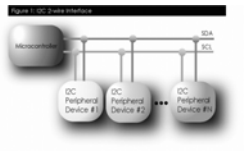
Inter-Integrated Circuit Bus

The I^2C Bus

Babak Kia
Adjunct Professor
Boston University
College of Engineering
Email: bkia -at- bu.edu

History of I^2C

- **The I2C bus was developed in the early 1980's by Philips Semiconductors**
- **It is an acronym for Inter-IC bus and is accepted by the industry as the de facto inter chip communication standard**
- **Its original purpose was to provide an easy way to connect a CPU to peripheral chips in a TV-set**
- **However today I2C is used in a slew of communication applications ranging from LCD communication to Factory Automation**



The I2C Bus Protocol

- **The I2C bus is a two wire interface (SDA for Serial Data and SCL for Serial Clock), along with a common ground connection**
- **Every device connected to the I2C bus has a unique address, and can act as either a receiver or a transmitter**
- **Unlike other serial interfaces, I2C is a multi master bus. The bus master is established at the time of commnication**

How it works – START and STOP

- Transmission is initiated when a microcontroller issues a START condition
- The START condition occurs when the chip pulls the SDA line low before pulling the SCL line low
- Conversely, the STOP condition is signaled when the microcontroller first releases the SCL line and then releases the SDA line
- Alternately, the microcontroller can take advantage of a process known as “repeated start”, where a message contains multiple START conditions
- However, a STOP condition always signals the termination of a message
- START acts as an attention signal, making all remaining ICs wait for trailing transmission

How it works – ADDRESS

- After a START condition has been signaled, the bus master will then send the ADDRESS of the device it wants to communicate with, along with the indication of a Read or Write operation
- This only occurs on the first byte of transmission, sending a 7-bit address followed by a 1-bit R/W# indication
- Bit 7 of the byte is the MSB of the address, and bit 0 of the address is the R/W# bit
- A number of addresses have been reserved for special operations, one of the more commonly used ones being used for Extended Addressing Mode

How it works – Acknowledge

- Any time a bus master transmits an address or data on the bus, it must be acknowledged by a slave through an ACK cycle
- This occurs if the address transmitted on the bus matches the unique address of the device, or consequently when data has been sent to a slave which has already been addressed
- For data transmissions, the slave device provides an Acknowledge cycle by pulling the SDA line low immediately after it has received the 8th bit of the transmission
- For address transmissions, the slave does so as soon as it has validated the address

How it works - Arbitration

- **The I2C is a multi-master bus, and therefore the arbitration must account for data corruption which occurs as a result of multiple masters simultaneously initiating transfers on the bus**
- **All masters on the bus monitor the START and STOP conditions and so corruption of data once a START condition has been signaled is really not an issue (it *is* an issue when one of the bus masters misses the START condition…)**
- **In order for the Bus Master to transmit a bit, it must pull one (or both) of the lines high. If on this event the line does not stay high, this indicates that another bus master is pulling it low and therefore is the master**
- **In this event it backs off until a STOP condition is detected**