ECEN302 : Embedded Systems
Assignment 2 Submission

Daniel Eisen : 300447549

October 15, 2020

1. ***Explain how a Cache works and how it improves the performance of a microprocessor.***

   CPU cache is relatively small block(s) of memory (to other system memory) that sits very close to the processor. It stores copies of the data from frequently used main memory locations as well as instructions that are [expected] frequently used.

   Due to the cache being faster and with much lower latency than system memory the this dramatically speeds up general processing/access times of this 'hot' data.

   The cache as a whole is usually broken up into a cache hierarchy of L1, L2, ...LN levels.

2. ***What is the main purpose of a Memory Management Unit?***

   The MMU primarily performs that translation of virtual memory addresses to the physical address. It also serves to performs memory management, various levels of memory protection and access control as well as the cache control.

3. ***What does the acronym JTAG stand for? Explain the uses of a JTAG interface.***

   JTAG stands for *Joint Test Action Group* and is standard that defines a hardware interface and on device controller/interfacing hooks for debugging and low-overhead searial communication.

   It can be used for program debugging, stepping through code and reading register values. It can facilitate direct flash programming of a device, and perform peripheral connection boundary scans. If a system contains multiple JTAG compatible device these can be chained together and simultaneously controlled.

4. ***Explain at least 4 of the many things that need to be considered when developing an embedded system for a product.***

   - Power Usage/Consumption
   - Security
   - Memory Management/Allocation
   - Long Term Reliability/Ease of access to troubleshoot/repair

5. ***Explain 4 things you can do with a JTAG unit while debugging code.***

   The use of the JTAG port in debugging are:

   Traditional external connection testing of the chip. This can serve as a lower barrier to entry method of formfilling a similar role of a fully blown logic analyser in debugging the communions/connection between the device and its peripherals.

   Additionally, due to the JTAG's interface being able to directly read/write a devices registers it can also be used as a means of programming the flash, and enabling step by step debugging and breakpoints via the TAP controller/FSM.

6. ***ADCs and other devices often communicate with the microprocessor via a serial interface. What do the acronyms I2C and SPI stand for? Explain the differences between these two interface methods.***

   Both I^2C (Inter-Integrated Circuit) and SPI (Serial Peripheral Interface) are synchronous serial protocols.

   I^2C is half duplex lower speed transition that only requires 2 wires. It distinguishes devices via a unique device address.

   SPI is higher speed and is full-duplex, it distinguishes device via a dedicated 'slave select' line that the master device selects to interface with the peripheral. Due to this it is a 4 wire protocol that is simpler to implement in hardware/software.

7. ***The first time an optical shaft encoder was connected to a microprocessor the engineer used an external interrupt but with it configured as a level edge triggered interrupt. Explain why that was not a good idea.***

   Having the interrupt configured to level triggered means that it can be constantly triggering interrupt. Either if the encoder is stationary and a constant low level is output or just multiple triggers for the duration of a pulse. It should be set to edge triggered to ensure the ISR is not constantly/multiple triggered.

8. ***What does the acronym PCIe stand for? Give an overview of how this interface works and how it uses the advantages of both parallel and serial communications.***

   PCIe stands for *Peripheral Component Interconnect Express*. Unlike its precursors PCI and PCI-X, PCIe utilises multiple *lanes* of non-synchronised serial data streams. By enabling a device to split it total sum of data it needs to transfer into multiple lanes the transfer rate can be increases hugely when compared to a single path serial connection without the set bit-width restriction of a hard parallel bus and the need for bit synchronisation as data it tagged on each lane for realigning upon decode and issues with length restriction due to interference.

9. ***Discuss the differences between "bare metal" and a Linux operating system and give an example of when each is more appropriate.***

   Bare metal will offer better or at least matched performance with the same hardware than with Linux and is easy and fast for small applications when utilising a trusted environment/library framework. It also offer solid realisably due to code being more deterministic without the influence of a MMU or task scheduler etc. However as the size and function of a project grows, a purely bare metal with exponentially grow in complexity, all function including basic/standard must be adapted to be hardware specific.

   Bare metal is therefore suited to small, lower power/low compute requirement applications such as LED controllers etc.

   Embedded Linux offer a level of hardware independence, multi-threading/task scheduling, easier scaling and by utilising prebuilt drivers and libs the easy use of thing like networking, internet connectively etc. However, the kernel requires a minimum (more powerful) hardware and is and is more complex for small applications and the is the introduction of system updates and security protection.

   For larger projects that require future feature support/updaters an embedded Linux model is more beneficial such as a remote display billboard.