EE443 - Embedded Systems

Exercise - 1

Microprocessor Basics

1. What are the three common addressing modes? Describe how the data is accessed in each addressing mode.

2. What are the advantages of **relative addressing** method compared to **absolute addressing**?

3. What are the functions of the Arithmetic and Logic Unit (ALU)?

4. What is the purpose or usage of data received from memory when the following microprocessor blocks are the target?

a) Controller
b) PC
c) Registers
d) ALU
e) Address Buffer

5. What is the purpose or usage of address or data sent to memory for the following transfers from one of the microprocessor blocks to the memory?

a) From PC to memory address
b) From PC to memory data
c) From a register to memory data
d) From a register to memory address
e) From Address Buffer to memory address

6. What are the execution steps including the opcode fetch cycle for the machine instructions given below. Assume a typical 8-bit microprocessor where the memory data size is 8 bits and address size is 16 bits.

a) Add 8-bit immediate data to register-A.
b) Unconditional branch (Jump or GoTo) to a target address that follows the opcode.
c) Store register-A contents into a memory location using direct addressing.

For all data transfer steps indicate

- usage of the byte(s) being transferred (i.e. opcode, operand, low-address, or high-address),
- source and target of the transfer (i.e. Controller, PC, Register, ALU, Addr-Buffer, Memory-Addr, or Memory-Data)

7. How many **memory read** and/or **memory write operations** are required to execute the following instructions on a typical 8-bit microprocessor with 16-bit memory address?

a) `Jump  <addr>;  unconditional jump`
b) `Add  A, #<data>;  immediate load`

c) `Add  A, <addr>;  direct addressing`
d) `Load  A, <addr>;  direct addressing`
e) `JumpIZ  <addr>;  conditional jump`
f) `Load  A, @<reg-X);  register-indirect addressing`
g) `Store  A, @<reg-X);  register-indirect addressing`
h) `Store  A, <addr>;  direct addressing`
i) `Call  <addr>;  store PC and jump to <addr>`
j) `Return;  retrieve stored PC value`

For example, "`Load  A, #<data>`" takes two memory cycles, one cycle to read the opcode, and one cycle to read the immediate data.