```
 1: unit dataentry_vhfmem;
 2:
 3: {$mode objfpc}{$H+}
 4:
 5: //=============================================================================================
 6: //
 7: //  DataEntry_VHFMEM.pas
 8: //
 9: //  Calls: AppCOnstants
10: //         AppTypes
11: //         AppVariables
12: //         DataEntry
13: //         Mem_VHF : LoadVHFStringGrid
14: //         Utilities : GetToneIndexFromToneNr
15: //
16: //  Called By: DataEntry : TfrmDataEntry.FormActivate
17: //                         TfrmDataEntry.bbtResetClick
18: //                         TfrmDataEntry.bbtSaveClick
19: //
20: //  Ver: 1.0.0
21: //
22: //  Date: 11 Aug 2013
23: //
24: //=============================================================================================
25:
26:
27: interface
28:
29: uses
30:   Classes, Dialogs, SysUtils,
31:   // Application Units
32:   AppConstants, AppTypes, AppVariables, MEM_VHF, Utilities;
33:
34: procedure DataEntry_VHFMEM_Init;
35: procedure DataEntry_VHFMEM_Save;
36:
37: implementation
38:
39: uses
40:   DataEntry;
41:
42: //=============================================================================================
43: procedure DataEntry_VHFMEM_Init;
44:
45: var
46:   vbytToneNr : Byte;
47:
48: begin
49:
50:     frmDataEntry.Caption := frmDataEntry.cstrMemFormTitle;
51:     frmDataEntry.edtSource.Text := Format('VHF%.2d',[frmDataEntry.vbytChannelNumber]);
52:
53:     //=====================
54:     // Set the RX Frequency
55:     //=====================
56:
57:     if Length(gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber,
58:               gcbytRXFrequencyField]) > 0 then
59:               frmDataEntry.edtRXFrequency.Text :=
```

```
60:       Copy(gvstrFAVChannelDataArray[frmDataEntry.vbytChannelNumber,
61:         gcbytRXFrequencyField], 3, 3) +
62:       '.' +
63:       Copy(gvstrFAVChannelDataArray[frmDataEntry.vbytChannelNumber,
64:         gcbytRXFrequencyField], 6, 3)
65:     else
66:       frmDataEntry.edtRXFrequency.Text := '';
67:
68:     //=========================
69:     // Set the Band Radio buttons
70:     //=========================
71:
72:     if gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber,
73:                                 gcbytVFOField] = gcstrUHF then
74:       frmDataEntry.rbtUHF.Checked := True
75:     else
76:       frmDataEntry.rbtVHF.Checked := True;
77:
78:     //=========================
79:     // Set the Shift Radio buttons
80:     //=========================
81:
82:     if gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber,
83:                                 gcbytShiftField] = gcstrShiftPlus then
84:       frmDataEntry.rbtPlus.Checked := True
85:     else if gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber,
86:                                 gcbytShiftField] = gcstrShiftMinus then
87:       frmDataEntry.rbtMinus.Checked := True
88:     else
89:       frmDataEntry.rbtSimplex.Checked := True;
90:
91:     //====================================
92:     // Set the Shift offset and calulate and Display the TX Frequency
93:     //====================================
94:
95:      if Length(frmDataEntry.edtRXFrequency.Text) > 0 then
96:      begin
97:        frmDataEntry.edtTXFrequency.Text := frmDataEntry.CalculateTXFrequency;
98:        frmDataEntry.SetShiftOffset;
99:      end
100:     else
101:       frmDataEntry.edtTXFrequency.Text := '';
102:
103:    //=====================================================
104:    // Set the Tone Checkboxes and Tone Frequency Combobox
105:    //=====================================================
106:    if gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber,
107:                                gcbytToneField] = gcstrOn then
108:      frmDataEntry.rbtTone.Checked := True
109:    else if gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber,
110:                                gcbytCTCSSField] = gcstrOn then
111:      frmDataEntry.rbtCTCSS.Checked := True
112:    else
113:      frmDataEntry.rbtNoTones.Checked := True;
114:
115:    //=========================================================
116:    // Determine the correct Index and set the Tone Freq Combobox
117:    //=========================================================
118:
```

```
119:     showmessage('gcbytToneNrField = ' + gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber
     ,
120:                                 gcbytToneNrField]);
121:
122:     if gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber,
123:                                 gcbytToneNrField] = '' then vbytToneNr := 1
124:     else vbytToneNr := StrToInt(gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber,
125:                                 gcbytToneNrField]);
126:     frmDataEntry.cbxTones.ItemIndex := GetToneIndexFromToneNr( vbytToneNr);
127:
128:     // Now Clear the Text field if there is no Tone function selected.
129:     if frmDataEntry.rbtNoTones.Checked then
130:       frmDataEntry.cbxTones.Text := '';
131:
132:     //=====================================
133:     //  Set the DTSS Checkbox and Code field
134:     //=====================================
135:     if gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber,
136:                                   gcbytDTSSField] = gcstrOn then
137:     begin
138:       frmDataEntry.chkDTSS.Checked := True;
139:       frmDataEntry.EnableDTSSCode;
140:     end
141:     else
142:     begin
143:       frmDataEntry.chkDTSS.Checked := False;
144:       frmDataEntry.DisableDTSSCode;
145:     end;
146:
147:     frmDataEntry.SetDTSSCode;
148:
149:     //======================
150:     // Set the Scan Checkbox
151:     //======================
152:
153:     if gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber,
154:                                   gcbytScanField] = gcstrOn then
155:       frmDataEntry.chkScan.Checked := True
156:     else
157:       frmDataEntry.chkScan.Checked := False;
158:
159:     //===============================
160:     // Set the RF Power Radio buttons
161:     //===============================
162:
163:     if gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber,
164:                                     gcbytRFPowerField] = gcstrRFPowerLow then
165:        frmDataEntry.rbtRFPowerLow.Checked := True
166:     else if gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber,
167:                                     gcbytShiftField] = gcstrRFPowerMedium then
168:       frmDataEntry.rbtRFPowerMedium.Checked := True
169:     else
170:       frmDataEntry.rbtRFPowerHigh.Checked := True;
171:
172:     //====================
173:     // Set the Channel Name
174:     //====================
175:
176:     frmDataEntry.edtChannelName.Text := gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber
```

```
    ,
177:                                        gcbytChannelNameField];
178:
179:     //=================
180:     // Set the Comments
181:     //=================
182:
183:     frmDataEntry.edtComments.Text := gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber,
184:                                      gcbytCommentsField];
185:
186: //    DisplayDataArray(drtVHFMEM, frmDataEntry.vbytChannelNumber);
187:
188: end;// procedure DataEntry_VHFMEM_Init
189:
190: //=============================================================================================
191: procedure DataEntry_VHFMEM_Save;
192:
193: var
194:   vstrTStr : string;
195:
196: begin
197:
198:   //=====================================================================
199:   // We populate the Favourite Array using the Favourite Button number
200:   // as the primary key and the Field number as the sceondary key.
201:   //=====================================================================
202:
203:   // VFO/Band
204:   if frmdataEntry.rbtVHF.Checked then
205:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytVFOField] := gcstrVHF
206:   else
207:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytVFOField] := gcstrUHF;
208:
209:   // RX Frequency
210:   vstrTStr := '00' +
211:               Copy(frmDataEntry.edtRXFrequency.Text,1,3) +
212:               Copy(frmDataEntry.edtRXFrequency.Text,5,3) +
213:               '000';
214:   gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytRXFrequencyField] :=
215:     vstrTStr;
216:
217:   // Step Size
218:   if frmdataEntry.rbtVHF.Checked then
219:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytStepField] :=
220:       gcstrVHFStep
221:   else
222:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytStepField] :=
223:       gcstrUHFStep;
224:
225:   // Shift Indicator
226:   if frmDataEntry.rbtSimplex.Checked then
227:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytShiftField] :=
228:       gcstrShiftSimplex
229:   else if frmDataEntry.rbtPlus.Checked then
230:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytShiftField] :=
231:       gcstrShiftPlus
232:   else
233:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytShiftField] :=
234:       gcstrShiftMinus;
```

```
235:
236:   // Reverse switch
237:   // It is not configurable. It may only be toggled by the GUI button
238:   // so we default it to Off
239:   gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytReverseField] :=
240:     gcstrOff;
241:
242:   // Tone and CTCSS switch
243:   // Although there are two data fields, they are mutually exclusive so we handle
244:   // them together. They may both be Off, but only one of them may be on at a time
245:   if frmDataEntry.rbtNoTones.checked then
246:   begin
247:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytToneField] :=
248:       gcstrOff;
249:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytCTCSSField] :=
250:       gcstrOff;
251:   end
252:   else if frmDataEntry.rbtTone.checked then
253:   begin
254:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytToneField] :=
255:       gcstrOn;
256:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytCTCSSField] :=
257:       gcstrOff;
258:   end
259:   else
260:   begin
261:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytToneField] :=
262:       gcstrOff;
263:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytCTCSSField] :=
264:       gcstrOn;
265:   end;// if frmDataEntry.rbtNoTones.checked
266:
267:   // DTSS switch
268:   if frmDataEntry.chkDTSS.checked then
269:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytDTSSField] :=
270:       gcstrOn
271:   else
272:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytDTSSField] :=
273:       gcstrOff;
274:
275:   // Tone Number
276:   if frmDataEntry.rbtTone.checked then
277:   begin
278:     vstrTStr := IntToStr(GetToneNrFromIndex(frmDataEntry.cbxTones.ItemIndex));
279:     if StrToInt(vstrTStr) > 10 then
280:       gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytToneNrField] := vstrTStr
281:     else
282:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytToneNrField] := '0' +
283:                                                                   vstrTStr;
284:   end
285:   else
286:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytToneNrField] := '01';
287:
288:      // DTSS Code
289:   if frmDataEntry.chkDTSS.checked then
290:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytDTSSCodeField] :=
291:       frmDataEntry.edtDTSSCode.Text
292:   else
293:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytDTSSCodeField] := '000';
```

```
294:
295:   // CTCSS Nr
296:   if frmDataEntry.rbtCTCSS.checked then
297:   begin
298:
299:     showmessage('CTCSS Nr - ' + IntToStr(frmDataEntry.cbxTones.ItemIndex));
300:
301:     vstrTStr := IntToStr(GetToneNrFromIndex(frmDataEntry.cbxTones.ItemIndex));
302:     if StrToInt(vstrTStr) > 10 then
303:       gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytCTCSSNrField] := vstrTStr
304:     else
305:       gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytCTCSSNrField] := '0' +
306:                                                                                      vstrTStr;
307:   end
308:   else
309:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytCTCSSNrField] := '01';
310:
311:   // Shift Offset
312:   if frmDataEntry.rbtSimplex.checked then
313:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber,
314:                              gcbytShiftOffsetField] := '000000000'
315:   else
316:   begin
317:     if frmDataEntry.rbtVHF.Checked then
318:       gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber,
319:                                gcbytShiftOffsetField] := '000600000'
320:     else
321:       gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber,
322:                                gcbytShiftOffsetField] := '005000000';
323:   end;// if frmDataEntry.rbtSimplex.checked
324:
325:   // Scan switch
326:   if frmdataEntry.chkScan.Checked then
327:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytScanField] := gcstrOn
328:   else
329:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytScanField] := gcstrOff;
330:
331:   // RF Power
332:   if frmDataEntry.rbtRFPowerLow.Checked then
333:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytRFPowerField] :=
334:       gcstrRFPowerLow
335:   else if frmDataEntry.rbtRFPowerMedium.Checked then
336:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytRFPowerField] :=
337:       gcstrRFPowerMedium
338:   else
339:     gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytRFPowerField] :=
340:       gcstrRFPowerHigh;
341:
342:   // Now we save the Channel Name
343:   gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytChannelNameField] :=
344:     frmDataEntry.edtChannelName.Text;
345:
346:    // Now the Comments
347:    gvstrVHFChannelDataArray[frmDataEntry.vbytChannelNumber, gcbytCommentsField] :=
348:      frmDataEntry.edtComments.Text;
349:
350: //   DisplayDataArray(drtVHFMEM, frmDataEntry.vbytChannelNumber);
351:
352:    LoadVHFStringGrid;
```

```
353:
354: end;// procedure DataEntry_VHFMEM_Save;
355:
356: //=============================================================================================
357: end.// unit DataEntry_VHFMEM;
358:
359:
```