

```

'*****
'* Name      : ALGC.pbp                               *
'* Version  : 1.0                                     *
'*          : a PIC116F628A                           *
'* Dave bruckenstein Analog Light Guide Clock
'*****
#CONFIG
__config _INTOSC_OSC_NOCLKOUT & _WDT_ON & _PWRTE_ON & _MCLRE_OFF & _LVP_OFF &
_BOREN_OFF
#ENDCONFIG

'----- Variables -----

Green      VAR BYTE[13]
Red        VAR BYTE[13]
Blue      VAR BYTE[13]
Bright    VAR BYTE      ' 0 to 31 MAX
Speed     VAR WORD
Location  VAR BYTE
Color     VAR BYTE
Hour     VAR BYTE
Mins     VAR BYTE
HRsav    VAR BYTE
MINSsav  VAR BYTE
Min_5    VAR BYTE      ' 5 minute value for clock hand
X        VAR BYTE
SecTime  VAR BYTE      ' seconds counter

'----- ALIAS -----

Dat      VAR PortA.0      ' DotStar data
Clk      VAR PortA.1      ' DotStar clock

MinSW    VAR portB.0      ' Minute set switch (active LOW)
HourSW   VAR PortB.1      ' Hour set switch (active LOW)
Hz1      VAR PORTB.5      ' RTC 1Hz CLOCK
scl      VAR PORTB.6      ' RTC SERIAL CLOCK
sda      VAR PORTB.7      ' RTC SERIAL DATA

'----- Initialization -----

TRISA = %00000000      'I/O config
TrisB = %11111111

CCP1CON = 0           ' 16F627/8A turn off capture/compare mode (set digital I/O)
CMCON = 7             ' 16F627/8A disable A/D and comparators !!!!!!!
OPTION_REG.7 = 0      ' 16F627/8A bit 7 = 0 ENABLES weak pull-ups on portB

' SETUP RTC
PAUSE 10 ' let power settle
SHIFTOUT Dat,Clk,5,[0,0,0,0] ' start frame
FOR X = 1 TO 13 ' fill LEDs
    SHIFTOUT Dat,Clk,5,[Bright + %11100000,0,0,0]' clear all LEDs first
NEXT X
' Init 1Hz clock in the DS1307
' SET CONTROL REGISTER FOR 1Hz OUTPUT ON CLK
I2CWRITE sda,scl,$D0,$07,[$10]
PAUSE 100 ' wait for write to complete

```

```

' SET TIME MANUALLY AT PROGRAMMING TIME - once only
  READ 0,X
  IF X = 1 THEN GOTO Main ' skip time set if already done
  I2CWRITE sda,scl,$D0,$00,[$00,$05,$09,$0,$11,$07,$19]
'
'                               SECS,Mins,Hour,DAY,DATE,MONTH,YEAR
  PAUSE 100 ' wait for write to complete
  WRITE 0,1 ' set flag that clock data was written

Init:
  FOR X = 0 TO 14 : Red[X] = 0 : Green[X] = 0 : Blue[X] = 0 : NEXT X ' clear
LEDS
  SHIFTOUT Dat,Clk,5,[0,0,0,0] ' start frame
  FOR X = 1 TO 13 ' fill LEDs
    SHIFTOUT Dat,Clk,5,[Bright + %11100000,0,0,0]' clear all LEDs first
  NEXT X
  GOSUB Show ' show current time

Main:
  GOSUB Show ' update display every second
  WHILE Hz1 = 0 ' read 1Hz signal from DS1307
    IF (MinSW = 0) OR (HourSW = 0) THEN
      GOSUB SetTime
      GOSUB Show
    ENDIF
  WEND
  WHILE Hz1 = 1 ' read 1Hz signal from DS1307
    IF (MinSW = 0) OR (HourSW = 0) THEN
      GOSUB SetTime
      GOSUB Show
    ENDIF
  WEND
  GOTO Main

Show:
' get current time from DS1307
I2CREAD sda,scl,$D0,$00,[X,MINsav,HRsav,X,X,X,X]
'
'                               SECS,mins,Hour,DAY,DATE,MONTH,YEAR
GOSUB BCDtoDEC ' convert

Min_5 = (Mins/5)
IF Min_5 < 1 THEN Min_5 = 12 ' fix 0 value to show at 12:00
IF Hour > 12 THEN Hour = Hour - 12 ' fix 24 hour bug

' clear all LEDs
FOR X = 0 TO 14 : Red[X] = 0 : Green[X] = 0 : Blue[X] = 0 : NEXT X
' initialise clock hand colors: HOURS=Red MINS=Blue
Red[Min_5] = 100 ' set minute color + BRIGHTNESS 0-255
Blue[Hour] = 255 ' set hour color + BRIGHTNESS 0-255
Bright = 31 ' max overall brightness = 31 (1-31 range)

SHIFTOUT Dat,Clk,5,[0,0,0,0] ' start frame
FOR X = 1 TO 13 ' fill LEDs
  SHIFTOUT Dat,Clk,5,[Bright + %11100000,Red[X],Blue[X],Green[X]]' clock
NEXT X
RETURN

SetTime:
I2CREAD sda,scl,$D0,$00,[X,MINsav,HRsav,X,X,X,X] ' get time from DS1307
'
'                               SECS,mins,Hour,DAY,DATE,MONTH,YEAR

```

```
IF HourSW = 0 THEN
  HRSav = HRSav + 1
  IF HRSav > 12 THEN HRSav = 1
  WHILE HourSW = 0 : WEND
  PAUSE 1
ENDIF

IF MinSW = 0 THEN
  MINSav = MINSav + 1 ' increment
  IF (MINSav & %1111) > 10 THEN ' BCD low byte is over 10
    MINSav = (MINSav + %00010000) ' BCD incerment high byte
    MINSav = (MINSav - %00001010) ' BCD clear low byte
  ENDIF
  IF MINSav > %01100000 THEN ' if over 60
    MINSav = 1 ' wrap around ' jump to 1
    ' hrsav = hrsav + 1 ' and increment hour hand
    ' if hrsav > 24 then hrsav = 1
  ENDIF
  WHILE MinSW = 0 : WEND
  PAUSE 1
ENDIF
I2CWRITE sda,scl,$D0,$00,[X,MINSav,HRSav,X,X,X,X]
' SECS,Minute,Hour,DAY,DATE,MONTH,YEAR
PAUSE 100
RETURN

BCDtoDEC:
' convert BCD to decimal
Mins = (((MINSav & %11110000) >> 4) * 10) + (MINSav & %1111)
Hour = (((HRSav & %11110000) >> 4) * 10) + (HRSav & %1111)
RETURN

END
```