SPI Communications System

# Overview

The SPI Communications System is responsible for giving the Z80 access to two major components:

* An MPU that offers data relating to a mouse, keyboard, analog joysticks, and several unassigned I/O pins.
* An SDCARD *(Not yet implemented)* .

## Input Interface

There is an external PIC chip (MPU) which is tracking the current state of all keyboard/mouse/joystick input. It is constantly updating its’ own internal state based on what keys are pressed, where the mouse is, etc.

There is a PACKET\_NUMBER in the FPGA that can be set by the Z80. Once set, the Z80 will periodically monitor the actual packet data, waiting to see the same PACKET\_NUMBER appear within it. When it does, you know that the packet contains the latest data. Typical Z80 algorithm is as follows:

If( ioReg.PACKET\_NUMBER == ioRam.PACKET\_NUMBER )

{

// A new packet has arrived since last time I looked!

ProcessInputPacket();

ioReg.PACKET\_NUMBER++;

}

While the FPGA is pulling the new state, it is being exchanged through the SPI Interface as one big block of data, into one of the FPGA’s internal dual-ported RAMs. The Z80 has direct access to the other side of the dual-ported RAM, and is technically capable of reading it even when the pull is incomplete (i.e. invalid data exists). Therefore, it's highly advised to follow the above algorithm when reading input.

## SDCARD Interface

The SDCARD could be accessed through the SPI interface as well, *but this system is not yet designed*. A hardware modification would be necessary, as the SDCARD must now be read by the IO MPU, rather than the FPGA (since the IO MPU is now the SPI master).

# SPI Interface

The FPGA acts as a slave SPI device. It has an /SS signal from the IO MPU.