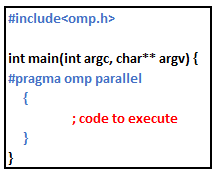
* **(46p) Foundation**

Introduction to Parallel Computing and the Raspberry Pi.pdf)

* + **(5p) Identifying the components on the raspberry PI B+**
    - The Raspberry Pi Model B+ consist of a 1.4 GHz Quad-Core 64-bit ARM Cortex-A53 processor with 512 KB of shared L2 cache. It also contains 1 GB of SDRAM, which is shared with the GPU. For input/output functionalities, it has four USB 2.0 ports, a 15-pin MIPI camera interface connector, one HDMI port for video/audio out, a 3.5 mm phone jack for audio input/output, a (300 Mbit/s) Gigabit Ethernet port, built in 2.4/5 GHz dual band Wi-Fi technology, Bluetooth 4.2, a Micro-USB port for power input (5 V), a MicroSD slot, and a 40-pin GPIO for various functionalities. **(1)**
  + **(5p) How many cores does the Raspberry Pi’s B+ CPU have**
    - The Raspberry Pi B+ contains four ARMv7 Processors (4th Revision). **(2)**
  + **(8p) List four main differences between X86 (CISC) and ARM Raspberry PI (RISC).**
  + **Justify you answer and use your own words (do not copy and paste)**

1. The Intel x86 supports a larger Instruction Set:
   * + - The Intel x86 is a CISC (Complex Instruction Set Computing) processor and its instruction set contains about 1000 different kinds of operation mnemonics that it can execute. **(3)** The Raspberry PI ARM processor is a RISC (Reduced Instruction Set Computing) processor and its instruction set contains less than 100 mnemonic instructions. **(4)** The CISC processors’ instructions are more complex and allow for more options in its instructions to manipulate data. CISC instruction set also contains a better support for memory access than its RISC counterpart.
2. CISC processors have less registers than RISC processors.
   * + - This is due to the fact that CISC processors’ have a more extensive instructions that support data management and does not need more registers to do this sort of tasks.**(4)** The ARM processor contains 16 integer registers and 32 floating point registers compared to the Intel x86 which only contains 8 General-Purpose registers, and 8 Special-Purpose registers. **(5)**
3. ARM processors provide conditional execution
   * + - The ARM processor also provides the capabilities for condition execution. This allows for an instruction to be executed or ignored based on the result of a status of a flag in the flag register. This option is not available in the Intel x86 processor. **(6)**
4. ARM processors provide Bi-Endian capabilities
   * + - ARM processors version 3 and above have BI-Endian capabilities. This allows for the processor to switch between little-endian and big-endian format when accessing information from memory. This option is controlled by either a hardware setting or software. **(7)** Intel x86 processors do not support this capabilities.
   * **(6p) what is the difference between sequential and parallel computation and identify the practical significance of each?**
     + Sequential or serial computing involves the process of breaking a problem into individual parts and executing them sequentially from start to finish. This process would be very similar to a factory assembly line where a complex problem can be executed one step at a time.
     + Parallel computation on the other hand, would also involve breaking the problem into individual parts. But unlike sequential computing, the parallel process would execute the instructions concurrently by coordinating and managing every aspect of the process. Parallel computation is preferred since it completes tasks in less time than Sequential computing.
   * **(5p) identify the basic form of data and task parallelism in computational problems.**
     + Data Parallelism is the equal distribution of work load between several processors. In this system, the problem is broken into tasks, then the processors execute the same tasks on different data until all tasks are completed. In Task Parallelism, the problem is broken into tasks but the tasks are distributed to the processors all having different task to perform. The data is then given to the processors; sometimes, data having to have through multiple processors to be fully processed. Task Parallelism’s major challenge is in the distribution of work load between the processors, since some tasks make take longer or heavier load of data, in execution time. Of the two parallelisms, Data Parallelism is more favorable.
   * **(6p) Explain the differences between processes and threads.**
     + A process is a set of tasks that executes the same set of instructions but each process runs independently of each other without shearing memory, while threads are part of a specific process that share the same memory between each other in that specific process.
   * **(3p) what is OpenMP and what is OpenMP pragmas?**
     + OpenMP is a standard that set the parameters for compilers to adhere to in order to take advantage of parallel processing. The OpenMP standard uses implements a Thread Pool pattern where the tasks are split into threads. These threads are then ran concurrently by each processor.
     + OpenMp pragmas is an implementation of the OpenMP standard which are directives to the compiler to implement the OpenMP libraries in order to create and manage the threads. The following is the pattern to implement an OpenMp pragmas in C Language:



* + **(4p) what applications benefit from multi-core (list four)?**
    - Database servers benefit greatly from multi-core technology. The need for the processor to execute the different request tasks makes it ideal for a multi-core system because the same.
    - Web servers also have a great need for extra processing power in order to accommodate the heavy amount of request from the network, and be able to process the individual requests.
    - Multimedia applications, especially with the increase of resolution in recent years, are applications ideal for multi-core processing. The great amount of processing work that takes for processing images, video, and sound will be better handled by a multi-core system.
    - A software antivirus program can also benefit from this type of processing since it constantly needs to perform different operations on different files.
  + **(4p) Why Multicore? (why not single core, list four)**
    - One of the limitations in making smaller single processors that execute more instructions per clock cycle is that it generates a lot of heat. Do to this fact, there is a physical limitation of number of instructions a single processor can execute in a given number of time. Multicore, tackles this problem by utilizing multiple processors to handle a set of tasks and getting more done in less time without the same limitations as the single processor.
    - With the development of new multi-threaded applications, it is necessary to provide the necessary hardware to accommodate such needs. It is inevitable the need for multi-core systems, since the limitations of single-core systems are eminent.
    - Multi-core also provide the hardware support needed for rendering today’s high definition videos and other multimedia. These type of software generate heavy workloads that would be handled better in a multi-core system.
    - Deep pipelined circuits provide a technique which divides a task into steps and executing the steps by different stages of the clock cycle in a single-core processor to provide a sort of parallelism. This technique produces high temperatures in the CPU. When temperatures exceed a certain threshold, there is the possibility to damage the CPU or for the CPU to malfunction. The limitations of this technique, provides the opportunity for multi-core processors.

**Citation:**

1. Package box
2. **Command Prompt: cat /proc/cpuinfo**
3. Heule, Stefan C. “Stefan c Heule.” *How Many x86-64 Instructions Are There Anyway?*, stefanheule.com/blog/how-many-x86-64-instructions-are-there-anyway/.
4. **“Arm Processor vs. Intel Processor” Handout**
5. “ARM Assembler in Raspberry Pi - RPi Labs.” *ARM Assembler in Raspberry Pi ·*, rpi.science.uoit.ca/lab/arm/#registers.
6. *ARM Information Center*, infocenter.arm.com/help/index.jsp?topic=%2Fcom.arm.doc.kui0100a%2Farmasm\_cegihjgh.htm.
7. “Endianness.” *Wikipedia*, Wikimedia Foundation, 30 Sept. 2018, en.wikipedia.org/wiki/Endianness.