Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

599 Menlo Drive, Suite 100
Rocklin, California 95765, USA
Office: (916) 624-8333
Fax: (916) 624-8003

General: info@parallax.com
Technical: support@parallax.com
Web Site: www.parallax.com
Educational: www.stampsinclass.com

Sensirion SHT11 Sensor Module (#28018)

Precision Temperature and Humidity Measurement

Introduction

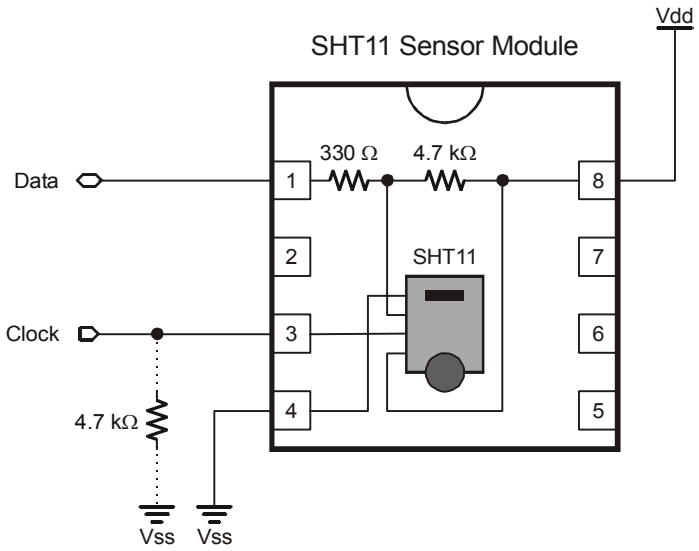
When it comes to precision temperature and humidity measurement, Sensirion (www.sensirion.com) has simplified the process their SHT1x sensor series. Through a two-wire serial interface, both temperature and humidity can be read with excellent response time and accuracy. Parallax has simplified the use of the SHT11 by mounting it in a user-friendly 8-pin DIP module. The module includes a data-line pull-up and series limiter making it possible to connect directly to the BASIC or Javelin Stamp.

Features

- Temperature range: -40 °F (-40 °C) to +254.9 °F (+123.8 °C)
- Temp. accuracy: +/- 0.5 °C @ 25 °C
- Humidity range: 0 to 100% RH
- Absolute RH accuracy: +/- 3.5% RH
- Low power consumption (typically 30 µW)

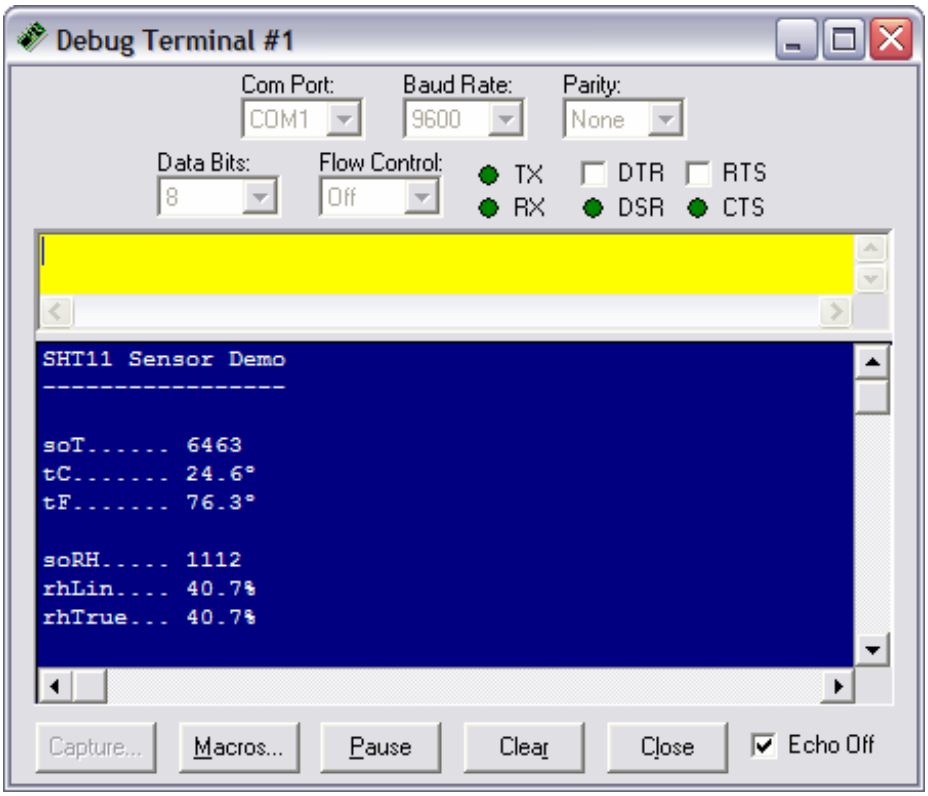
Connections

The SHT11 is interfaced to the Stamp over two I/O pins. The 4.7 kΩ pull-down resistor on the clock is optional but may be required if your application experiences sensor lock-up.



BASIC Stamp Application

The following BASIC Stamp application will read the SHT11 sensor module and display sensor counts, converted temperature and calibrated humidity. When running, the program output will appear as shown below:




```
' =============================================================================
'
'   File..... SHT11_Demo.BS2
'   Purpose... Interface to Sensirion SHT11 temperature/humidity sensor
'   Author.... Parallax
'   E-mail.... support@parallax.com
'   Started...
'   Updated... 19 JUL 2003
'
'   {$STAMP BS2}
'   {$PBASIC 2.5}
'
' =============================================================================


' -----------------------------------------------------------------------------
' Program Description
' -----------------------------------------------------------------------------
'
```

```
' This program demonstrates the interface and conversion of SHT11/15 data
' to usable program values.  This program uses advanced math features of
' PBASIC, specifically the ** operator.
'
' For detailed application information on the use and application of the
' ** operator, see Dr. Tracy Allen's web page at this link:
'
' -- http://www.emesystems.com/BS2math1.htm
'
' For SHT11/15 documentation and app notes, visit:
'
' -- http://www.sensirion.com


' -----------------------------------------------------------------------------
' Revision History
' -----------------------------------------------------------------------------


' -----------------------------------------------------------------------------
' I/O Definitions
' -----------------------------------------------------------------------------

ShtData         PIN     1                       ' bi-directional data
Clock           PIN     0


' -----------------------------------------------------------------------------
' Constants
' -----------------------------------------------------------------------------

ShtTemp         CON     %00011                  ' read temperature
ShtHumi         CON     %00101                  ' read humidity
ShtStatW        CON     %00110                  ' status register write
ShtStatR        CON     %00111                  ' status register read
ShtReset        CON     %11110                  ' soft reset

Ack             CON     0
NoAck           CON     1

No              CON     0
Yes             CON     1

DegSym          CON     186                     ' degrees symbol for DEBUG


' -----------------------------------------------------------------------------
' Variables
' -----------------------------------------------------------------------------
```

```
ioByte          VAR     Byte                    ' data from/to SHT11
ackBit          VAR     Bit                     ' ack/nak from/to SHT11
toDelay         VAR     Byte                    ' timeout delay timer
timeOut         VAR     Bit                     ' timeout status

soT             VAR     Word                    ' temp counts from SHT11
tC              VAR     Word                    ' temp - Celcius
tF              VAR     Word                    ' temp - Fahrenheit

soRH            VAR     Word                    ' humidity counts
rhLin           VAR     Word                    ' humidity; linearized
rhTrue          VAR     Word                    ' humidity; compensated

status          VAR     Byte                    ' status byte


' -----------------------------------------------------------------------------
' EEPROM Data
' -----------------------------------------------------------------------------


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

Initialize:
  GOSUB SHT_Connection_Reset                    ' reset device connection
  PAUSE 250                                     ' let DEBUG window open
  DEBUG CLS,
        "SHT11 Sensor Demo", CR,
        "-----------------", CR


' -----------------------------------------------------------------------------
' Program Code
' -----------------------------------------------------------------------------

Main:
  DO
    GOSUB SHT_Measure_Temp
    DEBUG CRSRXY, 0, 3,
          "soT..... ", DEC soT, CR,
          "tC..... ", DEC (tC / 10), ".", DEC1 tC, DegSym, " ", CR,
          "tF..... ", DEC (tF / 10), ".", DEC1 tF, DegSym, " "

    GOSUB SHT_Measure_Humidity
    DEBUG CRSRXY, 0, 7,
          "soRH..... ", DEC soRH, CR,
```

```
          "rhLin.... ", DEC (rhLin / 10), ".", DEC1 rhLin, "% ", CR,
          "rhTrue... ", DEC (rhTrue / 10), ".", DEC1 rhTrue, "% "

    PAUSE 1000
  LOOP
  END


' -----------------------------------------------------------------------------
' Subroutines
' -----------------------------------------------------------------------------

' connection reset: 9 clock cyles with ShtData high, then start sequence
'
SHT_Connection_Reset:
  SHIFTOUT ShtData, Clock, LSBFirst, [$FFF\9]

' generates SHT11 "start" sequence
'           _____          _____
' ShtData       |________|
'              ___      ___
' Clock    ___|   |___|   |___
'
SHT_Start:
  INPUT ShtData                                 ' let pull-up take high
  LOW Clock
  HIGH Clock
  LOW ShtData
  LOW Clock
  HIGH Clock
  INPUT ShtData
  LOW Clock
  RETURN


' measure temperature
' -- celcius = raw * 0.01 - 40
' -- fahrenheit = raw * 0.018 - 40
'
SHT_Measure_Temp:
  GOSUB SHT_Start                               ' alert device
  ioByte = ShtTemp                              ' temperature command
  GOSUB SHT_Write_Byte                          ' send command
  GOSUB SHT_Wait                                ' wait for measurement
  ackBit = Ack                                  ' another read follows
  GOSUB SHT_Read_Byte                           ' get MSB
  soT.HighByte = ioByte
  ackBit = NoAck                                ' last read
  GOSUB SHT_Read_Byte                           ' get LSB
```

```
  soT.LowByte = ioByte

  ' Note: Conversion factors are multiplied by 10 to return the
  '       temperature values in tenths of degrees

  tC = soT ** $1999 - 400                       ' convert to tenths C
  tF = soT ** $2E14 - 400                       ' convert to tenths F
  RETURN


' measure humidity
'
SHT_Measure_Humidity:
  GOSUB SHT_Start                               ' alert device
  ioByte = ShtHumi                              ' humidity command
  GOSUB SHT_Write_Byte                          ' send command
  GOSUB SHT_Wait                                ' wait for measurement
  ackBit = Ack                                  ' another read follows
  GOSUB SHT_Read_Byte                           ' get MSB
  soRH.HighByte = ioByte
  ackBit = NoAck                                ' last read
  GOSUB SHT_Read_Byte                           ' get LSB
  soRH.LowByte = ioByte

  ' linearize humidity
  '   rhLin = (soRH * 0.0405) - (soRH^2 * 0.0000028) - 4
  '
  ' for the BASIC Stamp:
  '   rhLin = (soRH * 0.0405) - (soRH * 0.002 * soRH * 0.0014) - 4
  '
  ' Conversion factors are multiplied by 10 to return tenths
  '
  rhLin = (soRH ** $67AE) - (soRH ** $83 * soRH ** $5B) - 40

  ' temperature compensated humidity
  '   rhTrue = (tc - 25) * (soRH * 0.00008 + 0.01) + rhLin
  '
  ' Conversion factors are multiplied by 10 to return tenths
  ' -- simplified
  '
  rhTrue = (tC - 250) * (soRH ** $34) + rhLin
  RETURN


' sends "status"
'
SHT_Write_Status:
  GOSUB SHT_Start                               ' alert device
  ioByte = ShtStatW                             ' write to status reg cmd
```

```
  GOSUB SHT_Write_Byte                          ' send command
  ioByte = status
  GOSUB SHT_Write_Byte
  RETURN


' returns "status"
'
SHT_Read_Status:
  GOSUB SHT_Start                               ' alert device
  ioByte = ShtStatW                             ' write to status reg cmd
  GOSUB SHT_Read_Byte                           ' send command
  ackBit = NoAck                                ' only one byte to read
  GOSUB SHT_Read_Byte
  RETURN


' sends "ioByte"
' returns "ackBit"
'
SHT_Write_Byte:
  SHIFTOUT ShtData, Clock, MSBFirst, [ioByte]   ' send byte
  SHIFTIN  ShtData, Clock, LSBPre, [ackBit\1]   ' get ack bit
  RETURN


' returns "ioByte"
' sends "ackBit"
'
SHT_Read_Byte:
  SHIFTIN  ShtData, Clock, MSBPre, [ioByte]     ' get byte
  SHIFTOUT ShtData, Clock, LSBFirst, [ackBit\1] ' send ack bit
  INPUT ShtData                                 ' release data line
  RETURN


' wait for device to finish measurement (pulls data line low)
' -- timeout after ~1/4 second
'
SHT_Wait:
  INPUT ShtData                                 ' data line is input
  timeOut = No                                  ' assume no timeout
  FOR toDelay = 1 TO 250                        ' wait ~1/4 second
    IF (ShtData = 0) THEN EXIT
    PAUSE 1
  NEXT
  IF (toDelay = 250) THEN timeOut = Yes         ' loop completed = timeout
  RETURN
```

```
' reset SHT11/15 with soft reset
'
SHT_Soft_Reset:
  GOSUB SHT_Connection_Reset                    ' reset the connection
  ioByte = ShtReset                             ' reset command
  ackBit = NoAck                                ' only one byte to send
  GOSUB SHT_Write_Byte                          ' send it
  PAUSE 11                                      ' wait at least 11 ms
  RETURN
```

In high humidity applications, the SHT11 heater can be switched on briefly to prevent condensation. Another use of the heater is to test the operation of the sensor: by reading before enabling the heater and immediately after the sensor can be verified by noting a higher temperature and lower humidity. The following subroutines can be used to switch the SHT11 heater on and off.

```
Heater_On:
  status = %00000100                            ' heater bit = On
  GOSUB SHT_Write_Status
  RETURN


Heater_Off:
  status = %00000000                            ' heater bit = Off
  GOSUB SHT_Write_Status
  RETURN
```