

## Game Pad Interface

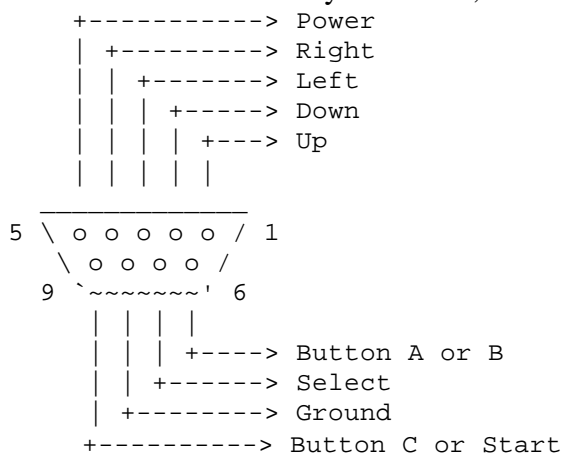
*Based on the Sega Genesis controller (circa 1989)*

The Sega Genesis controller originally was part of the first 16-bit game system available on the market (with great games like Altered Beast). For the final project, this controller is an ideal interface because it conforms to a common serial interface and it is extremely simple to decode. The interface is based on a 74HC157 quad 2:1 multiplexer and the game pad inputs. A select line determines functionality between different inputs. Since there are only nine pins connected to the controller and two of them take up power and ground, this only leaves seven pins to encode eight functions.

The signals of the interface are listed below:

- Up Signal = Pin 1 (with Select HIGH or LOW).
- Down Signal = Pin 2 (with Select HIGH or LOW).
- Left Signal = Pin 3 (with Select HIGH).
- Right Signal = Pin 4 (with Select HIGH).
- Button A Signal = Pin 6 (with Select LOW).
- Button B Signal = Pin 6 (with Select HIGH).
- Button C Signal = Pin 9 (with Select HIGH).
- Start button Signal = Pin 9 (with Select LOW).

To connect the controller to your circuit, the following diagram can be used:



Power and ground should be connected to the +5V supply and ground connections on the Xstend board.

The schematic gamepad.sch will allow you to create a module for the game pad. This module contains a SELECT input and outputs for each of the controller input functions. You can feed the SELECT input a signal that goes high and low at the appropriate times in order to use all of the buttons on the controller. If you do not need all of the button inputs, then either apply power or ground to this input. The schematic contains all of the

pins needed to interface to the controller. The FPGA pin assignments are listed below, and the pins are labeled as FPGA pin, Proto pin, and the number written on the Xstend board:

- RIGHT = 54/82/82
- LEFT = 56/83/83
- C\_START = 63/84/84
- A\_B = 64/3/1
- DOWN = 27/50/53
- UP = 28/51/54
- SELECT = 30/69/70

From this list and the connection diagram above, you should be able to correctly connect the game pad serial output to the XSA-100 board FPGA pins.

Points to Consider:

- Note that this is an asynchronous device because it is not tied to your system clock. The outputs of the module will go high when they are pressed, so whatever logic you create to read the outputs will synchronize them with the rest of the signals in the circuit
- These signals are **not debounced**. When a button is pressed, it may transition several times before settling to its correct value. For the game arrows (UP, DOWN, LEFT, RIGHT), this should not be a problem because these functions will only ever have one meaning in your design. The buttons however might take on more functionality, and it would be a problem if you accidentally had multiple button presses when you only expected one of them. There is a schematic called `debounce.sch` that debounces the signal by reading the first edge and then delaying the output signal long enough so that the switch has stabilized. You can use this if you are worried about switch bounce.
- The directional controls will allow you to press a diagonal direction which causes two of the signals (UP+LEFT, UP+RIGHT, DOWN+LEFT, DOWN+RIGHT) to be on at the same time. You can mitigate this problem easily by choosing an overriding direction if both are high. So you might always go LEFT, if LEFT and either UP or DOWN is high.