BS2p24/40 Demo Board (#45187)
BASIC Stamp® Microcontroller Demonstration Platform

Introduction

The BS2p24/40 Demo Board is designed as a low-cost platform to test and experiment with the advanced features of the BASIC Stamp 2p (24- or 40-pin) and BASIC Stamp 2pe microcontrollers. In particular: 16 additional I/O pins, working with standard parallel LCDs (with Hitachi HD44780 or compatible driver), Philips $I^2C^®$ protocol, and the Maxim/Dallas 1-Wire® protocol. The BS2p24/40 Demo Board is not exclusive to the BS2p/pe family; it is physically and electrically compatible with all 24-pin BS2-series BASIC Stamp and Javelin Stamp microcontrollers.

The purpose of this document is to introduce the BS2p24/40 Demo Board and explain the particular configuration requirements for connecting parallel LCDs, Philips I^2C™ devices, and Maxim/Dallas 1-Wire devices.

Packing List

- BS2p24/40 Demo Board
- Jumper wires   800-00016
- Documentation

Features

- Standard 2.1 mm center-positive DC power jack
- DB9-F connector for serial programming
- On-board 5-volt regulator (LM2940 low-dropout)
- Solderless breadboard
- 14-pin IDC connector for parallel LCD
- Contrast potentiometer for LCD
- iButton® socket (for F5 MicroCan)
- RJ-11 jack for Dallas/Maxim "Blue Dot Receptor" (DS1402D-DR8 / DS1402RP8) cable
- Selectable pull-up resistors for 1-Wire bus and pull-down for LCD enable pin
- Connections for P0 – P15, A0 – A15, Vin, Vdd, Vss and 1-Wire connections
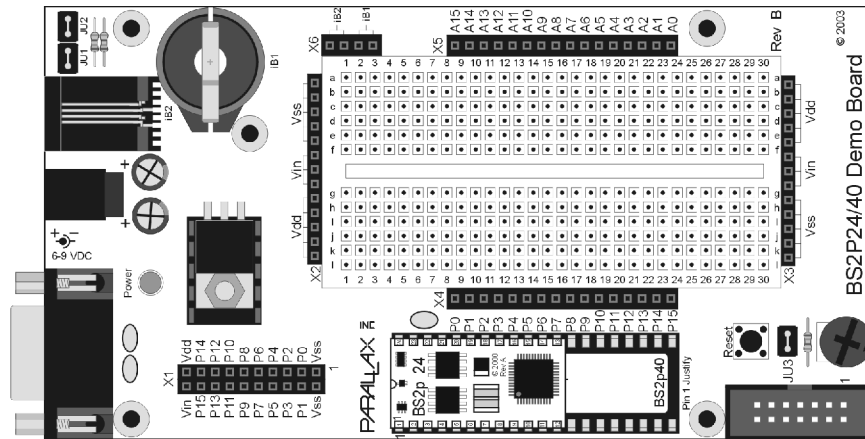
Compatible Accessories (Not Included)

- 2 x 16 Parallel LCD   603-00006
- 4 x 20 Parallel LCD   603-00004
- Blue Dot Receptor Cable   805-00004
- BS2p Plus Pack   45184

Mechanical Dimensions (Rev. C PCB)

- PCB   5.95" x 3.05" (151 mm x 77 mm)
- Mounting Holes   4.00" x 2.75" (102 mm x 70 mm)

BS2p24/40 Demo Board with BS2p24 Installed

Connections and Jumpers

The BS2p40-IC's 32 I/O lines are accessible through two sockets on either edge of the solderless breadboard. The main I/O (P0 – P15) group is available through socket X4, and the auxiliary I/O (A0 – A15) group is available through socket X5. These sockets, and the others on the board, are compatible with 22-guage hookup wire, as well as Parallax flexible jumpers. Vin (unregulated input), Vdd (regulated 5 volts) and Vss (ground) are made available through socket X2 at the top edge of the breadboard and X3 at the bottom edge of the breadboard.

When using a 24-pin BASIC Stamp or Javelin Stamp module in the demo board be sure to align pin 1 of the module in the socket as indicated on the PCB. And note that when using a 24-pin module, pins P8 through P15 are accessed via sockets A8 through A15.

Connection to the onboard iButton socket is through X6-iB1. By installing jumper JU1 a 4.7 KΩ pull-up is connected to this line. A "Blue Dot Receptor" cable (not included) may be connected to iB2 (RJ-11 jack) and connected to the BASIC Stamp microcontroller through X6-iB2. Jumper JU2 may be installed to connect a 4.7 KΩ pull-up to the iB2 connection.

When using a parallel LCD with the demo board, install jumper JU3 to enable the 4.7 kΩ pull-down resistor on the LCD's Enable (E) pin. Note that this jumper should be removed when not using the LCD connector as it pulls P0 to Vss through 4.7 kΩ and may affect some circuits.

I2C is patented technology owned by Philips. The end user is responsible for determining whether his or her I2C application requires licensing by Philips.

http://www.semiconductors.philips.com/buses/i2c/licensing/index.html.

Experimenting with Parallel LCDs

Parallel LCDs, especially those controlled by the Hitachi HD44780 (or compatible) driver, have become as commonplace as LEDs in many embedded applications. The BS2p family and BS2pe microcontroller supports the use of parallel LCDs with the LCDCMD, LCDOUT, and LCDIN instructions.

This experiment demonstrates proper multi-line LCD initialization, the creation and use of custom characters, and a convenient subroutine for printing right-justified numbers on the LCD. Connect a 2x16 LCD to the box header, install jumper JU3, and then adjust the contrast pot for the best view on the LCD. No other connections are required for this program.

Source Code

```
' ==========================================================================
'
'   File...... P24-40_LCD_DEMO.BSP
'   Purpose... Basic LCD Demo - Multi-line mode with custom characters
'   Author.... Parallax, Inc.
'             Copyright (c) 2004, All Rights Reserved
'   E-mail.... support@parallax.com
'   Started...
'   Updated... 17 JUN 2004
'
'   {$STAMP BS2p}
'   {$PBASIC 2.5}
'
' ==========================================================================



' -----[ Program Description ]----------------------------------------------
'
' This program demonstrates the use of the multi-line initialization and
' the use of custom characters.  When using the standard 5x7 font, the LCD
' will hold up to eight customer characters.
'
' To run this program on the BS2p24/40 Demo Board, connect the LCD and
' install Jumper JU3. Adjust contrast pot for best display.
'
' Refer to the Hitachi HD44780 documentation for details on LCD control.


' -----[ I/O Definitions ]--------------------------------------------------

E               PIN    0                       ' connect LCD to OutL


' -----[ Constants ]--------------------------------------------------------

#DEFINE _LcdReady = ($STAMP = BS2P) OR ($STAMP = BS2PE)
```

```
NoCmd           CON     $00
LcdCls          CON     $01                         ' clear the LCD
LcdHome         CON     $02                         ' move cursor home
LcdCrsrL        CON     $10                         ' move cursor left
LcdCrsrR        CON     $14                         ' move cursor right
LcdDispL        CON     $18                         ' shift chars left
LcdDispR        CON     $1C                         ' shift chars right


LcdDDRam        CON     $80                         ' Display Data RAM control
LcdCGRam        CON     $40                         ' Character Generator RAM
LcdLine1        CON     $80                         ' DDRAM address of line 1
LcdLine2        CON     $C0                         ' DDRAM address of line 2



' -----[ Variables ]-------------------------------------------------

addr            VAR     Byte                        ' address pointer
char            VAR     Byte                        ' character sent to LCD
idx             VAR     Byte                        ' loop counter
value           VAR     Word                        ' value for RJ_Print
width           VAR     Nib                         ' width of RJ field
pad             VAR     Nib                         ' space pad for field



' -----[ EEPROM Data ]-----------------------------------------------

CC0             DATA    $0E, $1F, $1C, $18, $1C, $1F, $0E, $00  ' char 0
CC1             DATA    $0E, $1F, $1F, $18, $1F, $1F, $0E, $00  ' char 1
CC2             DATA    $0E, $1F, $1F, $1F, $1F, $1F, $0E, $00  ' char 2



' -----[ Initialization ]--------------------------------------------

Initialize:
  NAP 5                                             ' let LCD self-initialize
  DIRL = %11111110                                  ' setup pins for LCD

LCD_Init:
  #IF _LcdReady #THEN
    LCDCMD E, %00110000 : PAUSE 5                   ' 8-bit mode
    LCDCMD E, %00110000 : PAUSE 0
    LCDCMD E, %00110000 : PAUSE 0
    LCDCMD E, %00100000 : PAUSE 0                   ' 4-bit mode
    LCDCMD E, %00101000 : PAUSE 0                   ' multi-line mode
    LCDCMD E, %00001100 : PAUSE 0                   ' no crsr, no blink
    LCDCMD E, %00000110                             ' inc crsr, no disp shift
  #ELSE
    #ERROR "Program requires BS2p or BS2pe"
  #ENDIF
```

```
Download_Chars:                                    ' download custom chars
  LCDCMD E, LcdCGRam                                ' prepare to write CG data
  FOR idx = CC0 TO (CC2 + 7)                        ' build 3 custom chars
    READ idx, char                                 ' get byte from EEPROM
    LCDOUT E, NoCmd, [char]                         ' put into LCD CG RAM
  NEXT


' -----[ Program Code ]------------------------------------------------

Main:
  LCDCMD E, LcdCls                                  ' blank screen, home cursor
  PAUSE 500
  LCDOUT E, LcdLine1, ["PARALLAX"]                  ' easy banner
  LCDOUT E, LcdLine2, ["THE BASIC STAMP"]
  PAUSE 2000

Animation:
  FOR addr = 0 TO 15                               ' cover 16 characters
    FOR idx = 0 TO 4                               ' 5 characters in cycle
      LOOKUP idx, [2, 1, 0, 1, " "], char          ' get new character
      LCDOUT E, (LcdLine2 + addr), [char]          ' write it
      PAUSE 75                                      ' frame delay
    NEXT
  NEXT

Counter_Demo:
  LCDOUT E, LcdCls, ["Counter"]
  width = 5                                         ' field is 5-wide
  FOR value = 1 TO 12345
    LCDCMD E, (LcdLine2 + 11)                       ' position cursor
    GOSUB RJ_Print                                  ' right justify
    PAUSE 100
  NEXT

  GOTO Main

  END


' -----[ Subroutines ]------------------------------------------------

' Prints right-justified value on LCD
' -- preposition cursor
' -- put width of field in 'width' (1 - 5)
' -- put value to print in 'value' (0 - 65535)
```

```
RJ_Print:
  LOOKDOWN value, >=[10000, 1000, 100, 10, 0], pad
  pad = pad – (5 – width)
  LCDOUT E, NoCmd, [REP " "\pad, DEC value]  RETURN
```

Experimenting with Philips I^{2}C Components

There are myriad components that communicate via the Philips I^{2}C bus, and the BASIC Stamp 2p family and BASIC Stamp 2pe makes connecting to these parts pretty simple with the I2COUT and I2CIN instructions. When using I^{2}C components, the SDA and SCL lines must be pulled up to Vdd (5 volts), usually through 4.7 kΩ resistors. The BS2p/BS2pe limits the SDA and SCL connections to pins 0 and 1 or 8 and 9, respectively. When using the BS2p40, I^{2}C commands may also use the auxiliary I/O (AUXIO) pins, doubling the possible number of I^{2}C connections. Refer to the Parallax Stamp Manual (version 2.0 or later) or BASIC Stamp Editor help file for details.

Project Schematic

Note: When using iB1 and iB2 pull-ups, be sure to remove any iButton devices from the demo board.
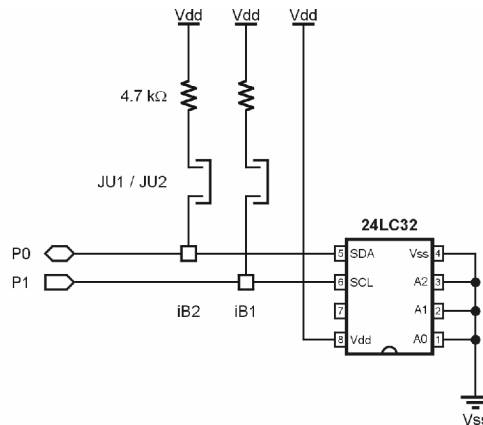
Demo Board Connections
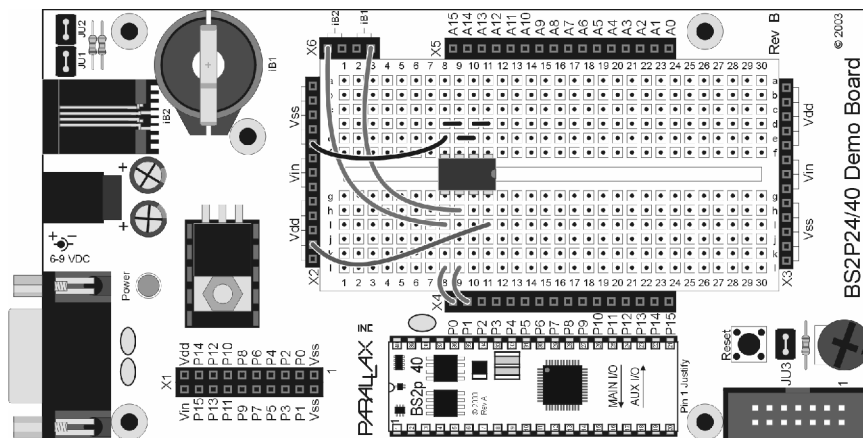
Source Code

```
' =========================================================================
'
'   File...... P24-40_24LC32.BSP
'   Purpose... Demonstrates I2COUT and I2CIN
'   Author.... Parallax, Inc.
'             Copyright (c) 2004, All Rights Reserved
'   E-mail.... support@parallax.com
'   Started...
'   Updated... 17 JUN 2004
'
'   {$STAMP BS2p}
'   {$PBASIC 2.5}
'
' =========================================================================



' -----[ Program Description ]---------------------------------------------
'
' Writes to and reads from I2C EEPROM.  Data is displayed in the DEBUG
' window.



' -----[ I/O Definitions ]-------------------------------------------------

SDA             PIN     0                       ' SDA on 0; SCL on 1



' -----[ Constants ]-------------------------------------------------------

#DEFINE _I2cReady = ($STAMP = BS2P) OR ($STAMP = BS2PE)

MaxEE           CON     4095                    ' highest EE address



' -----[ Variables ]-------------------------------------------------------

addr            VAR     Word                    ' EE address
addrHi          VAR     addr.HIGHBYTE
addrLo          VAR     addr.LOWBYTE
rVar            VAR     Word                    ' for random number
tOut            VAR     Byte                    ' test value to LCD
tIn             VAR     Byte                    ' test value read from LCD
value           VAR     Word                    ' value for RJ_Print
width           VAR     Nib                     ' width of RJ field
pad             VAR     Nib                     ' space pad for field
```

```
' -----[ Initialization ]------------------------------------------------

Reset:
  #IF (_I2cReady) #THEN
    DEBUG CLS,
          "I2C Demo", CR, CR,
          "Address... ", CR,
          "I2C Out... ", CR,
          "I2C In.... ", CR,
          "Status.... "
  #ELSE
    #ERROR "Program requires BS2p or BS2pe"
  #ENDIF


' -----[ Program Code ]-------------------------------------------------

Main:
  FOR addr = 0 TO MaxEE STEP 5                   ' create addresses
    RANDOM rVar                                  ' create "random" value
    tOut = rVar.HIGHBYTE
    I2COUT SDA, $A0, addrHi\addrLo, [tOut]       ' write value
    PAUSE 100
    I2CIN SDA, $A1, addrHi\addrLo, [tIn]         ' read it back

    DEBUG CRSRXY, 11, 2, IHEX3 addr              ' report results
    width = 4
    value = tOut
    DEBUG CRSRXY, 11, 3
    GOSUB RJ_Print
    value = tIn
    DEBUG CRSRXY, 11, 4
    GOSUB RJ_Print
    DEBUG CRSRXY, 11, 5
    IF (tIn = tOut) THEN
      DEBUG "Pass"
    ELSE
      DEBUG "Fail"
    ENDIF

    PAUSE 250
  NEXT

  PAUSE 2000
  DEBUG CR, CR, "Done!"

  END
```

© Parallax, Inc. • BS2p24/40 Demo Board (#45187) • 10/2004

```
' -----[ Subroutines ]-------------------------------------------------


' Prints right-justified value on DEBUG screen
' -- preposition cursor
' -- put width of field in 'width' (1 - 5)
' -- put value to print in 'value' (0 - 65535)

RJ_Print:
  LOOKDOWN value, >=[10000, 1000, 100, 10, 0], pad
  pad = pad - (5 - width)
  DEBUG REP " "\pad, DEC value
  RETURN
```

Experimenting With Dallas/Maxim 1-Wire Components

The BASIC Stamp 2p family and BS2pe microcontroller simplifies connecting to Dallas/Maxim 1-Wire components with the OWOUT and OWIN commands. The purpose of this experiment is to demonstrate these 1-Wire commands by reading the unique ROM code from an iButton device. When using 1-Wire components, the bus pin must be pulled-up to Vdd (5 volts). The standard value for this pull-up is 4.7 kΩ, and the BS2p24/40 demo board makes these pull-ups available via JU1 (for iB1) and JU2 (for iB2).

Project Schematic
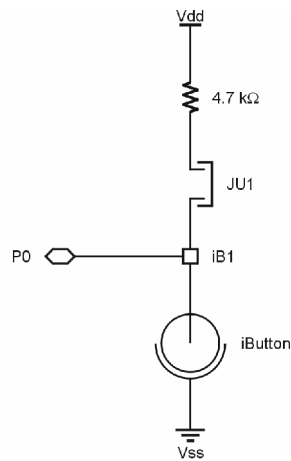
Demo Board Connections
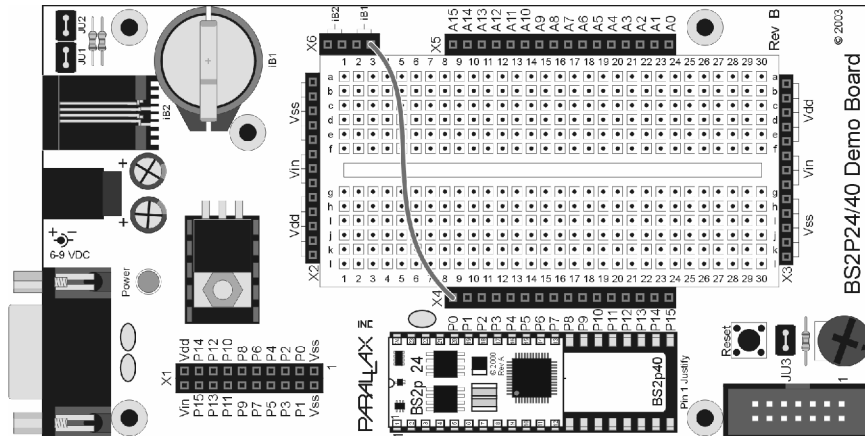
Source Code

```
' =============================================================================
'
'   File...... P24-40_1WIRE_ID.BSP
'   Purpose... 1-Wire device identification
'   Author.... Parallax, Inc.
'             Copyright (c) 2004, All Rights Reserved
'   E-mail.... support@parallax.com
'   Started...
'   Updated... 17 JUN 2004
'
'   {$STAMP BS2p}
'   {$PBASIC 2.5}
'
' =============================================================================



' -----[ Program Description ]-------------------------------------------------
'
' Reads and displays the serial number from a 1-Wire device.



' -----[ I/O Definitions ]-----------------------------------------------------

OW              PIN     0                       ' 1-Wire bus connection



' -----[ Constants ]-----------------------------------------------------------

#DEFINE _OwReady = ($STAMP = BS2P) OR ($STAMP = BS2PE)

OwFERst         CON     %0001                   ' Front-End Reset
OwBERst         CON     %0010                   ' Back-End Reset
```

```
OwBitMode        CON      %0100
OwHighSpd        CON      %1000

SearchROM        CON      $F0
ReadROM          CON      $33                        ' read ID, serial num, CRC

NoDevice         CON      %11                        ' no device present


' -----[ Variables ]-------------------------------------------------

idx              VAR      Byte                       ' loop counter
romData          VAR      Byte(8)                    ' ROM data from device
devCheck         VAR      Nib                        ' device check return ocde


' -----[ Initialization ]--------------------------------------------

Reset:
  #IF (_OwReady) #THEN
    DEBUG CLS,
          "1-Wire S/N Reader", CR, CR
  #ELSE
    #ERROR "Program requires BS2p or BS2pe"
  #ENDIF


' -----[ Program Code ]----------------------------------------------

Main:
  GOSUB Get_SN                                       ' read serial number
  IF (romData(7) < $FF) THEN                         ' if valid
    FOR idx = 7 TO 0
      DEBUG HEX2 romData(idx)                        '   show S/N data
      IF ((idx = 7) OR (idx = 1)) THEN
        DEBUG CR                                     ' CR after CRC and SN
      ENDIF
    NEXT
  ELSE
    DEBUG "Bad or no device"                         ' else questionable
  ENDIF

  END


' -----[ Subroutines ]-----------------------------------------------

' Read 8-byte serial number from 1-Wire device
```

```
Get_SN:
  OWOUT OW, OwFERst, [ReadROM]                ' send Read ROM command
  OWIN  OW, OwBERst, [STR romData\8]          ' read serial number & CRC
  RETURN
```

Additional Experiments

For more experiments with the BS2p family and BS2p24/40 Demo Board, you may wish to download the BS2p "Plus Pack" documentation and code samples from Parallax, free of charge. You may also order the BS2p "Plus Pack" (#45184) via the Internet or directly from Parallax.

Using Other Parallax Microcontrollers

The BS2p24/40 Demo Board is electrically compatible with all Parallax 24-pin microcontrollers, including the Javelin Stamp microcontroller. The demo board is particularly useful for experiments with parallel LCDs (download the StampWorks experiments from Parallax for LCD examples using the BS2, BS2e and BS2sx). Please note that 1-Wire interfacing is not possible with the BS2, BS2e, BS2sx or Javelin without an external co-processor. I^2C communications with these microcontrollers is possible with code.

When using a 24-pin Parallax microcontroller with the BS2p24/40 Demo Board, be sure to align the module as shown in the diagram below. Also note that when using a 24-pin module, pins P8 through P15 are accessed via sockets A8 through A15.