

```

1: unit TMVFiles;
2:
3: {$mode objfpc}{$H+}
4:
5: //=====
6: //
7: //   TMVFiles.pas
8: //
9: //   Calls: AppConstants
10: //           AppVariables
11: //           DataEntry
12: //           StatusBar : DisplayTMVFileStatus
13: //           TMVFiles_DTMF : ParseDTMFRecord
14: //                       MakeDTMFRecord
15: //           TMVFiles_FAV : ParseFAVRecord
16: //                       MakeFAVRecord
17: //           TMVFiles_UHF : ParseUHFRecord
18: //                       MakeUHFRecord
19: //           TMVFiles_VHF : ParseVHFRecord
20: //                       MakeVHFRecord
21: //           Utilities : GetToneFrequencyFromToneNr
22: //
23: //   Called By: Init : Initialize
24: //               Final : Finalize
25: //               Main : TfrmMain.mnuFileNewClick
26: //                       TfrmMain.mnuFileOpenClick
27: //                       TfrmMain.mnuFileSaveClick
28: //                       TfrmMain.mnuFileSaveAsClick
29: //           TMVFiles_VHF : ParseVHFRecord
30: //
31: //   Ver: 1.0.0
32: //
33: //   Date: 28 Aug 2013
34: //
35: //=====
36:
37: interface
38:
39: uses
40:   Classes, Dialogs, Messages, SysUtils,
41:   // Application Units
42:   AppConstants, AppTypes, AppVariables, StatusBar, TMVFiles_DTMF, TMVFiles_FAV,
43:   TMVFiles_UHF, TMVFiles_VHF, Utilities;
44:
45: procedure NewTMVFile;
46: procedure OpenTMVFile(vstrFName : string);
47: procedure SaveTMVFile;
48: procedure SaveTMVFileAs;
49: procedure WriteTMVFile(vstrFileName : string);
50:
51: implementation
52:
53: uses
54:   DataEntry, Main;
55:
56: var
57:   TMVFile : TextFile;
58:
59: //-----

```

```

60: //      CLOSE TMV FILE
61: //-----
62: procedure CloseTMVFile;
63: begin
64:   Close(TMVFile);
65: end;// procedure CloseTMVFile;
66:
67: //=====
68: //      OPEN TMV FILE
69: //=====
70: procedure OpenTMVFile(vstrFName : string);
71:
72: var
73:   vstrTStr : string;
74:   vbytRecNr : Byte;
75:
76: begin
77:
78:   if Length(vstrFName) = 0 then
79:   begin
80:     frmMain.OpenDialog1.Title := 'Open TMV File';
81:     frmMain.OpenDialog1.InitialDir := gvstrAppPath + gcstrTMV7FileDir;
82:     frmMain.OpenDialog1.Filter := 'TMVFiles|*.tmv';
83:     frmMain.OpenDialog1.DefaultExt := 'tmv';
84:
85:     if frmMain.OpenDialog1.Execute then
86:       gvstrTMVFileName := frmMain.OpenDialog1.FileName
87:     else
88:       begin
89:         ShowMessage('Cancelled');
90:         Exit;
91:       end;// if frmMain.OpenDialog1.Execute
92:   end;// if Length(vstrFName) = 0
93:
94:   // Open the file for Reading
95:   gvstrTMVFileName := vstrFName;
96:   Assign(TMVFile, gvstrTMVFileName);
97:   Reset(TMVFile);
98:
99:   // Read the first line to validate the file. If it is [TMVFile Ver. 1.0.0] it is
100:  // a valid file
101:  ReadLn(TMVFile, vstrTStr);
102:
103:  if vstrTStr <> '[TMVFile Ver. 1.0.0]' then
104:  begin
105:    ShowMessage('Invalid File');
106:    Exit;
107:  end;// if vsrtTStr <> '[TMVFile Ver. 1.0.0]'
108:
109:  // Now bypass the VHF header
110:  ReadLn(TMVFile, vstrTStr);
111:
112:  // and read all of the VHF data into the VHF array
113:  ReadLn(TMVFile, vstrTStr);
114:  vbytRecNr := 1;
115:  Repeat
116:    ParseVHFRecord(vbytRecNr, vstrTStr);
117:    ReadLn(TMVFile, vstrTStr);
118:    vbytRecNr := vbytRecNr + 1;

```

```

119:   until vstrTStr = gcstrTMV7UHFHeader;
120:
121:   // Read all of the UHF data into the UHF array
122:   ReadLn(TMVFile, vstrTStr);
123:   vbytRecNr := 1;
124:   Repeat
125:     ParseUHFRecord(vbytRecNr, vstrTStr);
126:     ReadLn(TMVFile, vstrTStr);
127:     vbytRecNr := vbytRecNr + 1;
128:   until vstrTStr = gcstrTMV7FAVHeader;
129:
130:   // Read all of the FAV data into the FAV array
131:   ReadLn(TMVFile, vstrTStr);
132:   vbytRecNr := 1;
133:   Repeat
134:     ParseFAVRecord(vbytRecNr, vstrTStr);
135:     ReadLn(TMVFile, vstrTStr);
136:     vbytRecNr := vbytRecNr + 1;
137:   until vstrTStr = gcstrTMV7DTMFHeader;
138:
139:   // and read all of the DTMF data into the DTMF array
140:   vbytRecNr := 1;
141:   Repeat
142:     ReadLn(TMVFile, vstrTStr);
143:     ParseDTMFRecord(vbytRecNr, vstrTStr);
144:     vbytRecNr := vbytRecNr + 1;
145:   until EOF(TMVFile);
146:
147:   // Close the TMV File
148:   CloseTMVFile;
149:
150:   // Set Data Changed flag
151:   gvstrTMVDataChanged := False;
152:
153:   // Update the Button captions
154:   frmMain.bbtFAV01.Caption := gvstrFAVChannelDataArray[1, gcbytChannelNameField];
155:   frmMain.bbtFAV02.Caption := gvstrFAVChannelDataArray[2, gcbytChannelNameField];
156:   frmMain.bbtFAV03.Caption := gvstrFAVChannelDataArray[3, gcbytChannelNameField];
157:   frmMain.bbtFAV04.Caption := gvstrFAVChannelDataArray[4, gcbytChannelNameField];
158:   frmMain.bbtFAV05.Caption := gvstrFAVChannelDataArray[5, gcbytChannelNameField];
159:   frmMain.bbtFAV06.Caption := gvstrFAVChannelDataArray[6, gcbytChannelNameField];
160:   frmMain.bbtFAV07.Caption := gvstrFAVChannelDataArray[7, gcbytChannelNameField];
161:   frmMain.bbtFAV08.Caption := gvstrFAVChannelDataArray[8, gcbytChannelNameField];
162:   frmMain.bbtFAV09.Caption := gvstrFAVChannelDataArray[9, gcbytChannelNameField];
163:   frmMain.bbtFAV10.Caption := gvstrFAVChannelDataArray[10, gcbytChannelNameField];
164:   frmMain.bbtFAV11.Caption := gvstrFAVChannelDataArray[11, gcbytChannelNameField];
165:   frmMain.bbtFAV12.Caption := gvstrFAVChannelDataArray[12, gcbytChannelNameField];
166:
167:   // Update the Status Bar
168:   DisplayTMVFileStatus;
169:
170: end; // procedure OpenTMVFile;
171:
172: //=====
173: //      WRITE TMV FILE
174: //=====
175: procedure WriteTMVFile(vstrFileName : string);
176:
177: var

```

```

178:   vbytTemp : Integer;
179:
180: begin
181:
182:     // Open the file for Writing
183:   Assign(TMVFile, vstrFileName);
184:   Rewrite(TMVFile);
185:
186:   Writeln(TMVFile, '[TMVFile Ver. 1.0.0]');
187:
188:   Writeln(TMVFile, gcstrTMV7VHFHeader);
189:   for vbytTemp := 1 to gcbytMaxVHFChannels do
190:     Writeln(TMVFile, MakeVHFRecord(vbytTemp));
191:
192:   Writeln(TMVFile, gcstrTMV7UHFHeader);
193:   for vbytTemp := 1 to gcbytMaxUHFChannels do
194:     Writeln(TMVFile, MakeUHFRecord(vbytTemp));
195:
196:   Writeln(TMVFile, gcstrTMV7FAVHeader);
197:   for vbytTemp := 1 to gcbytMaxFAVChannels do
198:     Writeln(TMVFile, MakeFAVRecord(vbytTemp));
199:
200:   Writeln(TMVFile, gcstrTMV7DTMFHeader);
201:   for vbytTemp := 1 to gcbytMaxDTMFCodes do
202:     Writeln(TMVFile, MakeDTMFRecord(vbytTemp));
203:
204:   // Close the TMV File
205:   CloseTMVFile;
206:
207:   // Set Data Changed flag
208:   gvstrTMVDataChanged := False;
209:
210:   // Update the Status Bar
211:   DisplayTMVFileStatus;
212:
213: end; // procedure WriteTMVFile;
214:
215: //=====
216: //           NEW TMV FILE
217: //=====
218: procedure NewTMVFile;
219:
220: var
221:   vintTempRec : Integer;
222:   vbytTempField : Byte;
223:
224: begin
225:
226:   // First we clear all channel arrays
227:   // UHF Data Array
228:   For vintTempRec := 1 to gcbytMaxUHFChannels do
229:     begin
230:       For vbytTempField := gcbytVFOField to gcbytMaxChannelFieldCount do
231:         gvstrUHFChannelDataArray[vintTempRec, vbytTempField] := '';
232:       end; // For vintTempRec := 1 to gcbytMaxUHFChannels do
233:
234:       // VHF Data Array
235:       For vintTempRec := 1 to gcbytMaxVHFChannels do
236:         begin

```

```

237:     For vbytTempField:= gcbytVFOField to gcbytMaxChannelFieldCount do
238:         gvstrVHFChannelDataArray[vintTempRec, vbytTempField] := '';
239:     end;// For vintTempRec := 1 to gcbytMaxVHFChannels do
240:
241:     // FAV Data Array
242:     For vintTempRec := 1 to gcbytMaxFAVChannels do
243:     begin
244:         For vbytTempField := gcbytVFOField to gcbytMaxChannelFieldCount do
245:             gvstrFAVChannelDataArray[vintTempRec, vbytTempField] := '';
246:         end;// For vintTempRec := 1 to gcbytMaxFAVChannels do
247:
248:         // Now the DTMF Data Array
249:         For vintTempRec := 1 to gcbytMaxDTMFCodes do
250:         begin
251:             gvstrDTMFCodeDataArray[vintTempRec] := '';
252:         end;// For vintTempRec := 1 to gcbytMaxDTMFCodes do
253:
254:         // Now we create the file and write the data
255:         WriteTMVFile(gvstrTMVFileName);
256:
257:         // Clear the FAV buttona
258:         frmMain.bbtFav01.Caption := '';
259:         frmMain.bbtFav02.Caption := '';
260:         frmMain.bbtFav03.Caption := '';
261:         frmMain.bbtFav04.Caption := '';
262:         frmMain.bbtFav05.Caption := '';
263:         frmMain.bbtFav06.Caption := '';
264:         frmMain.bbtFav07.Caption := '';
265:         frmMain.bbtFav08.Caption := '';
266:         frmMain.bbtFav09.Caption := '';
267:         frmMain.bbtFav10.Caption := '';
268:         frmMain.bbtFav11.Caption := '';
269:         frmMain.bbtFav12.Caption := '';
270:
271:         // Updaste the Status bar
272:         DisplayTMVFileStatus;
273:
274:     end;// procedure NewTMVFile;
275:
276: //=====
277: procedure SaveTMVFile;
278: begin
279:     WriteTMVFile(gvstrTMVFileName);
280: end;// procedure SaveTMVFile;
281:
282: //=====
283: procedure SaveTMVFileAs;
284: begin
285:
286:     frmMain.SaveDialog1.Title := 'Save the TMV File as';
287:     frmMain.SaveDialog1.InitialDir := gvstrAppPath + gcstrTMV7FileDir;
288:     frmMain.SaveDialog1.Filter := 'TMVFiles|*.tmv';
289:     frmMain.SaveDialog1.DefaultExt := 'tmv';
290:
291:     if frmMain.SaveDialog1.Execute then
292:     begin
293:         gvstrTMVFileName := frmMain.SaveDialog1.FileName;
294:         WriteTMVFile(gvstrTMVFileName);
295:         DisplayTMVFileStatus;

```

```
296:     end
297:     else
298:         ShowMessage('Save TMV File As Cancelled');
299:
300: end;// procedure SaveTMVFileAs;
301:
302: //=====
303: end.// unit TMVFiles;
304:
```