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

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

Sensirion SHT11 Sensor (#28018)

Temperature and Humidity Measurement

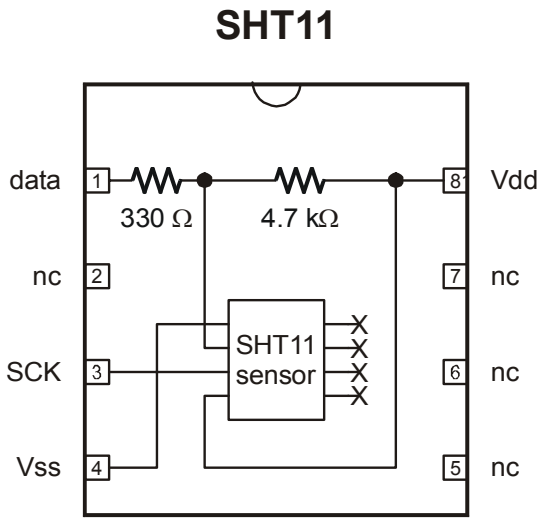
Introduction

The SHT11 is an 8-pin dip package sensor providing pre-calibrated relative humidity and temperature digital output. The Parallax module consists of the surface-mounted Sensirion SHT11 sensor and two resistors. The SHT11 relative humidity and temperature is output over a 14-bit range via serial interface. Please see the datasheet from the Parallax website or www.sensirion.com.

- Power supply DC 2.4, 5 or 5.5 V
- Supply current measuring 550 µA; average 2 uA; sleep 0.3 1 µA
- Size**:** L 12.8 mm; W 11.2 mm; pin height 11.5mm; board height 4.0 mm

Connections

The SHT11 is interfaced to the BASIC Stamp over two I/O pins. In the example below you will use I/O P1 for the DATA and I/O P0 for the Clock/SCK pin.



```
'{$STAMP BS2}
'{$PBASIC 2.5}
' ================================================================================
'
'   File...... SHT11 Simple.BS2
'   Purpose... Interface to Sensirion SHT11 temperature/humidity sensor
'   E-mail.... support@parallax.com
'   Date...... 3/03/03
'
'
'
' ================================================================================
'
'
' --------------------------------------------------------------------------------
' Program Description
' --------------------------------------------------------------------------------
'
' This program demonstrates the interface to the Sht1X and displays the raw data
'
' --------------------------------------------------------------------------------
' I/O Definitions
' --------------------------------------------------------------------------------

ShtData         CON     1                       ' bi-directional data
Clock           CON     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 (wait 11 ms after)

Ack             CON     0
NoAck           CON     1

No              CON     0
Yes             CON     1

MoveTo          CON     2                       ' for DEBUG control
ClrRt           CON     11                      ' clear DEBUG line to right

DegSym          CON     186                     ' degrees symbol for DEBUG


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

ioByte          VAR     Byte                    ' data from/to SHT1x
ackBit          VAR     Bit                     ' ack/nak from/to SHT1x
toDelay         VAR     Byte                    ' timeout delay timer
timeOut         VAR     Bit                     ' timeout status

soT             VAR     Word                    ' temp counts from SHT1x
soRH            VAR     Word                    ' humidity counts from SHT1x
rhLin           VAR     Word                    ' humidity; linearized
```

```
rhTrue          VAR     Word                    ' humidity; temp compensated
status          VAR     Byte                    ' SHT1x status byte


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


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

Initialize:
  GOSUB SHT_Connection_Reset                    ' reset device connection

  PAUSE 250                                     ' let DEBUG window open


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

Sensor_Demo:
  GOSUB SHT_Measure_Temp


Main:
  DEBUG CLS
  DEBUG "SHT1x Demo", CR
  DEBUG "----------", CR

Main2:
  GOSUB SHT_Measure_Temp
  DEBUG MoveTo, 0, 3
  DEBUG "Raw Data Temp: "
  DEBUG DEC soT, ClrRt, CR


  GOSUB SHT_Measure_Humidity
 ' DEBUG MoveTo, 0, 7
  DEBUG "Raw Data Humidity: "
  DEBUG DEC soRH, ClrRt, CR


  PAUSE 1000                                    ' minimum delay between readings
  GOTO Main2




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

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

'
SHT_Start:
  INPUT ShtData                                 ' let pull-up take line high
```

```
  LOW Clock
  HIGH Clock
  LOW ShtData
  LOW Clock
  HIGH Clock
  INPUT ShtData
  LOW Clock
  RETURN


SHT_Measure_Temp:
  GOSUB SHT_Start                                ' alert device
  ioByte = ShtTemp                               ' temperature command
  GOSUB SHT_Write_Byte                           ' send command
  GOSUB SHT_Wait                                 ' wait until measurement done
  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
  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 until measurement done
  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

  RETURN



SHT_Write_Status:
  GOSUB SHT_Start                                ' alert device
  ioByte = ShtStatW                              ' write to status reg command
  GOSUB SHT_Write_Byte                           ' send command
  ioByte = status
  GOSUB SHT_Write_Byte
  RETURN



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

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



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


SHT_Wait:
  INPUT ShtData                                 ' data line is input
  FOR toDelay = 1 TO 250                        ' give ~1/4 second to finish
    timeOut = Ins.LowBit(ShtData)               ' scan data line
    IF (timeOut = No) THEN SHT_Wait_Done        ' if low, we're done
    PAUSE 1
  NEXT

SHT_Wait_Done:
  RETURN


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
```

```
' =============================================================================
'
'   File...... SHT11 Advanced.bs2
'   Purpose... Interface to Sensirion SHT11 temperature/humidity sensor
'   E-mail.... support@parallax.com
'   Date...... 12/30/02
'
'   {$STAMP BS2}
'
' =============================================================================
' -----------------------------------------------------------------------------
' Program Description
' -----------------------------------------------------------------------------
'
' This program demonstrates the interface and conversion of SHT11 data to
' usable program values.
'
' For detailed information on the use and application of the ** operator,
' see Tracy Allen's web page at this link:
'
' -- http://www.emesystems.com/BS2math1.htm
'
' For Tracy's SHT1x code [very advanced]:
'
' -- http://www.emesystems.com/OL2sht1x.htm
'
' For SHT11/15 documentation and app notes, visit:
'
' -- http://www.sensirion.com


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

ShtData         CON     1                       ' bi-directional data
Clock           CON     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 (wait 11 ms after)

Ack             CON     0
NoAck           CON     1

No              CON     0
Yes             CON     1

MoveTo          CON     2                       ' for DEBUG control
ClrRt           CON     11                      ' clear DEBUG line to right

DegSym          CON     186                     ' degrees symbol for DEBUG


' -----------------------------------------------------------------------------
```

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

ioByte          VAR     Byte                    ' data from/to SHT1x
ackBit          VAR     Bit                     ' ack/nak from/to SHT1x
toDelay         VAR     Byte                    ' timeout delay timer
timeOut         VAR     Bit                     ' timeout status

soT             VAR     Word                    ' temp counts from SHT1x
tC              VAR     Word                    ' temp - celcius
tF              VAR     Word                    ' temp - fahrenheit

soRH            VAR     Word                    ' humidity counts from SHT1x
rhLin           VAR     Word                    ' humidity; linearized
rhTrue          VAR     Word                    ' humidity; temp compensated

status          VAR     Byte                    ' SHT1x status byte


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


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

Initialize:
  GOSUB SHT_Connection_Reset                    ' reset device connection

  PAUSE 250                                     ' let DEBUG window open
  DEBUG CLS
  DEBUG "SHT1x Demo", CR
  DEBUG "----------", CR

  ' GOTO Main                                   ' skip heater demo

' -----------------------------------------------------------------------------
' Program Code
' -----------------------------------------------------------------------------
Sensor_Demo:
  GOSUB SHT_Measure_Temp
  DEBUG MoveTo, 0, 3
  DEBUG "tF..... "
  DEBUG DEC (tF / 10), ".", DEC1 tF, DegSym, ClrRt, CR

  GOSUB SHT_Measure_Humidity
  DEBUG "rhLin... "
  DEBUG DEC (rhLin / 10), ".", DEC1 rhLin, "%", ClrRt, CR, CR

Heater_On:
  DEBUG "SHT1x heater on", CR
  status = %00000100                            ' heater bit = On
  GOSUB SHT_Write_Status
  DEBUG "Waiting 2 seconds", CR
  PAUSE 2000

Heater_Off:
  DEBUG "SHT1x heater off", CR, CR
  status = %00000000                            ' heater bit = Off
  GOSUB SHT_Write_Status
```

```
  GOSUB SHT_Measure_Temp
  DEBUG "tF...... "
  DEBUG DEC (tF / 10), ".", DEC1 tF, DegSym, ClrRt, CR

  GOSUB SHT_Measure_Humidity
  DEBUG "rhLin... "
  DEBUG DEC (rhLin / 10), ".", DEC1 rhLin, "%", ClrRt, CR, CR

  PAUSE 5000

Main:
  DEBUG CLS
  DEBUG "SHT1x Demo", CR
  DEBUG "----------", CR

Main2:
  GOSUB SHT_Measure_Temp
  DEBUG MoveTo, 0, 3
  DEBUG "soT...... "
  DEBUG DEC soT, ClrRt, CR
  DEBUG "tC....... "
  DEBUG DEC (tC / 10), ".", DEC1 tC, DegSym, ClrRt, CR
  DEBUG "tF....... "
  DEBUG DEC (tF / 10), ".", DEC1 tF, DegSym, ClrRt

  GOSUB SHT_Measure_Humidity
  DEBUG MoveTo, 0, 7
  DEBUG "soRH..... "
  DEBUG DEC soRH, ClrRt, CR
  DEBUG "rhLin.... "
  DEBUG DEC (rhLin / 10), ".", DEC1 rhLin, "%", ClrRt, CR
  DEBUG "rhTrue... "
  DEBUG DEC (rhTrue / 10), ".", DEC1 rhTrue, "%", ClrRt

  PAUSE 1000                                    ' minimum delay between readings
  GOTO Main2

  END


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

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

' generates SHT1x "start" sequence
'          _____       _____
' ShtData       |_____|
'             ___   ___
' Clock    ___|   |___|   |___
'
SHT_Start:
  INPUT ShtData                                 ' let pull-up take line high
  LOW Clock
  HIGH Clock
  LOW ShtData
  LOW Clock
  HIGH Clock
  INPUT ShtData
```

```
  LOW Clock
  RETURN


' measure temperature
' -- celcius = soT * 0.01 - 40
' -- fahrenheit = soT * 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 until measurement done
  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 / 10 - 400                           ' convert to tenths C
  tF = soT ** 11796 - 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 until measurement done
  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 * 0.004 * soRH * 0.0007) - 4
  '
  ' Conversion factors are multiplied by 10 and then rounded to
  ' return tenths
  '
  rhLin = (soRH ** 26542)
  rhLin = rhLin - ((soRH ** 3468) * (soRH ** 3468) + 50 / 100)
  rhLin = rhLin - 40

  ' temperature compensated humidity
  '   rhTrue = (tC - 25) * (soRH * 0.00008 + 0.01) + rhLin
  '
  ' Conversion factors are multiplied by 100 to improve accuracy and then
  ' rounded off.
  '
  rhTrue = ((tC / 10 - 25) * (soRH ** 524 + 1) + (rhLin * 10)) + 5 / 10
  RETURN
```

```
' sends "status"
'
SHT_Write_Status:
  GOSUB SHT_Start                                ' alert device
  ioByte = ShtStatW                              ' write to status reg command
  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 command
  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
  FOR toDelay = 1 TO 250                         ' give ~1/4 second to finish
    timeOut = Ins.LowBit(ShtData)                ' scan data line
    IF (timeOut = No) THEN SHT_Wait_Done         ' if low, we're done
    PAUSE 1
  NEXT

SHT_Wait_Done:
  RETURN


' reset SHT1x 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
```