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

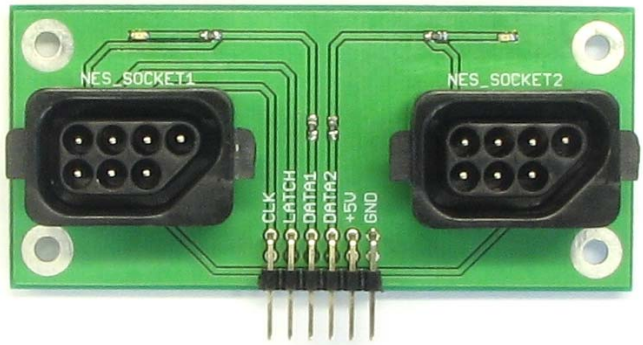
Web Site: www.parallax.com
Forums: forums.parallax.com
Sales: sales@parallax.com
Technical: support@parallax.com

Office: (916) 624-8333
Fax: (916) 624-8003
Sales: (888) 512-1024
Tech Support: (888) 997-8267

NES Gamepad Controller Adapter (#32368)

The NES Gamepad Controller Adapter is a printed circuit board with two Nintendo® gamepad compatible sockets routed to a dual row of pins with breadboard spacing. This adapter makes it convenient to connect up to two gamepad controllers to a breadboard for gamepad-controlled projects.

NOTE: This adapter is designed for gamepads only. It does not have signal lines for Nintendo Zappers and Power Pads.



Features

- Breadboard-friendly 0.1 inch pin spacing
- Works great with Nintendo gamepad style controllers (#32365)
- Plated holes for mounting in a project box
- Two adjacent SIP headers for stability
- Data connection indicator LEDs
- Compatible with BASIC Stamp®, SX, and Propeller™ microcontrollers
- Built-in series resistors protect 3.3 V Propeller I/O pin inputs

Key Specifications

- Power requirements: +5 VDC, 25 mA
- Communication: TTL synchronous serial
- Dimensions: 3.1 x 1.6 x 0.91 in (79 x 41 x 23 mm)
- Operating temperature: 32 to 131 °F (0 to 55 °C)

Application Ideas

- Gamepad controlled robot
- Gamepad controlled mechanical arm
- Video game system on a breadboard

Quick Start Circuit

Figure 1 shows a schematic that can be used with Nintendo gamepad controllers and the BASIC Stamp 2 and Propeller test programs featured in the Source Code section. P4 and P5 drive the clock and latch lines. P6 receives data from NES_SOCKET1, and P7 receives data from NES_SOCKET2.

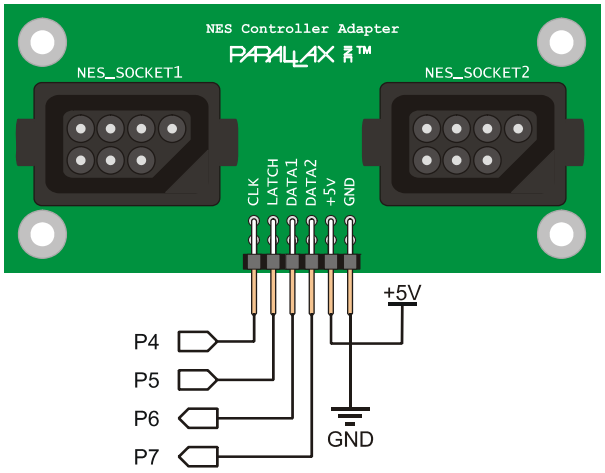
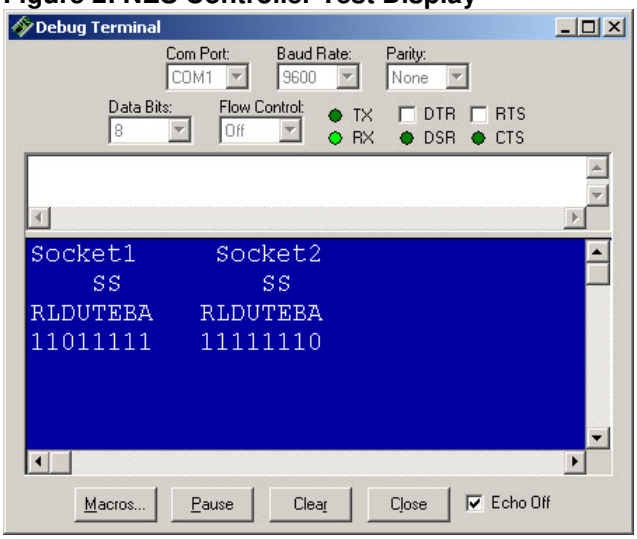


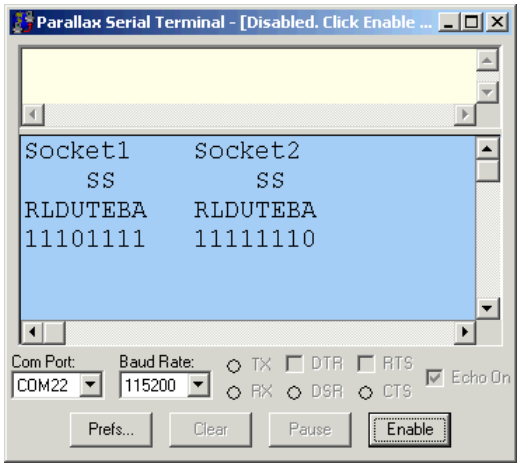
Figure 1: Quick Start Schematic

Connecting and Testing

Figure 2 shows test results for the BASIC Stamp and Propeller microcontrollers while they execute the test programs featured in the Source Code section. Messages from the BASIC Stamp are displayed by the Debug Terminal (left), and messages from the Propeller chip are displayed by the Parallax Serial Terminal (right). The buttons from left to right are Right, Left, Down, Up, Start, Select, B, and A. When one of those buttons is pressed and held, the corresponding bit displays a 0; otherwise, it displays a 1. The AA button causes the A bit to cycle on/off while, and likewise with the BB button. The SLOW button toggles the ST bit. Press it once, and the ST bit cycles on/off; press it again, and it returns to 1.

Figure 2: NES Controller Test Display





- ✓ Build the circuit shown in Figure 1.
- ✓ Download the Source Code from the 32368 product page at www.parallax.com.
- ✓ Load it into your microcontroller.
- ✓ Use the Debug Terminal (BASIC Stamp) or the Parallax Serial Terminal (Propeller) to display the status display from your microcontroller. See Figure 2.
- ✓ Try pressing the buttons on the controller and examine the way each bit responds.

Resources and Downloads

The NES Gamepad Controller Adapter product page has this PDF, source code, and a link to Chapter 6 in the Hydra Game Development System manual for additional information about the Nintendo gamepad. Go to www.parallax.com and "search" for 32368.

Device Information

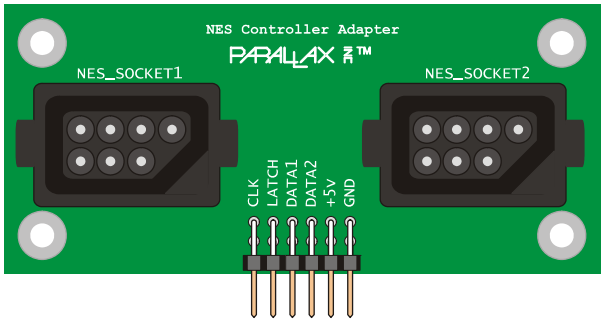
Pin Definitions

Figure 3 shows the module pinout for the pin definitions listed in Table 1. Ratings are not listed because they will be specific to the make and manufacture of the gamepad that gets connected to the adapter.

Table 1: Pin Definitions

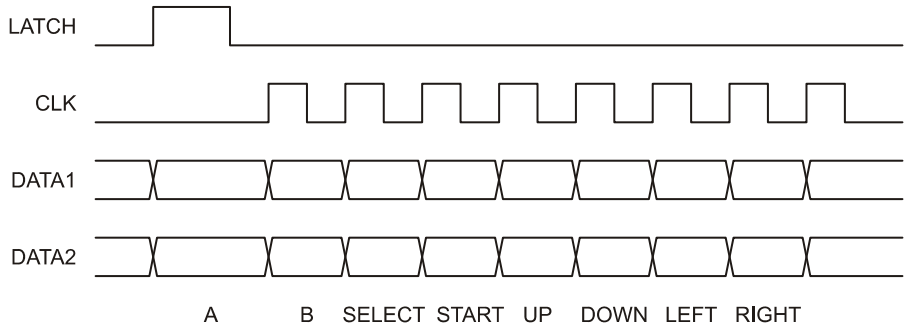
Name	Function
CLK	Clock input
LATCH	Latch input
DATA1	Data output for NES_SOCKET1
DATA2	Data output for NES_SOCKET2
+5V	Positive 5 V supply connection
GND	Ground 0 V supply connection

Figure 3: Pinout

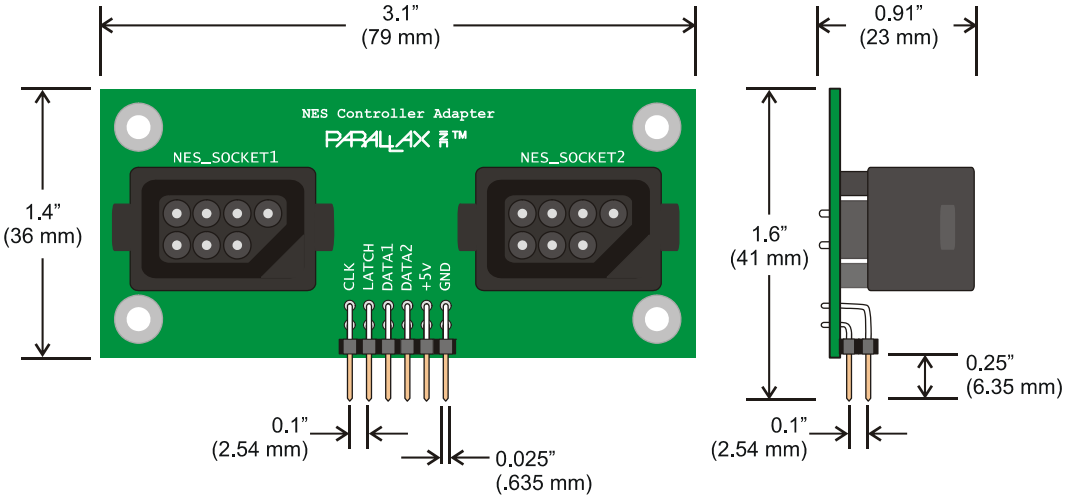


Communication Protocol

A pulse to the LATCH pin causes the gamepad to latch the button values and transmit the first (A button) bit. The rising edges of the next seven clock pulses deliver the remaining bits (B through RIGHT buttons). Although both data lines can be shifted in simultaneously, the microcontroller does not necessarily have to record both data lines at once. For example, the BASIC Stamp 2 test code in the source code section checks each DATA line individually with separate PULSOUT and SHIFTIN commands. Note that since the first data value is available before the first clock pulse, the PBASIC SHIFTIN command uses the LSBPRE argument to specify that the data line transmits the latest value as a pre-clock pulse.



Module Dimensions



Source Code

BASIC Stamp® 2 Program

```
' Test NES Controllers.bs2
' Display NES controller bit states, see Parallax NES Controller
' Adapter (Item# 32368) documentation at www.parallax.com.

' {$STAMP BS2}                                      ' BASIC Stamp 2 module
' {$PBASIC 2.5}                                     ' PBASIC 2.5 language

value          VAR     Byte(2)                     ' Store NES data

PAUSE 1000                                         ' Wait a second
DEBUG "Socket1    Socket2", CR,                    ' Display headings
      "    SS         SS", CR,
      "RLDUTEBA   RLDUTEBA", CR

LOW 5                                              ' Initialize LATCH
LOW 4                                              ' Initialize CLK

DO                                                 ' Main loop

  PULSOUT 5, 10                                    ' Latch the data
  SHIFTIN 6, 4, LSBPRE, [value(0) \8]              ' Shift in socket 1 data

  PULSOUT 5, 10                                    ' Latch the data
  SHIFTIN 7, 4, LSBPRE, [Value(1) \8]              ' Shift in socket 1 data

  DEBUG CRSRXY, 0, 3, BIN8 value(0),               ' Display values
        CRSRX, 11, BIN8 value(1)

LOOP                                               ' Repeat main loop
```

Propeller Test Application

```
''File: Test Two NES with PST.spin
''Test two NES controllers and display with Parallax Serial Terminal.
''Requires the NES.spin from John Abshier's NES Controller object.  It's available from the
''Parallax Object Exchange at obex.parallax.com.

CON                                                     ' Constant declaration block

  _clkmode = xtal1 + pll16x                             ' 5 MHz crystal → 80 MHz system clock
  _xinfreq = 5_000_000
                                                        ' I/O pin & power connections
  DATA2       = 7                                       '          P7 ←——————< DATA2
  DATA1       = 6                                       '          P6 ←——————< DATA1
  LATCH       = 5                                       '          P5 >——————→ CLK
  CLK         = 4                                       '          P4 >——————→ LATCH
                                                        '         5 V >——————→ +5V
                                                        '   (Vss) GND >——————→ GND

OBJ                                                     ' Object declaration block

  pst      : "Parallax Serial Terminal"                 ' Serial Terminal display
  nes[2]   : "NES"                                      ' NES object

PUB Main  | buttons[2], index, dataPin                  '' Main method

  ' Initialize display
  pst.Start(115200)                                     ' Start Parallax Serial Terminal
  pst.Str(String("Socket1    Socket2", pst#NL))         ' Display headings
  pst.Str(String("    SS         SS", pst#NL))
  pst.Str(String("RLDUTEBA   RLDUTEBA", pst#NL))

  ' Initialize NES Objects
  repeat index from 0 to 1                              ' Loop through 2 NES objects
    dataPin := lookupz(index: DATA1, DATA2)             ' Get data pin
    NES[index].Init(LATCH, datapin, CLK)                ' Get type of game pad

  ' Main Loop
  repeat                                                ' Main loop
    pst.Home                                            ' Cursor to top-left home position
    repeat index from 0 to 1                            ' Loop through 2 NES objects
      buttons[index] := nes[index].buttons              ' Get button states from game pad
      pst.Position(11*index, 3)
      pst.Bin(buttons[index], 8)                        ' Display button states
      Pst.NewLine                                       ' Carriage return
    waitcnt(clkfreq/10+cnt)                             ' 1/10 s delay
```