/array.c, 399
extern/libembroidery/src/compress.c, 401
 extern/libembroidery/src/embroidery.h, 403, 436
 extern/libembroidery/src/embroidery_internal.h, 443, 483
 extern/libembroidery/src/encoding.c, 490
 extern/libembroidery/src/fill.c, 493
 extern/libembroidery/src/formats.c, 498
 extern/libembroidery/src/formats/format_100.c, 502
 extern/libembroidery/src/formats/format_10o.c, 502
 extern/libembroidery/src/formats/format_art.c, 503
 extern/libembroidery/src/formats/format_bmc.c, 503
 extern/libembroidery/src/formats/format_bro.c, 504
 extern/libembroidery/src/formats/format_cnd.c, 504
 extern/libembroidery/src/formats/format_col.c, 505
 extern/libembroidery/src/formats/format_csd.c, 506
 extern/libembroidery/src/formats/format_csv.c, 507
 extern/libembroidery/src/formats/format_dat.c, 508
 extern/libembroidery/src/formats/format_dem.c, 509
 extern/libembroidery/src/formats/format_dsb.c, 509
 extern/libembroidery/src/formats/format_dst.c, 510
 extern/libembroidery/src/formats/format_dsz.c, 512
 extern/libembroidery/src/formats/format_dxf.c, 512
 extern/libembroidery/src/formats/format_edr.c, 513
 extern/libembroidery/src/formats/format_emd.c, 514
 extern/libembroidery/src/formats/format_exp.c, 514
 extern/libembroidery/src/formats/format_exy.c, 515
 extern/libembroidery/src/formats/format_eyc.c, 515
 extern/libembroidery/src/formats/format_fxy.c, 516
 extern/libembroidery/src/formats/format_gc.c, 516
 extern/libembroidery/src/formats/format_gnc.c, 517
 extern/libembroidery/src/formats/format_gt.c, 517
 extern/libembroidery/src/formats/format_hus.c, 518
 extern/libembroidery/src/formats/format_inb.c, 519
 extern/libembroidery/src/formats/format_inf.c, 519
 extern/libembroidery/src/formats/format_jef.c, 520
 extern/libembroidery/src/formats/format_ksm.c, 521
 extern/libembroidery/src/formats/format_max.c, 521
 extern/libembroidery/src/formats/format_mit.c, 522
 extern/libembroidery/src/formats/format_new.c, 523
 extern/libembroidery/src/formats/format_ofm.c, 523
 extern/libembroidery/src/formats/format_pcd.c, 524
 extern/libembroidery/src/formats/format_pcm.c, 525
 extern/libembroidery/src/formats/format_pcq.c, 525
 extern/libembroidery/src/formats/format_pcs.c, 526
 extern/libembroidery/src/formats/format_pec.c, 526
 extern/libembroidery/src/formats/format_pel.c, 528
 extern/libembroidery/src/formats/format_pem.c, 528
 extern/libembroidery/src/formats/format_pes.c, 529
 extern/libembroidery/src/formats/format_phb.c, 531
 extern/libembroidery/src/formats/format_phc.c, 532
 extern/libembroidery/src/formats/format_plt.c, 532
 extern/libembroidery/src/formats/format_rgb.c, 533
 extern/libembroidery/src/formats/format_sew.c, 533
 extern/libembroidery/src/formats/format_shv.c, 534
 extern/libembroidery/src/formats/format_sst.c, 534
 extern/libembroidery/src/formats/format_stx.c, 535
 extern/libembroidery/src/formats/format_svg.c, 535
 extern/libembroidery/src/formats/format_t01.c, 537
 extern/libembroidery/src/formats/format_t09.c, 537

extern/libembroidery/src/formats/format_tap.c, 538
extern/libembroidery/src/formats/format_thr.c, 538
extern/libembroidery/src/formats/format_txt.c, 539
extern/libembroidery/src/formats/format_u00.c, 539
extern/libembroidery/src/formats/format_u01.c, 540
extern/libembroidery/src/formats/format_vip.c, 540
extern/libembroidery/src/formats/format_vp3.c, 542
extern/libembroidery/src/formats/format_xxx.c, 543
extern/libembroidery/src/formats/format_zsk.c, 544
extern/libembroidery/src/geometry.c, 544
extern/libembroidery/src/geometry/arc.c, 546
extern/libembroidery/src/geometry/circle.c, 549
extern/libembroidery/src/geometry/ellipse.c, 550
extern/libembroidery/src/geometry/functions.c, 552
extern/libembroidery/src/geometry/line.c, 553
extern/libembroidery/src/geometry/path.c, 553
extern/libembroidery/src/geometry/polygon.c, 553
extern/libembroidery/src/geometry/polyline.c, 553
extern/libembroidery/src/geometry/rect.c, 554
extern/libembroidery/src/geometry/text.c, 554
extern/libembroidery/src/geometry/vector.c, 556
extern/libembroidery/src/image.c, 558
extern/libembroidery/src/main.c, 559
extern/libembroidery/src/pattern.c, 570
extern/libembroidery/src/thread-color.c, 575
exy, 16, 515
eys, 471, 516

F10Pressed
 CmdPrompt, 90
 CmdPromptInput, 103

F11Pressed
 CmdPrompt, 90
 CmdPromptInput, 103

F12Pressed
 CmdPrompt, 90
 CmdPromptInput, 103

F1Pressed
 CmdPrompt, 90
 CmdPromptInput, 103

F2Pressed
 CmdPrompt, 91
 CmdPromptInput, 104

F3Pressed
 CmdPrompt, 91
 CmdPromptInput, 104

F4Pressed
 CmdPrompt, 91
 CmdPromptInput, 104

F5Pressed
 CmdPrompt, 91
 CmdPromptInput, 104

F6Pressed
 CmdPrompt, 91
 CmdPromptInput, 104

F7Pressed
 CmdPrompt, 91
 CmdPromptInput, 104

F8Pressed

CmdPrompt, 91
CmdPromptInput, 104

F9Pressed
 CmdPrompt, 91
 CmdPromptInput, 104

factor
 UndoableScaleCommand, 315

fat
 _bcf_file, 53

fatEntries
 _bcf_file_fat, 54

fatEntryCount
 _bcf_file_fat, 54

fatSectorCount
 _bcf_file_difat, 53

fatSectorEntries
 _bcf_file_difat, 54

fieldEdited
 PropertyEditor, 254

fieldNewText
 PropertyEditor, 256

fieldNoText
 PropertyEditor, 256

fieldOffText
 PropertyEditor, 256

fieldOldText
 PropertyEditor, 256

fieldOnText
 PropertyEditor, 256

fieldVariesText
 PropertyEditor, 256

fieldYesText
 PropertyEditor, 256

file_toolbar
 embroidermodder.h, 345
 mainwindow-toolbars.cpp, 381

fileExtension
 MdiWindow, 217

fileMenu
 MainWindow, 205

filename
 EmbView_, 149

fileWasLoaded
 MdiWindow, 222

fill.c
 dragon_curve, 494
 embPattern_combine, 494
 embPattern_convertGeometry, 494
 embPattern_crossstitch, 494
 embPattern_horizontal_fill, 494
 embPattern_stitchArc, 494
 embPattern_stitchCircle, 494
 embPattern_stitchEllipse, 495
 embPattern_stitchPath, 495
 embPattern_stitchPolygon, 495
 embPattern_stitchPolyline, 495
 embPattern_stitchRect, 496
 embPattern_stitchText, 496

embPolygon_reduceByDistance, 496
 embPolygon_reduceByNth, 496
 generate_dragon_curve, 496
 greedy_algorithm, 496
 hilbert_curve, 496
 hilbert_curve_l_system, 497
 join_short_stitches, 497
 lindenmayer_system, 497
 rules, 497
 save_points_to_pattern, 497
 threshold_method, 497
 filled
 DimLeaderObject, 116
 findIndex
 PolygonObject, 236
 PolylineObject, 242
 findMdiWindow
 MainWindow, 182
 firstDifatSectorLocation
 _bcf_file_header, 55
 firstDirectorySectorLocation
 _bcf_file_header, 55
 firstMiniFATSectorLocation
 _bcf_file_header, 55
 firstRun
 UiObject_, 306
 flag
 EmbGeometry_, 133
 FLAG_CIRCLE
 main.c, 561
 FLAG_CIRCLE_SHORT
 main.c, 562
 FLAG_COMBINE
 main.c, 562
 FLAG_CROSS_STITCH
 main.c, 562
 FLAG_ELLIPSE
 main.c, 562
 FLAG_ELLIPSE_SHORT
 main.c, 562
 FLAG_FILL
 main.c, 562
 FLAG_FILL_SHORT
 main.c, 562
 FLAG_FORMATS
 main.c, 562
 FLAG_FORMATS_SHORT
 main.c, 562
 FLAG_FULL_TEST_SUITE
 main.c, 562
 FLAG_HELP
 main.c, 562
 FLAG_HELP_SHORT
 main.c, 562
 FLAG_HILBERT_CURVE
 main.c, 562
 FLAG_LINE
 main.c, 562
 FLAG_LINE_SHORT
 main.c, 562
 FLAG_POLYGON
 main.c, 562
 FLAG_POLYGON_SHORT
 main.c, 562
 FLAG_POLYLINE
 main.c, 562
 FLAG_POLYLINE_SHORT
 main.c, 563
 FLAG QUIET
 main.c, 563
 FLAG QUIET_SHORT
 main.c, 563
 FLAG_RENDER
 main.c, 563
 FLAG_RENDER_SHORT
 main.c, 563
 FLAG_SATIN
 main.c, 563
 FLAG_SATIN_SHORT
 main.c, 563
 FLAG_SIERPINSKI_TRIANGLE
 main.c, 563
 FLAG_SIMULATE
 main.c, 563
 FLAG_STITCH
 main.c, 563
 FLAG_STITCH_SHORT
 main.c, 563
 FLAG_TEST
 main.c, 563
 FLAG_TO
 main.c, 563
 FLAG_TO_SHORT
 main.c, 563
 FLAG_VERBOSE
 main.c, 563
 FLAG_VERBOSE_SHORT
 main.c, 563
 FLAG_VERSION
 main.c, 563
 FLAG_VERSION_SHORT
 main.c, 563
 flagList
 EmbPath_, 139
 flags
 EmbStitch_, 144
 Flared
 DimLeaderObject, 112
 Fletching
 DimLeaderObject, 112
 floatingChanged
 CmdPrompt, 91
 floatingChangedToolBar
 MainWindow, 183
 fname
 UiObject_, 306

focusWidget
 PropertyEditor, 256
 UndoEditor, 316
forceRepaint
 MdiArea, 210
 SelectBox, 270
format_100.c
 read100, 502
 write100, 502
format_10o.c
 read10o, 503
 write10o, 503
format_art.c
 readArt, 503
 writeArt, 503
format_bmc.c
 readBmc, 504
 writeBmc, 504
format_bro.c
 readBro, 504
 writeBro, 504
format_cnd.c
 readCnd, 505
 writeCnd, 505
format_col.c
 readCol, 506
 writeCol, 506
format_csd.c
 _subMask, 507
 _xorMask, 507
 BuildDecryptionTable, 506
 csd_decryptArray, 507
 CsdSubMaskSize, 506
 CsdXorMaskSize, 506
 DecodeCsdByte, 507
 readCsd, 507
 writeCsd, 507
format_csv.c
 csvStitchFlagToStr, 508
 csvStrToStitchFlag, 508
 readCsv, 508
 writeCsv, 508
format_dat.c
 readDat, 508
 writeDat, 508
format_dem.c
 readDem, 509
 writeDem, 509
format_dsb.c
 readDsb, 510
 writeDsb, 510
format_dst.c
 cci, 511
 decode_record_flags, 511
 encode_record, 512
 readDst, 512
 set_dst_variable, 512
 writeDst, 512
format_dsz.c
 readDsz, 512
 writeDsz, 512
format_dxf.c
 readDxf, 513
 readLine, 513
 writeDxf, 513
format_edr.c
 readEdr, 513
 writeEdr, 513
format_emd.c
 emdDecode, 514
 readEmd, 514
 writeEmd, 514
format_exp.c
 expDecode, 514
 readExp, 514
 writeExp, 515
format_exy.c
 decode_exy_flags, 515
 readExy, 515
 writeExy, 515
format_eyc.c
 readEys, 515
 writeEys, 516
format_fxy.c
 readFxy, 516
 writeFxy, 516
format_gc.c
 readGc, 516
 writeGc, 517
format_gnc.c
 readGnc, 517
 writeGnc, 517
format_gt.c
 readGt, 517
 writeGt, 518
format_hus.c
 husCompressData, 518
 husDecodeByte, 518
 husDecodeStitchType, 518
 husDecompressData, 518
 husEncodeByte, 518
 husEncodeStitchType, 518
 readHus, 519
 writeHus, 519
format_inb.c
 readInb, 519
 writeInb, 519
format_inf.c
 readInf, 519
 writeInf, 520
format_jef.c
 jefDecode, 520
 jefEncode, 520
 jefGetHoopSize, 520
 jefSetHoopFromId, 520
 read_hoop, 520

readJef, 521
 writeJef, 521
format_ksm.c
 ksmEncode, 521
 readKsm, 521
 writeKsm, 521
format_max.c
 max_header, 522
 readMax, 522
 writeMax, 522
format_mit.c
 readMit, 522
 writeMit, 523
format_new.c
 readNew, 523
 writeNew, 523
format_ofm.c
 ofmDecode, 524
 ofmReadBlockHeader, 524
 ofmReadClass, 524
 ofmReadColorChange, 524
 ofmReadExpanded, 524
 ofmReadLibrary, 524
 ofmReadThreads, 524
 readOfm, 524
 writeOfm, 524
format_pcd.c
 readPcd, 525
 writePcd, 525
format_pcm.c
 readPcm, 525
 writePcm, 525
format_pcq.c
 readPcq, 526
 writePcq, 526
format_pcs.c
 readPcs, 526
 writePcs, 526
format_pec.c
 pecEncode, 527
 pecEncodeJump, 527
 pecEncodeStop, 527
 readPec, 527
 readPecStitches, 527
 writeImage, 527
 writePec, 527
 writePecStitches, 527
format_pel.c
 readPel, 528
 writePel, 528
format_pem.c
 readPem, 528
 writePem, 528
format_pes.c
 pes_version, 531
 pes_version_strings, 531
 pesWriteEmbOneSection, 529
 pesWriteSewSegSection, 529
 readDescriptions, 529
 readFeatherPatterns, 529
 readHoopName, 529
 readImageString, 530
 readMotifPatterns, 530
 readPes, 530
 readPESHeaderV10, 530
 readPESHeaderV5, 530
 readPESHeaderV6, 530
 readPESHeaderV7, 530
 readPESHeaderV8, 530
 readPESHeaderV9, 530
 readProgrammableFills, 530
 readThreads, 530
 writePes, 531
format_phb.c
 readPhb, 531
 writePhb, 531
format_phc.c
 readPhc, 532
 writePhc, 532
format_plt.c
 readPlt, 532
 writePlt, 532
format_rgb.c
 readRgb, 533
 writeRgb, 533
format_sew.c
 readSew, 533
 sewDecode, 533
 writeSew, 533
format_shv.c
 readShv, 534
 shvDecode, 534
 shvDecodeShort, 534
 writeShv, 534
format_sst.c
 readSst, 534
 writeSst, 535
format_stx.c
 readStx, 535
 stxReadThread, 535
 writeStx, 535
format_svg.c
 attributeList, 536
 current_element_id, 536
 currentAttribute, 536
 currentValue, 536
 n_attributes, 536
 readSvg, 536
 svgCreator, 536
 svgExpect, 536
 svgMultiValue, 536
 writeSvg, 536
format_t01.c
 readT01, 537
 writeT01, 537
format_t09.c

readT09, 537
writeT09, 537
format_tap.c
decode_tap_record_flags, 538
encode_tap_record, 538
readTap, 538
writeTap, 538
format_thr.c
readThr, 538
writeThr, 539
format_txt.c
readTxt, 539
writeTxt, 539
format_u00.c
readU00, 539
writeU00, 540
format_u01.c
readU01, 540
writeU01, 540
format_vip.c
readVip, 541
vipCompressData, 541
vipDecodeByte, 541
vipDecodeStitchType, 541
vipDecodingTable, 541
vipDecompressData, 541
vipEncodeByte, 541
vipEncodeStitchType, 541
writeVip, 541
format_vp3.c
readVp3, 542
vp3Decode, 542
vp3DecodeInt16, 542
vp3PatchByteCount, 542
vp3ReadHoopSection, 542
vp3ReadString, 543
vp3WriteString, 543
vp3WriteStringLen, 543
writeVp3, 543
format_xxx.c
readXxx, 543
writeXxx, 543
xxxDecodeByte, 543
xxxEncodeDesign, 544
xxxEncodeStitch, 544
xxxEncodeStop, 544
format_zsk.c
readZsk, 544
writeZsk, 544
formatFilterOpen
 MainWindow, 205
formatFilterSave
 MainWindow, 205
formats.c
 binaryWriteInt, 499
 binaryWriteIntBE, 499
 binaryWriteShort, 499
 binaryWriteUInt, 499
 binaryWriteUIntBE, 499
 binaryWriteUShort, 499
 binaryWriteUShortBE, 499
 emb_identify_format, 500
 embFormat_getExtension, 500
 embPattern_read, 500
 embPattern_readAuto, 500
 embPattern_write, 500
 embPattern_writeAuto, 500
 formatTable, 501
 fpad, 501
 fread_int16, 501
 fread_int32_be, 501
 fread_uint16, 501
 imageWithFrame, 502
 safe_free, 501
 formatTable
 embroidery.h, 435
 formats.c, 501
 formatType
 SaveObject, 269
Fortron, 471, 516
fpad
 embroidery_internal.h, 466
 formats.c, 501
fread_int16
 embroidery_internal.h, 466
 formats.c, 501
fread_int32_be
 embroidery_internal.h, 467
 formats.c, 501
fread_uint16
 embroidery_internal.h, 467
 formats.c, 501
fromCenter
 UndoableNavCommand, 312
fromTransform
 UndoableNavCommand, 312
Fufu_Polyester
 embroidery.h, 416
Fufu_Rayon
 embroidery.h, 416
full_test_matrix
 embroidery.h, 433
functions.c
 degrees, 552
 emb_round, 552
 radians, 552
fxv, 16, 471, 516
g
 EmbColor_, 128
general_check_for_updates
 Settings_, 275
general_current_tip
 Settings_, 275
general_icon_size
 Settings_, 275
general_icon_theme

Settings_, 276
 general_language
 Settings_, 276
 general_mdi_bg_color
 Settings_, 276
 general_mdi_bg_logo
 Settings_, 276
 general_mdi_bg_texture
 Settings_, 276
 general_mdi_bg_use_color
 Settings_, 276
 general_mdi_bg_use_logo
 Settings_, 276
 general_mdi_bg_use_texture
 Settings_, 276
 general_system_help_browser
 Settings_, 276
 general_tip_of_the_day
 Settings_, 276
 generate_dragon_curve
 fill.c, 496
 geometry
 EmbArray_, 125
 EmbLayer_, 136
 EmbPattern_, 140
 geometry.c
 embGeometry_boundingRect, 545
 embGeometry_free, 545
 embGeometry_init, 545
 embGeometry_move, 545
 embGeometry_vulcanize, 545
 get_action_index
 embroidermodder.h, 344
 mainwindow-toolbars.cpp, 381
 get_trim_bounds
 main.c, 568
 getAction
 MainWindow, 183
 getApplication
 MainWindow, 183
 getArcCenter
 arc.c, 549
 embroidery.h, 433
 getArcDataFromBulge
 arc.c, 549
 embroidery.h, 434
 getCircleCircleIntersections
 circle.c, 550
 embroidery.h, 434
 getCircleTangentPoints
 circle.c, 550
 embroidery.h, 434
 getCurrentColor
 MainWindow, 183
 MdiWindow, 217
 getCurrentFile
 MdiWindow, 217
 getCurrentLayer
 MainWindow, 183
 MdiWindow, 217
 getGetCurrentLineType
 MainWindow, 184
 MdiWindow, 218
 getCurrentLineWeight
 MainWindow, 184
 MdiWindow, 218
 getCurrentText
 CmdPrompt, 91
 GetFile
 embroidery_internal.h, 467
 main.c, 568
 getFileSeparator
 MainWindow, 184
 getHistory
 CmdPrompt, 91
 getInfo
 EmbDetailsDialog, 129
 getMdiArea
 MainWindow, 184
 getPrefix
 CmdPrompt, 92
 getScene
 MdiWindow, 218
 getShortCurrentFile
 MdiWindow, 218
 getUndoStack
 View, 322
 getView
 MdiWindow, 218
 gnc, 16, 471, 517
 Gold Thread, 471, 518
 Great Notions, 471, 517
 greedy_algorithm
 fill.c, 496
 GREEN_TERM_COLOR
 embroidery_internal.h, 457
 grid_center
 Settings_, 276
 grid_center_on_origin
 Settings_, 276
 grid_color
 Settings_, 276
 grid_color_match_crosshair
 Settings_, 276
 grid_load_from_file
 Settings_, 276
 grid_mode
 EmbView_, 149
 grid_show_on_load
 Settings_, 276
 grid_show_origin
 Settings_, 276
 grid_size
 Settings_, 276
 grid_size_radius
 Settings_, 277

grid_spacing
 Settings_, 277
grid_spacing_angle
 Settings_, 277
grid_spacing_radius
 Settings_, 277
grid_type
 EmbView_, 149
 Settings_, 277
gridColor
 View, 327
gridPath
 View, 327
gripBaseObj
 View, 328
gripColorCool
 View, 328
gripColorHot
 View, 328
gripEdit
 ArcObject, 68
 BaseObject, 77
 CircleObject, 84
 DimLeaderObject, 113
 EllipseObject, 120
 ImageObject, 158
 LineObject, 167
 PathObject, 227
 PointObject, 232
 PolygonObject, 236
 PolylineObject, 242
 RectObject, 260
 TextSingleObject, 300
gripIndex
 PolygonObject, 238
 PolylineObject, 244
grippingActive
 View, 328
gripSize
 View, 328
group_box_arc_geometry
 data.cpp, 332
 embroidermodder.h, 345
group_box_arc_geometry_entries
 data.cpp, 332
 embroidermodder.h, 345
group_box_arc_misc
 data.cpp, 333
group_box_arc_misc_entries
 data.cpp, 333
group_box_data
 data.cpp, 333
 embroidermodder.h, 345
group_box_ellipse_geometry
 data.cpp, 333
 embroidermodder.h, 345
group_box_ellipse_geometry_entries
 data.cpp, 333

embroidermodder.h, 345
group_box_general
 data.cpp, 333
group_box_general_entries
 data.cpp, 333
group_box_list
 data.cpp, 333
 embroidermodder.h, 345
GroupBoxData
 embroidermodder.h, 339
GroupBoxData_, 152
 icon_name, 152
 key, 152
 label, 153
 map_signal, 153
 object, 153
 type, 153
groupBoxes
 data.cpp, 333
 embroidermodder.h, 345
gscene
 MdiWindow, 222
 SaveObject, 269
 View, 328
gt, 16, 471, 518
gview
 MdiWindow, 222
 UndoableAddCommand, 308
 UndoableDeleteCommand, 308
 UndoableGripEditCommand, 309
 UndoableMirrorCommand, 310
 UndoableMoveCommand, 311
 UndoableNavCommand, 313
 UndoableRotateCommand, 314
 UndoableScaleCommand, 315
handleMoved
 CmdPromptHandle, 96
handlePressed
 CmdPromptHandle, 96
handleReleased
 CmdPromptHandle, 97
Happy, 538
hash
 Action___, 60
hashDeletedObjects
 View, 328
haveExtraDIFATSectors
 main.c, 568
header
 _bcf_file, 53
HEART_MODE_NUM_POINTS
 embroidermodder.h, 343
HEART_MODE_STYLE
 embroidermodder.h, 343
HEART_MODE_XSCALE
 embroidermodder.h, 343
HEART_MODE_YSCALE
 embroidermodder.h, 343

height
 _vp3Hoop, 58
 EmblImage_, 135
 help
 MainWindow, 184
 help_toolbar
 mainwindow-toolbars.cpp, 382
 helpMenu
 MainWindow, 205
 Hemingworth_Polyester
 embroidery.h, 416
 hex_code
 thread_color_, 303
 hideAllGroups
 PropertyEditor, 254
 hideUnimplemented
 MainWindow, 184
 hilbert_curve
 embroidery.h, 434
 fill.c, 496
 hilbert_curve_l_system
 fill.c, 497
 historyAppended
 CmdPrompt, 92
 CmdPromptHistory, 99
 home
 EmbPattern_, 140
 HOOP_110X110
 embroidery_internal.h, 457
 HOOP_126X110
 embroidery_internal.h, 457
 HOOP_140X200
 embroidery_internal.h, 457
 HOOP_230X200
 embroidery_internal.h, 457
 HOOP_50X50
 embroidery_internal.h, 457
 hoop_height
 EmbPattern_, 140
 hoop_padding, 153
 bottom, 153
 left, 153
 right, 153
 top, 153
 hoop_width
 EmbPattern_, 140
 hoopSize
 ThredHeader_, 305
 hoopX
 ThredExtension_, 304
 hoopY
 ThredExtension_, 304
 hour
 EmbTime_, 147
 Huffman, 154
 default_value, 154
 lengths, 154
 nlengths, 154
 ntable, 154
 table, 154
 table_width, 154
 huffman
 embroidery_internal.h, 460
 huffman_build_table
 compress.c, 403
 embroidery_internal.h, 467
 huffman_lookup
 compress.c, 403
 huffman_lookup_data
 compress.c, 403
 huffman_table_lookup
 embroidery_internal.h, 467
 hus, 16, 518
 hus_compress
 compress.c, 403
 embroidery_internal.h, 467
 hus_decompress
 compress.c, 403
 embroidery_internal.h, 467
 hus_thread
 embroidery.h, 416
 husCompressData
 format_hus.c, 518
 husDecodeByte
 format_hus.c, 518
 husDecodeStitchType
 format_hus.c, 518
 husDecompressData
 format_hus.c, 518
 husEncodeByte
 format_hus.c, 518
 husEncodeStitchType
 format_hus.c, 518
 Husqvarna_Viking, 518, 534
 husThreads
 embroidery.h, 435
 thread-color.c, 576
 i_value
 Parameter_, 223
 icon
 Action_, 60
 icon_name
 GroupBoxData_, 152
 icon_toolbar
 mainwindow-toolbars.cpp, 382
 iconDir
 PropertyEditor, 256
 UndoEditor, 317
 iconResize
 MainWindow, 184
 iconSize
 PropertyEditor, 256
 UndoEditor, 317
 id
 UiObject_, 306
 UndoableNavCommand, 312

image.c
 image_diff, 558
 writeImage, 559
image_diff
 image.c, 558
ImageObject, 155
 ~ImageObject, 157
 allGripPoints, 158
 gripEdit, 158
 ImageObject, 157
init, 158
mouseSnapPoint, 158
objectArea, 158
objectBottomLeft, 158
objectBottomRight, 158
objectHeight, 158
objectTopLeft, 159
objectTopRight, 159
objectWidth, 159
paint, 159
setObjectRect, 159
Type, 157
type, 159
updatePath, 159
updateRubber, 159
vulcanize, 159
ImageWidget, 160
 ~ImageWidget, 161
 ImageWidget, 160
 img, 162
 load, 161
 paintEvent, 161
 save, 161
imageWithFrame
 embroidery_internal.h, 483
 formats.c, 502
img
 ImageWidget, 162
imgWidget
 PreviewDialog, 245
inb, 16, 472, 519
Inbro, 472, 519
inf, 472, 520
init
 ArcObject, 68
 CircleObject, 84
 DimLeaderObject, 113
 EllipseObject, 120
 ImageObject, 158
 LineObject, 167
 PathObject, 227
 PointObject, 232
 PolygonObject, 236
 PolylineObject, 242
 RectObject, 261
 TextSingleObject, 300
input_data
 Compress, 109
input_length
 Compress, 109
interface.cpp
 debug_message, 379
Isacord_Polyester
 embroidery.h, 416
Isafil_Rayon
 embroidery.h, 416
isBlinking
 CmdPromptInput, 106
isCommandActive
 CmdPrompt, 92
 MainWindow, 184
isLwtEnabled
 View, 322
isRapidFireEnabled
 CmdPrompt, 92
isRealEnabled
 View, 323
isShiftPressed
 MainWindow, 185
Janome, 520, 533
jef, 16, 520
jef_thread
 embroidery.h, 416
jefDecode
 format_jef.c, 520
jefEncode
 format_jef.c, 520
jefGetHoopSize
 format_jef.c, 520
jefSetHoopFromId
 format_jef.c, 520
jefThreads
 embroidery.h, 435
 thread-color.c, 576
join_short_stitches
 fill.c, 497
JUMP
 embroidery.h, 416
key
 GroupBoxData_, 152
ksm, 16, 521
ksmEncode
 format_ksm.c, 521
L_system
 embroidery.h, 420
label
 GroupBoxData_, 153
labelTipOfDay
 MainWindow, 205
lastCmd
 CmdPromptInput, 106
lastCommand
 CmdPrompt, 92
layer

EmbPattern_, 140
 LayerManager, 162
 ~LayerManager, 163
 addLayer, 163
 LayerManager, 162
 layerModel, 163
 layerModelSorted, 163
 treeView, 164
 layerManager
 MainWindow, 185
 layerModel
 LayerManager, 163
 layerModelSorted
 LayerManager, 163
 layerPrevious
 MainWindow, 185
 layerSelector
 MainWindow, 205
 layerSelectorIndexChanged
 MainWindow, 185
 layoutState
 MainWindow, 206
 left
 _vp3Hoop, 58
 EmbRect_, 142
 hoop_padding, 153
 left2
 _vp3Hoop, 58
 leftBrush
 SelectBox, 271
 leftBrushColor
 SelectBox, 271
 leftPen
 SelectBox, 271
 leftPenColor
 SelectBox, 271
 leftSiblingId
 _bcf_directory_entry, 51
 length
 EmbArray_, 125
 EmbSatinOutline_, 143
 ThredHeader_, 305
 lengths
 Huffman, 154
 LIBEMBROIDERY_EMBEDDED_VERSION
 embroidery.h, 416
 lindenmayer_system
 embroidery.h, 434
 fill.c, 497
 line
 BaseObject, 77
 EmbGeometry_, 133
 line.c
 embLine_intersectionPoint, 553
 embLine_normalVector, 553
 embLine_toVector, 553
 lineEditBlockX
 property-editor.cpp, 390
 lineEditBlockY
 property-editor.cpp, 390
 lineEditEllipseCenterX
 property-editor.cpp, 390
 lineEditEllipseCenterY
 property-editor.cpp, 390
 lineEditEllipseDiameterMajor
 property-editor.cpp, 390
 lineEditEllipseDiameterMinor
 property-editor.cpp, 390
 lineEditEllipseRadiusMajor
 property-editor.cpp, 390
 lineEditEllipseRadiusMinor
 property-editor.cpp, 390
 lineEditImageHeight
 property-editor.cpp, 390
 lineEditImageName
 property-editor.cpp, 390
 lineEditImagePath
 property-editor.cpp, 390
 lineEditImageWidth
 property-editor.cpp, 390
 lineEditImageX
 property-editor.cpp, 390
 lineEditImageY
 property-editor.cpp, 390
 lineEditInfiniteLineVectorX
 property-editor.cpp, 390
 lineEditInfiniteLineVectorY
 property-editor.cpp, 391
 lineEditInfiniteLineX2
 property-editor.cpp, 391
 lineEditInfiniteLineY1
 property-editor.cpp, 391
 lineEditInfiniteLineY2
 property-editor.cpp, 391
 lineEditLineAngle
 property-editor.cpp, 391
 lineEditLineDeltaX
 property-editor.cpp, 391
 lineEditLineDeltaY
 property-editor.cpp, 391
 lineEditLineEndX
 property-editor.cpp, 391
 lineEditLineEndY
 property-editor.cpp, 391
 lineEditLineLength
 property-editor.cpp, 391
 lineEditLineStartX
 property-editor.cpp, 391
 lineEditLineStartY
 property-editor.cpp, 391
 lineEditPathArea
 property-editor.cpp, 391
 lineEditPathLength
 property-editor.cpp, 391
 lineEditPathVertexX
 property-editor.cpp, 391

lineEditPathVertexY
 property-editor.cpp, 391
lineEditPointX
 property-editor.cpp, 391
lineEditPointY
 property-editor.cpp, 391
lineEditPolygonCenterX
 property-editor.cpp, 392
lineEditPolygonCenterY
 property-editor.cpp, 392
lineEditPolygonDiameterSide
 property-editor.cpp, 392
lineEditPolygonDiameterVertex
 property-editor.cpp, 392
lineEditPolygonInteriorAngle
 property-editor.cpp, 392
lineEditPolygonRadiusSide
 property-editor.cpp, 392
lineEditPolygonRadiusVertex
 property-editor.cpp, 392
lineEditPolylineArea
 property-editor.cpp, 392
lineEditPolylineLength
 property-editor.cpp, 392
lineEditPolylineVertexX
 property-editor.cpp, 392
lineEditPolylineVertexY
 property-editor.cpp, 392
lineEditRayVectorX
 property-editor.cpp, 392
lineEditRayVectorY
 property-editor.cpp, 392
lineEditRayX1
 property-editor.cpp, 392
lineEditRayX2
 property-editor.cpp, 392
lineEditRayY1
 property-editor.cpp, 392
lineEditRayY2
 property-editor.cpp, 392
lineEditRectangleArea
 property-editor.cpp, 392
lineEditRectangleCorner1X
 property-editor.cpp, 393
lineEditRectangleCorner1Y
 property-editor.cpp, 393
lineEditRectangleCorner2X
 property-editor.cpp, 393
lineEditRectangleCorner2Y
 property-editor.cpp, 393
lineEditRectangleCorner3X
 property-editor.cpp, 393
lineEditRectangleCorner3Y
 property-editor.cpp, 393
lineEditRectangleCorner4X
 property-editor.cpp, 393
lineEditRectangleCorner4Y
 property-editor.cpp, 393
lineEditRectangleHeight
 property-editor.cpp, 393
lineEditRectangleWidth
 property-editor.cpp, 393
lineEdits
 data.cpp, 333
 embroidermodder.h, 345
lineEditTextMultiX
 property-editor.cpp, 393
lineEditTextMultiY
 property-editor.cpp, 393
lineEditTextSingleContents
 property-editor.cpp, 393
lineEditTextSingleHeight
 property-editor.cpp, 393
lineEditTextSingleRotation
 property-editor.cpp, 393
lineEditTextSingleY
 property-editor.cpp, 393
LineObject, 164
 ~LineObject, 166
 allGripPoints, 167
 gripEdit, 167
 init, 167
 LineObject, 166
 mouseSnapPoint, 167
 objectAngle, 167
 objectDeltaX, 167
 objectDeltaY, 167
 objectEndPoint1, 167
 objectEndPoint2, 168
 objectLength, 168
 objectMidPoint, 168
 objectSavePath, 168
 objectX1, 168
 objectX2, 168
 objectY1, 168
 objectY2, 168
 paint, 168
 setObjectEndPoint1, 168
 setObjectEndPoint2, 169
 setObjectX1, 169
 setObjectX2, 169
 setObjectY1, 169
 setObjectY2, 169
 Type, 166
 type, 169
 updateRubber, 169
 vulcanize, 169
lineStyle
 DimLeaderObject, 112
lineStyleAngle
 DimLeaderObject, 116
lineStyleLength
 DimLeaderObject, 116
lineStylePath
 DimLeaderObject, 117
LINETO

embroidery_internal.h, 457
 lineType
 EmbGeometry_, 133
 EmbLine_, 137
 EmbPath_, 139
 EmbPoint_, 141
 linetypeSelector
 MainWindow, 206
 linetypeSelectorIndexChanged
 MainWindow, 185
 lineWeightPen
 BaseObject, 77
 linewidthSelector
 MainWindow, 206
 linewidthSelectorIndexChanged
 MainWindow, 185
 listMdiWin
 MainWindow, 206
 listTipOfTheDay
 MainWindow, 206
 load
 ImageWidget, 161
 LoadCommand
 MainWindow, 185
 loadFatFromSector
 embroidery_internal.h, 468
 main.c, 568
 loadFile
 MdiWindow, 218
 loadFormats
 MainWindow, 185
 loadRulerSettings
 View, 323
 logPromptInput
 MainWindow, 185
 MdiWindow, 218
 LSYSTEM, 170
 alphabet, 170
 axiom, 170
 constants, 170
 rules, 170
 lwt_default_lwt
 Settings_, 277
 lwt_mode
 EmbView_, 150
 lwt_real_render
 Settings_, 277
 lwt_show_lwt
 Settings_, 277
 lwtPen
 BaseObject, 81
 Madeira_Polyester
 embroidery.h, 416
 Madeira_Rayon
 embroidery.h, 416
 magicCode
 VipHeader_, 331
 main
 embroidermodder.cpp, 334
 main.c
 bcf_difat_create, 564
 bcf_directory_free, 564
 bcf_file_free, 564
 bcfFile_read, 564
 bcfFileFat_create, 564
 bcfFileHeader_read, 564
 binaryReadString, 564
 binaryReadUnicodeString, 565
 black_thread, 570
 check_header_present, 565
 CompoundFileDirectory, 565
 CompoundFileDirectoryEntry, 565
 copy_trim, 565
 difatEntriesInHeader, 570
 emb_error, 570
 emb_optOut, 565
 emb_readline, 566
 emb_verbose, 570
 embArc_print, 566
 embColor_distance, 566
 embColor_read, 566
 embColor_write, 566
 embConstantPi, 570
 embSatinOutline_generateSatinOutline, 566
 embSatinOutline_renderStitches, 566
 embThread_findNearestColor, 567
 embThread_findNearestThread, 567
 embThread_getRandom, 567
 embTime_initNow, 567
 embTime_time, 567
 embVector_print, 567
 entriesInDifatSector, 568
 FLAG_CIRCLE, 561
 FLAG_CIRCLE_SHORT, 562
 FLAG_COMBINE, 562
 FLAG_CROSS_STITCH, 562
 FLAG_ELLIPSE, 562
 FLAG_ELLIPSE_SHORT, 562
 FLAG_FILL, 562
 FLAG_FILL_SHORT, 562
 FLAG_FORMATS, 562
 FLAG_FORMATS_SHORT, 562
 FLAG_FULL_TEST_SUITE, 562
 FLAG_HELP, 562
 FLAG_HELP_SHORT, 562
 FLAG_HILBERT_CURVE, 562
 FLAG_LINE, 562
 FLAG_LINE_SHORT, 562
 FLAG_POLYGON, 562
 FLAG_POLYGON_SHORT, 562
 FLAG_POLYLINE, 562
 FLAG_POLYLINE_SHORT, 563
 FLAG QUIET, 563
 FLAG QUIET_SHORT, 563
 FLAG_RENDER, 563
 FLAG_RENDER_SHORT, 563

FLAG_SATIN, 563
FLAG_SATIN_SHORT, 563
FLAG_SIERPINSKI_TRIANGLE, 563
FLAG_SIMULATE, 563
FLAG_STITCH, 563
FLAG_STITCH_SHORT, 563
FLAG_TEST, 563
FLAG_TO, 563
FLAG_TO_SHORT, 563
FLAG_VERBOSE, 563
FLAG_VERBOSE_SHORT, 563
FLAG_VERSION, 563
FLAG_VERSION_SHORT, 563
get_trim_bounds, 568
GetFile, 568
haveExtraDIFATSectors, 568
loadFatFromSector, 568
NUM_FLAGS, 564
parseDIFATSectors, 568
parseDirectoryEntryName, 568
parseTime, 568
readFullSector, 569
readNextSector, 569
sectorSize, 569
seekToSector, 569
sizeOfChainingEntryAtEndOfDifatSector, 570
sizeOfDifatEntry, 570
sizeOfDirectoryEntry, 570
sizeOfFatEntry, 570
stringInArray, 569
WHITESPACE, 570
write_24bit, 569
mainWidget
 EmbDetailsDialog, 129
mainWin
 embroidermodder.h, 344
 MainWindow, 206
 mainwindow.cpp, 384
 MdiArea, 213
 MdiWindow, 223
 Settings_Dialog, 292
 StatusBarButton, 295
 View, 328
MainWindow, 170
 ~MainWindow, 177
 about, 178
 actionHash, 204
 activeCommand, 178
 activeMdiWindow, 178
 activeScene, 178
 activeUndoStack, 178
 activeView, 178
 actuator, 178
 buttonTipOfTheDayClicked, 179
 changelog, 179
 checkBoxTipOfTheDay, 204
 checkBoxTipOfTheDayStateChanged, 179
 checkForUpdates, 179
closeEvent, 179
closeToolBar, 179
colorSelector, 204
colorSelectorIndexChanged, 179
copy, 180
create_icon, 180
create_toolbar, 180
createAllActions, 180
createAllMenus, 180
createAllToolbars, 180
createEditMenu, 180
createFileMenu, 180
createHelpMenu, 181
createHelpToolbar, 181
createIconToolbar, 181
createLayerToolbar, 181
createPanToolbar, 181
createPromptToolbar, 181
createPropertiesToolbar, 181
createSettingsMenu, 181
createTextToolbar, 181
createViewMenu, 181
createWindowMenu, 182
cut, 182
cutCopyObjectList, 205
dayVision, 182
deletePressed, 182
designDetails, 182
disableMoveRapidFire, 182
disablePromptRapidFire, 182
docIndex, 205
dockPropEdit, 205
dockUndoEdit, 205
doNothing, 182
editMenu, 205
enableMoveRapidFire, 182
enablePromptRapidFire, 182
escapePressed, 182
exit, 182
fileMenu, 205
findMdiWindow, 182
floatingChangedToolBar, 183
formatFilterOpen, 205
formatFilterSave, 205
getAction, 183
getApplication, 183
getCurrentColor, 183
getCurrentLayer, 183
getCurrentLineType, 184
getCurrentLineWidth, 184
getFileSeparator, 184
getMdiArea, 184
help, 184
helpMenu, 205
hideUnimplemented, 184
iconResize, 184
isCommandActive, 184
isShiftPressed, 185

labelTipOfTheDay, 205
layerManager, 185
layerPrevious, 185
layerSelector, 205
layerSelectorIndexChanged, 185
layoutState, 206
linetypeSelector, 206
linetypeSelectorIndexChanged, 185
lineweightSelector, 206
lineweightSelectorIndexChanged, 185
listMdiWin, 206
listTipOfTheDay, 206
LoadCommand, 185
loadFormats, 185
logPromptInput, 185
mainWin, 206
MainWindow, 177
makeLayerActive, 185
mdiArea, 206
menuHash, 206
myFileSeparator, 206
nativeAddArc, 186
nativeAddCircle, 186
nativeAddDimLeader, 186
nativeAddEllipse, 186
nativeAddHorizontalDimension, 186
nativeAddImage, 186
nativeAddInfiniteLine, 187
nativeAddLine, 187
nativeAddPath, 187
nativeAddPoint, 187
nativeAddPolygon, 187
nativeAddPolyline, 187
nativeAddRay, 188
nativeAddRectangle, 188
nativeAddRegularPolygon, 188
nativeAddRoundedRectangle, 188
nativeAddSlot, 188
nativeAddTextMulti, 189
nativeAddTextSingle, 189
nativeAddToSelection, 189
nativeAddTriangle, 189
nativeAddVerticalDimension, 189
nativeAlert, 190
nativeAllowRubber, 190
nativeAppendPromptHistory, 190
nativeBlinkPrompt, 190
nativeCalculateAngle, 190
nativeCalculateDistance, 190
nativeClearRubber, 190
nativeClearSelection, 190
nativeCopySelected, 190
nativeCutSelected, 191
nativeDeleteSelected, 191
nativeDisableMoveRapidFire, 191
nativeDisablePromptRapidFire, 191
nativeEnableMoveRapidFire, 191
nativeEnablePromptRapidFire, 191
nativeEndCommand, 191
nativeExit, 191
nativeInitCommand, 191
nativeMessageBox, 192
nativeMirrorSelected, 192
nativeMouseX, 192
nativeMouseY, 192
nativeMoveSelected, 192
nativeNumSelected, 192
nativePasteSelected, 193
nativePerpendicularDistance, 193
nativePreviewOff, 193
nativePreviewOn, 193
nativePrintArea, 193
nativeQSnapX, 193
nativeQSnapY, 193
nativeRotateSelected, 194
nativeScaleSelected, 194
nativeSetBackgroundColor, 194
nativeSetCrossHairColor, 194
nativeSetCursorShape, 194
nativeSetGridColor, 194
nativeSetPromptPrefix, 194
nativeSetRubberMode, 195
nativeSetRubberPoint, 195
nativeSetRubberText, 195
nativeSpareRubber, 195
nativeTipOfTheDay, 195
nativeVulcanize, 195
newFile, 195
nightVision, 195
numOfDocs, 206
onCloseMdiWin, 195
onCloseWindow, 196
onWindowActivated, 196
openFile, 196
openFilePath, 206
openFilesSelected, 196
openrecentfile, 196
panDown, 197
panLeft, 197
panMenu, 207
panpoint, 197
panrealtime, 197
panRight, 197
panUp, 197
paste, 197
pickAddModeToggled, 197
platformString, 197
print, 197
prompt, 207
promptHistoryAppended, 197
promptInputNext, 198
promptInputPrevious, 198
quit, 198
readSettings, 198
recentMenu, 207
recentMenuAboutToShow, 198

redo, 198
resizeEvent, 198
run_script, 198
run_script_file, 199
saveasfile, 199
savefile, 199
setShiftPressed, 199
setShiftReleased, 199
setTextAngle, 200
setTextBold, 200
setTextFont, 200
setTextItalic, 200
setTextOverline, 200
setTextSize, 200
setTextStrikeOut, 200
setTextUnderline, 200
settingsDialog, 200
settingsMenu, 207
settingsPrompt, 200
setUndoCleanIcon, 201
shiftKeyPressedState, 207
statusbar, 207
stub_implement, 201
stub_testing, 201
textFontSelector, 207
textFontSelectorCurrentFontChanged, 201
textSizeSelector, 207
textSizeSelectorIndexChanged, 201
tipOfDay, 201
toggleGrid, 201
toggleLwt, 201
toggleRuler, 201
toolbarEdit, 207
toolbarFile, 207
toolbarHash, 207
toolbarHelp, 208
toolbarIcon, 208
toolbarLayer, 208
toolbarPan, 208
toolbarPrompt, 208
toolbarProperties, 208
toolbarText, 208
toolbarView, 208
toolbarZoom, 208
undo, 201
updateAllViewBackgroundColors, 202
updateAllViewCrossHairColors, 202
updateAllViewGridColors, 202
updateAllViewRulerColors, 202
updateAllViewScrollBars, 202
updateAllViewSelectBoxColors, 202
updateMenuToolbarStatusbar, 202
updatePickAddMode, 202
validFileFormat, 202
viewMenu, 208
whatsThisContextHelp, 203
windowMenu, 208
windowMenuAboutToShow, 203
windowMenuActivated, 203
wizardTipOfDay, 209
writeSettings, 203
zoomAll, 203
zoomCenter, 203
zoomDynamic, 203
zoomExtents, 204
zoomIn, 204
zoomMenu, 209
zoomOut, 204
zoomPrevious, 204
zoomRealtime, 204
zoomScale, 204
zoomSelected, 204
zoomWindow, 204
mainwindow-settings.cpp
 read_configuration, 380
 SettingsDir, 380
 SettingsPath, 380
 to_string_vector, 380
 write_setting, 380, 381
mainwindow-toolbars.cpp
 edit_toolbar, 381
 file_toolbar, 381
 get_action_index, 381
 help_toolbar, 382
 icon_toolbar, 382
 pan_toolbar, 382
 view_toolbar, 382
 zoom_toolbar, 382
mainwindow.cpp
 _mainWin, 384
 action_table, 384
 config, 385
 convert_args_to_type, 383
 Error, 384
 mainWin, 384
 Parameter, 383
 read_settings, 384
 read_string_setting, 384
 Todo, 384
 tokenize, 384
 validRGB, 384
major_tick_seperation
 Settings_, 277
majorVersion
 _bcf_file_header, 55
makeLayerActive
 MainWindow, 185
manufacturer_code
 thread_color_, 303
map_signal
 GroupBoxData_, 153
mapSignal
 PropertyEditor, 254
Marathon_Polyester
 embroidery.h, 416
Marathon_Rayon

embroidery.h, 416
 max, 522
 max_header
 format_max.c, 522
 MAX_STITCHES
 embroidery.h, 417
 MAX_THREADS
 embroidery.h, 417
 maxNumberOfDirectoryEntries
 _bcf_directory, 50
 maxPoints
 UiObject_, 306
 MdiArea, 209
 ~MdiArea, 210
 bgColor, 213
 bgLogo, 213
 bgTexture, 213
 cascade, 210
 forceRepaint, 210
 mainWin, 213
 MdiArea, 210
 mouseDoubleClickEvent, 211
 paintEvent, 211
 setBackgroundColor, 211
 setBackgroundLogo, 211
 setBackgroundTexture, 211
 tile, 212
 useBackgroundColor, 212
 useBackgroundLogo, 212
 useBackgroundTexture, 212
 useColor, 213
 useLogo, 213
 useTexture, 213
 zoomExtentsAllSubWindows, 212
 mdiArea
 MainWindow, 206
 MdiWindow, 223
 MdiWindow, 213
 ~MdiWindow, 215
 closeEvent, 216
 curColor, 222
 curFile, 222
 curLayer, 222
 curLineType, 222
 curLineWeight, 222
 currentColorChanged, 216
 currentLayerChanged, 216
 currentLinetypeChanged, 216
 currentLineweightChanged, 216
 deletePressed, 217
 designDetails, 217
 escapePressed, 217
 fileExtension, 217
 fileWasLoaded, 222
 getCurrentColor, 217
 getCurrentFile, 217
 getCurrentLayer, 217
 getCurrentLineType, 218
 getCurrentLineWidth, 218
 getScene, 218
 getShortCurrentFile, 218
 getView, 218
 gscene, 222
 gview, 222
 loadFile, 218
 logPromptInput, 218
 mainWin, 223
 mdiArea, 223
 MdiWindow, 215
 myIndex, 223
 onWindowActivated, 219
 print, 219
 printer, 223
 promptHistory, 223
 promptHistoryAppended, 219
 promptInputList, 223
 promptInputNext, 219
 promptInputNum, 223
 promptInputPrevious, 219
 promptInputPrevNext, 219
 saveBMC, 219
 saveFile, 220
 sendCloseMdiWin, 220
 setCurrentColor, 220
 setCurrentFile, 220
 setCurrentLayer, 220
 setCurrentLineType, 221
 setCurrentLineWidth, 221
 setViewBackgroundColor, 221
 setViewCrossHairColor, 221
 setViewGridColor, 221
 setViewRulerColor, 221
 setViewSelectBoxColors, 221
 showViewScrollBars, 221
 sizeHint, 221
 updateColorLinetypeLinewidth, 222
 Mega 2560 or another board with equal or, 23
 Melco, 505, 509, 514, 524
 menu_name
 Action_, 60
 menu_position
 Action_, 60
 menuHash
 MainWindow, 206
 mergeWith
 UndoableNavCommand, 312
 metric
 EmbView_, 150
 Metro_Polyester
 embroidery.h, 417
 mid
 EmbArc_, 124
 miniSectorShift
 _bcf_file_header, 56
 miniStreamCutoffSize
 _bcf_file_header, 56

minorVersion
 _bcf_file_header, 56

minPoints
 UiObject_, 306

minute
 EmbTime_, 147

mirror
 UndoableMirrorCommand, 310

mirrorLine
 UndoableMirrorCommand, 310

mirrorSelected
 View, 323

mit, 472, 523

mitDecodeStitch
 embroidery_internal.h, 468
 encoding.c, 492

mitEncodeStitch
 embroidery_internal.h, 468
 encoding.c, 492

Mitsubishi, 472, 523

mode
 UiObject_, 306

modifiedTime
 _bcf_directory_entry, 52

modifierName
 ThredExtension_, 304

month
 EmbTime_, 147

mouseDoubleClickEvent
 MdiArea, 211
 View, 323

mouseMoveEvent
 CmdPromptHandle, 97
 View, 323

mousePressEvent
 CmdPromptHandle, 97
 View, 323

mouseReleaseEvent
 CmdPromptHandle, 97
 View, 323

mouseSnapPoint
 ArcObject, 68
 BaseObject, 77
 CircleObject, 85
 DimLeaderObject, 113
 EllipseObject, 120
 ImageObject, 158
 LineObject, 167
 PathObject, 227
 PointObject, 232
 PolygonObject, 237
 PolylineObject, 242
 RectObject, 261
 TextSingleObject, 300

moveAction
 View, 323

movePoint
 View, 328

moveResizeHistory
 CmdPromptSplitter, 108

moveSelected
 View, 323

MOVETO
 embroidery_internal.h, 457

moveY
 CmdPromptHandle, 97

movingActive
 View, 328

myFileSeparator
 MainWindow, 206

myIndex
 MdiWindow, 223

n_attributes
 format_svg.c, 536

N_PES_VERSIONS
 embroidery_internal.h, 457

n_selected
 EmbView_, 150

name
 EmblImage_, 135
 EmbLayer_, 136
 SvgAttribute_, 297
 thread_color_, 303

nativeAddArc
 MainWindow, 186

nativeAddCircle
 MainWindow, 186

nativeAddDimLeader
 MainWindow, 186

nativeAddEllipse
 MainWindow, 186

nativeAddHorizontalDimension
 MainWindow, 186

nativeAddImage
 MainWindow, 186

nativeAddInfiniteLine
 MainWindow, 187

nativeAddLine
 MainWindow, 187

nativeAddPath
 MainWindow, 187

nativeAddPoint
 MainWindow, 187

nativeAddPolygon
 MainWindow, 187

nativeAddPolyline
 MainWindow, 187

nativeAddRay
 MainWindow, 188

nativeAddRectangle
 MainWindow, 188

nativeAddRegularPolygon
 MainWindow, 188

nativeAddRoundedRectangle
 MainWindow, 188

nativeAddSlot

MainWindow, 188
nativeAddTextMulti
 MainWindow, 189
nativeAddTextSingle
 MainWindow, 189
nativeAddToSelection
 MainWindow, 189
nativeAddTriangle
 MainWindow, 189
nativeAddVerticalDimension
 MainWindow, 189
nativeAlert
 MainWindow, 190
nativeAllowRubber
 MainWindow, 190
nativeAppendPromptHistory
 MainWindow, 190
nativeBlinkPrompt
 MainWindow, 190
nativeCalculateAngle
 MainWindow, 190
nativeCalculateDistance
 MainWindow, 190
nativeClearRubber
 MainWindow, 190
nativeClearSelection
 MainWindow, 190
nativeCopySelected
 MainWindow, 190
nativeCutSelected
 MainWindow, 191
nativeDeleteSelected
 MainWindow, 191
nativeDisableMoveRapidFire
 MainWindow, 191
nativeDisablePromptRapidFire
 MainWindow, 191
nativeEnableMoveRapidFire
 MainWindow, 191
nativeEnablePromptRapidFire
 MainWindow, 191
nativeEndCommand
 MainWindow, 191
nativeExit
 MainWindow, 191
nativeInitCommand
 MainWindow, 191
nativeMessageBox
 MainWindow, 192
nativeMirrorSelected
 MainWindow, 192
nativeMouseX
 MainWindow, 192
nativeMouseY
 MainWindow, 192
nativeMoveSelected
 MainWindow, 192
nativeNumSelected
 MainWindow, 192
nativePasteSelected
 MainWindow, 193
nativePerpendicularDistance
 MainWindow, 193
nativePreviewOff
 MainWindow, 193
nativePreviewOn
 MainWindow, 193
nativePrintArea
 MainWindow, 193
nativeQSnapX
 MainWindow, 193
nativeQSnapY
 MainWindow, 193
nativeRotateSelected
 MainWindow, 194
nativeScaleSelected
 MainWindow, 194
nativeSetBackgroundColor
 MainWindow, 194
nativeSetCrossHairColor
 MainWindow, 194
nativeSetCursorShape
 MainWindow, 194
nativeSetGridColor
 MainWindow, 194
nativeSetPromptPrefix
 MainWindow, 194
nativeSetRubberMode
 MainWindow, 195
nativeSetRubberPoint
 MainWindow, 195
nativeSetRubberText
 MainWindow, 195
nativeSpareRubber
 MainWindow, 195
nativeTipOfDay
 MainWindow, 195
nativeVulcanize
 MainWindow, 195
navType
 UndoableNavCommand, 313
needle_speed
 Settings_, 277
negativeXHoopSize
 VipHeader_, 331
negativeYHoopSize
 VipHeader_, 331
new, 472, 523
newFile
 MainWindow, 195
next
 _bcf_directory_entry, 52
nightVision
 MainWindow, 195
nlenghts
 Huffman, 154

NoArrow
 DimLeaderObject, 112

NoLine
 DimLeaderObject, 112

NORMAL
 embroidery.h, 417

normalPath
 PathObject, 229

NUM_FLAGS
 main.c, 564

numberOfBytesRemaining
 _vp3Hoop, 58

numberOfColors
 _vp3Hoop, 58

 VipHeader_, 331

numberOfDifatSectors
 _bcf_file_header, 56

numberOfDirectorySectors
 _bcf_file_header, 56

numberOfEntriesInDifatSector
 embroidery_internal.h, 468

numberOfEntriesInFatSector
 _bcf_file_fat, 54

numberOfFATSectors
 _bcf_file_header, 56

numberOfFormats
 embroidery.h, 417

numberOfMiniFatSectors
 _bcf_file_header, 56

numberOfStitches
 VipHeader_, 331

numOfDocs
 MainWindow, 206

numPoints
 UiObject_, 306

numSelected
 View, 323

numStiches
 ThredHeader_, 305

OBJ_COLOR
 embroidermodder.h, 339

OBJ_KEYS
 embroidermodder.h, 339

OBJ_LAYER
 embroidermodder.h, 339

OBJ_LTYPE
 embroidermodder.h, 339

OBJ_LTYPE_CENTER
 embroidermodder.h, 340

OBJ_LTYPE_CONT
 embroidermodder.h, 340

OBJ_LTYPE_DOT
 embroidermodder.h, 340

OBJ_LTYPE_FISHBONE
 embroidermodder.h, 340

OBJ_LTYPE_HIDDEN
 embroidermodder.h, 340

OBJ_LTYPE_PHANTOM
 embroidermodder.h, 340

OBJ_LTYPE_RUNNING
 embroidermodder.h, 340

OBJ_LTYPE_SATIN
 embroidermodder.h, 340

OBJ_LTYPE_VALUES
 embroidermodder.h, 339

OBJ_LTYPE_ZIGZAG
 embroidermodder.h, 340

OBJ_LWT
 embroidermodder.h, 339

OBJ_LWT_01
 embroidermodder.h, 340

OBJ_LWT_02
 embroidermodder.h, 340

OBJ_LWT_03
 embroidermodder.h, 340

OBJ_LWT_04
 embroidermodder.h, 340

OBJ_LWT_05
 embroidermodder.h, 340

OBJ_LWT_06
 embroidermodder.h, 340

OBJ_LWT_07
 embroidermodder.h, 340

OBJ_LWT_08
 embroidermodder.h, 340

OBJ_LWT_09
 embroidermodder.h, 340

OBJ_LWT_10
 embroidermodder.h, 340

OBJ_LWT_11
 embroidermodder.h, 340

OBJ_LWT_12
 embroidermodder.h, 340

OBJ_LWT_13
 embroidermodder.h, 340

OBJ_LWT_14
 embroidermodder.h, 340

OBJ_LWT_15
 embroidermodder.h, 340

OBJ_LWT_16
 embroidermodder.h, 340

OBJ_LWT_17
 embroidermodder.h, 340

OBJ_LWT_18
 embroidermodder.h, 340

OBJ_LWT_19
 embroidermodder.h, 340

OBJ_LWT_20
 embroidermodder.h, 340

OBJ_LWT_21
 embroidermodder.h, 340

OBJ_LWT_22
 embroidermodder.h, 340

embroidermodder.h, 340
OBJ_LWT_23
 embroidermodder.h, 340
OBJ_LWT_24
 embroidermodder.h, 340
OBJ_LWT_BYBLOCK
 embroidermodder.h, 340
OBJ_LWT_BYLAYER
 embroidermodder.h, 340
OBJ_LWT_DEFAULT
 embroidermodder.h, 340
OBJ_LWT_VALUES
 embroidermodder.h, 340
OBJ_NAME
 embroidermodder.h, 339
OBJ_RUBBER
 embroidermodder.h, 339
OBJ_RUBBER_CIRCLE_1P_DIA
 embroidermodder.h, 341
OBJ_RUBBER_CIRCLE_1P_RAD
 embroidermodder.h, 341
OBJ_RUBBER_CIRCLE_2P
 embroidermodder.h, 341
OBJ_RUBBER_CIRCLE_3P
 embroidermodder.h, 341
OBJ_RUBBER_CIRCLE_TTR
 embroidermodder.h, 341
OBJ_RUBBER_CIRCLE_TTT
 embroidermodder.h, 341
OBJ_RUBBER_DIMLEADER_LINE
 embroidermodder.h, 341
OBJ_RUBBER_ELLIPSE_LINE
 embroidermodder.h, 341
OBJ_RUBBER_ELLIPSE_MAJOR_DIAMETER_MINOR_RADIUS_SNAP_VALUES
 embroidermodder.h, 341
OBJ_RUBBER_ELLIPSE_MAJOR_RADIUS_MINOR_RADIUS_OBJ_TYPE
 embroidermodder.h, 341
OBJ_RUBBER_ELLIPSE_ROTATION
 embroidermodder.h, 341
OBJ_RUBBER_GRIP
 embroidermodder.h, 341
OBJ_RUBBER_IMAGE
 embroidermodder.h, 341
OBJ_RUBBER_LINE
 embroidermodder.h, 341
OBJ_RUBBER_OFF
 embroidermodder.h, 341
OBJ_RUBBER_ON
 embroidermodder.h, 341
OBJ_RUBBER_POLYGON
 embroidermodder.h, 341
OBJ_RUBBER_POLYGON_CIRCUMSCRIBE
 embroidermodder.h, 341
OBJ_RUBBER_POLYGON_INSCRIBE
 embroidermodder.h, 341
OBJ_RUBBER_POLYLINE
 embroidermodder.h, 341
OBJ_RUBBER_RECTANGLE
 embroidermodder.h, 340
embroidermodder.h, 341
OBJ_RUBBER_TEXTSINGLE
 embroidermodder.h, 341
OBJ_RUBBER_VALUES
 embroidermodder.h, 340
OBJ_SNAP_APPINTERSECTION
 embroidermodder.h, 341
OBJ_SNAP_CENTER
 embroidermodder.h, 341
OBJ_SNAP_ENDPOINT
 embroidermodder.h, 341
OBJ_SNAP_EXTENSION
 embroidermodder.h, 341
OBJ_SNAP_INSERTION
 embroidermodder.h, 341
OBJ_SNAP_INTERSECTION
 embroidermodder.h, 341
OBJ_SNAP_MIDPOINT
 embroidermodder.h, 341
OBJ_SNAP_NEAREST
 embroidermodder.h, 341
OBJ_SNAP_NODE
 embroidermodder.h, 341
OBJ_SNAP_NULL
 embroidermodder.h, 341
OBJ_SNAP_PARALLEL
 embroidermodder.h, 341
OBJ_SNAP_PERPENDICULAR
 embroidermodder.h, 341
OBJ_SNAP_QUADRANT
 embroidermodder.h, 341
OBJ_SNAP_TANGENT
 embroidermodder.h, 341
OBJ_TYPE_SNAP_VALUES
 embroidermodder.h, 341
OBJ_TYPE_OBJ_TYPE
 embroidermodder.h, 339
OBJ_TYPE_ARC
 embroidermodder.h, 342
OBJ_TYPE_BASE
 embroidermodder.h, 342
OBJ_TYPE_BLOCK
 embroidermodder.h, 342
OBJ_TYPE_CIRCLE
 embroidermodder.h, 342
OBJ_TYPE_DIMALIGNED
 embroidermodder.h, 342
OBJ_TYPE_DIMANGULAR
 embroidermodder.h, 342
OBJ_TYPE_DIMARCLENGTH
 embroidermodder.h, 342
OBJ_TYPE_DIMDIAMETER
 embroidermodder.h, 342
OBJ_TYPE_DIMLEADER
 embroidermodder.h, 342
OBJ_TYPE_DIMLINEAR
 embroidermodder.h, 342
OBJ_TYPE_DIMORDINATE
 embroidermodder.h, 342

embroidermodder.h, 342
OBJ_TYPE_DIMRADIUS
 embroidermodder.h, 342
OBJ_TYPE_ELLIPSE
 embroidermodder.h, 342
OBJ_TYPE_ELLIPSEARC
 embroidermodder.h, 342
OBJ_TYPE_GRID
 embroidermodder.h, 342
OBJ_TYPE_HATCH
 embroidermodder.h, 342
OBJ_TYPE_IMAGE
 embroidermodder.h, 342
OBJ_TYPE_INFINITELINE
 embroidermodder.h, 342
OBJ_TYPE_LINE
 embroidermodder.h, 342
OBJ_TYPE_NULL
 embroidermodder.h, 342
OBJ_TYPE_PATH
 embroidermodder.h, 342
OBJ_TYPE_POINT
 embroidermodder.h, 342
OBJ_TYPE_POLYGON
 embroidermodder.h, 342
OBJ_TYPE_POLYLINE
 embroidermodder.h, 342
OBJ_TYPE_RAY
 embroidermodder.h, 342
OBJ_TYPE_RECTANGLE
 embroidermodder.h, 342
OBJ_TYPE_RUBBER
 embroidermodder.h, 342
OBJ_TYPE_SLOT
 embroidermodder.h, 342
OBJ_TYPE_SPLINE
 embroidermodder.h, 342
OBJ_TYPE_TEXTMULTI
 embroidermodder.h, 342
OBJ_TYPE_TEXTSINGLE
 embroidermodder.h, 342
OBJ_TYPE_VALUES
 embroidermodder.h, 342
object
 EmbGeometry_, 133
 GroupBoxData_, 153
 UndoableAddCommand, 308
 UndoableDeleteCommand, 308
 UndoableGripEditCommand, 309
 UndoableMirrorCommand, 310
 UndoableMoveCommand, 311
 UndoableRotateCommand, 314
 UndoableScaleCommand, 315
object-arc.cpp
 rotate_vector, 385
object_index
 UiObject_, 306
objectAngle
 DimLeaderObject, 113
 LineObject, 167
objectArcLength
 ArcObject, 69
objectArea
 ArcObject, 69
 CircleObject, 85
 ImageObject, 158
 RectObject, 261
objectBottomLeft
 ImageObject, 158
 RectObject, 261
objectBottomRight
 ImageObject, 158
 RectObject, 261
objectCenter
 BaseObject, 77
objectCenterX
 BaseObject, 77
objectCenterY
 BaseObject, 77
objectChord
 ArcObject, 69
objectCircumference
 CircleObject, 85
objectClockwise
 ArcObject, 69
objectColor
 BaseObject, 77
objectColorRGB
 BaseObject, 78
objectCopyPath
 PathObject, 227
 PolygonObject, 237
 PolylineObject, 242
objectDeltaX
 DimLeaderObject, 114
 LineObject, 167
objectDeltaY
 DimLeaderObject, 114
 LineObject, 167
objectDiameter
 CircleObject, 85
objectDiameterMajor
 EllipseObject, 120
objectDiameterMinor
 EllipseObject, 120
objectEndAngle
 ArcObject, 69
objectEndPoint
 ArcObject, 70
objectEndPoint1
 DimLeaderObject, 114
 LineObject, 167
objectEndPoint2
 DimLeaderObject, 114
 LineObject, 168
objectEndX

ArcObject, 70
objectEndY
 ArcObject, 70
objectHeight
 EllipseObject, 121
 ImageObject, 158
 RectObject, 261
objectID
 BaseObject, 78
objectIncludedAngle
 ArcObject, 70
objectLength
 DimLeaderObject, 114
 LineObject, 168
objectLineType
 BaseObject, 78
objectLineWidth
 BaseObject, 78
objectMidPoint
 ArcObject, 70
 DimLeaderObject, 114
 LineObject, 168
objectMidX
 ArcObject, 71
objectMidY
 ArcObject, 71
objectPath
 BaseObject, 78
objectPen
 BaseObject, 78
objectPos
 PathObject, 227
 PointObject, 232
 PolygonObject, 237
 PolylineObject, 242
 RectObject, 261
 TextSingleObject, 300
objectQuadrant0
 CircleObject, 85
 EllipseObject, 121
objectQuadrant180
 CircleObject, 85
 EllipseObject, 121
objectQuadrant270
 CircleObject, 85
 EllipseObject, 121
objectQuadrant90
 CircleObject, 85
 EllipseObject, 121
objectRadius
 ArcObject, 71
 CircleObject, 85
objectRadiusMajor
 EllipseObject, 121
objectRadiusMinor
 EllipseObject, 121
objectRubberMode
 BaseObject, 78
objectRubberPoint
 BaseObject, 78
objectRubberText
 BaseObject, 78
objectSavePath
 CircleObject, 85
 EllipseObject, 121
 LineObject, 168
 PathObject, 227
 PointObject, 232
 PolygonObject, 237
 PolylineObject, 242
 RectObject, 261
objectSavePathList
 TextSingleObject, 301
objectStartAngle
 ArcObject, 71
objectStartPoint
 ArcObject, 71
objectStartX
 ArcObject, 71
objectStartY
 ArcObject, 72
objectTextJustifyList
 TextSingleObject, 301
objectTopLeft
 ImageObject, 159
 RectObject, 262
objectTopRight
 ImageObject, 159
 RectObject, 262
objectType
 _bcf_directory_entry, 52
ObjectTypeRootEntry
 embroidery_internal.h, 457
ObjectTypeStorage
 embroidery_internal.h, 457
ObjectTypeStream
 embroidery_internal.h, 457
ObjectTypeUnknown
 embroidery_internal.h, 457
objectWidth
 EllipseObject, 121
 ImageObject, 159
 RectObject, 262
objectX
 PathObject, 227
 PointObject, 232
 PolygonObject, 237
 PolylineObject, 243
 TextSingleObject, 301
objectX1
 DimLeaderObject, 114
 LineObject, 168
objectX2
 DimLeaderObject, 114
 LineObject, 168
objectY

PathObject, 227
PointObject, 232
PolygonObject, 237
PolylineObject, 243
TextSingleObject, 301
objectY1
 DimLeaderObject, 114
 LineObject, 168
objectY2
 DimLeaderObject, 114
 LineObject, 168
objID
 BaseObject, 81
objLine
 BaseObject, 81
objPen
 BaseObject, 81
objRubberMode
 BaseObject, 81
objRubberPoints
 BaseObject, 81
objRubberTexts
 BaseObject, 81
objText
 TextSingleObject, 302
objTextBackward
 TextSingleObject, 302
objTextBold
 TextSingleObject, 303
objTextFont
 TextSingleObject, 303
objTextItalic
 TextSingleObject, 303
objTextJustify
 TextSingleObject, 303
objTextOverline
 TextSingleObject, 303
objTextPath
 TextSingleObject, 303
objTextSize
 TextSingleObject, 303
objTextStrikeOut
 TextSingleObject, 303
objTextUnderline
 TextSingleObject, 303
objTextUpsideDown
 TextSingleObject, 303
ofm, 524
ofmDecode
 format_ofm.c, 524
ofmReadBlockHeader
 format_ofm.c, 524
ofmReadClass
 format_ofm.c, 524
ofmReadColorChange
 format_ofm.c, 524
ofmReadExpanded
 format_ofm.c, 524
ofmReadLibrary
 format_ofm.c, 524
ofmReadThreads
 format_ofm.c, 524
onCloseMdiWin
 MainWindow, 195
onCloseWindow
 MainWindow, 196
onWindowActivated
 MainWindow, 196
 MdiWindow, 219
Open
 DimLeaderObject, 112
openFile
 MainWindow, 196
openFilesPath
 MainWindow, 206
openFilesSelected
 MainWindow, 196
openrecentfile
 MainWindow, 196
opensave_custom_filter
 Settings_, 277
opensave_open_format
 Settings_, 277
opensave_open_thumbnail
 Settings_, 277
opensave_recent_directory
 Settings_, 277
opensave_recent_list_of_files
 Settings_, 277
opensave_recent_max_files
 Settings_, 277
opensave_save_format
 Settings_, 277
opensave_save_thumbnail
 Settings_, 277
opensave_trim_dst_num_jumps
 Settings_, 278
operator+
 embroidermodder.h, 344
operator-
 embroidermodder.h, 344
origin
 EmbView_, 150
originPath
 View, 328
ortho_mode
 EmbView_, 150
paint
 ArcObject, 72
 CircleObject, 86
 DimLeaderObject, 114
 EllipseObject, 121
 ImageObject, 159
 LineObject, 168
 PathObject, 227
 PointObject, 232

PolygonObject, 237
 PolylineObject, 243
 RectObject, 262
 TextSingleObject, 301
paintEvent
 ImageWidget, 161
 MdiArea, 211
 SelectBox, 270
pan_toolbar
 mainwindow-toolbars.cpp, 382
panDistance
 View, 328
panDown
 MainWindow, 197
 View, 323
panLeft
 MainWindow, 197
 View, 323
panMenu
 MainWindow, 207
panningActive
 View, 328
panningPointActive
 View, 328
panningRealTimeActive
 View, 328
panPoint
 View, 323
panpoint
 MainWindow, 197
panRealTime
 View, 323
panrealtime
 MainWindow, 197
panRight
 MainWindow, 197
 View, 324
panStart
 View, 324
panStartX
 View, 328
panStartY
 View, 328
Pantone
 embroidery.h, 417
panUp
 MainWindow, 197
 View, 324
Parameter
 mainwindow.cpp, 383
Parameter_, 223
 i_value, 223
 r_value, 224
 s_value, 224
parseDIFATSectors
 main.c, 568
parseDirectoryEntryName
 main.c, 568
parseTime
 main.c, 568
paste
 MainWindow, 197
 View, 324
pasteClip
 CmdPromptInput, 104
pasteDelta
 View, 328
pasteObjectItemGroup
 View, 329
pastePressed
 CmdPrompt, 92
 CmdPromptInput, 104
pastingActive
 View, 329
path
 EmbGeometry_, 133
 EmblImage_, 135
path_desc
 UiObject_, 306
PathObject, 224
 ~PathObject, 226
 allGripPoints, 226
 gripEdit, 227
 init, 227
 mouseSnapPoint, 227
 normalPath, 229
 objectCopyPath, 227
 objectPos, 227
 objectSavePath, 227
 objectX, 227
 objectY, 227
 paint, 227
 PathObject, 226
 setObjectPos, 228
 setObjectX, 228
 setObjectY, 228
 Type, 226
 type, 228
 updatePath, 228
 updateRubber, 228
 vulcanize, 228
pattern
 EmbView_, 150
pattern.c
 convert, 571
 embPattern_addCircleAbs, 571
 embPattern_addEllipseAbs, 571
 embPattern_addLineAbs, 572
 embPattern_addPathAbs, 572
 embPattern_addPointAbs, 572
 embPattern_addPolygonAbs, 572
 embPattern_addPolylineObjectAbs, 572
 embPattern_addRectAbs, 572
 embPattern_addStitchAbs, 572
 embPattern_addStitchRel, 572
 embPattern_addThread, 572

embPattern_calcBoundingBox, 573
embPattern_center, 573
embPattern_changeColor, 573
embPattern_color_count, 573
embPattern_combineJumpStitches, 573
embPattern_copyPolylinesToStitch_list, 573
embPattern_copystitch_listToPolylines, 573
embPattern_correctForMaxStitchLength, 573
embPattern_create, 573
embPattern_designDetails, 573
embPattern_end, 574
embPattern_fixColorCount, 574
embPattern_flip, 574
embPattern_flipHorizontal, 574
embPattern_flipVertical, 574
embPattern_free, 574
embPattern_hideStitchesOverLength, 574
embPattern_jumpStitches, 574
embPattern_lengthHistogram, 574
embPattern_loadExternalColorFile, 574
embPattern_maximumStitchLength, 574
embPattern_minimumStitchLength, 574
embPattern_movePolylinesToStitch_list, 575
embPattern_movestitch_listToPolylines, 575
embPattern_realStitches, 575
embPattern_scale, 575
embPattern_totalStitchLength, 575
embPattern_trimStitches, 575
pattern_index
 Settings_, 278
 UiObject_, 306
pcd, 16, 473, 525
pcm, 16, 473, 525
pcm_thread
 embroidery.h, 417
pcmThreads
 embroidery.h, 436
 thread-color.c, 576
pcq, 16, 473, 526
pos, 16, 473, 526
pec, 16, 473, 527
pec_thread
 embroidery.h, 417
pecEncode
 format_pec.c, 527
pecEncodeJump
 format_pec.c, 527
pecEncodeStop
 format_pec.c, 527
pecThreadCount
 embroidery.h, 436
 thread-color.c, 576
pecThreads
 embroidery.h, 436
 thread-color.c, 577
pel, 16, 473, 528
pem, 16, 474, 528
pes, 16, 531
PES0001
 embroidery_internal.h, 457
PES0020
 embroidery_internal.h, 457
PES0022
 embroidery_internal.h, 457
PES0030
 embroidery_internal.h, 457
PES0040
 embroidery_internal.h, 458
PES0050
 embroidery_internal.h, 458
PES0055
 embroidery_internal.h, 458
PES0056
 embroidery_internal.h, 458
PES0060
 embroidery_internal.h, 458
PES0070
 embroidery_internal.h, 458
PES0080
 embroidery_internal.h, 458
PES0090
 embroidery_internal.h, 458
PES0100
 embroidery_internal.h, 458
pes_version
 format_pes.c, 531
pes_version_strings
 format_pes.c, 531
pesWriteEmbOneSection
 format_pes.c, 529
pesWriteSewSegSection
 format_pes.c, 529
Pfaff, 436, 473, 475, 521, 522, 525, 526, 537, 542, 543
pfaffDecode
 embroidery_internal.h, 468
 encoding.c, 492
pfaffEncode
 embroidery_internal.h, 468
 encoding.c, 493
phb, 16, 474, 531
phc, 16, 474, 532
pickAdd
 PropertyEditor, 256
pickAddModeToggled
 MainWindow, 197
 PropertyEditor, 254
pickBoxSize
 View, 329
pivotX
 UndoableRotateCommand, 314
pivotY
 UndoableRotateCommand, 314
platformString
 MainWindow, 197
plt, 475, 532
point

EmbGeometry_, 133
pointList
 EmbPath_, 139
PointObject, 229
 ~PointObject, 231
 allGripPoints, 231
 gripEdit, 232
 init, 232
 mouseSnapPoint, 232
 objectPos, 232
 objectSavePath, 232
 objectX, 232
 objectY, 232
 paint, 232
 PointObject, 231
 setObjectPos, 232, 233
 setObjectX, 233
 setObjectY, 233
Type, 231
type, 233
updateRubber, 233
vulcanize, 233
polar_mode
 EmbView_, 150
polygon
 EmbGeometry_, 133
PolygonObject, 233
 ~PolygonObject, 236
 allGripPoints, 236
 findIndex, 236
 gripEdit, 236
 gripIndex, 238
 init, 236
 mouseSnapPoint, 237
 normalPath, 238
 objectCopyPath, 237
 objectPos, 237
 objectSavePath, 237
 objectX, 237
 objectY, 237
 paint, 237
 PolygonObject, 236
 setObjectPos, 237
 setObjectX, 238
 setObjectY, 238
Type, 236
type, 238
updatePath, 238
updateRubber, 238
vulcanize, 238
polyline
 EmbGeometry_, 134
PolylineObject, 239
 ~PolylineObject, 241
 allGripPoints, 242
 findIndex, 242
 gripEdit, 242
 gripIndex, 244
 init, 242
 mouseSnapPoint, 242
 normalPath, 244
 objectCopyPath, 242
 objectPos, 242
 objectSavePath, 242
 objectX, 243
 objectY, 243
 paint, 243
 PolylineObject, 241
 setObjectPos, 243
 setObjectX, 243
 setObjectY, 243
Type, 241
type, 243
updatePath, 243
updateRubber, 244
vulcanize, 244
position
 EmbAlignedDim_, 123
 EmbAngularDim_, 123
 EmbArcLengthDim_, 124
 EmbBlock_, 126
 EmbDiameterDim_, 130
 EmblImage_, 135
 EmblInfiniteLine_, 136
 EmbLeaderDim_, 137
 EmbLinearDim_, 138
 EmbOrdinateDim_, 138
 EmbPoint_, 141
 EmbRadiusDim_, 141
 EmbRay_, 142
 EmbTextMulti_, 145
 EmbTextSingle_, 146
positiveXHoopSize
 VipHeader_, 331
positiveYHoopSize
 VipHeader_, 331
precisionAngle
 PropertyEditor, 257
precisionLength
 PropertyEditor, 257
prefix
 CmdPromptInput, 106
pressPoint
 View, 329
pressResizeHistory
 CmdPromptSplitter, 108
pressY
 CmdPromptHandle, 97
preview
 settings-dialog.cpp, 398
PREVIEW_CLONE_NULL
 embroidermodder.h, 342
PREVIEW_CLONE_RUBBER
 embroidermodder.h, 342
PREVIEW_CLONE_SELECTED
 embroidermodder.h, 342

PREVIEW_CLONE_VALUES
embroidermodder.h, 342

PREVIEW_MODE_MOVE
embroidermodder.h, 343

PREVIEW_MODE_NULL
embroidermodder.h, 343

PREVIEW_MODE_ROTATE
embroidermodder.h, 343

PREVIEW_MODE_SCALE
embroidermodder.h, 343

PREVIEW_MODE_VALUES
embroidermodder.h, 342

previewActive
View, 329

previewData
View, 329

PreviewDialog, 244
~PreviewDialog, 245
imgWidget, 245
PreviewDialog, 244

previewMode
View, 329

previewObjectItemGroup
View, 329

previewObjectList
View, 329

previewOff
View, 324

previewOn
View, 324

previewPoint
View, 329

print
MainWindow, 197
MdiWindow, 219

printArcResults
embroidery_internal.h, 468

printer
MdiWindow, 223

printing_default_device
Settings_, 278

printing_disable_bg
Settings_, 278

printing_use_last_device
Settings_, 278

privacy_policy.md, 577

processInput
CmdPrompt, 92
CmdPromptInput, 104

prompt
MainWindow, 207

prompt_bg_color
Settings_, 278

prompt_font_family
Settings_, 278

prompt_font_size
Settings_, 278

prompt_font_style

Settings_, 278

prompt_save_history
Settings_, 278

prompt_save_history_as_html
Settings_, 278

prompt_save_history_filename
Settings_, 278

prompt_text_color
Settings_, 278

promptDivider
CmdPrompt, 95

promptHistory
CmdPrompt, 95
MdiWindow, 223

promptHistoryAppended
MainWindow, 197
MdiWindow, 219

promptInput
CmdPrompt, 95

promptInputList
MdiWindow, 223

promptInputNext
MainWindow, 198
MdiWindow, 219

promptInputNum
MdiWindow, 223

promptInputPrevious
MainWindow, 198
MdiWindow, 219

promptInputPrevNext
MdiWindow, 219

promptSplitter
CmdPrompt, 95

promptVBoxLayout
CmdPrompt, 95

property-editor.cpp
comboBoxArcClockwise, 389
comboBoxPathClosed, 389
comboBoxPathVertexNum, 389
comboBoxPolylineClosed, 389
comboBoxPolylineVertexNum, 389
comboBoxTextSingleBackward, 390
comboBoxTextSingleJustify, 390
comboBoxTextSingleUpsideDown, 390
lineEditBlockX, 390
lineEditBlockY, 390
lineEditEllipseCenterX, 390
lineEditEllipseCenterY, 390
lineEditEllipseDiameterMajor, 390
lineEditEllipseDiameterMinor, 390
lineEditEllipseRadiusMajor, 390
lineEditEllipseRadiusMinor, 390
lineEditImageHeight, 390
lineEditImageName, 390
lineEditImagePath, 390
lineEditImageWidth, 390
lineEditImageX, 390
lineEditImageY, 390

lineEditInfiniteLineVectorX, 390
lineEditInfiniteLineVectorY, 391
lineEditInfiniteLineX2, 391
lineEditInfiniteLineY1, 391
lineEditInfiniteLineY2, 391
lineEditLineAngle, 391
lineEditLineDeltaX, 391
lineEditLineDeltaY, 391
lineEditLineEndX, 391
lineEditLineEndY, 391
lineEditLineLength, 391
lineEditLineStartX, 391
lineEditLineStartY, 391
lineEditPathArea, 391
lineEditPathLength, 391
lineEditPathVertexX, 391
lineEditPathVertexY, 391
lineEditPointX, 391
lineEditPointY, 391
lineEditPolygonCenterX, 392
lineEditPolygonCenterY, 392
lineEditPolygonDiameterSide, 392
lineEditPolygonDiameterVertex, 392
lineEditPolygonInteriorAngle, 392
lineEditPolygonRadiusSide, 392
lineEditPolygonRadiusVertex, 392
lineEditPolylineArea, 392
lineEditPolylineLength, 392
lineEditPolylineVertexX, 392
lineEditPolylineVertexY, 392
lineEditRayVectorX, 392
lineEditRayVectorY, 392
lineEditRayX1, 392
lineEditRayX2, 392
lineEditRayY1, 392
lineEditRayY2, 392
lineEditRectangleArea, 392
lineEditRectangleCorner1X, 393
lineEditRectangleCorner1Y, 393
lineEditRectangleCorner2X, 393
lineEditRectangleCorner2Y, 393
lineEditRectangleCorner3X, 393
lineEditRectangleCorner3Y, 393
lineEditRectangleCorner4X, 393
lineEditRectangleCorner4Y, 393
lineEditRectangleHeight, 393
lineEditRectangleWidth, 393
lineEditTextMultiX, 393
lineEditTextMultiY, 393
lineEditTextSingleContents, 393
lineEditTextSingleHeight, 393
lineEditTextSingleRotation, 393
lineEditTextSingleY, 393
tempArcObj, 393
tempBlockObj, 393
tempCircleObj, 394
tempDimAlignedObj, 394
tempDimAngularObj, 394
tempDimArcLenObj, 394
tempDimDiamObj, 394
tempDimLeaderObj, 394
tempDimLinearObj, 394
tempDimOrdObj, 394
tempDimRadiusObj, 394
tempEllipseArcObj, 394
tempEllipseObj, 394
tempHatchObj, 394
tempImageObj, 394
tempInfLineObj, 394
tempLineObj, 394
tempPathObj, 394
tempPointObj, 394
tempPolygonObj, 394
tempPolylineObj, 395
tempRayObj, 395
tempRectObj, 395
tempSplineObj, 395
tempTextMultiObj, 395
tempTextSingleObj, 395
toolButtonArcClockwise, 395
toolButtonBlockX, 395
toolButtonBlockY, 395
toolButtonImageHeight, 395
toolButtonImageName, 395
toolButtonImagePath, 395
toolButtonImageWidth, 395
toolButtonImageX, 395
toolButtonImageY, 395
toolButtonInfiniteLineVectorX, 395
toolButtonInfiniteLineVectorY, 395
toolButtonInfiniteLineX2, 395
toolButtonInfiniteLineY2, 396
toolButtonLineAngle, 396
toolButtonLineDeltaX, 396
toolButtonLineDeltaY, 396
toolButtonLineEndX, 396
toolButtonLineEndY, 396
toolButtonLineLength, 396
toolButtonLineStartX, 396
toolButtonLineStartY, 396
toolButtonPolygonDiameterSide, 396
toolButtonPolygonDiameterVertex, 396
toolButtonPolygonInteriorAngle, 396
toolButtonPolygonRadiusSide, 396
toolButtonPolygonRadiusVertex, 396
toolButtonPolylineArea, 396
toolButtonPolylineClosed, 396
toolButtonPolylineLength, 396
toolButtonPolylineVertexNum, 396
toolButtonPolylineVertexX, 397
toolButtonPolylineVertexY, 397
toolButtonRayVectorX, 397
toolButtonRayVectorY, 397
toolButtonRayX1, 397
toolButtonRayX2, 397
toolButtonRayY1, 397

toolButtonRayY2, 397
toolButtonTextMultiX, 397
toolButtonTextMultiY, 397
toolButtonTextSingleContents, 397
toolButtonTextSingleFont, 397
toolButtonTextSingleHeight, 397
toolButtonTextSingleJustify, 397
toolButtonTextSingleRotation, 397
PropertyEditor, 245
 ~PropertyEditor, 247
 clearAllFields, 247
 comboBoxSelected, 255
 createComboBox, 248
 createComboBoxSelected, 248
 createFontComboBox, 248
 createGroupBox, 248
 createGroupBoxGeneral, 248
 createGroupBoxGeometryArc, 248
 createGroupBoxGeometryBlock, 249
 createGroupBoxGeometryCircle, 249
 createGroupBoxGeometryDimAligned, 249
 createGroupBoxGeometryDimAngular, 249
 createGroupBoxGeometryDimArcLength, 249
 createGroupBoxGeometryDimDiameter, 249
 createGroupBoxGeometryDimLeader, 250
 createGroupBoxGeometryDimLinear, 250
 createGroupBoxGeometryDimOrdinate, 250
 createGroupBoxGeometryDimRadius, 250
 createGroupBoxGeometryEllipse, 250
 createGroupBoxGeometryImage, 250
 createGroupBoxGeometryInfiniteLine, 250
 createGroupBoxGeometryLine, 251
 createGroupBoxGeometryPath, 251
 createGroupBoxGeometryPoint, 251
 createGroupBoxGeometryPolygon, 251
 createGroupBoxGeometryPolyline, 251
 createGroupBoxGeometryRay, 251
 createGroupBoxGeometryRectangle, 252
 createGroupBoxGeometryTextMulti, 252
 createGroupBoxGeometryTextSingle, 252
 createGroupBoxMiscArc, 252
 createGroupBoxMiscImage, 252
 createGroupBoxMiscPath, 252
 createGroupBoxMiscPolyline, 253
 createGroupBoxMiscTextSingle, 253
 createGroupBoxTextTextSingle, 253
 createLineEdit, 253
 createToolButton, 253
 createToolButtonPickAdd, 253
 createToolButtonQSelect, 253
 eventFilter, 254
 fieldEdited, 254
 fieldNewText, 256
 fieldNoText, 256
 fieldOffText, 256
 fieldOldText, 256
 fieldOnText, 256
 fieldVariesText, 256
fieldYesText, 256
focusWidget, 256
hideAllGroups, 254
iconDir, 256
iconSize, 256
mapSignal, 254
pickAdd, 256
pickAddModeToggled, 254
precisionAngle, 257
precisionLength, 257
PropertyEditor, 247
 propertyEditorButtonStyle, 257
selectedItemList, 257
setSelectedItems, 254
showGroups, 254
showOneType, 254
signalMapper, 257
togglePickAddMode, 255
toolButtonPickAdd, 257
toolButtonQSelect, 257
updateComboBoxBoollfVaries, 255
updateComboBoxStrlfVaries, 255
updateFontComboBoxStrlfVaries, 255
updateLineEditNumlfVaries, 255
updateLineEditStrlfVaries, 255
updatePickAddModeButton, 255
propertyEditorButtonStyle
 PropertyEditor, 257

qsnap_aperture_size
 Settings_, 278
qsnap_apparent
 Settings_, 278
qsnap_center
 Settings_, 278
qsnap_enabled
 Settings_, 278
qsnap_endpoint
 Settings_, 278
qsnap_extension
 Settings_, 279
qsnap_insertion
 Settings_, 279
qsnap_intersection
 Settings_, 279
qsnap_locator_color
 Settings_, 279
qsnap_locator_size
 Settings_, 279
qsnap_midpoint
 Settings_, 279
qsnap_mode
 EmbView_, 150
qsnap_nearest
 Settings_, 279
qsnap_node
 Settings_, 279
qsnap_parallel
 Settings_, 279

qsnap_perpendicular
 Settings_, 279

qsnap_quadrant
 Settings_, 279

qsnap_tangent
 Settings_, 279

qSnapActive
 View, 329

qsnapApertureSize
 View, 329

qsnapLocatorColor
 View, 329

qsnapLocatorSize
 View, 329

qSnapToggle
 View, 329

qtrack_mode
 EmbView_, 150

QUADTOCONTROL
 embroidery_internal.h, 458

QUADTOEND
 embroidery_internal.h, 458

quit
 MainWindow, 198

r
 EmbColor_, 128

r_value
 Parameter_, 224

radians
 embroidermodder.h, 344
 embroidery.h, 434
 functions.c, 552

radius
 EmbCircle_, 127
 EmbEllipse_, 131
 EmbRect_, 142

rapidFireEnabled
 CmdPromptInput, 106

rapidMoveActive
 View, 329

read100
 embroidery_internal.h, 469
 format_100.c, 502

read10o
 embroidery_internal.h, 469
 format_10o.c, 503

read_configuration
 mainwindow-settings.cpp, 380

read_hoop
 format_jef.c, 520

read_settings
 embroidermodder.h, 344
 mainwindow.cpp, 384

read_string_setting
 mainwindow.cpp, 384

readArt
 embroidery_internal.h, 469
 format_art.c, 503

readBmc
 embroidery_internal.h, 469
 format_bmc.c, 504

readBro
 embroidery_internal.h, 469
 format_bro.c, 504

readCnd
 embroidery_internal.h, 469
 format_cnd.c, 505

readCol
 embroidery_internal.h, 469
 format_col.c, 506

readCsd
 embroidery_internal.h, 469
 format_csd.c, 507

readCsv
 embroidery_internal.h, 469
 format_csv.c, 508

readDat
 embroidery_internal.h, 469
 format_dat.c, 508

readDem
 embroidery_internal.h, 469
 format_dem.c, 509

readDescriptions
 embroidery_internal.h, 470
 format_pes.c, 529

readDsb
 embroidery_internal.h, 470
 format_dsb.c, 510

readDst
 embroidery_internal.h, 470
 format_dst.c, 512

readDsz
 embroidery_internal.h, 470
 format_dsz.c, 512

readDxf
 embroidery_internal.h, 470
 format_dxf.c, 513

readEdr
 embroidery_internal.h, 470
 format_edr.c, 513

readEmd
 embroidery_internal.h, 470
 format_emd.c, 514

reader_state
 EmbFormatList_, 132

readExp
 embroidery_internal.h, 470
 format_exp.c, 514

readExy
 embroidery_internal.h, 470
 format_exy.c, 515

readEys
 embroidery_internal.h, 470
 format_eys.c, 515

readFeatherPatterns
 embroidery_internal.h, 471

format_pes.c, 529
readFullSector
 embroidery_internal.h, 471
 main.c, 569
readFxy
 embroidery_internal.h, 471
 format_fxy.c, 516
readGc
 embroidery_internal.h, 471
 format_gc.c, 516
readGnc
 embroidery_internal.h, 471
 format_gnc.c, 517
readGt
 embroidery_internal.h, 471
 format_gt.c, 517
readHoopName
 embroidery_internal.h, 471
 format_pes.c, 529
readHus
 embroidery_internal.h, 471
 format_hus.c, 519
readImageString
 embroidery_internal.h, 471
 format_pes.c, 530
readInb
 embroidery_internal.h, 472
 format_inb.c, 519
readInf
 embroidery_internal.h, 472
 format_inf.c, 519
readJef
 embroidery_internal.h, 472
 format_jef.c, 521
readKsm
 embroidery_internal.h, 472
 format_ksm.c, 521
readLine
 format_dxf.c, 513
readMax
 embroidery_internal.h, 472
 format_max.c, 522
readMit
 embroidery_internal.h, 472
 format_mit.c, 522
readMotifPatterns
 embroidery_internal.h, 472
 format_pes.c, 530
readNew
 embroidery_internal.h, 472
 format_new.c, 523
readNextSector
 embroidery_internal.h, 472
 main.c, 569
readOfm
 embroidery_internal.h, 472
 format_ofm.c, 524
readPcd
 embroidery_internal.h, 473
 format_pcd.c, 525
readPcm
 embroidery_internal.h, 473
 format_pcm.c, 525
readPcq
 embroidery_internal.h, 473
 format_pcq.c, 526
readPcs
 embroidery_internal.h, 473
 format_pcs.c, 526
readPec
 embroidery_internal.h, 473
 format_pec.c, 527
readPecStitches
 embroidery_internal.h, 473
 format_pec.c, 527
readPel
 embroidery_internal.h, 473
 format_pel.c, 528
readPem
 embroidery_internal.h, 473
 format_pem.c, 528
readPes
 embroidery_internal.h, 474
 format_pes.c, 530
readPESHeaderV10
 embroidery_internal.h, 474
 format_pes.c, 530
readPESHeaderV5
 embroidery_internal.h, 474
 format_pes.c, 530
readPESHeaderV6
 embroidery_internal.h, 474
 format_pes.c, 530
readPESHeaderV7
 embroidery_internal.h, 474
 format_pes.c, 530
readPESHeaderV8
 embroidery_internal.h, 474
 format_pes.c, 530
readPESHeaderV9
 embroidery_internal.h, 474
 format_pes.c, 530
readPhb
 embroidery_internal.h, 474
 format_phb.c, 531
readPhc
 embroidery_internal.h, 474
 format_phc.c, 532
readPlt
 embroidery_internal.h, 474
 format_plt.c, 532
readProgrammableFills
 embroidery_internal.h, 475
 format_pes.c, 530
readRgb
 embroidery_internal.h, 475

format_rgb.c, 533
readSettings
 MainWindow, 198
readSew
 embroidery_internal.h, 475
 format_sew.c, 533
readShv
 embroidery_internal.h, 475
 format_shv.c, 534
readSst
 embroidery_internal.h, 475
 format_sst.c, 534
readStx
 embroidery_internal.h, 475
 format_stx.c, 535
readSvg
 embroidery_internal.h, 475
 format_svg.c, 536
readT01
 embroidery_internal.h, 475
 format_t01.c, 537
readT09
 embroidery_internal.h, 475
 format_t09.c, 537
readTap
 embroidery_internal.h, 475
 format_tap.c, 538
readThr
 embroidery_internal.h, 476
 format_thr.c, 538
readThreads
 embroidery_internal.h, 476
 format_pes.c, 530
readTxt
 embroidery_internal.h, 476
 format_txt.c, 539
readU00
 embroidery_internal.h, 476
 format_u00.c, 539
readU01
 embroidery_internal.h, 476
 format_u01.c, 540
readVip
 embroidery_internal.h, 476
 format_vip.c, 541
readVp3
 embroidery_internal.h, 476
 format_vp3.c, 542
readXxx
 embroidery_internal.h, 476
 format_xxx.c, 543
readZsk
 embroidery_internal.h, 476
 format_zsk.c, 544
real_render
 EmbView_, 150
realRender
 BaseObject, 78
 recalculateLimits
 View, 324
recentMenu
 MainWindow, 207
recentMenuAboutToShow
 MainWindow, 198
rect
 BaseObject, 78
 EmbGeometry_, 134
rect.c
 embRect_area, 554
 embRect_init, 554
RectObject, 257
 ~RectObject, 260
 allGripPoints, 260
 gripEdit, 260
 init, 261
 mouseSnapPoint, 261
 objectArea, 261
 objectBottomLeft, 261
 objectBottomRight, 261
 objectHeight, 261
 objectPos, 261
 objectSavePath, 261
 objectTopLeft, 262
 objectTopRight, 262
 objectWidth, 262
 paint, 262
 RectObject, 260
 setObjectRect, 262
 Type, 260
 type, 262
 updatePath, 262
 updateRubber, 262
 vulcanize, 262
RED_TERM_COLOR
 embroidery_internal.h, 458
redo
 MainWindow, 198
 UndoableAddCommand, 307
 UndoableDeleteCommand, 308
 UndoableGripEditCommand, 309
 UndoableMirrorCommand, 310
 UndoableMoveCommand, 311
 UndoableNavCommand, 312
 UndoableRotateCommand, 313
 UndoableScaleCommand, 315
 UndoEditor, 316
redoPressed
 CmdPrompt, 92
 CmdPromptInput, 105
redoText
 UndoEditor, 316
rejectChanges
 Settings_Dialog, 290
releasePoint
 View, 329
releaseResizeHistory

CmdPromptSplitter, 108
releaseY
 CmdPromptHandle, 97
repeatAction
 View, 324
report
 embroidery.h, 434
reserved
 ThredExtension_, 304
 ThredHeader_, 305
reserved1
 _bcf_file_header, 56
reserved2
 _bcf_file_header, 56
RESET_TERM_COLOR
 embroidery_internal.h, 458
resizeEvent
 MainWindow, 198
resizeHistory
 CmdPromptHistory, 99
resizeTheHistory
 CmdPrompt, 92
reverse_byte_order
 encoding.c, 493
rgb, 16, 475, 533
right
 _vp3Hoop, 58
 EmbRect_, 143
 hoop_padding, 153
right2
 _vp3Hoop, 58
rightBrush
 SelectBox, 271
rightBrushColor
 SelectBox, 271
rightPen
 SelectBox, 271
rightPenColor
 SelectBox, 271
rightSiblingId
 _bcf_directory_entry, 52
RobisonAnton_Polyester
 embroidery.h, 417
RobisonAnton_Rayon
 embroidery.h, 417
rotate
 UndoableRotateCommand, 313
ROTATE_MODE_NORMAL
 embroidermodder.h, 343
ROTATE_MODE_REFERENCE
 embroidermodder.h, 343
rotate_vector
 object-arc.cpp, 385
rotateAction
 View, 324
rotateSelected
 View, 324
rotation
 EmbEllipse_, 131
 EmbRect_, 143
 UiObject_, 306
roundToMultiple
 View, 324
rubber_mode
 EmbView_, 150
rubberRoomList
 View, 329
ruler_color
 Settings_, 279
ruler_metric
 Settings_, 279
ruler_mode
 EmbView_, 151
ruler_pixel_size
 Settings_, 279
ruler_show_on_load
 Settings_, 279
ruler_width
 Settings_, 279
rulerColor
 View, 330
rulerMetric
 View, 330
rulerPixelSize
 View, 330
rules
 fill.c, 497
 LSYSTEM, 170
run_script
 MainWindow, 198
run_script_file
 MainWindow, 199
runCommand
 CmdPrompt, 92
 CmdPromptInput, 105
running
 Settings_, 279
s_value
 Parameter_, 224
safe_free
 embroidery_internal.h, 476
 formats.c, 501
save
 ImageWidget, 161
 SaveObject, 268
save_points_to_pattern
 fill.c, 497
saveasfile
 MainWindow, 199
saveBMC
 MdiWindow, 219
saveFile
 MdiWindow, 220
savefile
 MainWindow, 199
saveHistory

CmdPrompt, 92
 SaveObject, 263
 ~SaveObject, 264
 addArc, 264
 addBlock, 264
 addCircle, 264
 addDimAligned, 265
 addDimAngular, 265
 addDimArcLength, 265
 addDimDiameter, 266
 addDimLeader, 266
 addDimLinear, 266
 addDimOrdinate, 266
 addDimRadius, 266
 addEllipse, 266
 addEllipseArc, 266
 addGrid, 267
 addHatch, 267
 addImage, 267
 addInfiniteLine, 267
 addLine, 267
 addPath, 267
 addPoint, 267
 addPolygon, 267
 addPolyline, 268
 addRay, 268
 addRectangle, 268
 addSlot, 268
 addSpline, 268
 addTextMulti, 268
 addTextSingle, 268
 formatType, 269
 gscene, 269
 save, 268
 SaveObject, 264
 toPolyline, 269
 scale
 EmbView_, 151
 UiObject_, 306
 SCALE_MODE_NORMAL
 embroidermodder.h, 343
 SCALE_MODE_REFERENCE
 embroidermodder.h, 343
 scaleAction
 View, 324
 scaleSelected
 View, 324
 sceneGripPoint
 View, 330
 sceneMousePoint
 View, 330
 sceneMovePoint
 View, 330
 scenePressPoint
 View, 330
 sceneReleasePoint
 View, 330
 script
 Action___, 60
 second
 EmbTime_, 147
 sectionName
 StxThread_, 296
 sectorShift
 _bcf_file_header, 56
 sectorSize
 _bcf_file_difat, 54
 main.c, 569
 seekToSector
 main.c, 569
 selectable
 UiObject_, 306
 selectAll
 View, 324
 selectAllPressed
 CmdPrompt, 93
 CmdPromptInput, 105
 SelectBox, 269
 alpha, 271
 boxDir, 271
 dirBrush, 271
 dirPen, 271
 forceRepaint, 270
 leftBrush, 271
 leftBrushColor, 271
 leftPen, 271
 leftPenColor, 271
 paintEvent, 270
 rightBrush, 271
 rightBrushColor, 271
 rightPen, 271
 rightPenColor, 271
 SelectBox, 270
 setColors, 270
 setDirection, 271
 selectBox
 View, 330
 selected
 EmbView_, 151
 selectedItemList
 PropertyEditor, 257
 selectingActive
 View, 330
 selection_coolgrip_color
 Settings_, 280
 selection_grip_size
 Settings_, 280
 selection_hotgrip_color
 Settings_, 280
 selection_mode_pickadd
 Settings_, 280
 selection_mode_pickdrag
 Settings_, 280
 selection_mode_pickfirst
 Settings_, 280
 selection_pickbox_size

Settings_, 280
selectionChanged
 View, 325
sendCloseMdiWin
 MdiWindow, 220
SEQUIN
 embroidery.h, 417
set_dst_variable
 format_dst.c, 512
set_label_visibility
 Settings_Dialog, 290
set_object_color
 arc.c, 549
set_spinbox_visibility
 Settings_Dialog, 290
setBackgroundColor
 MdiArea, 211
 View, 325
setBackgroundLogo
 MdiArea, 211
setBackgroundTexture
 MdiArea, 211
setColors
 SelectBox, 270
setCornerButton
 View, 325
setCrossHairColor
 View, 325
setCrossHairSize
 View, 325
setCurrentColor
 MdiWindow, 220
setCurrentFile
 MdiWindow, 220
setCurrentLayer
 MdiWindow, 220
setCurrentLineType
 MdiWindow, 221
setCurrentLineWidth
 MdiWindow, 221
setCurrentText
 CmdPrompt, 93
setDirection
 SelectBox, 271
setGridColor
 View, 325
setHistory
 CmdPrompt, 93
setLine
 BaseObject, 79
setMainWin
 Application, 62
setMouseCoord
 StatusBar, 292
setObjectArea
 CircleObject, 86
setObjectCenter
 BaseObject, 79
 setObjectCenterX
 BaseObject, 79
 setObjectCenterY
 BaseObject, 79
 setObjectCircumference
 CircleObject, 86
 setObjectColor
 BaseObject, 79
 setObjectColorRGB
 BaseObject, 79
 setObjectDiameter
 CircleObject, 86
 setObjectDiameterMajor
 EllipseObject, 121
 setObjectDiameterMinor
 EllipseObject, 122
 setObjectEndAngle
 ArcObject, 72
 setObjectEndPoint
 ArcObject, 72, 73
 setObjectEndPoint1
 DimLeaderObject, 115
 LineObject, 168
 setObjectEndPoint2
 DimLeaderObject, 115
 LineObject, 169
 setObjectLineType
 BaseObject, 79
 setObjectLineWidth
 BaseObject, 79
 setObjectMidPoint
 ArcObject, 73
 setObjectPath
 BaseObject, 80
 setObjectPos
 PathObject, 228
 PointObject, 232, 233
 PolygonObject, 237
 PolylineObject, 243
 TextSingleObject, 301
 setObjectRadius
 ArcObject, 73
 CircleObject, 86
 setObjectRadiusMajor
 EllipseObject, 122
 setObjectRadiusMinor
 EllipseObject, 122
 setObjectRect
 ImageObject, 159
 RectObject, 262
 setObjectRubberMode
 BaseObject, 80
 setObjectRubberPoint
 BaseObject, 80
 setObjectRubberText
 BaseObject, 80
 setObjectSize
 EllipseObject, 122

setObjectStartAngle
 ArcObject, 73
setObjectStartPoint
 ArcObject, 73
setObjectText
 TextSingleObject, 301
setObjectTextBackward
 TextSingleObject, 301
setObjectTextBold
 TextSingleObject, 301
setObjectTextFont
 TextSingleObject, 301
setObjectTextItalic
 TextSingleObject, 301
setObjectTextJustify
 TextSingleObject, 301
setObjectTextOverline
 TextSingleObject, 301
setObjectTextSize
 TextSingleObject, 302
setObjectTextStrikeOut
 TextSingleObject, 302
setObjectTextStyle
 TextSingleObject, 302
setObjectTextUnderline
 TextSingleObject, 302
setObjectTextUpsideDown
 TextSingleObject, 302
setObjectX
 PathObject, 228
 PointObject, 233
 PolygonObject, 238
 PolylineObject, 243
 TextSingleObject, 302
setObjectX1
 DimLeaderObject, 115
 LineObject, 169
setObjectX2
 DimLeaderObject, 115
 LineObject, 169
setObjectY
 PathObject, 228
 PointObject, 233
 PolygonObject, 238
 PolylineObject, 243
 TextSingleObject, 302
setObjectY1
 DimLeaderObject, 115
 LineObject, 169
setObjectY2
 DimLeaderObject, 115
 LineObject, 169
setPrefix
 CmdPrompt, 93
setPromptBackgroundColor
 CmdPrompt, 93
setPromptFontFamily
 CmdPrompt, 93
setPromptFontSize
 CmdPrompt, 93
setPromptFontStyle
 CmdPrompt, 93
setPromptTextColor
 CmdPrompt, 93
setRect
 BaseObject, 80
setRubberMode
 View, 325
setRubberPoint
 View, 325
setRubberText
 View, 325
setRulerColor
 View, 325
setSelectBoxColors
 View, 325
setSelectedItems
 PropertyEditor, 254
setShiftPressed
 MainWindow, 199
setShiftReleased
 MainWindow, 199
setTextAngle
 MainWindow, 200
setTextBold
 MainWindow, 200
setTextFont
 MainWindow, 200
setTextItalic
 MainWindow, 200
setTextOverline
 MainWindow, 200
setTextSize
 MainWindow, 200
setTextStrikeOut
 MainWindow, 200
setTextUnderline
 MainWindow, 200
Settings
 embroidermodder.h, 339
settings
 embroidermodder.h, 346
 settings-dialog.cpp, 398
settings-dialog.cpp
 accept_, 398
 dialog, 398
 extensions, 398
 preview, 398
 settings, 398
Settings_, 272
 assets_dir, 274
 current_directory, 274
 debug_mode, 274
 display_bg_color, 274
 display_crosshair_color, 274
 display_crosshair_percent, 274

display_renderhint_aa, 274
display_renderhint_high_aa, 275
display_renderhint_noncosmetic, 275
display_renderhint_smooth_pix, 275
display_renderhint_text_aa, 275
display_scrollbar_widget_num, 275
display_selectbox_alpha, 275
display_selectbox_left_color, 275
display_selectbox_left_fill, 275
display_selectbox_right_color, 275
display_selectbox_right_fill, 275
display_show_scrollbars, 275
display_units, 275
display_use_opengl, 275
display_zoomscale_in, 275
display_zoomscale_out, 275
general_check_for_updates, 275
general_current_tip, 275
general_icon_size, 275
general_icon_theme, 276
general_language, 276
general_mdi_bg_color, 276
general_mdi_bg_logo, 276
general_mdi_bg_texture, 276
general_mdi_bg_use_color, 276
general_mdi_bg_use_logo, 276
general_mdi_bg_use_texture, 276
general_system_help_browser, 276
general_tip_of_the_day, 276
grid_center, 276
grid_center_on_origin, 276
grid_color, 276
grid_color_match_crosshair, 276
grid_load_from_file, 276
grid_show_on_load, 276
grid_show_origin, 276
grid_size, 276
grid_size_radius, 277
grid_spacing, 277
grid_spacing_angle, 277
grid_spacing_radius, 277
grid_type, 277
lwt_default_lwt, 277
lwt_real_render, 277
lwt_show_lwt, 277
major_tick_seperation, 277
needle_speed, 277
opensave_custom_filter, 277
opensave_open_format, 277
opensave_open_thumbnail, 277
opensave_recent_directory, 277
opensave_recent_list_of_files, 277
opensave_recent_max_files, 277
opensave_save_format, 277
opensave_save_thumbnail, 277
opensave_trim_dst_num_jumps, 278
pattern_index, 278
printing_default_device, 278
printing_disable_bg, 278
printing_use_last_device, 278
prompt_bg_color, 278
prompt_font_family, 278
prompt_font_size, 278
prompt_font_style, 278
prompt_save_history, 278
prompt_save_history_as_html, 278
prompt_save_history_filename, 278
prompt_text_color, 278
qsnap_aperture_size, 278
qsnap_apparent, 278
qsnap_center, 278
qsnap_enabled, 278
qsnap_endpoint, 278
qsnap_extension, 279
qsnap_insertion, 279
qsnap_intersection, 279
qsnap_locator_color, 279
qsnap_locator_size, 279
qsnap_midpoint, 279
qsnap_nearest, 279
qsnap_node, 279
qsnap_parallel, 279
qsnap_perpendicular, 279
qsnap_quadrant, 279
qsnap_tangent, 279
ruler_color, 279
ruler_metric, 279
ruler_pixel_size, 279
ruler_show_on_load, 279
ruler_width, 279
running, 279
selection_coolgrip_color, 280
selection_grip_size, 280
selection_hotgrip_color, 280
selection_mode_pickadd, 280
selection_mode_pickdrag, 280
selection_mode_pickfirst, 280
selection_pickbox_size, 280
shift_held, 280
shine_color, 280
show_about_dialog, 280
show_details_dialog, 280
show_editor, 280
show_open_file_dialog, 280
show_settings_editor, 280
stitch_time, 280
testing, 280
text_angle, 280
text_font, 280
text_size, 281
text_style_bold, 281
text_style_italic, 281
text_style_overline, 281
text_style_strikeout, 281
text_style_underline, 281
tick_depth, 281

ticks_color, 281
 to_open, 281
 use_translation, 281
 version, 281
 zoomInLimit, 281
 zoomOutLimit, 281
Settings_Dialog, 281
 ~Settings_Dialog, 284
 acceptChanges, 284
 addColorsToComboBox, 284
 buttonBox, 291
 buttonCustomFilterClearAll, 284
 buttonCustomFilterClearAllClicked, 284
 buttonCustomFilterSelectAll, 284
 buttonCustomFilterSelectAllClicked, 284
 buttonQSnapClearAll, 284
 buttonQSnapClearAllClicked, 284
 buttonQSnapSelectAll, 284
 buttonQSnapSelectAllClicked, 284
 checkBoxCustomFilterStateChanged, 284
 checkBoxDisableBGStateChanged, 284
 checkBoxGeneralMdiBGUseColorStateChanged, 285
 checkBoxGeneralMdiBGUseLogoStateChanged, 285
 checkBoxGeneralMdiBGUseTextureStateChanged, 285
 checkBoxGridCenterOnOriginStateChanged, 285
 checkBoxGridColumnMatchCrossHairStateChanged, 285
 checkBoxGridLoadFromFileStateChanged, 285
 checkBoxGridShowOnLoadStateChanged, 285
 checkBoxGridShowOriginStateChanged, 285
 checkBoxLwtRealRenderStateChanged, 285
 checkBoxLwtShowLwtStateChanged, 285
 checkBoxPromptSaveHistoryAsHtmlStateChanged, 285
 checkBoxPromptSaveHistoryStateChanged, 285
 checkBoxRenderHintAAStateChanged, 286
 checkBoxRenderHintHighAAStateChanged, 286
 checkBoxRenderHintNonCosmeticStateChanged, 286
 checkBoxRenderHintSmoothPixStateChanged, 286
 checkBoxRenderHintTextAAStateChanged, 286
 checkBoxRulerShowOnLoadStateChanged, 286
 checkBoxSelectionModePickAddStateChanged, 286
 checkBoxSelectionModePickDragStateChanged, 286
 checkBoxSelectionModePickFirstStateChanged, 286
 checkBoxShowScrollBarsStateChanged, 286
 checkBoxTipOfTheDayStateChanged, 286
 checkBoxUseOpenGLStateChanged, 286
 chooseDisplayBackgroundColor, 287
 chooseDisplayCrossHairColor, 287
 chooseDisplaySelectBoxLeftColor, 287
 chooseDisplaySelectBoxLeftFill, 287
 chooseDisplaySelectBoxRightColor, 287
 chooseDisplaySelectBoxRightFill, 287
 chooseGeneralMdiBackgroundColor, 287
 chooseGeneralMdiBackgroundLogo, 287
 chooseGeneralMdiBackgroundTexture, 287
 chooseGridColor, 287
 choosePromptBackgroundColor, 287
 choosePromptTextColor, 287
 chooseRulerColor, 287
 comboBoxGridTypeCurrentIndexChanged, 287
 comboBoxIconSizeCurrentIndexChanged, 287
 comboBoxIconThemeCurrentIndexChanged, 288
 comboBoxLanguageCurrentIndexChanged, 288
 comboBoxPromptFontFamilyCurrentIndexChanged, 288
 comboBoxPromptFontSizeCurrentIndexChanged, 288
 comboBoxQSnapLocatorColorCurrentIndexChanged, 288
 comboBoxRulerMetricCurrentIndexChanged, 288
 comboBoxScrollBarWidgetCurrentIndexChanged, 288
 comboBoxSelectionCoolGripColorCurrentIndexChanged, 288
 comboBoxSelectionHotGripColorCurrentIndexChanged, 288
 create_checkbox, 288
 create_float_spinbox, 288
 createTabDisplay, 289
 createTabFilesPaths, 289
 createTabGeneral, 289
 createTabGridRuler, 289
 createTabLineWeight, 289
 createTabOpenSave, 289
 createTabOrthoPolar, 289
 createTabPrinting, 289
 createTabPrompt, 289
 createTabQuickSnap, 289
 createTabQuickTrack, 289
 createTabSelection, 289
 createTabSnap, 289
 currentDisplayBackgroundColorChanged, 289
 currentDisplayCrossHairColorChanged, 289
 currentDisplaySelectBoxLeftColorChanged, 290
 currentDisplaySelectBoxLeftFillChanged, 290
 currentDisplaySelectBoxRightColorChanged, 290
 currentDisplaySelectBoxRightFillChanged, 290
 currentGeneralMdiBackgroundColorChanged, 290
 currentGridColorChanged, 290
 currentPromptBackgroundColorChanged, 290
 currentPromptTextColorChanged, 290
 currentRulerColorChanged, 290
 mainWin, 292
 rejectChanges, 290
 set_label_visibility, 290
 set_spinbox_visibility, 290
Settings_Dialog, 284

sliderQSnapApertureSizeValueChanged, 291
sliderQSnapLocatorSizeValueChanged, 291
sliderSelectionGripSizeValueChanged, 291
sliderSelectionPickBoxSizeValueChanged, 291
spinBoxDisplaySelectBoxAlphaValueChanged,
 291
spinBoxPromptFontSizeValueChanged, 291
spinBoxRecentMaxFilesValueChanged, 291
spinBoxRulerPixelSizeValueChanged, 291
spinBoxTrimDstNumJumpsValueChanged, 291
spinBoxZoomScaleInValueChanged, 291
spinBoxZoomScaleOutValueChanged, 291
tabWidget, 292
settingsDialog
 MainWindow, 200
SettingsDir
 mainwindow-settings.cpp, 380
settingsGrid
 StatusBarButton, 294
settingsLwt
 StatusBarButton, 294
settingsMenu
 MainWindow, 207
settingsOrtho
 StatusBarButton, 295
SettingsPath
 mainwindow-settings.cpp, 380
settingsPolar
 StatusBarButton, 295
settingsPrompt
 MainWindow, 200
settingsQSnap
 StatusBarButton, 295
settingsQTrack
 StatusBarButton, 295
settingsRuler
 StatusBarButton, 295
settingsSnap
 StatusBarButton, 295
setUndoCleanIcon
 MainWindow, 201
setViewBackgroundColor
 MdiWindow, 221
setViewCrossHairColor
 MdiWindow, 221
setViewGridColor
 MdiWindow, 221
setViewRulerColor
 MdiWindow, 221
setViewSelectBoxColors
 MdiWindow, 221
sew, 16, 533
sewDecode
 format_sew.c, 533
shape
 BaseObject, 80
shift_hold
 Settings_, 280
shiftKeyPressedState
 MainWindow, 207
shiftPressed
 CmdPrompt, 93
 CmdPromptInput, 105
shiftReleased
 CmdPrompt, 94
 CmdPromptInput, 105
shine_color
 Settings_, 280
shortcut
 Action_, 60
show_about_dialog
 Settings_, 280
show_details_dialog
 Settings_, 280
show_editor
 Settings_, 280
show_open_file_dialog
 Settings_, 280
show_settings_editor
 Settings_, 280
showGroups
 PropertyEditor, 254
showOneType
 PropertyEditor, 254
showScrollBars
 View, 325
showSettings
 CmdPrompt, 94
 CmdPromptInput, 105
showViewScrollBars
 MdiWindow, 221
shv, 16, 534
shv_thread
 embroidery.h, 417
shvDecode
 format_shv.c, 534
shvDecodeShort
 format_shv.c, 534
shvThreadCount
 embroidery.h, 436
 thread-color.c, 577
shvThreads
 embroidery.h, 436
 thread-color.c, 577
side1
 EmbSatinOutline_, 143
side2
 EmbSatinOutline_, 143
Sierra_Expanded, 471, 516
Sigma_Polyester
 embroidery.h, 417
signalMapper
 PropertyEditor, 257
signature
 _bcf_file_header, 56
sigVersion

ThredHeader_, 305
 simulate
 EmbView_, 151
 simulation_start
 EmbView_, 151
 Singer, 506, 544
 SINGLE_LINE_TEXT_MODE_JUSTIFY
 embroidermodder.h, 343
 SINGLE_LINE_TEXT_MODE_RAPID
 embroidermodder.h, 343
 SINGLE_LINE_TEXT_MODE_SETFONT
 embroidermodder.h, 343
 SINGLE_LINE_TEXT_MODE_SETGEOM
 embroidermodder.h, 343
 sizeHint
 MdiWindow, 221
 sizeOfChainingEntryAtEndOfDifatSector
 main.c, 570
 sizeOfDifatEntry
 main.c, 570
 sizeOfDirectoryEntry
 main.c, 570
 sizeOfFatEntry
 main.c, 570
 sliderQSnapApertureSizeValueChanged
 Settings_Dialog, 291
 sliderQSnapLocatorSizeValueChanged
 Settings_Dialog, 291
 sliderSelectionGripSizeValueChanged
 Settings_Dialog, 291
 sliderSelectionPickBoxSizeValueChanged
 Settings_Dialog, 291
 snap_mode
 EmbView_, 151
 SNOWFLAKE_MODE_NUM_POINTS
 embroidermodder.h, 343
 SNOWFLAKE_MODE_XSCALE
 embroidermodder.h, 344
 SNOWFLAKE_MODE_YSCALE
 embroidermodder.h, 344
 someInt
 SubDescriptor_, 297
 someNum
 SubDescriptor_, 297
 someOtherInt
 SubDescriptor_, 297
 SPARE_RUBBER_OFF
 embroidermodder.h, 343
 SPARE_RUBBER_PATH
 embroidermodder.h, 343
 SPARE_RUBBER_POLYGON
 embroidermodder.h, 343
 SPARE_RUBBER_POLYLINE
 embroidermodder.h, 343
 SPARE_RUBBER_VALUES
 embroidermodder.h, 343
 spareRubber
 View, 325
 spareRubberList
 View, 330
 spinBoxDisplaySelectBoxAlphaValueChanged
 Settings_Dialog, 291
 spinBoxPromptFontSizeValueChanged
 Settings_Dialog, 291
 spinBoxRecentMaxFilesValueChanged
 Settings_Dialog, 291
 spinBoxRulerPixelSizeValueChanged
 Settings_Dialog, 291
 spinBoxTrimDstNumJumpsValueChanged
 Settings_Dialog, 291
 spinBoxZoomScaleInValueChanged
 Settings_Dialog, 291
 spinBoxZoomScaleOutValueChanged
 Settings_Dialog, 291
 spline
 EmbGeometry_, 134
 sst, 16, 475, 535
 STAR_MODE_CENTER_PT
 embroidermodder.h, 343
 STAR_MODE_NUM_POINTS
 embroidermodder.h, 343
 STAR_MODE_RAD_INNER
 embroidermodder.h, 343
 STAR_MODE_RAD_OUTER
 embroidermodder.h, 343
 start
 EmbArc_, 124
 EmbBezier_, 126
 EmbLine_, 137
 startBlinking
 CmdPrompt, 94
 startCommand
 CmdPrompt, 94
 CmdPromptInput, 105
 startGripping
 View, 326
 startingSectorLocation
 _bcf_directory_entry, 52
 startResizeHistory
 CmdPromptHistory, 99
 startResizingTheHistory
 CmdPrompt, 94
 stateBits
 _bcf_directory_entry, 52
 StatusBar, 292
 setMouseCoord, 292
 StatusBar, 292
 statusBarGridButton, 292
 statusBarLwtButton, 293
 statusBarMouseCoord, 293
 statusBarOrthoButton, 293
 statusBarPolarButton, 293
 statusBarQSnapButton, 293
 statusBarQTrackButton, 293
 statusBarRulerButton, 293
 statusBarSnapButton, 293

statusbar
 MainWindow, 207
 StatusBarButton, 296
StatusBarButton, 293
 contextMenuEvent, 294
 disableLwt, 294
 disableReal, 294
 enableLwt, 294
 enableReal, 294
 mainWin, 295
 settingsGrid, 294
 settingsLwt, 294
 settingsOrtho, 295
 settingsPolar, 295
 settingsQSnap, 295
 settingsQTrack, 295
 settingsRuler, 295
 settingsSnap, 295
 statusbar, 296
 StatusBarButton, 294
 toggleGrid, 295
 toggleLwt, 295
 toggleOrtho, 295
 togglePolar, 295
 toggleQSnap, 295
 toggleQTrack, 295
 toggleRuler, 295
 toggleSnap, 295
statusBarGridButton
 StatusBar, 292
statusBarLwtButton
 StatusBar, 293
statusBarMouseCoord
 StatusBar, 293
statusBarOrthoButton
 StatusBar, 293
statusBarPolarButton
 StatusBar, 293
statusBarQSnapButton
 StatusBar, 293
statusBarQTrackButton
 StatusBar, 293
statusBarRulerButton
 StatusBar, 293
statusBarSnapButton
 StatusBar, 293
statustip
 Action__, 60
stitch
 EmbArray__, 125
 EmbGeometry__, 134
stitch_list
 EmbPattern__, 140
stitch_time
 Settings__, 280
stitchesJump
 EmbDetailsDialog, 130
stitchesReal
 EmbDetailsDialog, 130
stitchesTotal
 EmbDetailsDialog, 130
stitchesTrim
 EmbDetailsDialog, 130
stitchGranularity
 ThredExtension_, 304
STOP
 embroidery.h, 417
stopBlinking
 CmdPrompt, 94
 CmdPromptInput, 105
stopGripping
 View, 326
stopResizeHistory
 CmdPromptHistory, 99
stopResizingTheHistory
 CmdPrompt, 94
streamSize
 _bcf_directory_entry, 52
streamSizeHigh
 _bcf_directory_entry, 52
stringInArray
 embroidery_internal.h, 477
 main.c, 569
stringVal
 VipHeader_, 331
stub_implement
 MainWindow, 201
stub_testing
 MainWindow, 201
stx, 535
stxColor
 StxThread_, 296
stxReadThread
 format_stx.c, 535
StxThread
 embroidery_internal.h, 460
StxThread_
 colorCode, 296
 colorName, 296
 sectionName, 296
 stxColor, 296
 subDescriptors, 296
styleHash
 CmdPrompt, 95
SubDescriptor
 embroidery_internal.h, 460
SubDescriptor_
 colorCode, 296
 colorName, 297
 someInt, 297
 someNum, 297
 someOtherInt, 297
subDescriptors
 StxThread_, 296
subPathList
 TextSingleObject, 302

SULKY_RAYON
 embroidery.h, 417

SUNSTAR, 475, 535

SVG, 16, 536

SVG_ATTRIBUTE
 embroidery_internal.h, 458

SVG_CATCH_ALL
 embroidery_internal.h, 458

SVG_Colors
 embroidery.h, 417

SVG_CREATOR_EMBROIDERMODDER
 embroidery_internal.h, 458

SVG_CREATOR_ILLUSTRATOR
 embroidery_internal.h, 458

SVG_CREATOR_INKSCAPE
 embroidery_internal.h, 458

SVG_CREATOR_NULL
 embroidery_internal.h, 459

SVG_ELEMENT
 embroidery_internal.h, 459

SVG_EXPECT_ATTRIBUTE
 embroidery_internal.h, 459

SVG_EXPECT_ELEMENT
 embroidery_internal.h, 459

SVG_EXPECT_NULL
 embroidery_internal.h, 459

SVG_EXPECT_VALUE
 embroidery_internal.h, 459

SVG_MEDIA_PROPERTY
 embroidery_internal.h, 459

SVG_NULL
 embroidery_internal.h, 459

SVG_PROPERTY
 embroidery_internal.h, 459

SvgAttribute
 embroidery_internal.h, 460

SvgAttribute_, 297
 name, 297
 value, 297

svgCreator
 format_svg.c, 536

svgExpect
 format_svg.c, 536

svgMultiValue
 format_svg.c, 536

t01, 475, 537

t09, 475, 537

table
 Huffman, 154

table_width
 Huffman, 154

tabPressed
 CmdPrompt, 94
 CmdPromptInput, 105

tabWidget
 Settings_Dialog, 292

Tajima, 510

tap, 16, 538

tempArcObj
 property-editor.cpp, 393

tempBaseObj
 View, 330

tempBlockObj
 property-editor.cpp, 393

tempCircleObj
 property-editor.cpp, 394

tempDimAlignedObj
 property-editor.cpp, 394

tempDimAngularObj
 property-editor.cpp, 394

tempDimArcLenObj
 property-editor.cpp, 394

tempDimDiamObj
 property-editor.cpp, 394

tempDimLeaderObj
 property-editor.cpp, 394

tempDimLinearObj
 property-editor.cpp, 394

tempDimOrdObj
 property-editor.cpp, 394

tempDimRadiusObj
 property-editor.cpp, 394

tempEllipseArcObj
 property-editor.cpp, 394

tempEllipseObj
 property-editor.cpp, 394

tempHatchObj
 property-editor.cpp, 394

tempImageObj
 property-editor.cpp, 394

tempInflLineObj
 property-editor.cpp, 394

tempLineObj
 property-editor.cpp, 394

tempPathObj
 property-editor.cpp, 394

tempPointObj
 property-editor.cpp, 394

tempPolygonObj
 property-editor.cpp, 394

tempPolylineObj
 property-editor.cpp, 395

tempRayObj
 property-editor.cpp, 395

tempRectObj
 property-editor.cpp, 395

tempSplineObj
 property-editor.cpp, 395

tempTextMultiObj
 property-editor.cpp, 395

tempTextSingleObj
 property-editor.cpp, 395

testEmbCircle
 embroidery_internal.h, 477

testEmbCircle_2
 embroidery_internal.h, 477

testEmbFormat
 embroidery_internal.h, 477

testGeomArc
 embroidery_internal.h, 477

testing
 Settings_, 280

testMain
 embroidery.h, 435

testTangentPoints
 embroidery_internal.h, 477

testThreadColor
 embroidery_internal.h, 477

text
 EmbTextMulti_, 145
 EmbTextSingle_, 146
 UiObject_, 307

text.c
 textSingle_gripEdit, 554
 textSingle_mouseSnapPoint, 554
 textSingle_paint, 555
 textSingle_setJustify, 555
 textSingleSetTextBackward, 555
 textSingle_setTextBold, 555
 textSingle_setTextFont, 555
 textSingle_setTextItalic, 555
 textSingle_setTextOverline, 555
 textSingle_setTextSize, 555
 textSingle_setTextStrikeOut, 555
 textSingle_setTextStyle, 555
 textSingle_setTextUnderline, 555
 textSingle_setTextUpsideDown, 555
 textSingle_updateRubber, 555

text_angle
 EmbView_, 151
 Settings_, 280

text_font
 EmbView_, 151
 Settings_, 280

text_size
 EmbView_, 151
 Settings_, 281

text_style_bold
 EmbView_, 151
 Settings_, 281

text_style_italic
 EmbView_, 151
 Settings_, 281

text_style_overline
 EmbView_, 152
 Settings_, 281

text_style_strikeout
 EmbView_, 152
 Settings_, 281

text_style_underline
 EmbView_, 152
 Settings_, 281

textFont
 UiObject_, 307

textFontSelector
 MainWindow, 207

textFontSelectorCurrentFontChanged
 MainWindow, 201

textHeight
 UiObject_, 307

textJustify
 UiObject_, 307

textRotation
 UiObject_, 307

textSingle_gripEdit
 text.c, 554

textSingle_mouseSnapPoint
 text.c, 554

textSingle_paint
 text.c, 555

textSingle_setJustify
 text.c, 555

textSingle_setTextBackward
 text.c, 555

textSingle_setTextBold
 text.c, 555

textSingle_setTextFont
 text.c, 555

textSingle_setTextItalic
 text.c, 555

textSingle_setTextOverline
 text.c, 555

textSingle_setTextSize
 text.c, 555

textSingle_setTextStrikeOut
 text.c, 555

textSingle_setTextStyle
 text.c, 555

textSingle_setTextUnderline
 text.c, 555

textSingle_setTextUpsideDown
 text.c, 555

textSingle_updateRubber
 text.c, 555

TextSingleObject, 297
 ~TextSingleObject, 300
 allGripPoints, 300
 gripEdit, 300
 init, 300
 mouseSnapPoint, 300
 objectPos, 300
 objectSavePathList, 301
 objectTextJustifyList, 301
 objectX, 301
 objectY, 301
 objText, 302
 objTextBackward, 302
 objTextBold, 303
 objTextFont, 303
 objTextItalic, 303
 objTextJustify, 303
 objTextOverline, 303

objTextPath, 303
 objTextSize, 303
 objTextStrikeOut, 303
 objTextUnderline, 303
 objTextUpsideDown, 303
 paint, 301
 setObjectPos, 301
 setObjectText, 301
 setObjectTextBackward, 301
 setObjectTextBold, 301
 setObjectTextFont, 301
 setObjectTextItalic, 301
 setObjectTextJustify, 301
 setObjectTextOverline, 301
 setObjectTextSize, 302
 setObjectTextStrikeOut, 302
 setObjectTextStyle, 302
 setObjectTextUnderline, 302
 setObjectTextUpsideDown, 302
 setObjectX, 302
 setObjectY, 302
 subPathList, 302
 TextSingleObject, 300
 Type, 300
 type, 302
 updateRubber, 302
 vulcanize, 302
 textSizeSelector
 MainWindow, 207
 textSizeSelectorIndexChanged
 MainWindow, 201
 thr, 476, 539
 thread
 EmbArray_, 125
 EmbGeometry_, 134
 thread-color.c
 _dxsetColorTable, 576
 brand_codes, 576
 brand_codes_files, 576
 husThreads, 576
 jefThreads, 576
 pcmThreads, 576
 pecThreadCount, 576
 pecThreads, 577
 shvThreadCount, 577
 shvThreads, 577
 threadColor, 576
 threadColorName, 576
 threadColorNum, 576
 thread_color
 embroidery.h, 420
 thread_color_
 hex_code, 303
 manufacturer_code, 303
 name, 303
 thread_list
 EmbPattern_, 140
 ThreadArt_Polyester
 embroidery.h, 417
 ThreadArt_Rayon
 embroidery.h, 417
 threadColor
 embroidery.h, 435
 thread-color.c, 576
 threadColorName
 embroidery.h, 435
 thread-color.c, 576
 threadColorNum
 embroidery.h, 435
 thread-color.c, 576
 ThreaDelight_Polyester
 embroidery.h, 418
 threadLength
 _vp3Hoop, 58
 ThreadWorks, 476, 539
 ThredExtension
 embroidery_internal.h, 460
 ThredExtension_
 auxFormat, 304
 creatorName, 304
 hoopX, 304
 hoopY, 304
 modifierName, 304
 reserved, 304
 stitchGranularity, 304
 ThredHeader
 embroidery_internal.h, 460
 ThredHeader_
 hoopSize, 305
 length, 305
 numStiches, 305
 reserved, 305
 sigVersion, 305
 threshold_method
 fill.c, 497
 Tick
 DimLeaderObject, 112
 tick_depth
 Settings_, 281
 ticks_color
 Settings_, 281
 tile
 MdiArea, 212
 tipOfDay
 MainWindow, 201
 tmpHeight
 CmdPromptHistory, 99
 to_EmbVector
 embroidermodder.h, 344
 to_open
 Settings_, 281
 to_QPointF
 embroidermodder.h, 344
 to_string_vector
 mainwindow-settings.cpp, 380
 toCenter

UndoableNavCommand, 313
Todo
 mainwindow.cpp, 384
toggleGrid
 MainWindow, 201
 StatusBarButton, 295
 View, 326
toggleLwt
 MainWindow, 201
 StatusBarButton, 295
 View, 326
toggleOrtho
 StatusBarButton, 295
 View, 326
togglePickAddMode
 PropertyEditor, 255
togglePolar
 StatusBarButton, 295
 View, 326
toggleQSnap
 StatusBarButton, 295
 View, 326
toggleQTrack
 StatusBarButton, 295
 View, 326
toggleReal
 View, 326
toggleRuler
 MainWindow, 201
 StatusBarButton, 295
 View, 326
toggleSnap
 StatusBarButton, 295
 View, 326
tokenize
 mainwindow.cpp, 384
toolbar_name
 Action __, 60
toolbar_position
 Action __, 60
toolbarEdit
 MainWindow, 207
toolbarFile
 MainWindow, 207
toolbarHash
 MainWindow, 207
toolbarHelp
 MainWindow, 208
toolbarIcon
 MainWindow, 208
toolbarLayer
 MainWindow, 208
toolbarPan
 MainWindow, 208
toolbarPrompt
 MainWindow, 208
toolbarProperties
 MainWindow, 208
 toolbarText
 MainWindow, 208
 toolbarView
 MainWindow, 208
 toolbarZoom
 MainWindow, 208
 toolButtonArcClockwise
 property-editor.cpp, 395
 toolButtonBlockX
 property-editor.cpp, 395
 toolButtonBlockY
 property-editor.cpp, 395
 toolButtonImageHeight
 property-editor.cpp, 395
 toolButtonImageName
 property-editor.cpp, 395
 toolButtonImagePath
 property-editor.cpp, 395
 toolButtonImageWidth
 property-editor.cpp, 395
 toolButtonImageX
 property-editor.cpp, 395
 toolButtonImageY
 property-editor.cpp, 395
 toolButtonInfiniteLineVectorX
 property-editor.cpp, 395
 toolButtonInfiniteLineVectorY
 property-editor.cpp, 395
 toolButtonInfiniteLineX2
 property-editor.cpp, 395
 toolButtonInfiniteLineY2
 property-editor.cpp, 396
 toolButtonLineAngle
 property-editor.cpp, 396
 toolButtonLineDeltaX
 property-editor.cpp, 396
 toolButtonLineDeltaY
 property-editor.cpp, 396
 toolButtonLineEndX
 property-editor.cpp, 396
 toolButtonLineEndY
 property-editor.cpp, 396
 toolButtonLineLength
 property-editor.cpp, 396
 toolButtonLineStartX
 property-editor.cpp, 396
 toolButtonLineStartY
 property-editor.cpp, 396
 toolButtonPickAdd
 PropertyEditor, 257
 toolButtonPolygonDiameterSide
 property-editor.cpp, 396
 toolButtonPolygonDiameterVertex
 property-editor.cpp, 396
 toolButtonPolygonInteriorAngle
 property-editor.cpp, 396
 toolButtonPolygonRadiusSide
 property-editor.cpp, 396

toolButtonPolygonRadiusVertex
 property-editor.cpp, 396

toolButtonPolylineArea
 property-editor.cpp, 396

toolButtonPolylineClosed
 property-editor.cpp, 396

toolButtonPolylineLength
 property-editor.cpp, 396

toolButtonPolylineVertexNum
 property-editor.cpp, 396

toolButtonPolylineVertexX
 property-editor.cpp, 397

toolButtonPolylineVertexY
 property-editor.cpp, 397

toolButtonQSelect
 PropertyEditor, 257

toolButtonRayVectorX
 property-editor.cpp, 397

toolButtonRayVectorY
 property-editor.cpp, 397

toolButtonRayX1
 property-editor.cpp, 397

toolButtonRayX2
 property-editor.cpp, 397

toolButtonRayY1
 property-editor.cpp, 397

toolButtonRayY2
 property-editor.cpp, 397

toolButtons
 data.cpp, 333
 embroidermodder.h, 346

toolButtonTextMultiX
 property-editor.cpp, 397

toolButtonTextMultiY
 property-editor.cpp, 397

toolButtonTextSingleContents
 property-editor.cpp, 397

toolButtonTextSingleFont
 property-editor.cpp, 397

toolButtonTextSingleHeight
 property-editor.cpp, 397

toolButtonTextSingleJustify
 property-editor.cpp, 397

toolButtonTextSingleRotation
 property-editor.cpp, 397

tooltip
 Action __, 61

top
 _vp3Hoop, 58
 EmbRect __, 143
 hoop_padding, 153

top2
 _vp3Hoop, 59

toPolyline
 SaveObject, 269

toTransform
 UndoableNavCommand, 313

Toyota, 502

transactionSignatureNumber
 _bcf_file_header, 57

treeView
 LayerManager, 164

TRIM
 embroidery.h, 418

txt, 476, 539

Type
 ArcObject, 66
 BaseObject, 76
 CircleObject, 84
 DimLeaderObject, 112
 EllipseObject, 119
 ImageObject, 157
 LineObject, 166
 PathObject, 226
 PointObject, 231
 PolygonObject, 236
 PolylineObject, 241
 RectObject, 260
 TextSingleObject, 300

type
 ArcObject, 73
 BaseObject, 80
 CircleObject, 86
 DimLeaderObject, 115
 EllipseObject, 122
 EmbArray __, 125
 EmbFormatList __, 132
 EmbGeometry __, 134
 GroupBoxData __, 153
 ImageObject, 159
 LineObject, 169
 PathObject, 228
 PointObject, 233
 PolygonObject, 238
 PolylineObject, 243
 RectObject, 262
 TextSingleObject, 302
 UiObject __, 307

u00, 476, 540

u01, 16, 476, 540

ui_mode
 EmbView __, 152

UiMode
 embroidermodder.h, 343

UiObject
 embroidermodder.h, 339

UiObject __, 305
 center, 306
 color, 306
 command, 306
 controlPointLabels, 306
 controlPoints, 306
 firstRun, 306
 fname, 306
 id, 306
 maxPoints, 306

minPoints, 306
mode, 306
numPoints, 306
object_index, 306
path_desc, 306
pattern_index, 306
rotation, 306
scale, 306
selectable, 306
text, 307
textFont, 307
textHeight, 307
textJustify, 307
textRotation, 307
type, 307
undo
 MainWindow, 201
 UndoableAddCommand, 307
 UndoableDeleteCommand, 308
 UndoableGripEditCommand, 309
 UndoableMirrorCommand, 310
 UndoableMoveCommand, 311
 UndoableNavCommand, 312
 UndoableRotateCommand, 313
 UndoableScaleCommand, 315
 UndoEditor, 316
undo_history
 EmbView_, 152
UndoableAddCommand, 307
 gview, 308
 object, 308
 redo, 307
 undo, 307
 UndoableAddCommand, 307
UndoableDeleteCommand, 308
 gview, 308
 object, 308
 redo, 308
 undo, 308
 UndoableDeleteCommand, 308
UndoableGripEditCommand, 309
 after, 309
 before, 309
 gview, 309
 object, 309
 redo, 309
 undo, 309
 UndoableGripEditCommand, 309
UndoableMirrorCommand, 310
 gview, 310
 mirror, 310
 mirrorLine, 310
 object, 310
 redo, 310
 undo, 310
 UndoableMirrorCommand, 310
UndoableMoveCommand, 311
 dx, 311
dy, 311
gview, 311
object, 311
redo, 311
undo, 311
UndoableMoveCommand, 311
UndoableNavCommand, 312
 done, 312
 fromCenter, 312
 fromTransform, 312
 gview, 313
 id, 312
 mergeWith, 312
 navType, 313
 redo, 312
 toCenter, 313
 toTransform, 313
 undo, 312
 UndoableNavCommand, 312
UndoableRotateCommand, 313
 angle, 314
 gview, 314
 object, 314
 pivotX, 314
 pivotY, 314
 redo, 313
 rotate, 313
 undo, 314
 UndoableRotateCommand, 313
UndoableScaleCommand, 314
 dx, 315
 dy, 315
 factor, 315
 gview, 315
 object, 315
 redo, 315
 undo, 315
 UndoableScaleCommand, 314
UndoEditor, 315
 ~UndoEditor, 316
 addStack, 316
 canRedo, 316
 canUndo, 316
 focusWidget, 316
 iconDir, 317
 iconSize, 317
 redo, 316
 redoText, 316
 undo, 316
 UndoEditor, 316
 undoGroup, 317
 undoText, 316
 undoView, 317
 updateCleanIcon, 316
undoGroup
 UndoEditor, 317
undoPressed
 CmdPrompt, 94

CmdPromptInput, 105
 undoStack
 View, 330
 undoText
 UndoEditor, 316
 undoView
 UndoEditor, 317
 unknown
 VipHeader_, 331
 unknown2
 _vp3Hoop, 59
 unknown3
 _vp3Hoop, 59
 unknown4
 _vp3Hoop, 59
 updateAllViewBackgroundColors
 MainWindow, 202
 updateAllViewCrossHairColors
 MainWindow, 202
 updateAllViewGridColors
 MainWindow, 202
 updateAllViewRulerColors
 MainWindow, 202
 updateAllViewScrollBars
 MainWindow, 202
 updateAllViewSelectBoxColors
 MainWindow, 202
 updateArcRect
 ArcObject, 73
 updateCleanIcon
 UndoEditor, 316
 updateColorLinetypeLineweight
 MdiWindow, 222
 updateComboBoxBoolIfVaries
 PropertyEditor, 255
 updateComboBoxStrIfVaries
 PropertyEditor, 255
 updateCurrentText
 CmdPromptInput, 105
 updateFontComboBoxStrIfVaries
 PropertyEditor, 255
 updateLeader
 DimLeaderObject, 116
 updateLineEditNumIfVaries
 PropertyEditor, 255
 updateLineEditStrIfVaries
 PropertyEditor, 255
 updateMenuToolbarStatusbar
 MainWindow, 202
 updateMouseCoords
 View, 326
 updatePath
 ArcObject, 74
 CircleObject, 86
 EllipseObject, 122
 ImageObject, 159
 PathObject, 228
 PolygonObject, 238
 PolylineObject, 243
 RectObject, 262
 updatePickAddMode
 MainWindow, 202
 updatePickAddModeButton
 PropertyEditor, 255
 updateRubber
 ArcObject, 74
 CircleObject, 86
 DimLeaderObject, 116
 EllipseObject, 122
 ImageObject, 159
 LineObject, 169
 PathObject, 228
 PointObject, 233
 PolygonObject, 238
 PolylineObject, 244
 RectObject, 262
 TextSingleObject, 302
 updateStyle
 CmdPrompt, 94
 upPressed
 CmdPrompt, 94
 CmdPromptInput, 106
 usage
 embroidermodder.cpp, 334
 use_translation
 Settings_, 281
 useBackgroundColor
 MdiArea, 212
 useBackgroundLogo
 MdiArea, 212
 useBackgroundTexture
 MdiArea, 212
 useColor
 MdiArea, 213
 useLogo
 MdiArea, 213
 useTexture
 MdiArea, 213
 validFileFormat
 MainWindow, 202
 validRGB
 mainwindow.cpp, 384
 value
 SvgAttribute_, 297
 vector
 EmbGeometry_, 134
 vector.c
 embVector_add, 556
 embVector_angle, 556
 embVector_average, 556
 embVector_cross, 556
 embVector_distance, 557
 embVector_dot, 557
 embVector_length, 557
 embVector_multiply, 557
 embVector_normalize, 557

embVector_relativeX, 557
embVector_relativeY, 558
embVector_subtract, 558
embVector_transpose_product, 558
embVector_unit, 558
version
 embroidermodder.cpp, 334
 Settings_, 281
View, 317
 ~View, 320
 addObject, 320
 addToRubberRoom, 320
 alignScenePointWithViewPoint, 321
 allowRubber, 321
 allowZoomIn, 321
 allowZoomOut, 321
 center, 321
 centerAt, 321
 clearRubberRoom, 321
 clearSelection, 321
 contextMenuEvent, 321
 copy, 321
 copySelected, 321
 cornerButtonClicked, 321
 createGrid, 321
 createGridIso, 321
 createGridPolar, 321
 createGridRect, 321
 createObjectList, 322
 createOrigin, 322
 createRulerTextPath, 322
 crosshairColor, 327
 crosshairSize, 327
 cut, 322
 cutCopyMousePoint, 327
 deleteObject, 322
 deletePressed, 322
 deleteSelected, 322
 disableMoveRapidFire, 322
 drawBackground, 322
 drawForeground, 322
 enableMoveRapidFire, 322
 enterEvent, 322
 escapePressed, 322
 getUndoStack, 322
 gridColor, 327
 gridPath, 327
 gripBaseObj, 328
 gripColorCool, 328
 gripColorHot, 328
 grippingActive, 328
 gripSize, 328
 gscene, 328
 hashDeletedObjects, 328
 isLwtEnabled, 322
 isRealEnabled, 323
 loadRulerSettings, 323
 mainWin, 328
 mirrorSelected, 323
 mouseDoubleClickEvent, 323
 mouseMoveEvent, 323
 mousePressEvent, 323
 mouseReleaseEvent, 323
 moveAction, 323
 movePoint, 328
 moveSelected, 323
 movingActive, 328
 numSelected, 323
 originPath, 328
 panDistance, 328
 panDown, 323
 panLeft, 323
 panningActive, 328
 panningPointActive, 328
 panningRealTimeActive, 328
 panPoint, 323
 panRealTime, 323
 panRight, 324
 panStart, 324
 panStartX, 328
 panStartY, 328
 panUp, 324
 paste, 324
 pasteDelta, 328
 pasteObjectItemGroup, 329
 pastingActive, 329
 pickBoxSize, 329
 pressPoint, 329
 previewActive, 329
 previewData, 329
 previewMode, 329
 previewObjectItemGroup, 329
 previewObjectList, 329
 previewOff, 324
 previewOn, 324
 previewPoint, 329
 qSnapActive, 329
 qsnapApertureSize, 329
 qsnapLocatorColor, 329
 qsnapLocatorSize, 329
 qSnapToggle, 329
 rapidMoveActive, 329
 recalculateLimits, 324
 releasePoint, 329
 repeatAction, 324
 rotateAction, 324
 rotateSelected, 324
 roundToMultiple, 324
 rubberRoomList, 329
 rulerColor, 330
 rulerMetric, 330
 rulerPixelSize, 330
 scaleAction, 324
 scaleSelected, 324
 sceneGripPoint, 330
 sceneMousePoint, 330

sceneMovePoint, 330
scenePressPoint, 330
sceneReleasePoint, 330
selectAll, 324
selectBox, 330
selectingActive, 330
selectionChanged, 325
setBackgroundColor, 325
setCornerButton, 325
setCrossHairColor, 325
setCrossHairSize, 325
setGridColor, 325
setRubberMode, 325
setRubberPoint, 325
setRubberText, 325
setRulerColor, 325
setSelectBoxColors, 325
showScrollBars, 325
spareRubber, 325
spareRubberList, 330
startGripping, 326
stopGripping, 326
tempBaseObj, 330
toggleGrid, 326
toggleLwt, 326
toggleOrtho, 326
togglePolar, 326
toggleQSnap, 326
toggleQTrack, 326
toggleReal, 326
toggleRuler, 326
toggleSnap, 326
undoStack, 330
updateMouseCoords, 326
View, 320
view_state, 330
viewMousePoint, 330
vulcanizeObject, 326
vulcanizeRubberRoom, 327
wheelEvent, 327
willOverflowInt32, 327
willUnderflowInt32, 327
zoomExtents, 327
zoomIn, 327
zoomOut, 327
zoomSelected, 327
zoomToPoint, 327
zoomWindow, 327
zoomWindowActive, 330
view_state
 View, 330
view_toolbar
 embroidermodder.h, 346
 mainwindow-toolbars.cpp, 382
viewMenu
 MainWindow, 208
viewMousePoint
 View, 330
vip, 16, 436, 542
vipCompressData
 format_vip.c, 541
vipDecodeByte
 format_vip.c, 541
vipDecodeStitchType
 format_vip.c, 541
vipDecodingTable
 embroidery.h, 436
 format_vip.c, 541
vipDecompressData
 format_vip.c, 541
vipEncodeByte
 format_vip.c, 541
vipEncodeStitchType
 format_vip.c, 541
VipHeader
 embroidery_internal.h, 460
VipHeader_, 331
 attributeOffset, 331
 colorLength, 331
 magicCode, 331
 negativeXHoopSize, 331
 negativeYHoopSize, 331
 numberOfColors, 331
 numberOfStitches, 331
 postitiveXHoopSize, 331
 postitiveYHoopSize, 331
 stringVal, 331
 unknown, 331
 xOffset, 331
 yOffset, 332
vp3, 16, 543
vp3Decode
 format_vp3.c, 542
vp3DecodeInt16
 format_vp3.c, 542
vp3Hoop
 embroidery_internal.h, 460
vp3PatchByteCount
 format_vp3.c, 542
vp3ReadHoopSection
 format_vp3.c, 542
vp3ReadString
 format_vp3.c, 543
vp3WriteString
 format_vp3.c, 543
vp3WriteStringLen
 format_vp3.c, 543
vulcanize
 ArcObject, 74
 BaseObject, 80
 CircleObject, 86
 DimLeaderObject, 116
 EllipseObject, 122
 ImageObject, 159
 LineObject, 169
 PathObject, 228

PointObject, 233
PolygonObject, 238
PolylineObject, 244
RectObject, 262
TextSingleObject, 302
vulcanizeObject
 View, 326
vulcanizeRubberRoom
 View, 327

whatsThisContextHelp
 MainWindow, 203
wheelEvent
 View, 327
WHITESPACE
 main.c, 570
width
 _vp3Hoop, 59
 EmblImage_, 135
willOverflowInt32
 View, 327
willUnderflowInt32
 View, 327
windowMenu
 MainWindow, 208
windowMenuAboutToShow
 MainWindow, 203
windowMenuActivated
 MainWindow, 203
wizardTipOfTheDay
 MainWindow, 209
write100
 embroidery_internal.h, 477
 format_100.c, 502
write10o
 embroidery_internal.h, 477
 format_10o.c, 503
write_24bit
 embroidery_internal.h, 477
 encoding.c, 493
 main.c, 569
write_external_color_file
 EmbFormatList_, 132
write_setting
 mainwindow-settings.cpp, 380, 381
write_settings
 embroidermodder.h, 344
writeArt
 embroidery_internal.h, 478
 format_art.c, 503
writeBmc
 embroidery_internal.h, 478
 format_bmc.c, 504
writeBro
 embroidery_internal.h, 478
 format_bro.c, 504
writeCnd
 embroidery_internal.h, 478
 format_cnd.c, 505
writeCol
 embroidery_internal.h, 478
 format_col.c, 506
writeCsd
 embroidery_internal.h, 478
 format_csd.c, 507
writeCsv
 embroidery_internal.h, 478
 format_csv.c, 508
writeDat
 embroidery_internal.h, 478
 format_dat.c, 508
writeDem
 embroidery_internal.h, 478
 format_dem.c, 509
writeDsb
 embroidery_internal.h, 478
 format_dsb.c, 510
writeDst
 embroidery_internal.h, 478
 format_dst.c, 512
writeDsz
 embroidery_internal.h, 479
 format_dsz.c, 512
writeDxf
 embroidery_internal.h, 479
 format_dxf.c, 513
writeEdr
 embroidery_internal.h, 479
 format_edr.c, 513
writeEmd
 embroidery_internal.h, 479
 format_emd.c, 514
writeExp
 embroidery_internal.h, 479
 format_exp.c, 515
writeExy
 embroidery_internal.h, 479
 format_exy.c, 515
writeEys
 embroidery_internal.h, 479
 format_eys.c, 516
writeFxy
 embroidery_internal.h, 479
 format_fxy.c, 516
writeGc
 embroidery_internal.h, 479
 format_gc.c, 517
writeGnc
 embroidery_internal.h, 479
 format_gnc.c, 517
writeGt
 embroidery_internal.h, 479
 format_gt.c, 518
writeHus
 embroidery_internal.h, 480
 format_hus.c, 519
writeImage

format_pec.c, 527
image.c, 559
writeInb
 embroidery_internal.h, 480
 format_inb.c, 519
writeInf
 embroidery_internal.h, 480
 format_inf.c, 520
writeJef
 embroidery_internal.h, 480
 format_jef.c, 521
writeKsm
 embroidery_internal.h, 480
 format_ksm.c, 521
writeMax
 embroidery_internal.h, 480
 format_max.c, 522
writeMit
 embroidery_internal.h, 480
 format_mit.c, 523
writeNew
 embroidery_internal.h, 480
 format_new.c, 523
writeOfm
 embroidery_internal.h, 480
 format_ofm.c, 524
writePcd
 embroidery_internal.h, 480
 format_pcd.c, 525
writePcm
 embroidery_internal.h, 480
 format_pcm.c, 525
writePcq
 embroidery_internal.h, 481
 format_pcq.c, 526
writePcs
 embroidery_internal.h, 481
 format_pcs.c, 526
writePec
 embroidery_internal.h, 481
 format_pec.c, 527
writePecStitches
 embroidery_internal.h, 481
 format_pec.c, 527
writePel
 embroidery_internal.h, 481
 format_pel.c, 528
writePem
 embroidery_internal.h, 481
 format_pem.c, 528
writePes
 embroidery_internal.h, 481
 format_pes.c, 531
writePhb
 embroidery_internal.h, 481
 format_phb.c, 531
writePhc
 embroidery_internal.h, 481
 format_phc.c, 532
writePlt
 embroidery_internal.h, 481
 format_plt.c, 532
writer_state
 EmbFormatList_, 132
writeRgb
 embroidery_internal.h, 482
 format_rgb.c, 533
writeSettings
 MainWindow, 203
writeSew
 embroidery_internal.h, 482
 format_sew.c, 533
writeShv
 embroidery_internal.h, 482
 format_shv.c, 534
writeSst
 embroidery_internal.h, 482
 format_sst.c, 535
writeStx
 embroidery_internal.h, 482
 format_stx.c, 535
writeSvg
 embroidery_internal.h, 482
 format_svg.c, 536
writeT01
 embroidery_internal.h, 482
 format_t01.c, 537
writeT09
 embroidery_internal.h, 482
 format_t09.c, 537
writeTap
 embroidery_internal.h, 482
 format_tap.c, 538
writeThr
 embroidery_internal.h, 482
 format_thr.c, 539
writeTxt
 embroidery_internal.h, 482
 format_txt.c, 539
writeU00
 embroidery_internal.h, 483
 format_u00.c, 540
writeU01
 embroidery_internal.h, 483
 format_u01.c, 540
writeVip
 embroidery_internal.h, 483
 format_vip.c, 541
writeVp3
 embroidery_internal.h, 483
 format_vp3.c, 543
writeXxx
 embroidery_internal.h, 483
 format_xxx.c, 543
writeZsk
 embroidery_internal.h, 483

format_zsk.c, 544

x

- EmbStitch_, 144
- EmbVector_, 148

xOffset

- _vp3Hoop, 59
- VipHeader_, 331

xxx, 16, 544

xxxDecodeByte

- format_xxx.c, 543

xxxEncodeDesign

- format_xxx.c, 544

xxxEncodeStitch

- format_xxx.c, 544

xxxEncodeStop

- format_xxx.c, 544

y

- EmbStitch_, 145
- EmbVector_, 148

year

- EmbTime_, 147

YELLOW_TERM_COLOR

- embroidery_internal.h, 459

yOffset

- _vp3Hoop, 59
- VipHeader_, 332

Z102_Isacord_Polyester

- embroidery.h, 418

zoom_toolbar

- brodermodder.h, 346
- mainwindow-toolbars.cpp, 382

zoomAll

- MainWindow, 203

zoomCenter

- MainWindow, 203

zoomDynamic

- MainWindow, 203

zoomExtents

- MainWindow, 204
- View, 327

zoomExtentsAllSubWindows

- MdiArea, 212

zoomIn

- MainWindow, 204
- View, 327

zoomInLimit

- Settings_, 281

zoomMenu

- MainWindow, 209

zoomOut

- MainWindow, 204
- View, 327

zoomOutLimit

- Settings_, 281

zoomPrevious

- MainWindow, 204