

```
1: /*
2: Program to interface between VHFLOG contest logger and the Kenwood TS2000.
3:
4: Developed by WA3DRC starting 1/3/2017. Adapted for bandswitching by W3SZ 7/2017.
5:
6: This program is meant to intercept commands from VHFLOG and provide transverter
   bandswitching. VHFLOG thinks it is talking
7: to a CANAKIT board to switch bands. https://www.canakit.com/Media/Manuals/UK1104.pdf
8:
9: W3KM sends the first two digits of the new band with each band change in VHFLOG:
10:
11: */
12:
13: //include string handling library
14: #include <string.h>
15:
16: //define variables
17: String commandInputString = ""; // input buffer string to hold incoming data
18: boolean commandStringComplete = false; // true when the input string is complete
19: String command = ""; // incoming data string for parsing
20:
21: boolean hwCR = false; // true if '\r' has been received
22:
23: //define constant pin aliases
24: const int Pin50 = 2; //number of 50 MHz pin
25: const int Pin144 = 3; //number of 144 MHz pin
26: const int Pin222 = 4; //number of 222 MHz pin
27: const int Pin432 = 5; //number of 432 MHz pin
28: const int Pin902 = 6; //number of 902 MHz pin
29: const int Pin1296 = 8; //number of 1296 MHz pin
30: const int Pin2304 = A5; //number of 2304 MHz pin
31: const int Pin3G = A4; //number of 3GHz pin
32: const int Pin5G = A3; //number of 5GHz pin
33: const int Pin10G = A2; //number of 10GHz pin
34: const int Pin24G = A1; //number of 24GHz pin
35: const int Pin47G = A0; //number of 47GHz pin
36: const int Pin76G = 7; //number of 76GHz pin
37:
38: void setup() {
39:
40: // define GPIO pins as output pins
41: pinMode(Pin50,OUTPUT);
42: pinMode(Pin144,OUTPUT);
43: pinMode(Pin222,OUTPUT);
44: pinMode(Pin432,OUTPUT);
45: pinMode(Pin902,OUTPUT);
46: pinMode(Pin1296,OUTPUT);
47: pinMode(Pin2304,OUTPUT);
48: pinMode(Pin3G,OUTPUT);
49: pinMode(Pin5G,OUTPUT);
50: pinMode(Pin10G,OUTPUT);
51: pinMode(Pin24G,OUTPUT);
52: pinMode(Pin47G,OUTPUT);
53: pinMode(Pin76G,OUTPUT);
54:
55: //initialize all GPIO pin values to low
56: digitalWrite(Pin50,LOW);
57: digitalWrite(Pin144,LOW);
58: digitalWrite(Pin222,LOW);
59: digitalWrite(Pin432,LOW);
60: digitalWrite(Pin902,LOW);
61: digitalWrite(Pin1296,LOW);
62: digitalWrite(Pin2304,LOW);
63: digitalWrite(Pin3G,LOW);
64: digitalWrite(Pin5G,LOW);
65: digitalWrite(Pin10G,LOW);
```

```
66: digitalWrite(Pin24G,LOW);
67: digitalWrite(Pin47G,LOW);
68: digitalWrite(Pin76G,LOW);
69:
70: // define, start, flush serial port Serial 0
71: // VHF log will send commands to this port
72: Serial.begin(9600, SERIAL_8N1); // 9600/8/N/1
73: Serial.println("VHFLog/RoverLog Bandswitch");
74: Serial.println("By W3SZ");
75: Serial.println("Uses USB-Serial Port");
76: Serial.println("50 MHz thru 76 GHz");
77: delay(100);
78:
79: Serial.flush(); // clear buffers
80: }
81:
82: void loop() { //MAIN
83:
84: ///////////////////////////////////////////////////////////////////
85: // get VHFLOG command from serial0
86: if (commandStringComplete) {
87:     command = commandInputString;
88:     // save this new command then clear the input buffer
89:     // clear the string:
90:     commandInputString = "";
91:     //set string complete flag to false in preparation for next VHFLOG command;
92:     commandStringComplete = false;
93: }
94: ///////////////////////////////////////////////////////////////////
95: // now process the VHFLOG command
96: if (command.length() > 0){
97: ///////////////////////////////////////////////////////////////////
98:
99:     if (command.startsWith("50")) { // set band to 6m
100:         //set Pin50 high, all other pins low
101: digitalWrite(Pin50,HIGH);
102: digitalWrite(Pin144,LOW);
103: digitalWrite(Pin222,LOW);
104: digitalWrite(Pin432,LOW);
105: digitalWrite(Pin902,LOW);
106: digitalWrite(Pin1296,LOW);
107: digitalWrite(Pin2304,LOW);
108: digitalWrite(Pin3G,LOW);
109: digitalWrite(Pin5G,LOW);
110: digitalWrite(Pin10G,LOW);
111: digitalWrite(Pin24G,LOW);
112: digitalWrite(Pin47G,LOW);
113: digitalWrite(Pin76G,LOW);
114:     }
115:
116:     else if (command.startsWith("14")) { // set band to 2m
117:         //set Pin144 high, all other pins low
118: digitalWrite(Pin50,LOW);
119: digitalWrite(Pin144,HIGH);
120: digitalWrite(Pin222,LOW);
121: digitalWrite(Pin432,LOW);
122: digitalWrite(Pin902,LOW);
123: digitalWrite(Pin1296,LOW);
124: digitalWrite(Pin2304,LOW);
125: digitalWrite(Pin3G,LOW);
126: digitalWrite(Pin5G,LOW);
127: digitalWrite(Pin10G,LOW);
128: digitalWrite(Pin24G,LOW);
129: digitalWrite(Pin47G,LOW);
130: digitalWrite(Pin76G,LOW);
131:     }
```

```
132:
133:     else if (command.startsWith("22")) { // set band to 222
134:         //set Pin222 high, all other pins low
135: digitalWrite(Pin50,LOW);
136: digitalWrite(Pin144,LOW);
137: digitalWrite(Pin222,HIGH);
138: digitalWrite(Pin432,LOW);
139: digitalWrite(Pin902,LOW);
140: digitalWrite(Pin1296,LOW);
141: digitalWrite(Pin2304,LOW);
142: digitalWrite(Pin3G,LOW);
143: digitalWrite(Pin5G,LOW);
144: digitalWrite(Pin10G,LOW);
145: digitalWrite(Pin24G,LOW);
146: digitalWrite(Pin47G,LOW);
147: digitalWrite(Pin76G,LOW);
148:     }
149:
150:     else if (command.startsWith("43")) { // set band to 432
151:         //set Pin432 high, all other pins low
152: digitalWrite(Pin50,LOW);
153: digitalWrite(Pin144,LOW);
154: digitalWrite(Pin222,LOW);
155: digitalWrite(Pin432,HIGH);
156: digitalWrite(Pin902,LOW);
157: digitalWrite(Pin1296,LOW);
158: digitalWrite(Pin2304,LOW);
159: digitalWrite(Pin3G,LOW);
160: digitalWrite(Pin5G,LOW);
161: digitalWrite(Pin10G,LOW);
162: digitalWrite(Pin24G,LOW);
163: digitalWrite(Pin47G,LOW);
164: digitalWrite(Pin76G,LOW);
165:     }
166:
167:     else if (command.startsWith("90")) { // set band to 903
168:         //set Pin902 high, all other pins low
169: digitalWrite(Pin50,LOW);
170: digitalWrite(Pin144,LOW);
171: digitalWrite(Pin222,LOW);
172: digitalWrite(Pin432,LOW);
173: digitalWrite(Pin902,HIGH);
174: digitalWrite(Pin1296,LOW);
175: digitalWrite(Pin2304,LOW);
176: digitalWrite(Pin3G,LOW);
177: digitalWrite(Pin5G,LOW);
178: digitalWrite(Pin10G,LOW);
179: digitalWrite(Pin24G,LOW);
180: digitalWrite(Pin47G,LOW);
181: digitalWrite(Pin76G,LOW);
182:     }
183:
184:     else if (command.startsWith("12")) { // set band to 1296
185:         //set Pin1296 high, all other pins low
186: digitalWrite(Pin50,LOW);
187: digitalWrite(Pin144,LOW);
188: digitalWrite(Pin222,LOW);
189: digitalWrite(Pin432,LOW);
190: digitalWrite(Pin902,LOW);
191: digitalWrite(Pin1296,HIGH);
192: digitalWrite(Pin2304,LOW);
193: digitalWrite(Pin3G,LOW);
194: digitalWrite(Pin5G,LOW);
195: digitalWrite(Pin10G,LOW);
196: digitalWrite(Pin24G,LOW);
197: digitalWrite(Pin47G,LOW);
```

```
198: digitalWrite(Pin76G,LOW);
199:     }
200:
201:     else if (command.startsWith("23")) { // set band to 2304
202:         //set Pin2304 high, all other pins low
203:         digitalWrite(Pin50,LOW);
204:         digitalWrite(Pin144,LOW);
205:         digitalWrite(Pin222,LOW);
206:         digitalWrite(Pin432,LOW);
207:         digitalWrite(Pin902,LOW);
208:         digitalWrite(Pin1296,LOW);
209:         digitalWrite(Pin2304,HIGH);
210:         digitalWrite(Pin3G,LOW);
211:         digitalWrite(Pin5G,LOW);
212:         digitalWrite(Pin10G,LOW);
213:         digitalWrite(Pin24G,LOW);
214:         digitalWrite(Pin47G,LOW);
215:         digitalWrite(Pin76G,LOW);
216:     }
217:
218:     else if (command.startsWith("34")) { // set band to 3456
219:         //set Pin3G high, all other pins low
220:         digitalWrite(Pin50,LOW);
221:         digitalWrite(Pin144,LOW);
222:         digitalWrite(Pin222,LOW);
223:         digitalWrite(Pin432,LOW);
224:         digitalWrite(Pin902,LOW);
225:         digitalWrite(Pin1296,LOW);
226:         digitalWrite(Pin2304,LOW);
227:         digitalWrite(Pin3G,HIGH);
228:         digitalWrite(Pin5G,LOW);
229:         digitalWrite(Pin10G,LOW);
230:         digitalWrite(Pin24G,LOW);
231:         digitalWrite(Pin47G,LOW);
232:         digitalWrite(Pin76G,LOW);
233:     }
234:
235:     else if (command.startsWith("57")) { // set band to 5760
236:         //set Pin5G high, all other pins low
237:         digitalWrite(Pin50,LOW);
238:         digitalWrite(Pin144,LOW);
239:         digitalWrite(Pin222,LOW);
240:         digitalWrite(Pin432,LOW);
241:         digitalWrite(Pin902,LOW);
242:         digitalWrite(Pin1296,LOW);
243:         digitalWrite(Pin2304,LOW);
244:         digitalWrite(Pin3G,LOW);
245:         digitalWrite(Pin5G,HIGH);
246:         digitalWrite(Pin10G,LOW);
247:         digitalWrite(Pin24G,LOW);
248:         digitalWrite(Pin47G,LOW);
249:         digitalWrite(Pin76G,LOW);
250:     }
251:
252:     else if (command.startsWith("10")) { // set band to 10368
253:         //set Pin10G high, all other pins low
254:         digitalWrite(Pin50,LOW);
255:         digitalWrite(Pin144,LOW);
256:         digitalWrite(Pin222,LOW);
257:         digitalWrite(Pin432,LOW);
258:         digitalWrite(Pin902,LOW);
259:         digitalWrite(Pin1296,LOW);
260:         digitalWrite(Pin2304,LOW);
261:         digitalWrite(Pin3G,LOW);
262:         digitalWrite(Pin5G,LOW);
263:         digitalWrite(Pin10G,HIGH);
```

```
264: digitalWrite(Pin24G,LOW);
265: digitalWrite(Pin47G,LOW);
266: digitalWrite(Pin76G,LOW);
267:     }
268:
269:     else if (command.startsWith("24")) { // set band to 24 GHz
270:         //set Pin24G high, all other pins low
271: digitalWrite(Pin50,LOW);
272: digitalWrite(Pin144,LOW);
273: digitalWrite(Pin222,LOW);
274: digitalWrite(Pin432,LOW);
275: digitalWrite(Pin902,LOW);
276: digitalWrite(Pin1296,LOW);
277: digitalWrite(Pin2304,LOW);
278: digitalWrite(Pin3G,LOW);
279: digitalWrite(Pin5G,LOW);
280: digitalWrite(Pin10G,LOW);
281: digitalWrite(Pin24G,HIGH);
282: digitalWrite(Pin47G,LOW);
283: digitalWrite(Pin76G,LOW);
284:     }
285:
286:     else if (command.startsWith("47")) { // set band 47 GHz
287:         //set Pin47G high, all other pins low
288: digitalWrite(Pin50,LOW);
289: digitalWrite(Pin144,LOW);
290: digitalWrite(Pin222,LOW);
291: digitalWrite(Pin432,LOW);
292: digitalWrite(Pin902,LOW);
293: digitalWrite(Pin1296,LOW);
294: digitalWrite(Pin2304,LOW);
295: digitalWrite(Pin3G,LOW);
296: digitalWrite(Pin5G,LOW);
297: digitalWrite(Pin10G,LOW);
298: digitalWrite(Pin24G,LOW);
299: digitalWrite(Pin47G,HIGH);
300: digitalWrite(Pin76G,LOW);
301:     }
302:
303:     else if (command.startsWith("76")) { // set band to 76 GHz
304:         //set Pin76G high, all other pins low
305: digitalWrite(Pin50,LOW);
306: digitalWrite(Pin144,LOW);
307: digitalWrite(Pin222,LOW);
308: digitalWrite(Pin432,LOW);
309: digitalWrite(Pin902,LOW);
310: digitalWrite(Pin1296,LOW);
311: digitalWrite(Pin2304,LOW);
312: digitalWrite(Pin3G,LOW);
313: digitalWrite(Pin5G,LOW);
314: digitalWrite(Pin10G,LOW);
315: digitalWrite(Pin24G,LOW);
316: digitalWrite(Pin47G,LOW);
317: digitalWrite(Pin76G,HIGH);
318:     }
319:     // cleanup
320:     command = ""; // clear the VHFLOG command
321: }
322: //////////////////////////////////// END COMMANDS ////////////////////////////////////
323:
324:
325:     delay(25); // long enough for the radio to return its frequency
326:
327: } //END MAIN
328:
329:
```

```
330: /*
331:   SerialEvent occurs whenever a new data comes in the
332:   hardware serial RX. This routine is run between each
333:   time loop() runs, so using inside loop can
334:   response. Multiple bytes of data may be available.
335: */
336: void serialEvent() {
337:
338:   char commandInChar;
339:
340:   while (Serial.available()) { // interrupt generated by hardware serial port
341:     // get the new byte:
342:     commandInChar = (char)Serial.read();
343:
344:     // add it to the commandInputString:
345:     commandInputString += commandInChar; // append
346:
347:     // look for a carriage return, then a line feed; set a flag
348:     // so the main loop can do something about it:
349:     if (commandInChar == '\r') { // the commands all end with a CR and then a LF (13 10
350:       )
351:       hwCR = true;
352:     }
353:     if ( commandInChar == '\n') {
354:       if ( hwCR ) {
355:         hwCR = false; // cleanup
356:         commandStringComplete = true;
357:       }
358:     }
359:   }
360:
361:
362:
```