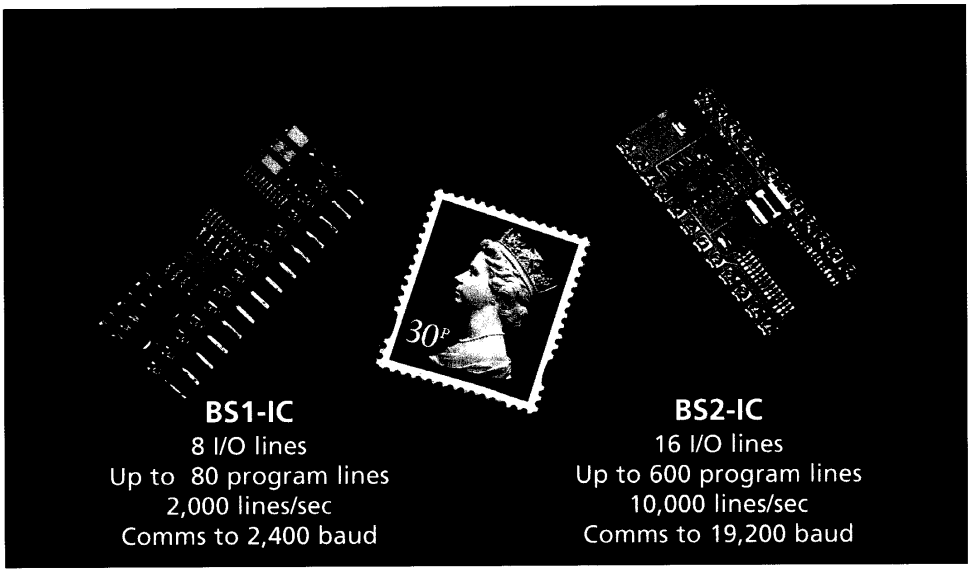
BASIC STAMPS *from* MILFORD INSTRUMENTS

Two new re-programmable BASIC Stamp Computers



BS1-IC
8 I/O lines
Up to 80 program lines
2,000 lines/sec
Comms to 2,400 baud

BS2-IC
16 I/O lines
Up to 600 program lines
10,000 lines/sec
Comms to 19,200 baud

BASIC Stamp Modules

BASIC computers as small as 4.5cm²

BASIC Stamps are small modules that run BASIC programs. Their low cost and powerful features make them perfect for many prototyping and control applications. Their ease of programming means greatly reduced development time yet allows specific features to be easily included.

Stamp Modules:

BS1-IC: (part number 1-102) 10x35mm surface mount board with convenient 14-pin SIP connection provides 8 I/O lines. On board 256 byte EEPROM can hold up to 80 instruction lines and programs are executed at 2,000 lines per second.

BS2-IC: (part number 1-103) 24-pin DIP format package provides 16 I/O lines. On board 2Kbyte EEPROM can hold up to 600 instruction lines and programs are executed at 10,000 lines per second.

Hardware: the brain of either module is an 18- or 28-pin BASIC interpreter. An 8-pin EEPROM holds a tokenized version of your program which is executed by the interpreter. The remainder of the circuit board is taken up with a 5-volt regulator, brown-out circuit and a 4- or 20-MHz resonator.

BASIC Language: the Stamps are programmed in a simple BASIC language. The language includes familiar instructions such as GOTO, FOR...NEXT, and IF...THEN as well as SBC instructions such as SERIN (serial input) PWM and BUTTON (button debounce). A full command list is included overleaf.

Programming: to write software for the Stamps, you'll need the Programming Package (part number 1-433). The package contains PC based editor software, programming cables, manual, extensive application notes and free technical support.

- BASIC language includes instructions for serial I/O, PWM, potentiometer input, pulse measurement, button debounce, tone generation, etc.
- Stamps have 8 or 16 I/O lines, each programmable as an input or output. Each can sink 25mA and source 20mA.
- Programs are stored in EEPROM so they are retained if power is removed and can be changed as often as required.
- Stamps need connect to PC only during programming, thereafter they run independently.
- Powered by 5-12 VDC or 9-volt battery.
- Consume down to just 2mA (typ.) or 20μA (sleep mode)
- BS1-IC comes in convenient 14-pin SIP package.
- BS2-IC has expanded I/O, more program memory and faster execution speed; all conveniently packaged in a 24-pin DIP format package.

To order or request information, please call Milford Instruments Tel: 01977 683665 or Fax: 01977 681465

BASIC Instruction Set

The Stamps' BASIC language includes both familiar program and SBC instructions.
Commands particular to the BS1 or BS2 are shown thus [ ]

PROGRAM FLOW

FOR.. TO.. NEXT	Establish a FOR-NEXT LOOP
BRANCH	Branch to address specified by offset
GOTO	Branch to address
GOSUB	Branch to subroutine at address.
RETURN	Return from subroutine
IF.. THEN	Compare and conditionally branch.

VARIABLE MANIPULATION

{LET}	Perform variable manipulation. Arithmetic operators, max,min and logical operators [16 bit maths plus SQR,SIN,COS,ABS for BS2 only]
LOOKUP	Lookup data specified by offset and store in variable.
LOOKDOWN	Find target's match number (0-N) and store in variable
RANDOM	Generate a pseudo-random number

SERIAL I/O

SEROUT	Output serial data from pin. Up to 2400 baud for BS1. Up to 19,200 baud for BS2 with both flow control and time-out facilities.
SERIN	Serial input data with qualifiers on specified pin. Up to 2,400 baud for BS1. Up to 19,200 baud for BS2 with both flow control and time-out facilites

INTERNAL EEPROM ACCESS

READ	Read EEPROM location into variable
WRITE	Write variable into EEPROM location
EEPROM [BS1]	Store data in EEPROM before downloading BASIC program
DATA [BS2]	Store data in EEPROM before downloading BASIC program

DIGITAL I/O

XOUT [BS2]	Output signals to the X-10 home automation specifications
SHIFTOUT [BS2]	Shift out data synchronously through defined pin
SHIFTIN [BS2]	Shift in data synchronously through defined pin
COUNT [BS2]	Count pin cycles in given time. Max frequency 125kHz
PULSOUT	Output pulse on pin for given time. (resolution; 2µs/BS2, 10µs/BS1)
PULSIN	Input pulse on pin. (resolution; 2µs/BS2, 10µs/BS1)
BUTTON	Perform debounce / auto repeat/ branch if button in target state
INPUT/OUTPUT	Make pin input /output
HIGH/LOW	Make pin high / low output
TOGGLE/REVERSE	Make pin output and toggle. / If output make input and vice versa

ANALOG

FREQOUT [BS2]	Output tone for given time (1Hz resolution)
DTMF [BS2]	Output touch tones: standard 16 key keypad format
PWM	Output PWM on pin (8-bit resolution). Use for pseudo analog output.
RCTIME	Input time to charge capacitor through resistor. [POT command on BS1]
SOUND[BS1]	Output note through pin

POWER DOWN

NAP	Enter low power mode for short period (up to 2.3 sec)
SLEEP	Enter low power mode for period (up to 65535 secs)
END	Power down until reset

MISCELLANEOUS

PAUSE	Pause execution for up to 65535 secs
DEBUG	Send data back to the PC for debugging purposes
STOP [BS2]	Stop (don't power down)

Note 1: LCD User-Interface Terminal

Introduction. This application note presents a program in Parallax BASIC that enables the BASIC Stamp to operate as a simple user-interface terminal.

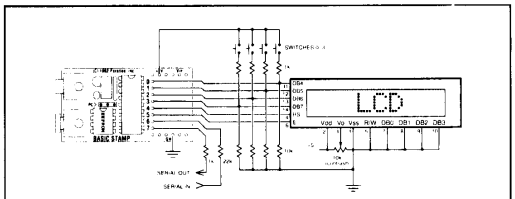
Background. Many systems use a central host computer to control remote functions. At various locations, users communicate with the main system via small terminals that display system status and accept inputs. The BASIC Stamp's ease of programming and built-in support for serial communications make it a good candidate for such user-interface applications.

The liquid-crystal display (LCD) used in this project is based on the popular Hitachi 44780 controller IC. These chips are at the heart of LCD's ranging in size from two lines of four characters (2x4) to 2x40.

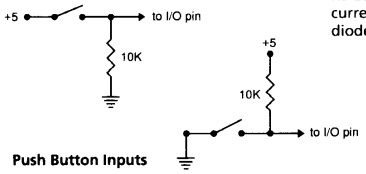
How it works. When power is first applied, the BASIC program initializes the LCD. It sets the display to print from left to right, and enables an underline cursor. To eliminate any stray characters, the program clears the screen.

After initialization, the program enters a loop waiting for the arrival of a character via the 2400-baud RS-232 interface. When a character arrives, it is checked against a short list of special characters (backspace, control-C, and return). If it is not one of these, the program prints it on the display, and re-enters the waiting-for-data loop.

If a backspace is received, the program moves the LCD cursor back one



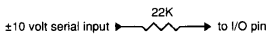
Schematic to accompany program TERMINAL.BAS.



Push Button Inputs

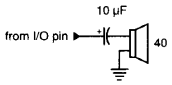
Top: active high
Right: active low
Instruction that uses these circuits: BUTTON.

RS-232 Serial Input. Uses a resistor to limit input current. Input voltage is limited by PIC's internal diodes. Instruction that uses this circuit: SERIN.



Audio Output. Uses a capacitor and speaker (or piezo) generates tones.

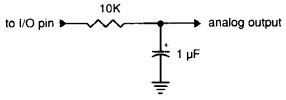
Instruction that uses this circuit: SOUND.



Remove capacitor if using piezo electric speaker

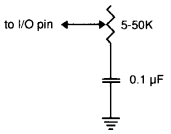
PWM Analog Output. Uses PWM to form a slow D/A convertor.

Instruction that uses this circuit: PWM.



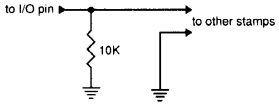
Potentiometer Input. Reads potentiometer by timing how long it takes to fill the capacitor.

Instruction that uses this circuit: POT.



Network Node. This circuit shows how to connect a Stamp to a network of multiple Stamps.

Instructions that might use this circuit SERIN, SEROUT.



BASIC Stamp Application Notes

Helpful application notes reduce learning time.
When you purchase the BASIC Stamp Programming Package, you'll not only receive the necessary hardware to program the Stamp, you'll also receive a collection of Stamp application notes; similar to the example shown:

BASIC Stamp Programming Language

Clear but powerful syntax speeds development time

Note 1: LCD User-Interface Terminal

```
Symbol   Sw_0   =   pin0              ' User input switches
Symbol   Sw_1   =   pin1              ' multiplexed w/LCD data lines.
Symbol   Sw_2   =   pin2
Symbol   Sw_3   =   pin3

' Set up the Stamp's I/O lines and initialize the LCD.
begin:   let pins = 0                 ' Clear the output lines
         let dirs = %01111111         ' One input, 7 outputs.
         pause 200                    ' Wait 200 ms for LCD to reset.

' Initialize the LCD in accordance with Hitachi's instructions for 4-bit interface.
i_LCD:   let pins = %00000011         ' Set to 8-bit operation.
         pulsout E,1                  ' Send data three times
         pause 10                     ' to initialize LCD.
         pulsout E,1
         pause 10
         pulsout E,1
         pause 10
         let pins = %00000010         ' Set to 4-bit operation.
         pulsout E,1                  ' Send above data three times.
         pulsout E,1
         pulsout E,1
         let char = 14                ' Set up LCD in accordance with
         gosub wr_LCD                 ' Hitachi instruction manual.
         let char = 6                 ' Turn on cursor and enable
         gosub wr_LCD                 ' left-to-right printing.
         let char = 1                 ' Clear the display.
         gosub wr_LCD
         high    RS                   ' Prepare to send characters.

' Main program loop: receive data, check for backspace, and display data on LCD.
main:    serin S_in,N2400,char        ' Main terminal loop.
         goto bksp
out:     gosub wr_LCD
         goto main

' Write the ASCII character in b3 to LCD.
wr_LCD:  let pins = pins & %00010000
         let b2 = char/16             ' Put high nibble of b3 into b2.
         let pins = pins | b2         ' OR the contents of b2 into pins.
         pulsout E,1                  ' Blip enable pin.
         let b2 = char & %00001111    ' Put low nibble of b3 into b2.
         let pins = pins & %00010000  ' Clear 4-bit data bus.
         let pins = pins | b2         ' OR the contents of b2 into pins.
         pulsout E,1                  ' Blip enable.
         return

' Backspace, rub out character by printing a blank.
bksp:    if char > 13 then out        ' Not a bksp or cr? Output character.
         if char = 3 then clear       ' Ctl-C clears LCD screen.
```

BASIC Stamp I/O Circuits

Simple circuits enable the Stamp to receive/send RS-232 data, read potentiometers, keypads etc.

To order or request information, please call Milford Instruments Tel: 01977 683665 or Fax: 01977 681465

111 285

BASIC Stamp Programming Package

(part number 1-433)
Includes everything necessary to program Stamp modules from your PC.

The package includes editor software, programming cables, documentation, application notes and free technical support.

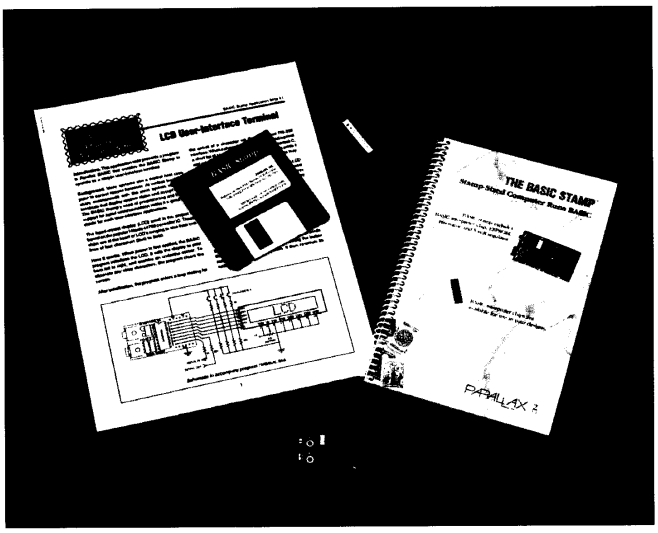
To use the Programming Package, you'll need an IBM PC or compatible with a 3.5" disk drive and DOS 2.0+. For programming, your PC must have an available parallel port for the BS1-IC and an available serial port for the BS2-IC.

Stamps are sold separately. Most customers purchase the Programming Package with one or two Stamp modules. You may also wish to purchase one or more corresponding project boards since they provide easy connection to the serial and parallel programming cables.

BASIC Stamp Project Boards

(part numbers 1-474 and 1-475)

Both Stamp modules have corresponding project boards. These optional boards provide an easy means to program and prototype with the Stamp modules. Both boards measure approx 35x85mm and provide prototyping area, Stamp connection area, 9-volt battery clips, I/O header, programming connector and D-9 connector area.
Project Board 1-474 - BS1-IC
Project Board 1-475 - BS2-IC



BASIC Stamp Chipsets

Various chipsets are available for the BS1 and BS2 and offer the cheapest means of building the Stamp into your own high-volume design. Please call for available options and pricing.

PIC Development Tools

We also stock a complete range of development tools for the popular PIC series of microcontrollers (as used in the Stamp). Please call to receive our separate brochure with full details of programmers, adaptors and true, real-time ICE's using genuine bondout chipsets.

	Programmer	Emulator	Assembler	Software Simulator	16C5x	16C61	16C64	16C71	16C74	16C84	17C42	Execution speed (MHz)	Programming Adaptors
PIC 16Cxx Programmer	●		●	●	●	●	●	●	●	●			ZIF, SOIC, SSOP, PLCC
BackDraft 17	●		●								●		PLCC
ClearView '5x		●	●	●	●							20	
ClearView 'xx		●	●	●		●	●	●	●	●		20	

MILFORD INSTRUMENTS – *Creative Products for Enquiring Minds*

Milford House, 120, High Street,
SOUTH MILFORD, LEEDS LS25 5AQ, ENGLAND.
Tel: 01977 683665 (24 hr) Fax: 01977 681465
Internet: Info@Milinst.Demon.co.uk