SparkFun Electronics

Sharing Ingenuity

Skip to content

Search… Search
Advanced search

- Board index ‹ Platform Specific ‹ PIC Microchip ‹ Code Snippets
- Change font size
- Print view

- FAQ
- Register
- Login

Altimeter using MPX4115, umFPU 3.1, DS18B20, and 16F88

Moderator: phalanx

Post a reply

Search this topic… Search

2 posts • Page **1** of **1**

Altimeter using MPX4115, umFPU 3.1, DS18B20, and 16F88

by **BertMan** » Tue Oct 04, 2011 7:57 pm

I built this this evening for use in a UAV project I am working on. I thought I might share it with the community. I tried to comment as much as I can. Don't forget the output filter caps as outlined in the MPX4115 datasheet or the altitude jumps a bit. I am temorarily using the built in oscilator of the PIC, hence the slow 4 MHz output. Probably wont get a significant boost in speed at 20 MHz because of the slow conversion rate from the DS18B20 anyways. Nathan, you can use this code if you want to add it to the umFPU page. It's tested 100%. Enjoy!

Code: Select all

```
OSCCON =  %01101100  ' Set to 4 MHZ internal oscilator

DEFINE      OSC 4
DEFINE      DEBUG_REG PORTA
DEFINE      DEBUG_BIT 2
DEFINE      DEBUG_BAUD 9600
```

```
DEFINE      DEBUG_MODE 0

FPUOUT      VAR     PORTB.2 ' umFPU in
FPUIN       VAR     PORTB.1 ' umFPU out
FPUCLK      VAR     PORTB.4 ' umFPU SPI clock
DQ          VAR     PORTB.3 ' DS18B20
TEMPRAW     VAR     WORD
BUSY        VAR     BIT
COLDBIT     VAR     TEMPRAW.BIT11
ALTCHAR     VAR     BYTE
DEBUGCHAR   VAR     BYTE
DATAWORD    VAR     word
DATAHIGH    VAR     dataWord.HIGHBYTE
DATALOW      VAR      dataword.LOWBYTE
FPU_STATUS   VAR      DATALOW
CALIBRATED  VAR     BIT

ADCVAL      CON     1       ' ADC register
ZEROOFFSET  CON     2       ' Calibration register
HPA         CON     3       ' hPa register
SLPRES      CON     4       ' Sealevel pressure register
HFTEMP      CON     5       ' Hypsometric formula temp register
ALT         CON     6       ' Altitude register
TEMPMUL     CON     7       ' Temperature multiplyer register
TEMPC       CON     8       ' Temperature register
SELECTA      CON      $01     ' Select register A
ATOF      CON      $1E      ' Convert ASCII to float, store in reg[0]
FTOA      CON      $1F      ' Convert float to ASCII
FSET      CON      $20      ' reg[A] = reg[nn]
FSET0      CON      $29      ' reg[A] = reg[0]
FADD      CON      $21      ' reg[A] = reg[A] + reg[nn]
FMUL      CON      $24      ' reg[A] = reg[A] * reg[nn]
FDIV      CON      $25      ' reg[A] = reg[A] / reg[nn]
FSUB      CON      $22      ' reg[A] = reg[A] - reg[nn]
FMUL0      CON      $2D      ' reg[A] = reg[A] * reg[0]
FSUB0      CON      $2B      ' reg[A] = reg[A] - reg[0]
FPOW0      CON      $30      ' reg[A] = reg[A] ** reg[0]
FADD0      CON      $2A      ' reg[A] = reg[A] + reg[0]
FDIV0      CON      $2E      ' reg[A] = reg[A] / reg[0]
ROUND      CON      $53      ' reg[A] = round(reg[A])
LOADUWORD   CON      $5C      ' reg[0] = float(unsigned word)
ADCMODE      CON      $D1      ' Set A/D trigger mode
ADCTRIG      CON      $D2      ' A/D manual trigger
ADCSCALE   CON      $D3      ' ADCscale[ch] = B
ADCLOAD      CON      $D5      ' reg[0] = float(ADCvalue[ch]) * ADCscale[ch]
SYNC      CON      $F0      ' Get synchronization byte
MSBFIRST   CON     1      ' shiftout mode
MSBPRE      CON     0      ' shiftin mode
SYNC_CHAR   CON      $5C      ' sync character
FNEG      CON      $3E      ' reg[A] = -reg[A]
READSTR      CON      $F2      ' Read string from string buffer
READSTATUS   CON      $F1      ' Read status byte
READDELAY   CON     15      ' read setup delay
```

```
:Init  ' Initialize FPU
DEBUG "Beginning Initialization", 13, 10
PAUSE 1000
SHIFTOUT FpuOut, FpuClk, MSBFIRST, [$FF, $FF, $FF, $FF, $FF, $FF, $FF, $FF, $FF, $FF,
0]
PAUSE 10
SHIFTOUT FpuOut, FpuClk, MSBFIRST, [SYNC]
PAUSEUS ReadDelay
SHIFTIN FpuIn, FpuClk, MSBPRE, [fpu_status]
CALIBRATED = 0

:CheckFPUStatus
IF fpu_status <> SYNC_CHAR THEN Init

:ConfigureFPU  ' Set FPU Config
SHIFTOUT FPUOUT, FPUCLK, 1, [ADCMODE, $1F] ' Set ADC mode to trigger mode
SHIFTOUT FPUOUT, FPUCLK, 1, [ADCSCALE, $01] ' Set ADC multiplier
DEBUG "Initialized", 13, 10
PAUSE 1000

:Main
DEBUGCHAR = 1
OWOUT DQ, 1, [$CC, $44] ' Get temperature and perform conversion

:WaitUp
OWIN DQ, 4, [BUSY]   ' Make sure temperature conversion is complete
If BUSY = 0 THEN WaitUp

OWOUT DQ, 1, [$CC, $BE] ' Read temparature from buffer
OWIN DQ, 2, [TEMPRAW.LOWBYTE, TEMPRAW.HIGHBYTE]

' Get temperature. Temperature sensor is in 12bit ADC mode so each bit is
' multiplied by .0625
IF COLDBIT = 1 THEN  ' Find out if it's negative celsius
  TEMPRAW = ~TEMPRAW   ' Bits are inversed when negative celsius
  SHIFTOUT FPUOUT, FPUCLK, MSBFIRST, [SELECTA, TEMPMUL, ATOF, "0.0625", $00, FSET0]
  SHIFTOUT FPUOUT, FPUCLK, MSBFIRST, [SELECTA, TEMPC, LOADUWORD, TEMPRAW.HIGHBYTE,
TEMPRAW.LOWBYTE+1, FSET0]
  SHIFTOUT FPUOUT, FPUCLK, MSBFIRST, [FMUL, TEMPMUL, FNEG, ROUND]
ELSE
  SHIFTOUT FPUOUT, FPUCLK, MSBFIRST, [SELECTA, TEMPMUL, ATOF, "0.0625", $00, FSET0]
  SHIFTOUT FPUOUT, FPUCLK, MSBFIRST, [SELECTA, TEMPC, LOADUWORD, TEMPRAW.HIGHBYTE,
TEMPRAW.LOWBYTE, FSET0]
  SHIFTOUT FPUOUT, FPUCLK, MSBFIRST, [FMUL, TEMPMUL, ROUND]
ENDIF

' Send temperature to debug
DEBUG "TEMP: "
SHIFTOUT FPUOUT, FPUCLK, MSBFIRST, [FTOA, 40, READSTR]
REPEAT
  SHIFTIN FPUIN, FPUCLK, 0, [DEBUGCHAR\8]
  DEBUG DEBUGCHAR
UNTIL DEBUGCHAR = 0
DEBUGCHAR = 1
```

```
DEBUG " C", 13, 10

' Pressure is calculated with a linear equation designed for a 12bit ADC:
' P = .0271 X ADC Value + 10.58. Once you have temperature and atomspheric
' pressure, you can calculate altitude using the Hypsometric formula:
' h = (((P0 / P) to the power of (1 / 5.257) - 1) X (T + 273.15)) / .0065
' where P0 = sealevel pressure in hPa, P = atmospheric pressure in hPa and T = temp in
Celsius
Shiftout FPUOUT, FPUCLK, MSBFIRST, [ADCTRIG, SELECTA, ADCVAL, ADCLOAD, 0, FSET0, ATOF]
' Trigger ADC
SHIFTOUT FPUOUT, FPUCLK, MSBFIRST, ["0.0269", $00, FMUL0, ATOF, "10.56", $00, FADD0] '
Perform linear equation
SHIFTOUT FPUOUT, FPUCLK, MSBFIRST, [ATOF, "10", $00, FMUL0, SELECTA, HPA, FSET,
ADCVAL] ' Convert kPa to hPa

' Send pressure to debug
DEBUG "PRESSURE: "
SHIFTOUT FPUOUT, FPUCLK, MSBFIRST, [FTOA, 61, READSTR]
REPEAT
  SHIFTIN FPUIN, FPUCLK, 0, [DEBUGCHAR\8]
  If DEBUGCHAR <> 0 THEN
    DEBUG DEBUGCHAR
  ENDIF
UNTIL DEBUGCHAR = 0
DEBUGCHAR = 1
DEBUG " hPa", 13, 10

' Calculate altitude
SHIFTOUT FPUOUT, FPUCLK, MSBFIRST, [SELECTA, SLPRES, ATOF, "1013.25", $00, FSET0] '
Next few lines are hypsometric formula
SHIFTOUT FPUOUT, FPUCLK, MSBFIRST, [SELECTA, HFTEMP, FSET, SLPRES, FDIV, HPA, ATOF]
SHIFTOUT FPUOUT, FPUCLK, MSBFIRST, ["0.190222560", $00, FPOW0, ATOF, "1", $00, FSUB0]
SHIFTOUT FPUOUT, FPUCLK, MSBFIRST, [SELECTA, ALT, FSET, TEMPC, ATOF, "273.15", $00]
SHIFTOUT FPUOUT, FPUCLK, MSBFIRST, [FADD0, FMUL, HFTEMP, ATOF, "0.0065", $00, FDIV0] '
ALT = Altitude in meters
SHIFTOUT FPUOUT, FPUCLK, MSBFIRST, [ATOF, "3.28008399", $00, FMUL0, ATOF, "10"] ' ALT
= Altitude in feet
SHIFTOUT FPUOUT, FPUCLK, MSBFIRST, [$00, FDIV0, ROUND, ATOF, "10", $00, FMUL0] ' Set
resolution to 10'

' This routine will calibrate current pressure to 0 ft
If CALIBRATED = 0 THEN
  SHIFTOUT FPUOUT, FPUCLK, MSBFIRST, [SELECTA, ZEROOFFSET, FSET, ALT]
  CALIBRATED = 1
ENDIF
SHIFTOUT FPUOUT, FPUCLK, MSBFIRST, [SELECTA, ALT, FSUB, ZEROOFFSET]

' Send altitude to debug
DEBUG "ALT: "
SHIFTOUT FPUOUT, FPUCLK, MSBFIRST, [FTOA, 50, READSTR]
REPEAT
  SHIFTIN FPUIN, FPUCLK, 0, [DEBUGCHAR\8]
  DEBUG DEBUGCHAR
UNTIL DEBUGCHAR = 0
```

```
DEBUG " FT", 13, 10

:TheEnd
Goto Main
```

Last edited by BertMan on Tue Oct 04, 2011 8:26 pm, edited 2 times in total.

BertMan

Posts: 19
Joined: Mon Sep 12, 2005 4:05 pm

Top

Re: Altimeter using MPX4114, umFPU 3.1, DS18B20, and 16F88

by **BertMan** » Tue Oct 04, 2011 8:00 pm

Sorry for the misaligned formatting. The code is aligned in my IDE so I'm not sure what's going on.

BertMan

Posts: 19
Joined: Mon Sep 12, 2005 4:05 pm

Top

Display posts from previous: All posts ▼ Sort by Post time ▼ Ascending ▼ Go

Display posts by user: All users ▼ Go

Post a reply
2 posts • Page **1** of **1**

Return to Code Snippets

Jump to: Code Snippets ▼ Go

Who is online

Users browsing this forum: No registered users and 1 guest

- Board index
- The team • Delete all board cookies • All times are UTC - 7 hours

Powered by phpBB® Forum Software © phpBB Group