Lab 3

The task was to build an experimental setup for a two-node distributed embedded system, consisting of a master and a slave node. We had to connect two DE0 boards with two additional boards based on the CAN bus and the Serial Parallel Interface (SPI). CAN is a controller area network, bus allows many microcontrollers and different types of devices to communicate with each other in real time and also without a host computer. It is a completely message-based protocol.

Using Qsys, we first designed the processor for the master node of the embedded system. Then, we downloaded the design to the Cyclone III FPGA on the master DE0 board. We implemented the TTC Scheduler code, from lab 2, on the DE0 board to ensure the design was working correctly. After, the LED flashed continuously, we made the connections from the CAN-SPI module to the on-chip SPI Master using jumper wires to connect the two boards. Finally, orange switches on the SPI-CAN Module were arranged so that 2, 4, and 8 were in ON positions, while all the remaining switches were in OFF positions.

SPI, I2C, and UART are ideal for communication between microcontrollers and between microcontrollers and sensors where large amounts of high-speed data don’t need to be transferred.

In serial communication, the bits are sent one by one through a single wire.

A benefit of SPI is the fact that data can be transferred without interruption. Any number of bits can be sent or received in a continuous stream. With I2C and UART, data is sent in packets, limited to a specific number of bits. Start and stop conditions define the beginning and end of each packet, so the data is interrupted during transmission.

Devices communicating via SPI are in a master-slave relationship. The master is the controlling device (usually a microcontroller), while the slave (usually a sensor, display, or memory chip) takes instruction from the master. The simplest configuration of SPI is a single master, single slave system, but one master can control more than one slave.

MOSI (Master Output/Slave Input) – Line for the master to send data to the slave.

MISO (Master Input/Slave Output) – Line for the slave to send data to the master.

SCLK (Clock) – Line for the clock signal.

SS/CS (Slave Select/Chip Select) – Line for the master to select which slave to send data to.

SPI communication is always initiated by the master since the master configures and generates the clock signal, one bit of data is transferred in each clock cycle. SPI is a synchronous communication protocol, because devices share a clock signal. The master can choose which slave it wants to talk to by setting the slave’s CS/SS line to a low voltage level. In the idle, non-transmitting state, the slave select line is kept at a high voltage level.

References:

<https://www.techopedia.com/definition/32255/controller-area-network-can>

<http://www.circuitbasics.com/basics-of-the-spi-communication-protocol/>