EME- 154 Mechatronics

Lecture Summary #6

4. Microprocessor Based Mechatronics Control System
4.1 Example of Fundamental Mechatronics System

4.2 General Architecture of the Mechatronics Control Hardware

What is the general structure of the hardware of Mechatronics Controller using single processor?

Control Logic Unit
A Control Logic Unit consists of a Microprocessor-based Logic Unit and Application Specific Interface circuits. The Control Logic Unit executes the program to operate the machine by responding the request or in-coming signals, and processing the information in the system. The information processing is mainly performed in the Microprocessor-based Logic Unit. Whenever physical signals need to be sent to and from the outside of the Control Logic Unit, the Application Specific Interface converts data to a physical signal (output) or a physical signal to data.

Memory System
A Memory System in the Microprocessor-based Logic Unit consists of two subsystems, system memory and user
A system memory mainly contains the system programs to execute the user defined control programs stored in the user memory. Various kinds of memory are used in memory systems:

- **DRAM (Dynamic Random Access Memory)** Readable andWritable in normal memory access speed, refresh operation needed to maintain the content.
- **SRAM (Static Random Access Memory)** Readable and Writable in normal memory access speed, no refresh operation needed to maintain the content.
- **Flash Memory**
- **EEPROM (Electrically Erasable Read Only Memory, also Flash Memory)** Readable in normal memory access speed. Writable by special write action arrangement. Erase can be made electrically, not by light.
- **ROM (Read Only Memory)**
- **PROM (Programmable Read Only Memory, Rare)** Readable in normal memory access speed. Writable by special write action arrangement. Erase can be made by exposing memory cell to ultraviolet light. no refresh operation needed to maintain the content.

**Signals handled in Application Specific Interface**

The input and output (I/O) signals to be handled by the Application Specific Interface are typically three kinds, digital I/O, pulse I/O and analog I/O. The digital I/O signals take only two-valued state (“zero” and “one” or “ON” and “OFF”). The state of the digital I/O signals are represented by the voltage level of the signals typically zero (state “0”) and five (state “1”) volts. Sometimes, they may take zero and 12 volts or zero and 24 volts. The transition time spent to change from one voltage level to the other is very short. The pulse signals mean the periodic digital signals which take repetitively “0” and “1” state. The time spent for maintaining “1” state is “ON-time” and the time for “0” state is “OFF-time.” One cycle period of the signal is ON-time plus OFF-time.

The frequency of the pulse signal is the inverse of the one cycle period measured in second. The duty of the pulse signal is represented by ON-time/(ON-time + OFF-time). The analog signals are the continuous variable voltage signals within a certain range of voltages. The analog input signal is converted into digital value by an Analog-to-Digital (A/D) converter in the Application Specific Interface. The analog output is made by converting digital value to analog value through a Digital-to-Analog (D/A) converter.

**Power Electronics Circuit**

A Power Electronics Circuits are provided to amplify output signals from Application Specific Interface to the level enough to drive the actuator because the signals from the Application Specific Interface are usually very low and weak (typically 5 volts and less than milliamperes). The amplification can be made by adjusting the constant power supply.

**Electric Actuator**

An Electric Actuator is an electrical-to-mechanical energy converter and usually attached to the mechanism to be controlled. The motion of the machine is expected to be “deterministic system” from control point of view but, unfortunately, the most of the real system cannot give the linear, time-independent, and deterministic characteristics.

**Signal Conditioning Circuit**

In order to monitor the mechanism behavior during the control, sensors are attached to some appropriate locations in the mechanism. The output range of a sensor varies from sensor to sensor. On the other hand, the acceptable level of the incoming signal at the Application Specific Interface is limited. Therefore, the signal conditioning circuits are provided to adjust the various levels of the incoming signals such that the incoming signals meet the acceptable range of the Application Specific Interface.
4.3 Typical configuration of mechatronics control hardware connected with typical sensors and Actuators.

5. Microprocessor System
5.1 Principle of programmable data processing in microprocessor by digital electronics

- What is a microprocessor?

A function box filled with programmable logic (digital) switches, which relates digital inputs and outputs with time clock (pulse)-based synchronization of digital signal transition.

![NPN Transistor Diagram]

If the Base is "ON" then Collector is connected to Emitter electrically.

If the Base is "OFF" then Collector is disconnected to Emitter electrically.
5.2 Basic Configuration of Microprocessor System

- What is the external view of a microprocessor system?

**Microprocessor systems:** it is advantageous to view the entire system – microprocessor, ROM, RWM, and I/O ports – as a collection of addressable registers.

**Registers** that reside within the microprocessor are internal registers, and that exist in the ROM, RWM, and I/O ports are external registers.

Program (Instruction Input), Data (Input, Output)

- What is an external bus system for a microprocessor?
  Address, data and control bus are referred to as the system bus of the microprocessor system bus. The bus is for sending the data, the data read and write signals, instructions to and from the external registers (memory and I/O).

5.3 Microprocessor Architecture

**System Architecture:** The collection of registers that constitutes a particular system and the data transfers that are possible among them make up the system architecture.

- What is the internal view of microprocessor?

**Microprocessor’s architecture:** The types of registers in the microprocessor and the possible data
transfers among them determine the microprocessor’s architecture.

• What is a difference between an internal bus system and an external bus system?
  An Internal bus is much faster than an external bus.

**Data transfer and transform**: A microprocessor system implements its functions by transferring and transforming data in registers of the system. Typically, transformations on data occur in internal registers, many of which are operational registers.

**Operational registers**: Operational registers differ from storage registers in that and their associated circuitry implement arithmetic or logic operations on the data contained in the register, thus transforming the data.

5.4 **General Architecture of microprocessor**

![Generalized Microprocessor Architecture](Diagram)

**ALU**: It executes the arithmetic or logic operations on one or two operands constitute the basic data transformations implemented in a microprocessor. The ALU is capable of performing the following operations on binary data:
1. Binary addition, subtraction, multiplication, division
2. Logical AND, OR, EX-OR
3. Complement
4. Rotate left or right

**Program Counter**: The program counter is an operational register that always holds the address of either the next instruction to be executed or the address of the next word of a multiword instruction that has not been completely fetched

**PSW**: The Program Status Word (PSW) contains status bits that reflect the current CPU state.
**Instruction Register:** It holds the op-code of the instruction code currently being fetched and executed.

**Control Unit:** It is the key sequential subsystem in the microprocessor itself and it controls and synchronizes all data transfers and transformations in the microprocessor system. The control unit uses inputs from a master clock to derive timing and control signals that regulate the transfers and transformations in the system associated with each instruction. The control unit also accepts, as input, control signals generated by other devices in the microprocessor system, which alter the state of the microprocessor.

**Stackpointer:** A stack pointer is for indicating the address of the stack location. A stack is a storage structure in which a microprocessor saves its register contents during subroutine calls and interrupts. The stack consists of a group of specifically allocated locations in external read write memory. The stack location is either the last location written into or the next location available to be written into. The stack pointer is designed in such a way that items are read from the stack in the reverse order from which they are written into it. This storage structure is referred to as a last-in, first-out, LIFO, stack. Access to data stored in the stack is, therefore, sequential, not random: Writing data into a stack is called a push operation, and reading data from a stack is called a pop operation. To retrieve a data item pushed onto the stack, all subsequent data items on the stack must be retrieved first.

**General Purpose Registers:** They are internal registers to be used to store temporally processed results in the microprocessor. The number of internal registers is limited.

**Temporary Registers:** The program counter is an operational register that always holds the address of either the next instruction to be executed or the address of the next word of a multiword instruction that has not been completely fetched.

### 5.5 Operation Principle of a Microprocessor

In order to operate a microprocessor, the sequential processing steps should be represented in the form of instructions codes.

Instruction codes, sometimes referred to as instructions are usually generated by compiling the software program such as programs written using C-language.

A single instruction consists of operation code (Opcode) and operand.

- **What are Operation code and Operand?**
  Opcode usually occupy one word length. It means if a microprocessor is 8-bit microprocessor, one word is 8 bit. All opcodes should be represented in 8-bit binary numbers. Therefore, opcode should be limited to 256 kinds.
  Operand is indicating the location from which the data to be processed is taken and the location to which the processed result is stored.

- **What is an instruction set?**
  Example of Instruction Sets of Microcontroller MCS-51 is shown in the lecture slide

- **What are typical instruction set?**
  - Data Transfer
  - Arithmetic Operation
  - Logical Operation
  - Program Branching
5.6 Mechanism of Microprocessor Operation

Sequential execution of a program (prearranged instructions) ---- instruction consist of operation code (command) and operand

• Operation of the instruction is cyclic ---- Instruction Fetch and Instruction Execution
  
  Fetch
  Moving one instruction code into the instruction register
  Decode the instruction code in control unit and find if the operand information is necessary for execution.
  If operand information is necessary, then operand information is read into the registers.

  Execution
  Then, the instruction is executed such as performing calculation, data transfer or program control.
  Then the next instruction code will be moved to instruction register.
  And repeat the same sequential processing.

Instruction Cache and Data Cache
The advantage of instruction cache and data cache memories have been explained in the lecture.

• What are the elements to enhance the performance of microprocessor?

  Clock frequency
  Bus width and distance
  Instruction decoding mechanism
  Data transfer mechanism
  Integration of external functional block into microprocessor chip
  Multiple register set
  Multi CPU Core
<table>
<thead>
<tr>
<th>$x$</th>
<th>$2^{**x}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>5</td>
<td>32</td>
</tr>
<tr>
<td>6</td>
<td>64</td>
</tr>
<tr>
<td>7</td>
<td>128</td>
</tr>
<tr>
<td>8</td>
<td>256</td>
</tr>
<tr>
<td>9</td>
<td>512</td>
</tr>
<tr>
<td>10</td>
<td>1,024 K</td>
</tr>
<tr>
<td>11</td>
<td>2,048 2K</td>
</tr>
<tr>
<td>12</td>
<td>4,096 4K</td>
</tr>
<tr>
<td>13</td>
<td>8,192 8K</td>
</tr>
<tr>
<td>14</td>
<td>16,384 16K</td>
</tr>
<tr>
<td>15</td>
<td>32,768 32K</td>
</tr>
<tr>
<td>16</td>
<td>65,536 64K</td>
</tr>
<tr>
<td>17</td>
<td>131,072 128K</td>
</tr>
<tr>
<td>18</td>
<td>262,144 256K</td>
</tr>
<tr>
<td>19</td>
<td>524,288 512K</td>
</tr>
<tr>
<td>20</td>
<td>1,048,576 1M</td>
</tr>
<tr>
<td>21</td>
<td>2,097,152 2M</td>
</tr>
<tr>
<td>22</td>
<td>4,194,304 4M</td>
</tr>
<tr>
<td>23</td>
<td>8,388,608 8M</td>
</tr>
<tr>
<td>24</td>
<td>16,777,216 16M</td>
</tr>
<tr>
<td>25</td>
<td>33,554,432 32M</td>
</tr>
<tr>
<td>26</td>
<td>67,108,864 64M</td>
</tr>
<tr>
<td>27</td>
<td>134,217,728 128M</td>
</tr>
<tr>
<td>28</td>
<td>268,435,456 256M</td>
</tr>
<tr>
<td>29</td>
<td>536,870,912 512M</td>
</tr>
<tr>
<td>30</td>
<td>1,073,741,824 1G</td>
</tr>
<tr>
<td>31</td>
<td>2,147,483,648 2G</td>
</tr>
<tr>
<td>32</td>
<td>4,294,967,296 4G</td>
</tr>
</tbody>
</table>