PAMANTASAN NG LUNGSOD NG MAYNILA
(University of the City of Manila)
Intramuros, Manila

MICROPROCESSOR (LECTURE)

Activity No. 2
Microprocessor and its Architecture

Score

Submitted by:
Malabago, Antonio Emmanuel C.
S 1:00-7:00PM / CPE 0412-2

Date Submitted
25-11-2023

Submitted to:
Engr. Maria Rizette H. Sayo

Instruction:

A. Briefly describe the enumerated categories.

1. Microcontrollers are categorized according to the following:

a. Number of bits

Microcontrollers can be differentiated based on the size of their data width expressed by the number of bits.

- An 8-bit microcontroller can only handle 0 to 255 bits. It can perform basic arithmetic and logical operations such as addition, subtraction, multiplication, and division [1]. It is used in low-power devices such as remote controls, toys, and simple sensors. Some examples of 8-bit microcontrollers are Intel 8031/8051, PIC1x, and Motorola MC68HC11 family [2].
- On the other hand, a 16-bit microcontroller has a bigger data width of 0 to 65,535 bits. It is typically used in arithmetic and logical operations where great precision and performance are necessary [1]. These include industrial automation, telecommunication, and navigation systems. Examples of 16-bit microprocessors are extended 8051XA, PIC2x, Intel 8096, and Motorola MC68HC12 families [2].
- Meanwhile, a 32-bit microcontroller can handle large amounts of data from 0 to 4,294,967,296 bits. It can handle complex arithmetic and logic operations in autonomous and controlled devices [1]. It is usually found in medical implants, engine systems, office machines, home appliances, and other embedded devices. Some of its examples are Intel/ATMEL 251 and the PIC3x family [2].

b. Memory

Likewise, microcontrollers differ in the availability of functional blocks in the chip.

- Embedded memory microcontrollers have complete functional blocks for programs and data memory, counters and timers, interrupt flags, and I/O ports [2]. Examples of this are the Intel 8051 microcontroller.
- On the other hand, the external memory microcontroller has incomplete functional blocks within the chip. For instance, it has no built-in program memory [2]. Hence, an external memory can be interfaced into the chip, and this interfaced circuit is also known as a glue circuit [3]. Examples of this are the Intel 8031 microcontroller.

c. Instruction set

Moreover, microcontrollers can also be distinguished according to the type of instruction set.

- CISC, or Complex Instruction Set Computer, is an instruction set that enables a single instruction keyword to perform multiple simple tasks. Its primary purpose is to develop a program in the least possible lines of assembly code [4]. For example, multiplication in CISC operates directly on memory without requiring the user to call load and store functions explicitly. However, it consumes more resources and power than RISC.
- RISC, or Reduced Instruction Set Computer, is an instruction set that consists only of simple instructions to reduce the program's operational time. Its main goal is to shorten the clock cycle per instruction by allowing each instruction to utilize

any available register or addressing modes, with simultaneous access to program and data [4]. RISC enables microprocessor design compatible with industrial applications. The table below shows an example of the difference between CISC and RISC instructions.

CISC	RISC
MULT 2:3, 5:2	LOAD A, 2:3
	LOAD B, 5.2
	PROD A, B
	STORE 2:3, A

Table 1. CISC and RISC instruction

d. Applications

Microprocessors can be found in various devices and machinery in different fields, such as industrial, mechanical, medical, and automation.

- Home appliances such as refrigerators, electric fans, microwave ovens, and rice cookers have microprocessors embedded to control and monitor certain environmental conditions that are detrimental to their efficient operation and precise control of their functions.
- In gadgets and devices, microprocessors serve as a brain for controlling the operations of small, embedded devices such as microphones, cameras, and sensors for optimal device operation and efficient device performance.
- New models of automobile engines utilize hundreds to thousands of microcontrollers in managing, controlling, and automating fuel injection, ignition timing, emission control, ventilation systems, emergency brake systems, and others.
- Wearable devices use microcontrollers to gather and process data from smartwatches and fitness trackers by providing necessary computational resources and storage while ensuring the devices' low-power consumption, convenience, and portability.
- Security systems require microcontrollers for operating surveillance cameras, theft detection systems, and alarm sensor systems placed in doors and windows of the house to maintain maximum security against robbery and theft.
- Biomedical implants use microprocessors to power, monitor, and control medical devices such as pacemakers, insulin pumps, hearing aids, and other devices that measure physiological factors.

2. Problems arise on the electrical characteristics of a bus. Discuss how noise immunity differs from bus loading.

In the context of microprocessors, a bus serves as an electronic pathway at which signals can travel at different speeds and register locations. This enables a communication pipeline between the computer components where information is transmitted and received between registers [5]. Internally, it is mainly utilized in facilitating the data transmission between the CPU and memory or other peripherals such as video cards, hard drives, and output devices such as mouse, keyboard, monitor, webcam, and others.

However, due to complex design architecture, several factors can affect the electrical characteristics of a bus, such as noise immunity and bus loading. Noise immunity refers to the capability of logic circuit components to withstand, resist, and neglect any unwanted electronic signals from the external environment [6]. Regarding bus communication, higher noise immunity means a higher noise resistance from external disturbances. These disturbances may come from various sources, such as electromagnetic interference from other components, radio-noise interference from other devices, and unwanted crosstalk between adjacent semiconductors. On the other hand, bus loading refers to the corresponding effect of adding multiple devices within the bus, resulting in additional capacitance and resistance that can negatively affect its overall performance. The increase in capacitance leads to distortion of signals and data discrepancies as it slows down the rise and fall intervals of the signal. Meanwhile, the excessive resistance can cause unexpected voltage drops or delays, which results in the deprivation of signal quality.

3. How are bus buffering techniques being done?

   Bus buffering techniques are applied to improve the overall performance and efficiency of the bus in transmitting electric signals or information between the components. It is typically applied if there is a need to match the speed and data flow rate between components by leveraging buffers as temporary storage for preloading the data [7]. First, the buffers are utilized to separate and strengthen the signals traveling within the bus to negate the effects of noise immunity and bus loading. Afterward, buffering techniques, either signal-ended or differential signaling, are implemented. Their main difference is that differential signaling uses at least two bus lines for each signal as it passes through bus repeaters for successive regeneration of signals to overcome signal degradation within the bus. The buffers can also be placed at regular intervals, and termination techniques can also be employed to ensure no distortion during signal propagation. Moreover, the user can utilize high-speed buses to improve the microprocessor's overall performance and signal quality. Also, it is important to ensure that the bus lines receive enough power to ensure that they are functioning efficiently. In addition, error handling and FIFO buffers can be implemented to apply corrective signal techniques and ensure data flow integrity.

4. Why are flags register necessary in the operation of the microprocessor?

   A flag register is a type of special register with flag bits of 0 or 1, depending on the characteristics of the resulting arithmetic or logical operations. The returned flag bits can be separated into status and control flags. Status flags describe the resulting value after the operation, and control flags are utilized to turn some features within the microprocessor on and off depending on the operation output [8]. Some of the usual flags that can be received from this register are zero, carry, sign, overflow, and parity flags. Zero flags are activated if the resulting value from the operation is zero. Meanwhile, the carry flag is set when the operation results to carry out or borrow. On the other hand, the sign flag indicates the sign of the resulting output, usually determined by the most significant bit [9]. Then, the overflow flag is activated if the arithmetic produces a very large result that cannot be stored by the destination. Lastly, the parity flag is set when the number of bits in the result is even. These flag registers are relevant in conditional branching in microprocessors and perform actions based on the result of the previous instructions. It is also essential for arithmetic and logical operations, error detection and handling, comparison of data, and sharing of resources when performing multitasking or multithreading.

References

[1] "Microcontrollers - Overview." Available: https://www.tutorialspoint.com/microprocessor/microcontrollers_overview.htm

[2] T. Agarwal, "Microcontrollers types : Advantages, Disadvantages & their applications," *ElProCus - Electronic Projects for Engineering Students*, Jan. 20, 2021. Available: https://www.elprocus.com/microcontrollers-types-and-applications/

[3] Rashikagupta, "Classification Of 8051 Microcontroller - Quick Learn," *Quick Learn*, Jan. 28, 2023. Available: https://quick-learn.in/classification-of-8051-microcontroller/

[4] "RISC vs. CISC." Available: https://cs.stanford.edu/people/eroberts/courses/soco/projects/risc/risccisc/

[5] "Bus: What's the importance of Buses on Computers | Lenovo Philippines." Available: https://www.lenovo.com/ph/en/glossary/bus/?orgRef=https%253A%252F%252Fwww.google.com%252F

[6] P. K. Chatterjee *et al.*, "Integrated circuits," in *Elsevier eBooks*, 2002, pp. 20–113. doi: 10.1016/b978-075067291-7/50022-4. Available: https://doi.org/10.1016/b978-075067291-7/50022-4

[7] P. Williams, "What is buffering? A simple buffering definition," *Bandwidth Place*, Aug. 21, 2020. Available: https://www.bandwidthplace.com/what-is-buffering-a-simple-buffering-definition-article/

[8] "Flag register of 8086 microprocessor." Available: https://www.tutorialspoint.com/flag-register-of-8086-microprocessor

[9] GeeksforGeeks, "Flag register of 8086 microprocessor," *GeeksforGeeks*, May 05, 2023. Available: https://www.geeksforgeeks.org/flag-register-8086-microprocessor/