

```

1 picaxe laptop demo program slot 0 'by Patrick Leiser
2 #picaxe 28x2
3 #slot 0
4 #revision 2
5 'pinout
6 '
7 reset | rst B.7 | LED (red)
8 | A.0 B.6 | Speakers
9 | A.1 B.5 |
10 | A.2 B.4 |
11 | A.3 B.3 |
12 serin |serin B.2| keyboard wakeup (currently unused)
13 serout |A.4 B.1|
14 0V | 0V B.0|
15 | res +V | +5V
16 | res 0V | 0V
17 SD card busy | C.0 C.7|
18 OLED serout | C.1 C.6|
19 | C.2 C.5|
20 hi2c | C.3 C.4| hi2c (keyboard and SD card)
21 '
22 #define 8mhz 'clock frequency is 8mhz
23 #define 16mhz 'too fast for display, overclock display to 32 Mhz?
24 #ifdef 8mhz
25 setfreq m8 'set clock frequency to 8MHz
26 symbol screenspeed=N2400_8 'N2400 at 8 MHz
27 symbol i2cspeed=i2cfast_8 'i2c 400khz at 8 MHz
28 #endif
29 #ifdef 16mhz
30 setfreq m16 'set clock frequency to 16 MHz
31 symbol screenspeed=N2400_16 'N2400 at 16 MHz
32 symbol i2cspeed=i2cfast_16 'i2c 400khz at 16 MHz
33 #endif
34 'i2c constants
35 symbol alfat=%10100100 'sd card reader
36 symbol keyboard=%01010100 'keyboard reading PICAXE 28X2
37 'system variables
38 symbol screenlevel=bit0 'unused scrolling function
39 symbol tempbit1 =bit1 'temporary bit 1
40 symbol tempbit2=bit2 'tempoary bit 2
41 symbol loopcount=b11 ' number of loops in for...next loops
42 symbol menupage=b21 'which page of menu you are on
43 'keyboard variables
44 symbol keycheck = b4 'will be one when data is valid, otherwise 0 or 255
45 symbol keyraw =b5 'data directly from keyboard
46 symbol keyascii = b6 'data from keyboard converted to ASCII
47 symbol keynum = b7 ' value of number keys 0-9
48 'wordedit variables
49 symbol cursorrow = b8 'part of row in wordedit
50 symbol cursorcol = b9 'part of colum in wordedit
51 symbol cursorpos = b10 'cursor position for OLED
52 '
53 symbol temp1=b14 'temporary byte (8 bits)1
54 symbol temp2=b15 'tempoaray byte 2
55 symbol tempw1=w7 'tempoary word (16 bits) 1 (overlaps temp1 and temp2)
56 symbol temp3=b16 'tempoary byte 3
57 symbol temp4=b17 'tempoary byte 4
58 symbol tempw2=w8 'tempoary word 2 (overlaps temp3 and temp4)
59 symbol temp5=b18 'tempoary byte 5
60 symbol temp6=b19 'tempoary byte 6
61 symbol tempw3=w9 'tempoary word 3 (overlaps temp5 and temp6

```

```

62 symbol charremove=b20           'amount of sd card data to remove eg !00
63 'symbol charremove1=b24         'single character to remove eg !
64 symbol pointertrack=w11        'track pointer read loops
65 symbol first=b24
66 symbol second=b25
67 'game variables
68 'symbol guesscount = b11
69 'symbol hitcount = b0
70 'symbol gamerandom = b12
71 'symbol gamerandom2 =b13
72 'symbol gamerandom3 =b14
73 'symbol gamerandom4 =b15
74 'symbol gamerandom5 =b16
75 'symbol gamerandom6 =b17
76 'symbol gamerandom7 =b18
77 'symbol gamerandom8 =b19
78
79
80 'math variables
81 'symbol math1=w3 'b4/b5
82 'symbol mathone=w3 'b4/b5
83 'symbol math2=w4 'b6/b7
84 'symbol mathtwo=w4 'b6/b7
85 'symbol mathresult=w6'b8/b9
86 'startup
87
88
89 if pinC.4=0 then                 ' if i2c read/write in progress
90     for loopcount = 0 to 15
91         toggle C.3                 'clear i2c bus read/write
92         pause 30
93         if pinC.4=1 then exit
94     next loopcount
95 endif
96 hi2csetup i2cmaster,alfat,i2cspeed,i2cbyte   'setup i2c comunication for SD card reader
97 'pause 200
98 hi2cout ("I M:",If)   'initialise sd card
99
100 setup:
101 hi2cin (b12)
102 if b12=255 then setup2
103 if b12 <> "!" then setup
104 hi2cin (b12,b12,b12)
105
106 hi2cout ("I M:",If)
107 setup2:
108 #ifdef 8mhz
109 'pause 600
110 tune B.6,4,($40,$42,$44,$45,$47,$02,$49)   'startup chime
111 #endif
112 #ifdef 16mhz
113 tune B.6,8,($40,$42,$44,$45,$47,$02,$49)   'startup chime
114 #endif
115 serout C.1, screenspeed, (254,128,254,1)'clear display
116
117 'read voltage
118 'CalibAdc10 tempw1   'read voltage of internal fixed voltage reference
119 gosub getvolts
120 'tempw1 = 10476 / tempw1   'convert to volts
121 if tempw1<38 then
122     gosub powerdisplay

```

```

123     sound B.6, (50,50,100,50)
124     gosub getakey
125 endif
126 #ifdef 8mhz
127 pause 30      'pause 30 ms
128 #endif
129 #ifdef 16mhz
130 pause 70
131 #endif
132
133 .....
134 'menus
135 main:
136 'clearvar:
137 'for bptr=5 to 255
138 '    @bptr=0
139 'next bptr
140 menu:      'main menu
141 serout C.1,screenspeed, (254,128,254,1)  'clear display
142     pause 30
143 select case menupage  'select page
144 case 0  'first page
145     serout C.1,screenspeed, ("1 Utilities",254,192,"2 Tests",254,148,"3 SD card",254,212,"4 favorites ",180)
146     'menu display
147     'serout C.1,N2400,(254,192,"2 Tests")
148     'serout C.1,N2400,(254,148,"3 SD card")
149     'serout C.1,N2400,(254,148,"3 games")
150     'serout C.1,N2400, (254,212,"4 favorites ",180)
151     'serout C.1,N2400, (254,212,"4 other")
152     'serout C.1,N2400,(254,212,"4 calculator")
153 case 1  '2nd page
154     serout C.1,screenspeed,("5 other")
155 else
156     menupage=0
157     goto menu
158 endselect
159 gosub getakey
160 branch keynum, (menu,utilitymenu, testmenu,gamemenu,favoritesmenu,othermenu)      'go to menu item
161 selected by key on keyboard
162 'if keynum = 10 then
163     select case keyraw  'shortcut keys
164     case 29  'w
165         goto wordeditsetup
166     case 45  'r
167         goto sdreadsetup
168     case 75  'l
169         goto sdinitializelist
170     case 33  'c
171         goto calculatorsetup
172     endselect
173 'endif
174 if keyraw=90 or keyraw= 114 then
175     inc menupage
176 elseif keyraw=102 or keyraw=117 then
177     dec menupage
178 endif
179 goto menu
180 utilitymenu:      'utility menu
181 serout C.1,screenspeed, (254,128)

```

```

182 serout C.1, screenspeed, (254,1)
183 pause 30
184 serout C.1,screenspeed, ("1 Wordedit")
185 serout C.1,N2400,(254,192,"2 calculator")
186 'serout C.1,N2400,(254,148,"3 ")
187
188 'serout C.1,N2400,(254,212,"4 ")
189
190 gsub getakey
191 branch keynum,(menu,wordeditsetup,calculatorsetup)
192
193 if keyraw= 118 or keyraw=108 then menu
194 goto utilitymenu
195
196
197 testmenu:      'test menu
198 serout C.1,screenspeed, (254,128)
199 serout C.1, screenspeed, (254,1)
200 pause 30
201 serout C.1,screenspeed, ("1 list characters")
202 serout C.1,screenspeed,(254,192,"2 Keyboard to screen")
203 serout C.1,screenspeed,(254,148,"3 read power supply")
204 serout C.1,screenspeed,(254,212,"4 System info")
205 'serout C.1,N2400, (254,212,"4 SD card read")
206 'serout C.1,N2400,(254,212,"4 calculator")
207
208 gsub getakey
209 branch keynum,(menu,listcharacters,keyboardtoscreen,readpower,sysinfo)',gettime)
210
211 if keyraw=118 or keyraw=108 then menu
212 goto testmenu
213
214 sdmenu:
215 gamemenu:  'sd card menu, used to be game menu
216 'run 1
217 serout C.1,N2400, (254,128,254,1)
218 select case menupage
219 case 0
220 serout C.1, screenspeed, ("1 read sdcard")
221 serout C.1, screenspeed, (254,192,"2 about PLOS")
222 serout C.1, screenspeed, (254,148,"3 add to visitor log")
223 serout C.1, screenspeed, (254,212,"4 view visitor log ",180)
224 'serout C.1,N2400, ("no games, press esc")
225 'serout C.1,N2400,(254,192,"- subtraction")
226 'serout C.1,N2400,(254,148,"* multiplication")
227 'serout C.1,N2400,(254,212,"/ division")
228
229 case 1
230     serout C.1, screenspeed, ("5 list sd files")
231     serout C.1, screenspeed, (254,192,"6 tutorial")
232 else
233     menupage=0
234     goto gamemenu
235 endselect
236 gsub getakey
237 branch keynum,
    (menu,sdreadsetup,sdaboutsetup,visitorlogappend,visitorlogreadsetup,sdinitializelist,tutorialsetup)
238 if keyraw =118 or keyraw=108 then menu
239 if keyraw=90 or keyraw= 114 then
240     inc menupage
241 elseif keyraw=102 or keyraw= 117 then

```

```

242     dec menupage
243 endif
244
245 goto sdmenu
246
247 favoritesmenu:           'shortcuts to sd card files ect menu
248 serout C.1,screenspeed, (254,128,254,1)
249 pause 30
250 serout C.1,screenspeed, ("1 ")
251
252 gosub getakey
253 branch keynum, (menu)
254
255 if keyraw = 118 or keyraw =108 then menu
256 goto favoritesmenu
257
258 othermenu:
259 serout C.1,screenspeed, (254,128)
260 serout C.1, screenspeed, (254,1)
261 pause 30
262 serout C.1,screenspeed, ("1 star animation")
263
264 gosub getakey
265 branch keynum, (menu,staranimatesetup)
266
267 if keyraw = 118 or keyraw =108 then menu
268 goto othermenu
269 .....
270 'sub program 1: wordedit
271 'a text editor like pages or word
272 wordeditsetup:
273 serout C.1, screenspeed, (254,14,254,128,254,1) 'cursor on, top of display, clear display
274 pause 30
275 cursorrow=0      'top of display
276 cursorcol=0     'top of display
277 wordedit:
278     gosub getakey      'get a key from keyboard
279
280     select case keyraw      'select raw key value
281     case 118                'esc
282         goto menu
283     case 108                'home
284         cursorrow=255
285         cursorcol=0
286     case 90                 'enter
287         cursorrow = 21
288     case 102                'backspace
289         if cursorrow = 0 then
290             dec cursorcol
291             cursorrow = 20
292         endif
293         cursorrow=cursorrow-1
294         gosub position
295             cursorrow=cursorrow-1
296         serout C.1,screenspeed, (" ")
297     case 13                 'tab
298         cursorrow=cursorrow+2
299         serout C.1, screenspeed, (" ")
300         @bptrinc=9         'tab
301         @ptrinc=9
302         @bptrinc=0         'null

```

```

303 @bptr=0 'null
304 case 107 'left arrow
305 if cursorrow = 0 then
306     dec cursorcol
307     cursorrow = 20
308 endif
309 cursorrow=cursorrow - 2
310 case 116 'right arrow
311 case 114 'down arrow
312 inc cursorcol
313 dec cursorrow
314 case 117 'up arrow
315 dec cursorcol
316 dec cursorrow
317 case 124
318     serout C.1, screenspeed, (254,128,"print screen")
319     goto printscreen
320 else
321     serout C.1,screenspeed, (keyascii)
322     @bptr=keyascii 'store contents of the screen for ability to save
323     @ptrinc=keyascii 'store contents of the screen for ability to save in progress
324 endselect
325
326 inc cursorrow
327 gosub position
328 goto wordedit
329 position: 'update position of cursor
330 if cursorrow > 19 then
331     inc cursorcol
332     cursorrow=0
333 endif
334 select case cursorcol
335 case 1
336     cursorpos=192
337     'serout C.1,N2400,(254,192)
338     'serout C.1,N2400,("ln 1")
339 case 2
340     cursorpos=148
341     'serout C.1,N2400,(254,148)
342     'serout C.1,N2400,("ln 2")
343 case 3
344     cursorpos=212
345     'serout C.1,N2400,(254,212)
346     'serout C.1,n2400,("ln 4")
347 else
348     cursorpos=128
349     'serout C.1,N2400,(254,128)
350     'serout C.1,N2400,("ln 0")
351     cursorcol=0
352 endselect
353 cursorpos = cursorrow+cursorpos
354 bptr=cursorpos
355 serout C.1, screenspeed, (254,cursorpos)
356 return
357 goto wordedit
358
359
360 printscreen: ' save the contents of the screen to SD card
361 hi2cout [alfat],("O 1A>M:\\screenshots",92) '92=\ \=\
362 serout C.1, screenspeed, (254,128,"name (not full path)")
363 gosub sdout

```

```

364 pointertrack= ptr
365
366 first = pointertrack / 16 + "0"      '4 msb plus ascii "0"
367 If first > "9" Then                 'if more than "9"
368     first = first + 7                'add seven (starting "A","B",etc)
369 EndIf
370
371 second = pointertrack & $0F + "0"
372 If second > "9" Then
373     second = second + 7
374 EndIf
375 hi2cout ("W 1>",first,second,lf) 'write handle 1, pointertrack characters,
376 sertxd (#pointertrack)
377 hi2cin (b12,b12,b12,b12)
378 for ptr=0 to pointertrack
379
380
381     hi2cout (@ptr)
382     sertxd (@ptr,#ptr)
383     for loopcount = 0 to 14
384         hi2cin (b12)
385     next loopcount
386 next ptr
387 'for bptr=128 to 147      'line 1
388 '     hi2cout ("W 1>1",lf) 'write handle 1, one character,
389 '     hi2cin (b12)
390 '     serout C.1,screenspeed, (b12)
391 '     hi2cin (b12)
392 '     serout C.1,screenspeed, (b12)
393 '     hi2cin (b12)
394 '     serout C.1,screenspeed, (b12)
395 '     hi2cin (b12)
396 '     serout C.1,screenspeed, (b12)
397 '     hi2cout(@bptr)
398 '     for loopcount = 0 to 14
399 '         hi2cin (b12)
400 '     next loopcount
401 'next bptr
402 'for bptr=192 to 211     'line 2
403 '     hi2cout ("W 1>1",lf)
404 '     hi2cin (b12,b12,b12,b12)
405 '     hi2cout(@bptr)
406 '     for loopcount = 0 to 14
407 '         hi2cin (b12)
408 '     next loopcount
409 'next bptr
410 'for bptr=148 to 167     'line 3
411 '     hi2cout ("W 1>1",lf)
412 '     hi2cin (b12,b12,b12,b12)
413 '     hi2cout(@bptr)
414 '     for loopcount = 0 to 14
415 '         hi2cin (b12)
416 '     next loopcount
417 'next bptr
418 'for bptr=212 to 241     'line 4
419 '     hi2cout ("W 1>1",lf)
420 '     hi2cin (b12,b12,b12,b12)
421 '     hi2cout(@bptr)
422 '     for loopcount = 0 to 14
423 '         hi2cin (b12)
424 '     next loopcount

```

```

425 'next bptr
426
427 hi2cout ("C 1",lf)
428 hi2cout ("C 1",lf)
429 hi2cin (b12,b12,b12,b12)
430
431 serout C.1, screenspeed, (254,128,"complete")
432 gosub getakey
433 goto wordeditsetup
434 .....
435 'sub program 2: keyboard to screen
436 keyboardtoscreen: 'show information about key on screen
437 gosub getakey 'read a key
438 serout C.1,screenspeed, (254,128,254,1) 'clear screen
439 serout C.1,screenspeed, (#keyraw," ",keyascii," ",#keyascii," ",#keynum) 'display information about key
440 if keyraw = 118 then 'if escape
441     #ifdef 8mhz
442     pause 1000 'pause a second
443     #endif
444     #ifdef 16mhz
445     pause 2000 '1 sec
446     #endif
447     goto menu 'go to menu
448 endif
449 goto keyboardtoscreen 'loop
450 .....
451
452 'sub program 3: list characters
453 listcharacters: 'list caracters avaiable for display
454 serout C.1,screenspeed,(254,128) ; move to start of first line
455 for b0 = 0 to 252 'all characters, no commands
456     serout C.1,screenspeed,(b0," ") 'display character and space
457     inc cursorrow 'next position on display
458     gosub position
459     gosub checkforkey 'check if esc has been pressed
460     if keyraw= 118 then menu
461 next b0
462
463 goto listcharacters
464
465 'gettime: 'unused routine to retrive time
466 'serout C.1, screenspeed, (254,128,254,1)
467 'pause 30
468 'hi2cout ("G D",lf)
469 'for loopcount =0 to 11
470 '    hi2cin (b12)
471 '    serout C.1, screenspeed, (b12)
472 'next loopcount
473 'serout C.1, screenspeed, (254,192)
474 'hi2cout ("G T",lf)
475 'for loopcount =0 to 8
476 '    hi2cin (b12)
477 '    serout C.1, screenspeed, (b12)
478 'next loopcount
479 'gosub getakey
480 'goto menu
481 'settime:
482 'hi2cout ("T B",lf)
483 'hi2cin (b12,b12,b12,b12)
484 'pause 200
485 'hi2cout ("S 42566DA0",lf)

```



```

486 'hi2cin (b12,b12,b12,b12)
487 'goto menu
488 .....
489 'voltage reader
490 readpower:
491 serout C.1,screenspeed, (254,128,254,1) 'clear display
492 pause 30
493 gosub getvolts
494 gosub powerdisplay
495 gosub getakey
496 goto main
497 getvolts:
498 CalibAdc10 tempw1
499 tempw1 = 10476 / tempw1
500 bintoascii tempw1,temp3,temp3,temp4 'lower byte of temp w1
501 return
502 powerdisplay:
503 'tempw1 = 52377 /tempw1 * 2
504 'serout C.1,screenspeed, (#tempw1)
505 serout C.1, screenspeed, ("power=",temp3,".",temp4," V")
506 serout C.1, screenspeed, (254,192,"full charge=4.5 V")
507 return
508 'gosub getakey
509 'goto main
510 .....
511 sdaboutsetup:
512 pause 30
513 hi2cout [alfat],("O OR>M:\\system\\about.txt",lf) '\\= \
514 goto presdread
515
516 sdreadsetup: 'read file at path indicated
517 serout C.1, screenspeed, (254,1) 'clear display
518 pause 30 'wait for display to clear
519 serout C.1, screenspeed, (254,128,"full file path",254,192) 'prompt user for full file path
520 hi2cout [alfat],("O OR>M:",92) 'open a file in handle 0, read mode with a path starting with M:\
521 gosub sdout 'gosub sdout (down two lines)
522 goto presdread 'goto sdreadsetup2
523 sdout: 'let the user choose file path
524 do 'start do loop
525 gosub getakey 'get a key
526 select case keyraw 'choose the value of raw key value
527 case 90
528     exit
529 case 93
530     hi2cout (92) '\\
531     serout C.1,screenspeed, (218) '\\
532 case 74, 76,84,91 'illegal characters
533 case 118 'escape or home
534     hi2cout (lf)
535     pause 50
536     hi2cout ("C 0",lf)
537
538 else
539     serout C.1, screenspeed, (keyascii)
540     hi2cout (keyascii)
541 endselect
542 loop
543 hi2cout (lf)
544 return
545
546 presdread: 'setup for read

```

```

547 serout C.1,screenspeed, (254,128,254,1) 'clear screen
548 pause 30 'wait for display to clear
549 cursorrow=0 'first character of screen
550 cursorpos=0 'top of screen
551 gosub position 'update position
552 sdread: 'read SD card
553 inc tempw3
554 'sertxd (" ",#tempw3," ")
555 'bug: read stops after 125th character.
556 cursorrow=0
557 gosub position
558 hi2cin (b12,b12,b12)
559 hi2cout ("R 0",23,">1",lf) 'read one byte of file '23=end of transmit block in ASCII
560 'serout C.1, N2400, (254,128,254,1) 'go to first line
561 pause 30
562 'sertxd ("R")
563 charactercutoff: 'cutoff error/sucess codes
564 do
565 hi2cin(b12) 'read a byte
566 loop until b12="!" 'loop until end of error codes
567 'sertxd ("1")
568 if loopcount >= 4 then
569 loopcount=loopcount+1 max 5
570 goto charactercutoff
571 if loopcount < 4 then charactercutoff 'loop four times
572 endif
573 if charremove>0 then 'if charremove is more than 0 loop that many times
574 dec charremove
575 'sertxd ("2")
576 goto charactercutoff
577 if tempbit1=1 then
578 ' tempbit1=0
579 ' goto charactercutoff
580 endif
581 if tempbit1=0 then
582 ' hi2cin (b12)
583 ' tempbit1=1
584 endif
585 hi2cin (b12,b12,b12)',b12,b12)',b12)
586 'sertxd ("L")
587 if tempbit2=0 then
588 ' hi2cin (b12)
589 ' tempbit2=1
590 endif
591 'serout C.1, N2400, (254,128) 'output to display
592 cursorrow=0
593 cursorcol=0
594 gosub position
595 do
596 hi2cin (b12)
597 select case b12
598 case "$"
599 exit
600 case lf,cr 'enter
601 cursorrow=0
602 inc cursorcol
603 if cursorcol >=4 then sdreadpageend 'end of page
604 case 23 'end of transmission
605 gosub getakey 'wait for keypress
606 hi2cout ("C 0",lf) 'close handle
607 goto menu 'goto menu

```

```

608 case 9
609     cursorrow=cursorrow+3
610 case 7     'bell
611     ' bell tone
612 tune B.6,8,($02,$07)
613 case <31
614 endselect
615 gosub position
616 serout C.1, screenspeed, (b12)
617 'elseif b12=10 then           'enter
618     '   inc cursorcol
619     '   cursorrow=0
620     '   gosub position
621 'if b12 <> "$" then
622 'endif
623 loop
624 charremove=1
625
626 inc cursorrow
627 if cursorrow>=20 and cursorcol >=3 then sdreadpageend 'end of page
628 if cursorcol >= 4 then sdreadpageend 'end of page
629 gosub position
630 'hi2cin (b12,b12,b12,b12)
631 goto sdread
632 sdreadpageend:           'end of page
633 do
634 gosub getakey
635 if keyraw=90 then           'if key is enter then
636     tempbit1=1
637     'charremove=charremove+1
638     tempbit2=1
639     cursorrow = 0 'top of display
640     cursorcol = 0 'top of display
641     gosub position ' update position
642     goto presdread           'goto next page
643 elseif keyraw=125 then
644     hi2cout ("P 0>0",lf)
645     goto presdread
646 'elseif keynum<10 then
647     'keynum=keynum*50
648     'gosub pageselect
649     hi2cout ("P 0> ",#b7,lf)
650     '   tempbit1=1
651     '   charremove=1
652     '   tempbit2=1
653     '   goto presdread
654 elseif keyraw=118 then 'escape or
655     hi2cout ("C 0",lf)
656     hi2cin (b12,b12,b12,b12)
657     'for loopcount= 0 to 17
658     'hi2cin (b12)
659     'next loopcount
660     goto menu
661 endif
662 loop
663
664
665 visitorlogappend:           'add to visitor log
666 serout C.1, screenspeed,(254,1) 'clear screen
667 pause 30'wait for screen to clear
668 do

```

```

669 '    pause 100
670 loop until pinC.0=0    'wait for buisy pin to go low
671 hi2cout [alfat],("O 1A>M:\\data\\visitor log.txt",lf)'open M:data\visitor log.txt in write mode file handle 1
672 hi2cin (b12,b12,b12,b12) 'read error code
673 serout C.1, screenspeed, (254,128,"initials?") 'prompt user for initials
674 gosub getakey 'get a key
675 serout C.1, screenspeed,(254,148,keyascii, ".") 'display key
676 b13=keyascii 'store data in b13
677 gosub getakey 'get a key
678 serout C.1, screenspeed,(keyascii, ".") 'display key
679 do
680 'pause 100
681 loop until pinC.0=0    'wait until busy bin is low
682 hi2cout ("W 1>5",lf) 'write handle 1, five characters
683 'pause 10
684 for loopcount=0 to 19 'repeat 20 times
685 hi2cin (b12) 'read a character
686 serout C.1, screenspeed, (b12) 'output to screen
687 inc cursorrow 'move cursor
688 gosub position 'update cursor position
689 next loopcount
690 'pause 1000
691 hi2cout (b13, ".",keyascii, ". ") 'write initials to SD card
692 serout C.1,screenspeed, (254,192) 'begining of display
693 'pause 1000
694 hi2cout ("C 1",lf) 'close file handle
695 pause 100
696 for loopcount=0 to 26 'repeat 27 times
697 hi2cin (b12) 'read character
698 serout C.1, screenspeed, (b12) 'output to display
699 'sertxd("3")
700 next loopcount 'repeat
701 serout C.1, screenspeed, ("complete") 'output to display "complete"
702 pause 1000 'pause a second
703 'toggle b.7
704 goto menu 'goto main menu
705
706
707 visitorlogreadsetup: 'read visitor log
708 'pause 30
709 hi2cout [alfat],("O 0R>M:\\data\\visitor log.txt",lf) 'open file on handle 0, read mode, path M:\data\visitor
log.txt
710 'pause 500
711 goto presdread 'setup for sd card read
712
713 tutorialsetup: 'read tutorial
714 'pause 30
715 hi2cout [alfat],("O 0R>M:\\system\\tutorial.txt",lf) 'open handle 0, read mode, file path
M:\system\tutorial.txt
716 'pause 500
717 goto presdread 'setup for sd card read
718
719 sdinitializelist: 'initalize list of files
720 serout C.1, screenspeed, (254,128,254,1) ' top of display, clear display
721 temp1=0
722 ptr=0
723 loopcount=0
724 tempbit1=0
725 pause 30
726 cursorrow=0
727 cursorcol=0

```

```

728 hi2cout [alfat],("@ M:",92)
729 hi2cin (b12,b12,b12)
730 serout C.1, screenspeed, ("file path:",254,192)
731 gsub sdout
732 serout C.1, screenspeed, (254,148,"press any key for",254,212,"manual mode")
733 for loopcount=0 to 3
734 pause 500
735 gsub checkforkey
736 if keyraw<> 0 then 'if key is pressed
737     tempbit1=1
738     exit
739 next loopcount
740 endif
741 listfiles:
742 serout C.1, screenspeed, (254,1) 'clear display
743 cursorrow=0
744 cursorcol=0
745 gsub position
746 hi2cout ("N",lf)
747 hi2cin (b12,b12,loopcount)
748 if b12="0" and loopcount="4" then 'if end of menu
749     hi2cin (b12)
750     goto menu
751 endif
752 loopcount=0
753 hi2cin (b12)
754 do
755     hi2cin (b12)
756     if b12= lf then 'next data field
757         inc cursorcol
758         cursorrow=0
759         inc loopcount
760
761     endif
762     'if loopcount >1 then' and loopcount <4 then
763     if b12<> "!" then 'if not end of command
764         serout C.1, screenspeed, (b12) 'output to display data received
765         if cursorcol=0 then
766             @ptr=b12'inc=b12
767             sertxd (@ptrinc)
768         endif
769         inc cursorrow
770         gsub position
771     else
772         hi2cin (b12,b12,b12)
773         cursorrow=0
774         cursorcol=0
775         gsub position
776         if tempbit1=0 then 'if automatic mode
777             #ifdef 8mhz
778                 pause 1000 'pause 1 second
779             #endif
780             #ifdef 16mhz
781                 pause 2000 '1000 at 16 mhz
782             #endif
783             gsub checkforkey
784             if keyraw=90 then 'enter
785                 ptr=0
786                 hi2cout [alfat],("O OR>M:",92) 'open a file in handle 0, read mode with a path starting with M:\
787                 do
788                     if @ptr=0 then

```

```

789         exit
790     endif
791     hi2cout (@ptrinc)
792 loop
793 hi2cout (lf)
794 goto presdread
795 endif
796     if keyraw=118 then menu
797     goto listfiles
798 endif
799 gosub getakey
800 if keyraw=90 then 'enter
801 ptr=0
802 hi2cout [alfat],("O OR>M:",92) 'open a file in handle 0, read mode with a path starting with M:\
803 do
804     if @ptr=0 then
805         exit
806     endif
807     sertxd (@ptr)
808     hi2cout (@ptrinc)
809 loop
810 hi2cout (lf)
811 goto presdread
812 endif
813 if keyraw= 118 then menu
814 'pause 1000
815 'sertxd ("@" )
816 goto listfiles
817 endif
818 if loopcount=4 then
819     exit
820 endif
821 loop
822 gosub getakey
823
824
825 if keyraw= 118 then menu 'if esc goto menu
826 'sertxd ("!")
827 goto listfiles 'repeat
828
829 .....
830 'calculator
831 calculatorsetup:
832 tempw1=0 'clear variables
833 tempw2=0 'clear variables
834 serout C.1, screenspeed, (254,128,254,14,254,1) 'top of display, cursor on, clear display
835 pause 30
836 calculatoroperand1:
837 gosub getakey 'get a key
838 if keynum>= 10 and keyraw<> 90 and keyraw<>118 then 'not number, not enter, not escape
839     'serout C.1,screenspeed, ("other")
840     goto calculatoroperand1
841
842 elseif keyraw= 118 then 'esc
843     goto menu
844 elseif keyraw= 90 then 'enter
845     goto calculatoroperator 'select operator (*+/- ect)
846 'elseif keyraw=102 then 'backspace
847 '     tempw1=tempw1/10 'remove character
848 '     serout C.1,screenspeed, (254,128,#tempw1,"")
849 '     goto calculatoroperand1

```

```

850 else
851     tempw1=tempw1*10 'shift left allowing for another digit
852     tempw1=tempw1+keynum 'add new digit
853     serout C.1, screenspeed, (254,128,#tempw1) 'display first operand
854     goto calculatoroperand1 'recive next digit
855 endif
856 calculatoroperator: 'select operator (+-* / ect)
857 serout C.1, screenspeed, (254,192)', "op")
858 gosub getakey
859 select case keyraw
860 case 85 '+' add
861     temp5=0
862     serout C.1,screenspeed, ("+")
863 case 78 '-'
864     temp5=1
865     serout C.1,screenspeed, ("-")
866 case 62 '*' multiply
867     temp5=2
868     serout C.1,screenspeed, ("*")
869 case 74 '/' divide
870     temp5=3
871     serout C.1,screenspeed, ("/")
872 case 54 '^ square root
873     temp5=4
874     serout C.1,screenspeed, ("square root")
875     goto calculate 'unary operator, skip reciving second value
876 case 45 'r random
877     temp5=5
878     goto calculate
879 case 118 'esc
880     goto menu
881 else
882     goto calculatoroperator
883 endselect
884 calculatoroperand2:
885 serout C.1,screenspeed, (254,148)
886 gosub getakey
887 if keynum >= 10 and keyraw <> 90 and keyraw <> 118 then 'not number, not enter
888     goto calculatoroperand2
889 elseif keyraw= 118 then
890     goto menu
891 elseif keyraw= 90 then
892     'serout C.1, screenspeed, ("enter")
893     goto calculate
894 'elseif keyraw=102 then
895 '     tempw2=tempw2/10
896 '     serout C.1,screenspeed, (254,148,#tempw2," ")
897 '     goto calculatoroperand2
898 else
899     tempw2=tempw2*10
900     tempw2=tempw2+keynum
901     serout C.1, screenspeed, (254,148,#tempw2)
902     goto calculatoroperand2
903 endif
904 goto calculatoroperand2
905 calculate:
906 select case temp5
907 case 0
908     let tempw1=tempw1+tempw2
909     serout C.1,screenspeed, ("+")
910 case 1

```

```

911     let tempw1=tempw1-tempw2
912     serout C.1,screenspeed, ("-")
913 case 2
914     let tempw1=tempw1*tempw2
915     serout C.1,screenspeed, ("*")
916 case 3
917     let tempw1=tempw1/tempw2
918     serout C.1,screenspeed, ("/")
919 case 4
920     let tempw1= SQR tempw1
921 case 5
922     if tempw1=0 then
923     touch16 0, temp1
924     touch16 0, temp2
925     endif
926     random tempw1
927 'else
928     serout C.1, screenspeed, ("error")
929 endselect
930 serout C.1,screenspeed, (254,212,#tempw1)',254,192,#tempw1)
931 do
932     gosub getakey
933     if keyraw=90 then
934         serout C.1, screenspeed, (254,1)
935         pause 30
936         serout C.1, screenspeed, (254,128,#tempw1)
937         tempw2=0
938         goto calculatoroperator
939     elseif keyraw=118 then
940         goto menu
941     elseif keyraw=102 then
942         goto calculatorsetup
943     endif
944 loop
945
946
947 'guesslocation:  'inefficeint and complicated, may come back to later
948 '     serout C.1, N2400, (254,14,254,128)
949 '     gosub getakey
950
951 '     readadc 17,gamerandom
952 '     readadc 17,gamerandom2
953 '     readadc 17,gamerandom3
954 '     readadc 17,gamerandom4
955
956 'guesslocation1:
957 '     random gamerandom
958 '     serout C.1, N2400, ("1")
959 '     select case gamerandom
960 '     case <128
961 '         goto guesslocation1
962 '     case <168
963 '         goto guesslocation2
964 '     case <192
965 '         goto guesslocation1
966 '     case <232
967 '         goto guesslocation2
968 '     else
969 '         goto guesslocation1
970 '     endselect
971

```



```

972 'guesslocation2:
973     random gamerandom2
974     serout C.1, N2400, ("2")
975     select case gamerandom2
976     case <148
977         goto guesslocation2
978     case <168
979         goto guesslocation3
980     case <192
981         goto guesslocation2
982     case <232
983         goto guesslocation3
984     else
985         goto guesslocation2
986     endselect
987
988 'guesslocation3:
989     random gamerandom3
990     serout C.1, N2400, ("3")
991     select case gamerandom3
992     case <148
993         goto guesslocation3
994     case <168
995         goto guesslocation4
996     case <192
997         goto guesslocation3
998     case <232
999         goto guesslocation4
1000    else
1001        goto guesslocation3
1002    endselect
1003
1004 'guesslocation4:
1005     random gamerandom4
1006     serout C.1, N2400, ("4")
1007     select case gamerandom4
1008     case <148
1009         goto guesslocation4
1010     case <168
1011         goto guesslocationplay
1012     case <192
1013         goto guesslocation4
1014     case <232
1015         goto guesslocationplay
1016     else
1017         goto guesslocation4
1018     endselect
1019
1020 'guesslocationplay:
1021     if hitcount= 15 then
1022         serout C.1,N2400,(254,128,"You win!",254,192,"it took ",#guesscount,"guesses")
1023         goto gamemenu
1024     endif
1025     gosub getakey
1026     select case keyraw
1027     case 118             'esc or home
1028         serout C.1, N2400, (254,1)
1029         pause 30
1030         goto gamemenu
1031     case 90             'enter
1032         goto locationcheck

```

```

1033 |     case 107                               'left arrow
1034 |         if cursorrow = 0 then
1035 |             dec cursorcol
1036 |             cursorrow = 20
1037 |         endif
1038 |         cursorrow=cursorrow - 1
1039 |     case 116                               'right arrow
1040 |         inc cursorrow
1041 |     case 114                               'down arrow
1042 |         inc cursorcol
1043 |         'dec cursorrow
1044 |     case 117                               'up arrow
1045 |         dec cursorcol
1046 |         'dec cursorrow
1047 |     endselect
1048 |     gosub position
1049 |         inc guesscount
1050 |     goto guesslocationplay
1051 |
1052 |'locationcheck:
1053 |     gosub position
1054 |         select case cursorpos
1055 |         case gamerandom
1056 |             serout C.1, N2400, ("X")
1057 |             bit8=1
1058 |             goto guesslocationplay
1059 |         case gamerandom2
1060 |             serout C.1, N2400, ("X")
1061 |             bit9=1
1062 |             goto guesslocationplay
1063 |         case gamerandom3
1064 |             serout C.1, N2400, ("X")
1065 |             bit10=1
1066 |             goto guesslocationplay
1067 |         case gamerandom4
1068 |             serout C.1, N2400, ("X")
1069 |             bit11=1
1070 |             goto guesslocationplay
1071 |         else
1072 |             serout C.1, N2400, ("O")
1073 |             inc cursorrow
1074 |             goto guesslocationplay
1075 |         endselect
1076 |         goto guesslocationplay
1077 |
1078 | .....
1079 | sysinfo:  'system info
1080 | serout C.1, screenspeed, (254,128,254,1)
1081 | pause 30
1082 | readrevision temp1
1083 | serout C.1, screenspeed, ("Revision #",#temp1)
1084 | readfirmware temp1
1085 | serout C.1, screenspeed, (254,192,"Firmware #",#temp1)
1086 | readsilicon temp1
1087 | serout C.1, screenspeed, (254,148,"Silicon #",#temp1)
1088 | 'readinternaltemp IT_4V5, 0 ,temp1      'M2 only, not X2
1089 | 'serout C.1, screenspeed, (254,148,"CPU temp",#temp1)
1090 | gosub getakey
1091 | goto menu
1092 | .....
1093 | staranimatesetup:  'star animation

```

```

1094 serout C.1, screenspeed, (254,14) 'turn on cursor
1095 'staranimate:
1096 do
1097 serout C.1, screenspeed, (" ")
1098 inc cursorrow
1099 gosub position
1100 gosub checkforkey
1101 if keyraw = 118 then
1102     exit
1103 endif
1104 loop
1105 goto menu
1106 .....
1107 #rem                                     'inefficient and complicated, may come back to later
1108 calculatormenu:
1109 serout C.1,N2400, (254,128)
1110 serout C.1, N2400, (254,1)
1111 pause 30
1112 serout C.1,N2400, ("+ addition")
1113 serout C.1,N2400,(254,192,"- subtraction")
1114 serout C.1,N2400,(254,148,"* multiplication")
1115 serout C.1,N2400,(254,212,"/ division")
1116 gosub getakey
1117 select case keyraw
1118 case 121                                     '+'
1119     serout C.1, N2400, (254,1)
1120     pause 30
1121     goto add
1122 case 123                                     '-'
1123     serout C.1, N2400, (254,1)
1124     pause 30
1125     goto subtract
1126 case 124                                     '*'
1127     serout C.1, N2400, (254,1)
1128     pause 30
1129     goto multiply
1130 case 74
1131     serout C.1, N2400, (254,1)
1132     pause 30
1133     goto divide
1134 case 118
1135     goto menu
1136 else
1137     goto calculatormenu
1138 endselect
1139 add:
1140     gosub getakey
1141     if math1<6500 then
1142         math1=math1*10
1143         select case keyraw
1144         case 22
1145             math1=math1+1
1146         case 30
1147             math1=math1+2
1148         case 38
1149             math1=math1+3
1150         case 37
1151             math1=math1+4
1152         case 46
1153             math1=math1+5
1154         case 54

```

```

1155         math1=math1+6
1156     case 61
1157         math1=math1+7
1158     case 62
1159         math1=math1+8
1160     case 70
1161         math1=math1+9
1162     case 69
1163     endselect
1164     else goto add
1165     endif
1166
1167 subtract:
1168 multiply:
1169 divide:
1170 goto calculatormenu
1171 #endrem
1172
1173 .....,.....
1174 checkforkey:           'check for a key, return even if no key pressed
1175 'hi2csetup i2cmaster,%01010100,i2cfast,i2cbyte 'setup i2c master with keyboard slave
1176 hi2cin [keyboard],0,(keycheck,keyraw,keyascii,keynum)
1177
1178     if keycheck<> 1 then
1179         keyraw=0
1180         keyascii=0
1181         keynum= 10
1182     else
1183         hi2cout 0,(2)
1184     endif
1185
1186 hi2csetup i2cmaster, alfat, i2cspeed, i2cbyte
1187 return
1188 .....,.....
1189 getakey:               'key detecting subroutine
1190 'hi2csetup i2cmaster,%01010100,i2cfast,i2cbyte 'setup i2c master with keyboard slave
1191 'sertxd("&")
1192 hi2cin [keyboard],0,(keycheck,keyraw,keyascii,keynum) 'get keyboard data formatted by keyboard
processor
1193 'sertxd(#b23,",",#b24,cr,lf)
1194 if keycheck<> 1 then 'if ready byte is not equal to 1
1195     'keyraw = 0
1196     'sertxd("#",#keycheck)
1197     pause 70 'pause 70 ms
1198     goto getakey 'loop
1199 else
1200     'sertxd("got a key")
1201     'sertxd("+")
1202     hi2cout 0,(2) 'reset keyboard status byte
1203 endif
1204 if keynum=11 then powersaversetup
1205 'sertxd("$")
1206 hi2csetup i2cmaster, alfat, i2cspeed, i2cbyte 'setup i2c for SD card reader
1207 'sertxd("-")
1208 return 'return to subroutine call
1209 .....,.....
1210 'go to power saving mode
1211 'powersaversetup: 'unused power saving routine
1212 'gosub scrollright
1213 'powersaver:
1214 'serout C.1, screenspeed, ("sleeping")

```

```

1215 'hintsetup %01000100
1216 'sleep 14                '65535 =~38 hours
1217 'high b.7
1218 'pause 250
1219 'low b.7
1220 'if pinB.2=0 then powersaver
1221 'serout C.1,screenspeed,("awake")
1222 'pause 100
1223 'gosub scrollleft
1224 'return
1225 .....
1226 'scrollright:   'unused scrolling routines
1227 'if screenlevel=0 then
1228 'for loopcount=0 to 19
1229 '    serout C.1,screenspeed,(254,24,254,168)
1230 'next loopcount
1231 'screenlevel=1
1232 'endif
1233 'return
1234
1235 'scrollleft:    'unused scrolling routine
1236 'if screenlevel=1 then
1237 'for loopcount=0 to 19
1238 '    serout C.1,screenspeed,(254,28)
1239 'next loopcount
1240 'screenlevel=0
1241 'endif
1242 'return
1243
1244 .....
1245 'pageselect:    'unused sd card page selecting routine
1246 'select case keynum
1247 'case 0
1248 '    temp1="0"
1249 '    temp2="0"
1250 '    'temp3="0"
1251 'case 1
1252 '    temp1="0"
1253 '    temp2="5"
1254 '    'temp3="0"
1255 'case 2
1256 '    temp1="0"
1257 '    temp2="A"
1258 '    'temp3="0"
1259 'case 3
1260 '    temp1="0"
1261 '    temp2="F"
1262 '    'temp3="0"
1263 'case 4
1264 '    temp1="1"
1265 '    temp2="4"
1266 '    'temp3="0"
1267 'case 5
1268 '    temp1="1"
1269 '    temp2="9"
1270 '    'temp3="0"
1271 'case 6
1272 '    temp1="1"
1273 '    temp2="E"
1274 '    'temp3="0"
1275 'case 7

```

```
1276 '    temp1="2"
1277 '    temp2="3"
1278 '    temp3="0"
1279 'case 8
1280 '    temp1="2"
1281 '    temp2="8"
1282 '    temp3="0"
1283 'case 9
1284 '    temp1="2"
1285 '    temp2="D"
1286 '    temp3="0"
1287 'endselect
1288 'return
1289 #ifdef 8mhz
1290 #ifdef 16mhz
1291 #error "both 8 and 16 mhz defined"    'if both clock frequencys defined create error message
1292 #endif
1293 #endif
1294
1295 #ifndef 8mhz
1296 #ifndef 16mhz
1297 #error "frequency not defined"    'if neither clock frequency defined create error message
1298 #endif
1299 #endif
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
```

```

1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360 'keyboard.bas keyboard monitoring program for PLOS laptop      by Patrick Leiser
1361 #picaxe 28x2
1362 symbol shift=bit0
1363 symbol caps=bit1
1364 symbol keyraw=b5
1365 symbol keycheck=b1
1366 symbol keyascii=b2
1367 symbol keynum=b3
1368 symbol powerstate=b4
1369 symbol repeatkey=b6
1370 symbol loopcount=w4
1371 EEPROM $00,("9?5312C?A864?~?")      ' Function keys
1372 EEPROM $10,("????Q!???ZSAW@?")    ' Main keyboard keys
1373 EEPROM $20,("?CXDE$??? VFTR%?")
1374 EEPROM $30,("?NBHGY^???MJU&*?")
1375 EEPROM $40,("?<KIO)(??>?L:P_?")
1376 EEPROM $50,("?",34,"{+????",181,"}|?") "'?'[=?????]?")  "\\=\ 34="
1377 EEPROM $60,("?",218,????",127,"?1?",127,"????")    ' Numeric keypad keys
1378 EEPROM $70,("0.",180,"5",126,179,??B+3-*9??")' * = print screen
1379 TABLE $00,("9?5312C?a864?")      ' Function keys
1380 TABLE $10,("????q1???zsaw2?")    ' Main keyboard keys
1381 TABLE $20,("?cxde43?? vfr5?")
1382 TABLE $30,("?nbhgy6???mju78?")
1383 TABLE $40,("?kio09??./!;p-?")
1384 TABLE $50,("?''[=????",181,"]?",92,"?") "'?'[=?????]?")  "\\=\
1385 TABLE $60,("?",218,????",127,"?1?",127,"????")    ' Numeric keypad keys
1386 TABLE $70,("0.",180,"5",126,179,??B+3-*9??")
1387 startup:
1388 'setfreq m8
1389 put 0,0
1390 hi2csetup i2cslave,%01010100
1391 pause 500
1392 'kbled %10000101      'disable LED blinking (enable capslock and scroll lock, disable numlock)
1393 kbled $80
1394 main:
1395 '   pause 100
1396   'sertxd("getting key ")
1397   kbin keyraw          'grab one character

```

```

1398 gotkey:
1399 if keyraw=128 then 'print screen (tries to capitalise instead without this)
1400 elseif keyraw=18 or keyraw=89 then
1401     shift=1
1402     goto main
1403 elseif keyraw = 88 then
1404     inc caps
1405     goto main
1406 endif
1407 'sertxd("got key ")
1408 'sertxd(#b0)
1409 'sertxd(cr)
1410 get 0,b1
1411 'if b1=1 then main 'if status=ready, discard character
1412
1413 put 1,keyraw 'if status <> ready, store character
1414 if caps=1 or shift=1 then
1415 read keyraw,b2 'read eeprom for capitalised ascii
1416 shift=0
1417 else
1418 readable keyraw,b2
1419 endif
1420 put 2,b2
1421 select case keyraw 'numbers or significant commands
1422 case 69
1423     b3=0
1424 case 22
1425     b3=1
1426 case 30
1427     b3=2
1428 case 38
1429     b3=3
1430 case 37
1431     b3=4
1432 case 46
1433     b3=5
1434 case 54
1435     b3=6
1436 case 61
1437     b3=7
1438 case 6
1439     b3=8
1440 case 70
1441     b3=9
1442 'case 63 'sleep button
1443     powerstate=1
1444     b3=11 'sleep command on master
1445 'case 94 'wake up button
1446     high A.3
1447     powerstate=0
1448     b3=12
1449     pause 500
1450     low A.3
1451 else
1452     b3=10
1453 endselect
1454 put 3,b3
1455 put 0,1 'update status=ready
1456 'pause 400
1457 do
1458 get 0,keycheck 'b1

```



```
1459     loop until keycheck=2
1460     'debug
1461     'goto main
1462
1463
1464 repeatcheck:     'prevent repeats if key is held too long
1465 kbin [450,norepeat], repeatkey
1466 'sertxd ("1")
1467 if repeatkey=keyraw then
1468     inc loopcount
1469     if loopcount>10 then
1470         loopcount=0
1471         goto main
1472     endif
1473     goto repeatcheck
1474 else'if repeatkey <> keyraw then
1475     keyraw=repeatkey
1476     repeatkey=0
1477     goto gotkey
1478 endif
1479 goto main
1480 norepeat:
1481     repeatkey=0
1482     goto main
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
```

```

1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540 ; AXE134 Serial 20x4 OLED using PICAXE-18M2*
1541 ; Emulates basic serial operation of the popular AXE033 module
1542 ; CPS, May 2011
1543 ; JB, Jan 2012
1544
1545 #picaxe 18M2
1546
1547
1548 ; Supported Commands
1549 ; 0-7, 8-15      CGRAM characters
1550 ; 16-252       normal ASCII characters, according to selected character map table
1551 ; 253, X        display 12 character pre-saved message from EEPROM memory, X can be 0-11
1552 ; 254, X        OLED command, X can be 0 to 255
1553 ; 255, X        control outputs C.2, C.1, C.0 (via lower 3 bits of X)
1554
1555 #define use_welcome      ; display the welcome message upon power up
1556
1557 symbol baud = N2400_16    ; Serial baud rate 2400,N,8,1. Note main program runs at 16MHz
1558
1559 symbol spare0           = C.0 ; spare output 0
1560 symbol spare1           = C.1 ; spare output 1
1561 symbol spare2           = C.2 ; spare output 2
1562 symbol RX                = C.5   ; serial receive pin
1563 symbol enable            = C.6   ; OLED enable
1564 symbol rs                 = C.7   ; OLED RS
1565
1566 ; OLED data pins are on B.0 to B.7
1567
1568 ; Store the 20 character user defined messages in EEPROM data memory
1569 ; First two messages are optionally used as welcome message
1570
1571 ; Please remember 4 line displays always use the strange 1-3-2-4 line layout.
1572
1573 EEPROM 00, (" PLOS starting up ")      ; store msg 0 in the EEPROM memory
1574 EEPROM 20, ("designed, built, and")    ; store msg 1 in the EEPROM memory
1575 EEPROM 40, (" programmed by ")        ; store msg 2 in the EEPROM memory
1576 EEPROM 60, (" Patrick Leiser ")      ; store msg 3 in the EEPROM memory
1577 EEPROM 80, ("This is msg 4 ")        ; store msg 4 in the EEPROM memory
1578 EEPROM 100, ("This is msg 5 ")       ; store msg 5 in the EEPROM memory
1579 EEPROM 120, ("This is msg 6 ")       ; store msg 6 in the EEPROM memory
1580 EEPROM 140, ("This is msg 7 ")       ; store msg 7 in the EEPROM memory

```

```

1581 EEPROM 160, ("This is msg 8 ") ; store msg 8 in the EEPROM memory
1582 EEPROM 180, ("This is msg 9 ") ; store msg 9 in the EEPROM memory
1583 EEPROM 200, ("This is msg 10 ") ; store msg 10 in the EEPROM memory
1584 EEPROM 220, ("This is msg 11 ") ; store msg 11 in the EEPROM memory
1585
1586 ;initialise OLED
1587 init:
1588     gosub OLED_init ; initialise OLED
1589
1590 ; display welcome message if desired
1591 #ifdef use_welcome
1592     let b1 = 0 ; message 0 on top line
1593     gosub msg ; do it
1594
1595     low rs ; command mode
1596     let pinsB = 192 ; move to line 2, instruction 192
1597     pulsout enable,1 ; pulse the enable pin to send data.
1598     high rs ; character mode again
1599
1600     let b1 = 1 ; message 1 on bottom line
1601     gosub msg ; do it
1602
1603
1604     low rs ; command mode
1605     let pinsB = 148 ; move to line 2, instruction 192
1606     pulsout enable,1 ; pulse the enable pin to send data.
1607     high rs ; character mode again
1608
1609     let b1 = 2 ; message 2 on bottom line
1610     gosub msg ; do it
1611     low rs ; command mode
1612     let pinsB = 212 ; move to line 2, instruction 192
1613     pulsout enable,1 ; pulse the enable pin to send data.
1614     high rs ; character mode again
1615
1616     let b1 = 3 ; message 3 on bottom line
1617     gosub msg ; do it
1618 #endif
1619
1620 ; main program loop, runs at 16MHz
1621
1622 main:
1623
1624     serin RX,baud,b1 ; wait for the next byte
1625
1626 ; NB keep character mode test as first item in this list to optimise speed
1627 if b1 < 253 then
1628     let pinsB = b1 ; output the data
1629     pulsout enable,1 ; pulse the enable pin to send data.
1630     goto main ; quickly loop back to top
1631 else if b1 = 254 then
1632     low rs ; change to command mode for next character
1633     serin RX,baud,b1 ; wait for the command byte
1634     let pinsB = b1 ; output the data
1635     pulsout enable,1 ; pulse the enable pin to send data.
1636     high rs ; back to character mode
1637     goto main ; quickly loop back to top
1638 else if b1 = 253 then
1639     serin RX,baud,b1 ; wait for the next byte
1640     gosub msg ; do the 16 character message
1641     goto main ; back to top

```

```

1642 else ; must be 255
1643     serin RX,baud,b1 ; wait for the next byte
1644     let pinsC = b1 & %00000111 | %10000000
1645     ; output the data on C.0 to C.1, keep RS high
1646     goto main ; back to top
1647 end if
1648
1649
1650 ; power on OLED initialisation sub routine
1651 OLED_init:
1652     let dirsC = %11000111 ; PortC 0,1,2,6,7 all outputs
1653     let dirsB = %11111111 ; PortB all outputs
1654
1655     ; Winstar OLED Module Initialisation
1656     ; according to WS0010 datasheet (8 bit mode)
1657
1658     pause 500 ; Power stabilisation = 500ms
1659
1660     ; Function set – select only one of these 4 character table modes
1661     ;let pinsB = %00111000 ; 8 bit, 2 line, 5x8 , English_Japanese table
1662     let pinsB = %00111001 ; 8 bit, 2 line, 5x8 , Western_European table1
1663     ;let pinsB = %00111010 ; 8 bit, 2 line, 5x8 , English_Russian table
1664     ;let pinsB = %00111011 ; 8 bit, 2 line, 5x8 , Western_European table2
1665
1666     pulsout enable,1 ;
1667
1668     let pinsB = %00001100 ; Display on, no cursor, no blink
1669     pulsout enable,1
1670
1671     let pinsB = %00000001 ; Display Clear
1672     pulsout enable,1
1673     pause 7 ; Allow 6.2ms to clear display
1674
1675     setfreq m16 ; now change to 16Mhz
1676
1677     let pinsB = %00000010 ; Return Home
1678     pulsout enable,1
1679
1680     let pinsB = %00000110 ; Entry Mode, ID=1, SH=0
1681     pulsout enable, 1
1682
1683     high rs ; Leave in character mode
1684     return
1685
1686
1687 ; display message from EEPROM sub routine
1688 ; message number 0–11 must be in b1 when called
1689 ; uses (alters) b1, b2, b3, b4
1690 msg:
1691     let b2 = b1 // 12 * 20 ; EEPROM start address is 0 to 11 multiplied by 20
1692     let b3 = b2 + 20 - 1 ; end address is start address + (20 - 1)
1693     for b4 = b2 to b3 ; for 20 times
1694         read b4,b1 ; read next character from EEPROM data memory into b1
1695         let pinsB = b1 ; output the data
1696         pulsout enable,1 ; pulse the enable pin to send data.
1697     next b4 ; next loop
1698     return
1699
1700
1701
1701 * note that this code came with the AXE134G OLED display I bought, I modified it to customise the startup
message

```

```
1702 '* the original code can be found at http://www.picaxe.com/downloads/axe134y.bas.txt
1703 '* the other two programs (the main program and the keyboard program) are original and written by me
1704 '*they can be found at http://patrickleiser.weebly.com/picaxe-computer.html
```