

```

001 ;Small Magnetic Loop Antenna Autotuner Program, Version 5b
Quarter-Quarter
002 ;steps and stepping back for the ;Model 3 (Standardized)
Autotuner Circuit.
003
004 ;Copyright: Andrew Cornwall / Enfield, Nova Scotia / Canada
005 ;This program is free for anyone to use for non-commercial
purposes.
006
007 ;Small Magnetic Loop Antenna Automatic Tuner Project Model 3
Circuit
008 ;This version of the Automatic Tuner program coincides with
the article,
009 ;'Experimental Automatic Tuner for Small Magnetic Loop
Antennas'
010 ;by Andrew Cornwall / VE1COR, published in the ; XXMonth-
XXMonth 2018;
011 ;edition of the "The Canadian Amateur" (TCA) magazine of the
012 ;Radio Amateurs of Canada. The article explains how to use
this program.
013
014 ;Notes:
015 ;The program assumes that a free-turning (360 degree) variable
016 ;capacitor is attached to the automatic tuner. It is possible
to modify the
017 ;circuit and program allowing for use with a 180 deg., half-
turn variable
018 ;capacitor. How to do this is explained on the website:
019 ;www.cornwaab.altervista.org.
020
021 ;The Model 3 circuit has optional provision to use a sampling
antenna relay to
022 ;reduce the current going into the automatic tuner's receiver
when high
023 ;wattage transmissions are taking place (i.e. not a low
wattage tuning
024 ;signal). Although the program provides for relay operation,
it does not
025 ;matter that a relay is omitted. Without a relay there will
not be the added
026 ;protection from possible overload, however, with transmission
of up to 100
027 ;relay protection may not be needed.
028 ;
029 ;This program is written for a PICAXE 18M2+ Microcontroller.
It should also
030 ;work with larger microcontrollers in the PICAXE family, but
the pin
031 ;assignments will have to be changed.
032
033 ; ***** Program Starts Below
*****

```

```

034  setfreq m32 ;This tells the 18M2+ to work at a 32 MHz
microcontroller
035  ;(internal) clock speed. This its maximum speed. I've run the
program at
036  ;slower rates, but why bother. The effect program
consideration of 32 MHz
037  ;is to speed up the pause statement by a factor of 8, e.g.
pause 8000 (ms)
038  ;is one second or 1000 ms.
039
040  SERTXD ("VE1COR Autotuner program, Direct Method Ver. 5b
(Quarter-Quarter)",CR,LF,"With Step Back",CR,LF)
041
042  ;VARIABLES - Variables starting with 'W' have 16 bits and can
range in
043  ;value from 0 to 65535, those starting with 'B' have 8 bits
and range from
044  ;0 to 255. All variables are integers.
045
046  symbol AVG_VOLTS=W0 ;AVG_VOLTS is the measure of Sample
Antenna radiated
047  ;signal
048  symbol DELTA=W1
049  symbol MAX_VOLTS=W2
050  symbol REF_VOLTS=W3
051  symbol VOLTS=W4
052  symbol M=W5
053  symbol TEMP_VOLTS=W6
054  symbol NUMBER_STEPS=W7
055  symbol SWEEP_VOLTS=W8
056  symbol MANUAL=B24
057  symbol SWEEP=B25
058  symbol J=B27
059  symbol N=B26
060
061  ;PINS - Pin names, e.g. LED_RED, are assigned here for inputs
or outputs.
062  ;Most pins of the 18M2+ can be used as either an input or
output. Whether
063  ;a pin is an input or output is determined by how it used in
the program.
064  ;Pin references are labeled starting with 'B' or 'C' in the
18M2+,
065  ;and represent specidic pin numbers 1 to 18 according to the
microcontroller's
066  ;diagram. Pin designations were made mostly for the
convenience of wiring.
067
068  symbol LED_RED=B.0 ;RED LED too high tuning signal. Output
HIGH or LOW
069  symbol LED_GREEN=B.1 ;GREEN LED manual mode. Output HIGH or
LOW

```

```

070  symbol LED_BLUE=B.2  ;BLUE LED too low tuning signal. OUTPUT
HIGH or LOW
071  symbol BUZZER=B.3  ;Buzzer. Output HIGH or LOW
072  symbol DIR=B.4  ;Determins turn direction of stepper motor.
Output HIGH or LOW
073  symbol STEPX=B.5  ;Move stepper moter one increment. Output
PULSE
074  symbol MS1=B.6  ;Determins stepper motor partial increment.
Output HIGH or LOW
075  symbol MS2=C.0  ;Determins stepper motor partial increment.
Output HIGH or LOW
076  symbol SLEEPX=B.7  ;Turns stepper motor ON or OFF. Outut HIGH
or LOW
077  symbol ADC=C.1  ;Analogue voltage input from receiver. Input
~0 to ~5 Volts
078  symbol RELAY=C.7; Optional relay - disconnects sampling
antenna from
079  ;receiver when tuning is completed. Output HIGH or LOW
080
081  ;Initial conditions
082  LOW LED_RED ;Off
083  LOW LED_BLUE ;Off
084  LOW LED_GREEN ;Off
085  LOW BUZZER ;Off
086  LOW STEPX ;Off
087  HIGH MS2 ;HIGH MS2 and LOW MS1 sets the stepper motor to move
at quarter step
088  LOW MS1
089  LOW SLEEPX ;NO power to the stepper motor when LOW
090  LOW DIR ;Direction at the Easy Driver needs to be something to
start,
091  ;which way does not matter
092  LOW RELAY ;Antenna disconnected to start
093  SWEEP=0 ;Sweep counts the number of active tuning tries later
on in the
094  ;program.
095  MANUAL=0 ;MANUAL MODE=1 when autotuner is in manual mode
096
097  NUMBER_STEPS=1640 ;1600 is a full turn for a 0.9 degree
stepper motor turning
098  ;at quarter step. The extra 40 steps accounts for possible
slipage in the
099  ;stepper motor-variable capacitor linkage. (I have not
encountered any.) The
100  ;NUMBER_STEPS variable can be changed when using a stepper
motor of a
101  ;different partial step (e.g. one-half step vs. one-quarter).
102
103  ; Below: pocessor self test and and announce time to start
tuning transmission
104  ;unless manual mode is desired.
105

```

```

for j=1 to 2
106     gosub BLINK_BLUE
107     gosub BLINK_RED
108     gosub BLINK_GREEN
109     gosub BLINK_BUZZER
110 next j
111
112 pause 16000 ;Wait 2 seconds for transmitter tuning signal to
start
113
114 MAX_VOLTS=0 ;Initialize maximum value variable
115
116 HIGH SLEEPX ;Turn on stepper motor power
117 HIGH RELAY ;Connect sampling antenna, if available, to the
receiver
118
119 For j=1 to 20 ;Move variable capacitor away from previous
tuned location.
120     pause 10
121     gosub MOVE
122 next j
123
124 ;Program section below finds maximum value of sample antenna
voltage
125 ;during one full turn of the variable capacitor.
126
127 BEGIN_MAX_VOLTS_SWEEP:
128 SERTXD ("STARTING MAX. VALUE SWEEP AT QUARTER-STEP",CR,LF)
129 for M=1 to NUMBER_STEPS
130     pause 20
131     gosub MOVE
132     gosub READ_AVG_VOLTS
133
134     if AVG_VOLTS>MAX_VOLTS then
135         MAX_VOLTS=AVG_VOLTS
136     endif
137
138     if AVG_VOLTS>10 then
139         sertxd ("TUNER POS: ",#M," VOLTS: ",#AVG_VOLTS, "
HIGHEST YET: ",#MAX_VOLTS,CR,LF)
140     endif
141 next M
142
143 ;Manual Mode triggered by near Zero MAX_VOLTS.
144
145 if MAX_VOLTS<20 then
146     LOW SLEEPX ;Turn off power to stepper motor.
147     MANUAL=1
148     sertxd ("STARTING MANUAL MODE",cr,lf)
149     HIGH LED_GREEN ;Green LED ON indicates autotuner is in
Manual Mode
150     MAX_VOLTS=0
151

```

```

152 REPEAT: ;Manual Mode continually returns to this point
153
154 gosub READ_AVG_VOLTS ;Record Maximum Volts so far
155     if AVG_VOLTS>MAX_VOLTS then
156         MAX_VOLTS=AVG_VOLTS
157     endif
158
159     sertxd ("MANUAL - VOLTS: ",#AVG_VOLTS,"    HIGHEST YET: "
, #MAX_VOLTS, cr, lf)
160     pause 2500 ;slow manual readings to about 3 times per
second
161     goto REPEAT
162 endif
163
164 ;FROM HERE IS ACTIVE AUTOTUNE MODE
165
166 sertxd (CR,LF,"ACTIVE TUNING AT QUARTER-STEP FOR APPROX.
MAX_VOLTS ",#MAX_VOLTS, CR,LF)
167
168 DELTA=MAX_VOLTS/20 ;DELTA is a 5% factor to reduce autotune
target in
169 ;succesive active tuning sweeps if necessary
170
171
172 TRY_ANOTHER_SWEEP: ;This is the point where the program
returns if there
173 ;is more than another active tuning sweep is needed
174
175 SWEEP=SWEEP+1 ;Keeps track of number of active tuning sweeps
that is occuring.
176
177 REF_VOLTS=MAX_VOLTS-DELTA ;Lower autotune target with each
active sweep
178 MAX_VOLTS=REF_VOLTS ; Lowers MAX_Volts to prepare for next
possible
179 ;active sweep
180
181 sertxd ("STARTING ACTIVE TUNING SWEEP: ",#SWEEP," - REF_VOLTS:
",#REF_VOLTS,CR,LF)
182
183 ;Announce Active tuning is about to start for next tuning
sweep - number
184 ;sweep indicated by buzzer beeps and LED blinks
185 for j=1 to SWEEP
186     gosub BLINK_BUZZER
187     gosub BLINK_RED
188     gosub BLINK_BLUE
189 next j
190
191 ;Start of an active tuning sweep process
192 for M=1 to NUMBER_STEPS
193     pause 10
194     gosub MOVE

```

```

195     gosub READ_AVG_VOLTS
196
197     ;Send data to computer, if attached, only for radiated signal
values greater
198     ;than 10
199     if AVG_VOLTS>10 then
200         sertxd ("TUNER POS: ",#SWEEP," / ", #M," TUNED VOLTS:
",#AVG_VOLTS, CR,LF)
201     endif
202
203     ;Check to see if radiated signal is greater than or equal to
the target, if
204     ;YES go to final autotuning stage. If NOT continue the tuning
sweep.
205     if AVG_VOLTS>=REF_VOLTS then
206         goto COMPLETE_TUNING
207     endif
208
209     Next M
210
211     ;If the target radiated signal was not encountered during the
active tuning
212     ;sweep, try again with a lower target for up to four tries.
213     if M>NUMBER_STEPS then
214         if SWEEP<4 then
215             goto TRY_ANOTHER_SWEEP
216         endif
217         sertxd ("* UNABLE TO TUNE",CR,LF)
218         LOW SLEEPX ;Turn off power to stepper motor
219         goto EXIT_TUNING
220     endif
221
222     COMPLETE_TUNING:
223
224     ;Stepping Backward action is below in case the autotuner parts
have
225     ;stored up tourque (even a little). The backwards step
dissipates the
226     ;tourque.
227     pause 20
228     toggle DIR     ;Change direction of stepper motor
229     pause 50
230
231     for j=1 to 4
232         SWEEP_VOLTS=AVG_VOLTS
233         gosub MOVE
234         gosub READ_AVG_VOLTS
235         sertxd ("STEPPING BACK. VOLTS: ",#AVG_VOLTS, CR,LF)
236         if AVG_VOLTS=>REF_VOLTS or SWEEP_VOLTS>AVG_VOLTS then
237             j=4
238         endif
239     next j

```

```

240
241
242 ;Active tuning should be completed now. The sequence below is:
turn off
243 ;power to stepper motor, perform a post tuning radiated signal
diagnostic
244 ;sound buzzer, and light status LEDs.
245
246 LOW SLEEPX ;turn off power to stepper motor
247
248 For J=1 to 8 ;Final diagnostic. Is radiated signal about
stable?
249 gosub READ_AVG_VOLTS
250 sertextd ("FINAL VOLTS: ",#AVG_VOLTS, CR,LF)
251 pause 50
252 Next J
253
254 EXIT_TUNING: ;This junction is used when autotuning has
failed. Radiated
255 ;signal diagnostic (above) is bypassed when tuning is
completed.
256
257 LOW RELAY ;turn off relay thus removing sampling antenna from
reciever
258
259 gosub BLINK_BUZZER
260
261 sertextd ("TUNING FINISHED",CR,LF)
262
263 if MAX_VOLTS>875 then ; Blink red LED for possible receiver
saturation
264     HIGH LED_BLUE ;Blue LED stays ON
265     gosub BLINK_BUZZER ;second buzzer sound to indicate
possible problem
266     sertextd ("* HIGH SIGNAL SATURATION MAY HAVE OCCURED",CR,LF)
267     STAY_HERE_4:
268     gosub BLINK_RED
269     goto STAY_HERE_4
270 endif
271
272 if MAX_VOLTS<45 then ; Blink Bluse LED for signal too low to
tune
273     HIGH LED_RED ;RED LED stays ON
274     gosub BLINK_BUZZER ;;second buzzer sound to indicate
possible problem
275     sertextd ("* SIGNAL TOO LOW TO TUNE",CR,LF)
276     STAY_HERE_3:
277     gosub BLINK_BLUE
278     goto STAY_HERE_3
279 endif
280
281 ;Both red and blue LEDs lit means tuning completed OK
282 HIGH LED_BLUE

```

```

283 HIGH LED_RED
284
285 STAY_HERE_5:
286 goto STAY_HERE_5
287 end
288
289 ;SUBROUTINES ARE BELOW
290
291 ;Blinking LEDs and Buzzer sounds are first
292 BLINK_BUZZER:
293 HIGH BUZZER
294 pause 400
295 LOW BUZZER
296 pause 400
297 return
298
299 BLINK_BLUE:
300 HIGH LED_BLUE
301 pause 400
302 LOW LED_BLUE
303 pause 400
304 return
305
306 BLINK_RED:
307 HIGH LED_RED
308 pause 400
309 LOW LED_RED
310 pause 400
311 return
312
313 BLINK_GREEN:
314 HIGH LED_GREEN
315 pause 400
316 LOW LED_GREEN
317 pause 400
318 return
319
320 ;The MOVE subroutine causes the stepper motor to turn one step
or partial
321 ;step by sending a pulse (low-high-low) to the stepper motor
controller
322 MOVE:
323 pause 10 ;was 20
324 HIGH STEPX
325 pause 10 ;was 20
326 LOW STEPX
327 return
328 ;READ_AVG_VOLTS examines the ADC voltage level at pin C.1 and
interprets
329 ;it as a variable with the range of 0 to 1023, does some
processine then
330 ;provides the value to the rest of the program.
331

```



```
READ_AVG_VOLTS:
332
333  if AVG_VOLTS>10 and MANUAL=0 then 'pause measuring 2/3 second
for
334  ;transmitter SWR foldback function to relent
335      pause 6600
336  endif
337
338  TEMP_VOLTS=0
339  AVG_VOLTS=0
340  VOLTS=0
341
342  ;Voltage is read four times and averaged to smooth
inconsistencies
343  for N=1 to 4
344  pause 10
345  readadc10 ADC,VOLTS
346  TEMP_VOLTS=VOLTS/4
347  AVG_VOLTS=AVG_VOLTS+TEMP_VOLTS
348  next N
349
350  if AVG_VOLTS>875 then; while tuning of possible saturation and
351  ;about 100 watt transmitter power limit
352      gosub BLINK_RED
353      sertxd ("Probable Saturation Measurement of : "
, #AVG_VOLTS, cr, lf)
354  endif
355
356  return
```