

```

1  '=====
2  '
3  '  VERSATILE  ENGINE  SOUND  UNIT
4  '
5  '  PROGRAM 2 OF 4 - SOUND GENERATION (PETROL)
6  '
7  '=====
8  '  VERSION 2.0  14th October 2008           © A.F.Bond 2008
9  '
10 '  (ONLY FOR USE IN CONJUNCTION WITH v2.0 SPEED DEMAND MODULE)
11 '=====
12 '
13 ' This module uses one of the PIC's ADCs to read a speed demand voltage and
14 ' then uses that value to create a variable frequency output signal. The output
15 ' signal comprises repeated cycles of a short burst of pseudo-random noise
16 ' (to simulate an exhaust pulse) followed by a longer period of silence. The
17 ' repetition rate of this cycle is organised to be proportional to the speed
18 ' demand voltage.
19 '
20 ' To give the engine a characteristic 'beat', a slightly different pitch of
21 ' pseudo-random noise is assigned to each of the cylinders before looping round
22 ' to repeat the sequence.
23 '
24 ' A jumper link is read at program launch and its state determines if the
25 ' cylinder selection mode is entered. If so, then the program succesively plays
26 ' a few seconds of fixed speed sound from the 2,3,4,5 or 6 cylinder engine
27 ' variants. Removing the link during one of these periods locks the chip into
28 ' that mode. The cylinder count value is then stored in the PICAXE's EEPROM ie
29 ' non-volatile memory, so that when the link is in the 'engine run' position the
30 ' cylinder count previously set by the user can be read at power up.
31 '
32 ' A counter is enabled whenever the engine is detected to be idling, and if a
33 ' time-out value is reached, the engine enters a several second phase where it
34 ' slowly runs down to a halt. Opening the throttle again causes a short 'cranking'
35 ' sound to be issued before the engine begins to run again.
36 '
37 '
38 ' NOTE: In the description below be aware that PIN1, PIN2 etc are virtual
39 ' pins and not to be confused with the device's physical pins, which in PIC
40 ' parlance are refered to as 'legs'
41 '
42 '=====
43
44
45 'VARIABLE DEFINITIONS
46
47 symbol pause=w3      'time between exhaust pulses: note spelling ("pause"
48                       'is a reserved word)
49
50
51 symbol running =b1    'engine running flag (1=running 0=stopped)
52
53 symbol demand=b2      'speed demand input, read from ADC on "pin4" (leg 3)
54
55 symbol kount=b3       'general purpose counter
56                       'note spelling ("count" is a reserved word)
57
58 symbol idle_count=w2   'counts revolutions at idle

```

```

59
60 symbol timeout = w6      'number of revolutions at idle before engine stops
61
62 symbol p=b8              'exhaust pitch (passed from sound lookup table)
63
64 symbol t=b9              'exhaust duration (passed from sound lookup table)
65
66 symbol cyl_count=b10     'number of cylinders in engine
67                          'note numbering begins at 0 as the lookup tables
68                          'have a zero'th element. The length of the tables
69                          'restricts the maximum cylinder count to six
70
71 symbol cownt=b11         'another general purpose counter
72                          'note spelling ("count" is a reserved word)
73
74
75
76
77 'CONSTANT DEFINITIONS
78
79 symbol neutral_LED=0     'neutral LED driven from "pin0" (leg 7) high=ON
80
81
82
83 'EXPERIMENT WITH THESE VALUES TO CHANGE THE DYNAMICS OF THE ENGINE
84 symbol tickover =7000    'as these values define the pause between each
85 symbol topspeed =200     'cylinder firing then their numerical sense is
86                          'inversely related to engine speed
87
88 symbol revrange =25      'this is a multiplier or 'gain' factor that is
89                          'applied to the speed demand input to the module
90                          'ie adjusts the throttle sensitivity from tickover
91                          'towards topspeed
92
93 symbol limit=tickover-topspeed 'a derived value which is used to limit the
94                          'top speed if the revrange multiplier is
95                          'set too high
96
97 symbol holt=20000        'the speed below tickover at which engine stops
98                          'note spelling ("halt" or "stop" are reserved words)
99
100 symbol rate = 1500      'rate of engine run-down to a stop
101
102
103
104 'EXPERIMENT WITH THESE VALUES TO CHANGE THE CHARACTER OF THE ENGINE
105 'values must lie in the range 128 to 255
106 symbol p1=235           'pitch of pseudo random sound for cylinder 1
107 symbol p2=133           '..... 2
108 symbol p3=238           '..... 3
109 symbol p4=243           '..... 4
110 symbol p5=230           '..... 5
111 symbol p6=244           '..... 6
112
113 symbol d1=2             'duration of pseudo random sound for cylinder 1
114 symbol d2=4             '..... 2
115 symbol d3=2             '..... 3
116 symbol d4=2             '..... 4

```

```

117 symbol d5=2          '.....5
118 symbol d6=2          '.....6
119
120
121
122 'CHIP PROGRAMMING DIRECTIVE (only executed when program is loaded into chip)
123
124 'Pre-load non-zero value into EEPROM storage location 0 (cylinder count)
125 data 0, (3)          'set default cylinder count to 4
126
127
128
129
130
131 '===== PROGRAM STARTS HERE =====
132
133 'INITIALISATION
134 setfreq m8            'set 8Mhz operation to yield 5uSec resolution
135
136
137 init:
138 if pin3=0 then        'check jumper link
139 'CYLINDER SELECT MODE SELECTED
140 high neutral_LED      'turn on LED
141 sound 1,(120,60)      'warning beep
142
143 for cyl_count=1 to 5  'step through all cylinder values (omit single cyl)
144
145     pause 2000         '1 sec pause between variants (8Mhz doubles values)
146     for kount=0 to 15  'run for sixteen engine cycles
147         for cownt= 0 to cyl_count      'for each cylinder in turn
148             lookup cownt,(p1,p2,p3,p4,p5,p6),p 'get its exhaust pitch
149             lookup cownt,(d1,d2,d3,d4,d5,d6),t 'get its exhaust duration
150             sound 1,(p,t)                'make exhaust noise
151             pause 60                      'silence until next cylinder fires
152         next cownt
153     next kount
154     if pin3=1 then        'has user selected the current cylinder variant?
155         write 0,cyl_count 'YES - store value in non-volatile memory
156         sound 1,(120,10)  'acknowledge with a beep
157         goto init2       'and jump out of cylinder select mode
158     end if
159     'NO - try next cylinder count
160 next cyl_count
161
162 goto init          'no selection made, loop round again
163
164 else
165 'ENGINE RUN MODE
166 read 0, cyl_count  'get cylinder count previously stored in EEPROM
167 end if
168
169
170 init2:
171 'set number of idle revs for approx 1 minute, adjusting for cylinder count
172 'note: variable is re-used rather than having to assign another word variable
173 timeout=cyl_count+1
174 timeout=290*4/timeout 'ie 290 is approx 1 minute for 4 cyl engine

```

```

175
176
177
178
179
180 'MAIN LOOP
181
182 main:
183 readadc 4,demand      'check speed demand status with ADC4
184
185 if running=1 then
186 'ENGINE RUNNING
187 'process speed demand value and assign value to 'pawse'
188
189 pawse=demand*revrange max limit 'set spread of speed, limiting if necessary
190
191 pawse=tickover-pawse      'numerical sense of pawse is inversely related
192                          'to speed (see above)
193
194
195
196 'generate a cycle of engine sound
197 for cownt=0 to cyl_count      'for each cylinder in turn
198     lookup cownt,(p1,p2,p3,p4,p5,p6),p 'get its exhaust pitch
199     lookup cownt,(d1,d2,d3,d4,d5,d6),t 'get its exhaust duration
200     sound 1,(p,t)             'make exhaust noise
201     pulsout 2,pawse           'silence until next cylinder fires
202 next
203
204 'check if speed demand is IDLE
205 if demand<10 then
206 'ENGINE IDLING
207     high neutral_LED          'LED on
208     idle_count=idle_count+1    'increment idle time-out counter
209 else
210 'NOT IDLING
211     low neutral_LED           'LED off
212     idle_count=0              'reset idle time-out counter
213 end if
214
215
216 'check if engine stop required
217 if idle_count>timeout then
218 'STOP ENGINE
219     for pawse=tickover to holt step rate 'gradually slow engine
220     'generate a cycle of engine sound
221     for cownt=0 to cyl_count      'for each cylinder in turn
222         lookup cownt,(p1,p2,p3,p4,p5,p6),p 'get its exhaust pitch
223         lookup cownt,(d1,d2,d3,d4,d5,d6),t 'get its exhaust duration
224         sound 1,(p,t)             'make exhaust noise
225         pulsout 2,pawse           'silence until next cylinder fires
226     next cownt
227 next pawse
228 running=0                    'clear "engine is running" flag
229 end if
230 'from here program drops through to final "goto main" statement
231
232

```

```

233 else
234 'ENGINE IS NOT RUNNING
235 'but has demand been increased from idle in the meantime?
236 if demand>10 then
237 'YES - START ENGINE
238 for count=150 to 25 step -25 'make primitive engine cranking noise
239 sound 1,(185,6)
240 sound 1,(205,3)
241 sound 1,(235,2) 'some experimentation may be of benefit!
242 sound 1,(254,2) 'or omit this for-next loop altogether
243 pause count 'for instant start
244 next
245 'but leave this next line in!
246 running=1 'set "engine is running" flag
247 else
248 'NO - ENGINE REMAINS STOPPED
249 toggle neutral_LED 'flicker neutral LED to signify stopped
250 pause 80 'sets rate of flicker
251 end if
252 end if
253
254 goto main
255
256
257
258
259 '=====
260 ' VERSION HISTORY
261 '=====
262
263 'v1.0 1st October 2008
264 'first release
265
266
267 'v2.0 14th October 2008
268 'Logarithmic lookup table function moved into the speed demand module, which
269 'allowed a longer lookup table to be used without code execution time problems
270 'as the speed demand code only has to run at the transmitter frame rate rather
271 'than the engine revolution rate. The resulting throttle response of the
272 'overall system is thereby much smoother.
273 '
274 'The program space saved by exporting the speed lookup table has been used to
275 'rework the speed demand processing to give single numeric values (for user
276 'experimentation) to set the tickover speed and the revs range plus a top speed
277 'limiter which operates if the revs range multiplier is set too high.
278 '
279 'The timing for main engine loop now uses 'pulsout' (to an unused output) to
280 'create the inter-cylinder delays rather than the 'pause' statement. Much
281 'larger values for an equivalent delay result, meaning that the speed resolution
282 '(ie size of the steps between successive values) is much improved.
283 '
284 'Again, utilising the increased program space, it has now been possible to vary
285 'the 'timeout' value with the cylinder count to yield the same idling period
286 'prior to cut-off regardless of the number of cylinders chosen.
287 '=====

```