Marks: 3 marks absolute Deadline : 17-4-2016

Microprocessor and computer architecture project instruction set includes

Datapath should contains 4 general purpose registers

**1 Byte Instruction**

|  |  |  |
| --- | --- | --- |
| **Opcode** | **Source** | **Destination** |
| XXXX | XX | XX |

**2 Byte Instruction (**where second byte is the address**)**

|  |  |  |
| --- | --- | --- |
| **Opcode** | **Source** | **Destination** |
| XXXX | XX | XX |
| **xxxx-xx-xx** | | |

🡨 🡨 Address

1. ADD (1 byte instruction)

**ADD:** It ADDS the contents of source and destination registers in the ALU and stores the result in destination register

1. Subtract (1 byte instruction)

**Subtract:** It SUBTRACTS the contents of the source and destination registers in the ALU stores the result in destination register

1. Multiply (1 byte instruction)

**Multiply:** It MULTIPLIES the contents of source and destination registers in the ALU and stores the result in destination register

1. And (1 byte instruction)

**And:** Forms the bit-wise AND of the contents of source and destination registers and store result in destination register

1. Not (1 byte instruction)

**Not:** Forms the bit-wise COMPLEMENT of the contents of source register and store result in destination register.

1. Unconditional Branch (2 byte instruction)

**Unconditional Branch:** Branches to a new address specified by the address of the second byte of the instruction. It means the second byte is load in the program counter. In this case the source and destination bits are don’t care

1. Conditional Branch based upon zero flag and overflow flag (2 byte instruction)

**Conditional Branch: (two instruction one is based on zero flag and other is based on overflow flag):** Branches to a new address **(if the Zero flag and overflow is asserted)** specified by the address of the second byte of the instruction. . It means the second byte is load in the program counter if condition is true.

1. Read direct from memory (2 byte instruction)

**Read Direct from Memory (RDM):** Load the contents of second byte from memory in the destination register specified in first byte.

1. Read indirect from memory (2 byte instruction)

**Read Indirect from Memory (RIDM):** Second byte contains the address of operand. Load the operand from the second byte in the desired destination register specified in the first byte.

1. Write indirect from memory (2 byte instruction)

**Write Indirect from Memory (WDM):** Writes the contents of source register to the address specified in the second byte.

1. Read direct from register (2 byte instruction)

**Read Direct from Register (RDR):** It reads the contents of second byte and stores it to the source register.

1. Read direct from memory (2 byte instruction)

**Read Indirect from Register (RDR):** The second byte contains the address of operand. It reads operand from memory and store it in destination register.

1. Add instruction for function call

**Function Call:** When the function is called, all the contents of the general purpose registers and the program counter will be saved in the memory. The PC will be loaded with the second byte of the address and the program executes accordingly.  
When the return occurs, all the contents are restored and popped out from the memory into their respective registers.

1. Add Interrupt cycle

**Interrupt Cycle:** Interrupt will be checked after every execution of the instruction and if the interrupt flag is set then you need to save the content of program counter and general purpose register.