Hardware & Software Co-design: Final Paper

Charlie Coleman

December 17, 2018

1 Serial Communication: I^{2}C

Inter-Integrated Circuit (I^{2}C) is a serial communication protocol created by NXP Semiconductors. As the name implies, the bus was created and is used to connect IC peripherals. The bus consists of master nodes, responsible for generating the clock signal & initiating conversations, and slave nodes, responsible for recieving the clock signal & responding to requests from the master nodes. I^{2}C can operate at multiple different speeds. They are Fast mode (400 Kbit/s), Fast-mode plus (1 Mbit/s), Hs-mode (3.4 Mbit/s), and Ultra Fast-mode (5 Mbit/s). All speeds supported by I^{2}C are backwards compatible, so slower slaves are able to connect to faster masters & vice versa [6].

I^{2}C supports the connection of many devices, with multiple masters/slaves. In order to specify communication to one device, a 7 bit address is send on the bus. These addresses are allocated by NXP, and must be purchased from them [3]. I^{2}C does not implement any CRCs or parity bits, so the only error checking is checking that `SDA` is constant for the duration of the positive clock period. Only one bit is generated for each clock cycle [6].

Plug-and-play functionality is not explicitely included within the I^{2}C specification, and therefore is not available by default in some devices. However, there are ways to implement plug and A master device can poll for addresses and if it recieves an ACK, that device has been added to the network. Master devices can also signal to other masters that they have been connected [3].

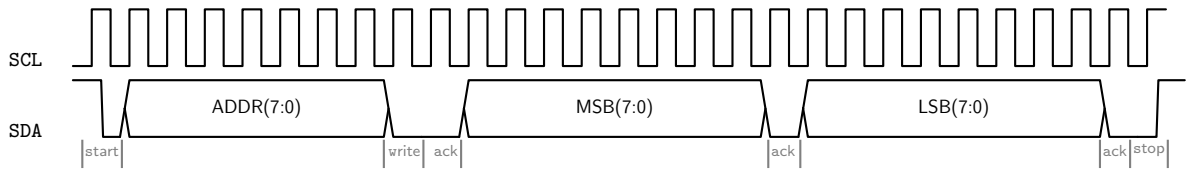

Figure 1: A write operation over an I^{2}C bus

2 Parallel Communication: PCI

The Peripheral Component Interconnect (PCI) Local Bus is a parallel communication bus developed originally in 1992. PCI aimed to allow high bandwidth between display peripherals within a PC. The PCI Bus supports one bus master with multiple devices. Each device shares a portion of memory with the master, which is used for control & status registers, the device ID + vendor ID, etc [5].

The PCI Local Bus supports a different transfer speeds depending on device clock speed and data width. A PCI device can operate at 33MHz or 66MHz, and can have a 32-bit or 64-bit data path. This allows for 3 different transfer speeds: 132 MB/s (32-bit @ 33MHz), 264 MB/s (32-bit @ 66MHz or 64-bit @ 33MHz), or 528 MHz (64-bit @ 66MHz) [2].

As stated above, each PCI target shares a small block of memory with the bus master. This block contains base address registers that help the bus master determine the required memory size of the PCI target. The target device will then be assigned an address range where the device will reside [2]. A PCI device can only perform DMA operations when it has requested and been granted bus master status. This allows for per-device optimization of DMA rather than relying on a 3rd party DMA controller [4].

Each PCI device has four interrupt lines, all asynchronous to the clock. A device can use any number of these lines [2]. PCI is also plug and play capable, as all of the configuration data is held on the card and is not needed to be set by jumpers. A few signals need to be asserted in order to allow the cards to be inserted and removed, but it does not require restarting the whole system [1].

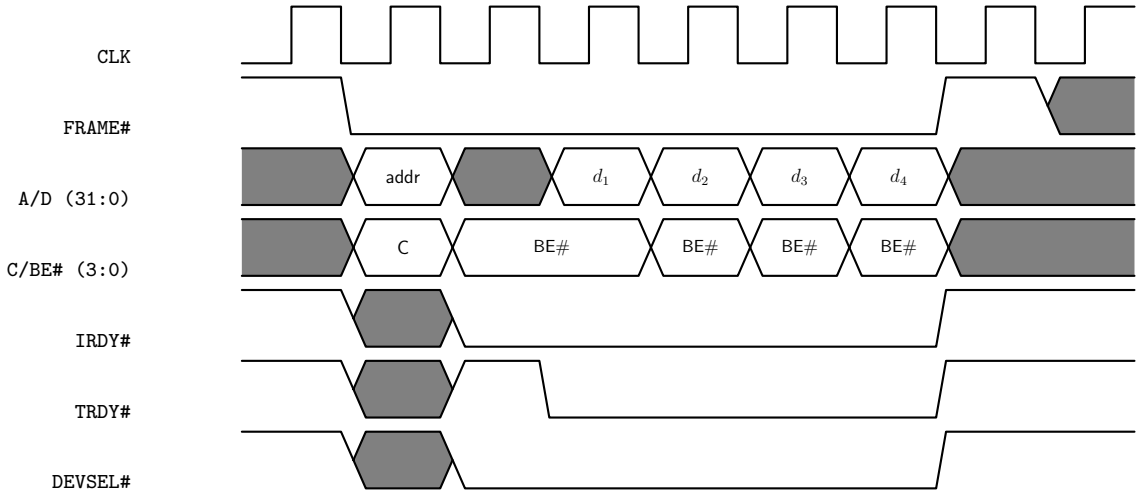

Figure 2: A 4-byte write operation over a PCI bus

References

[1] PCI Special Interest Group. *PCI Hot-Plug Specification*. 2001. URL: http://www.drydkim.com/MyDocuments/PCI%20Spec/specifications/pcihp1_1.pdf (visited on 12/17/2018).

[2] PCI Special Interest Group. *PCI Local Bus Specification*. 1998. URL: https://www.ics.uci.edu/~harris/ics216/pci/PCI_22.pdf (visited on 12/17/2018).

[3] Jean-Marc Irazabal and Steve Blozis. *I^2C Manual*. 2013. URL: https://www.nxp.com/docs/en/application-note/AN10216.pdf (visited on 12/16/2018).

[4] The Infinite Loop. *An overview of direct memory access*. 2014. URL: https://geidav.wordpress.com/2014/04/27/an-overview-of-direct-memory-access/ (visited on 12/17/2018).

[5] David A. Rusling. *Chapter 6: PCI - The Linux Documentation Project*. 1999. URL: http://tldp.org/LDP/tlk/dd/pci.html (visited on 12/17/2018).

[6] NXP Semiconductors. *I^2C-bus specification and user manual*. 2014. URL: https://www.nxp.com/docs/en/user-guide/UM10204.pdf (visited on 12/16/2018).