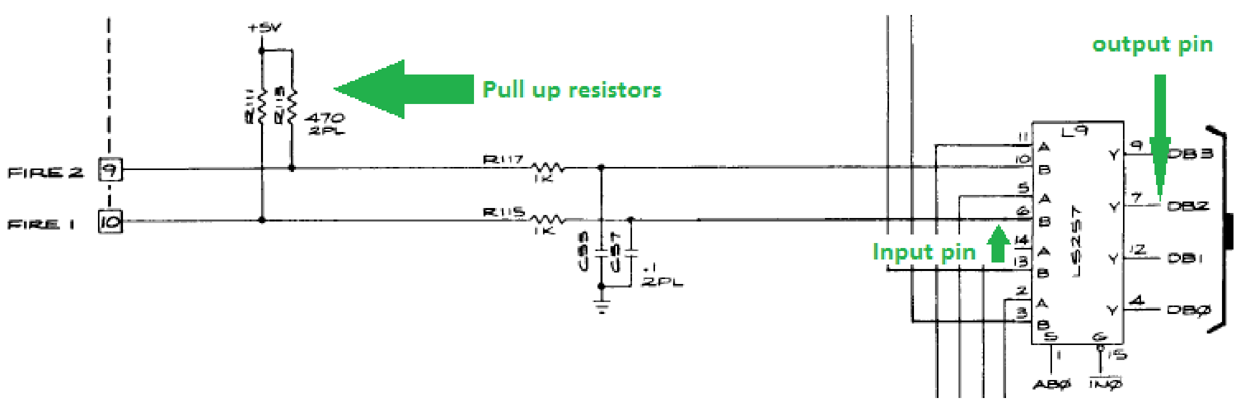
Lesson 7: Testing Inputs Version 1.03

Obviously an important component of an arcade game is it's inputs (joysticks, buttons etc). Without these working you simply cannot interact with the game. While it is fairly uncommon for inputs to break it is worth mentioning how to determine if they are working or not, and it's very easy to test them with nothing but a logic probe or even an multimeter in DC voltage mode. This section will allow you to test any digital (on/off) input from joystick buttons, pushbuttons to DIP switches. To do this you have to get out the schematics to see where the buttons inputs go to. Before we look at the schematics though there is some background we need. Buttons, joysticks and dipswitches work because when you press the button it closes a switch from the button input to the GROUND wire. There is a chip (often something like a 74LS244) that the wire to the switch goes to. Normally the input pin on this circuit is directly connected to a resistor that is directly connected to the +5V power line. This setup is called a *pull up resistor*. Normally when the switch is NOT activated the input to the chip (through the resistor) is held high or +5V via the resistor/power source. When you press the button the switch on the button closes the connection, creating a direct path from the input pin to the GROUND on the other side of the button.

So each buttons signal enters a chip via a pin, but the button signal also exits the chip on a pin on the way to the CPU via the *data bus*. The output pin should work just like the input, normally it will be high, however when you press the button it should go low. (It is also possible that these might be reversed on some boards too ie. Output is normally low, and goes high when pressed)



To test the inputs, first test the input to the pin, and verify it's high (+5V) when not pressed, and that it goes low when the button is pressed. If this happens you are good from the chip input to the button's switch and that switch is wired up correctly to ground. If you find that the input is normally LOW or that the input is normally HIGH, but never goes LOW, then physically check your button's microswitch and check that the wiring to the game board is OK.

If everything works OK, the next step is to check out output of the chip, and perform the test again. If you find any inconsistancies with the output (that is it's not high when button unpressed, or does not go low when the button is pressed), simply replace the chip in question.

Exercise 1: Test the Centipede *fire* button using a logic probe or multimeter

1. Power up your Centipede game board
2. If using a logic probe connect your logic probe power to +5 and GROUND as appropriate
3. If using a multimeter, turn your meter on and set it on DC voltage test
4. Now locate the chip and input pin that reads the fire button on the Centipede PCB board (pin 6 on the 74LS257 @ L9)
5. Place your logic probe on pin 6 (or multimeter red probe on pin 6 and black probe on any ground test point)
6. Verify you see +5V or “high logic state”
7. Push the fire button
8. Verify you see 0V (GROUND) or “low logic state”
9. Now locate the chip and output pin on the 74LS257 that reads the fire button (pin 7)
10. Place your logic probe on pin 7 (or multimeter red probe on pin 7 and black probe on any ground test point)
11. Verify you see +5V or “high logic state”
12. Push the fire button
13. Verify you see 0V (GROUND) or “low logic state”

Testing inputs with the Fluke 9010A.

As far as the CPU is concerned all inputs are simply values that the CPU reads from a certain memory address. The CPU constanty reads various memory addresses and looks to see what state the bits are in. These controls are usually mapped to one or more bits on one or more addresses. Normally when a button is open the bit is “1” and when closed the bit is “0”. On the Centipede PCB you can see from the memory map that address 0x0C01 is where many of the buttons (player 1 and 2 fire, player one and 2 start, as well as the coin detectors are mapped to). You can tell the Fluke to constanty read these addresses in “loop” mode while you press them and watch the values changes on the Fluke LED display.

MEMORY MAP										
HEXA-DECIMAL ADDRESS	**R/W**	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**	**FUNCTION**
0000-03FF		D	D	D	D	D	D	D	D	RAM
0400-07BF		D	D	D	D	D	D	D	D	Playfield RAM
07C0-07CF		D	D	D	D	D	D	D	D	Motion Object Picture
07D0-07DF		D	D	D	D	D	D	D	D	Motion Object Vert.
07E0-07EF		D	D	D	D	D	D	D	D	Motion Object Horiz.
07F0-07FF				D	D	D	D	D	D	Motion Object Color
0800	R	D	D	D	D	D	D	D	D	Option Switch 1 (0 = On)
0801	R	D	D	D	D	D	D	D	D	Option Switch 2 (0 = On)
0C00	R	D				D	D	D	D	Horizontal Mini-Trak Ball™ Inputs
	R		D							VBLANK (1 = VBlank)
	R			D						Self-Test (0 = On)
	R				D					Cocktail Cabinet (1 = Cocktail)
0C01	R	D	D	D						R,C,L Coin Switches (0 = On)
	R				D					SLAM (0 = On)
	R					D				Player 2 Fire Switch (0 = On)
	R						D			Player 1 Fire Switch (0 = On)
	R							D		Player 2 Start Switch (0 = On)
	R								D	Player 1 Start Switch (0 = On)
0C02	R	D				D	D	D	D	Vertical Mini-Trak Ball™ Inputs
0C03	R	D	D	D	D					Player 1 Joystick (R, L, Down, Up)
	R					D	D	D	D	Player 2 Joystick (0 = On)
1000-100F	R/W	D	D	D	D	D	D	D	D	Custom Audio Chip
1404	W					D	D	D	D	Playfield Color RAM
140C	W					D	D	D	D	Motion Object Color RAM
1600	W	D	D	D	D	D	D	D	D	EA ROM Address & Data Latch
1680	W					D	D	D	D	EA ROM Control Latch
1700	R	D	D	D	D	D	D	D	D	EA ROM Read Data
1800	W									IRQ Acknowledge
1C00	W	D								Left Coin Counter (1 = On)
1C01	W	D								Center Coin Counter (1 = On)
1C02	W	D								Right Coin Counter (1 = On)
1C03	W	D								Player 1 Start LED (0 = On)
1C04	W	D								Player 2 Start LED (0 = On)
1C07	W	D								Trak Ball™ Flip Control (0 = Player 1)
2000	W									WATCHDOG
2400	W									Clear Mini-Trak Ball™ Counters
2000-3FFF	R									Program ROM

Exercise 2: Test the Centipede "fire" button using a the Fluke 9010A

1. Setup the Fluke properly (you should know how to do this correctly by now, but remember power the Fluke on FIRST before the Centipede PCB)
2. In the Fluke hit the "Setup" Button, choose "more" until "Active Line Force" is selected, choose "No"
3. Hit "Bus Test" to make sure all is working OK
4. Hit the "read" key when asked what address to read choose 0x0C01 and hit "loop"
5. View the output on the Fluke LED display, the 3rd bit from the right is the player 1 fire button
6. Hold down the player 1 fire button; watch the bit change from 1 to 0.

Special thanks to KLOV users TROXEL and BARITONOMARCHETTO for reading through this guide, doing the exercises, catching many many typos, and generally making this a better document.