PC:

Program counter points to the address of the next instruction to be executed.It is connected to the bus via a bi-directional buffer.Generally the PC value is incremented in each fetch cycle .In CALL,JNZ,JNC instructions PC value is loaded from the Bus. While executing an instruction related to ADDRESS or IMMIDIATE ,PC value is incremented for getting the address or immediate value. In our implementation

PC is 8 bit. We used a counter to load and increment the PC value . As it is a synchronous block,It has a clock.

Control Signals:

Epc: when high PC value is transferred to BUS

Ipc:when high PC is incremented

Lpc and J: these signals are for loading address to PC. The configuration is given below.

|  |  |  |
| --- | --- | --- |
| J | Lpc | Operation |
| 0 | 0 | Do not Load |
| 0 | 1 | JNC |
| 1 | 0 | JNZ |
| 1 | 1 | Unconditional Load |

MAR:

This 8 bit register holds the address of the memory to be read off or written to.It is connected to the BUS with a unidirectional line from BUS to MAR.It is connected to the RAM through unidirectional buffer from MAR to RAM .It is also a synchronous buffer.We implemented MAR with flip flops and buffer.

Control Signals:

LM\_BAR: when 0 address is loaded from BUS to MAR in the next positive clock edge.

E\_MAR: when 1 address is sent to RAM

RAM and BootLoader:

In out implementation we used a RAM of 8 bit address and 8 bit data .So the memory has 256 bytes of capacity .Data can be read from and written to RAM.It is connected to MAR through a unidirectional line .It is connected to MDR through a bidirectional buffer.We implemented memory with IC 6116.Though RAM is a asynchronous block a clock is needed for toggling the write enable pin of ram

BootLoader:

Before running the code in memory We have to load the memory with the code and data. This program and data is stored in file . We use a ROM chip for storing the file. When the simulation starts a first we load the file stored in ROM which generates the address by a counter into the RAM. The file is written in Hex format and the extension is .BIN. RAM has a output signal BOOT which lets controller know that the boot loading is done by setting the value to 0.A clock is needed for the counter for generating address.

Control Signals:

MEM\_R\_BAR: when 0 data is read from memory from the address specified by MAR and sent to MDR

MEM\_W\_BAR:when 0 data from MDR is written in memory in the address of MAR

BOOT\_INPUT: when 1 boot is not finished

CLR: when 1 Booting is started in that clock cycle . and 0 for other times

MDR:

It is a 8-bit register.It is connected to BUS and RAM through bidirectional buffer . It holds the data read from memory and to be written to memory .It is a synchronous block .

Control Signals:

EMB: when 1 data is transferred to BUS from MDR

EMR: when 1 data is transferred to RAM from MDR for writing into RAM

LMB\_BAR: when 0 data is loaded in MDR from BUS in the next positive clock edge.

LMR\_BAR: when 0 data is loaded from RAM from address of MAR in the next positive clock edge.