

# Specification Document

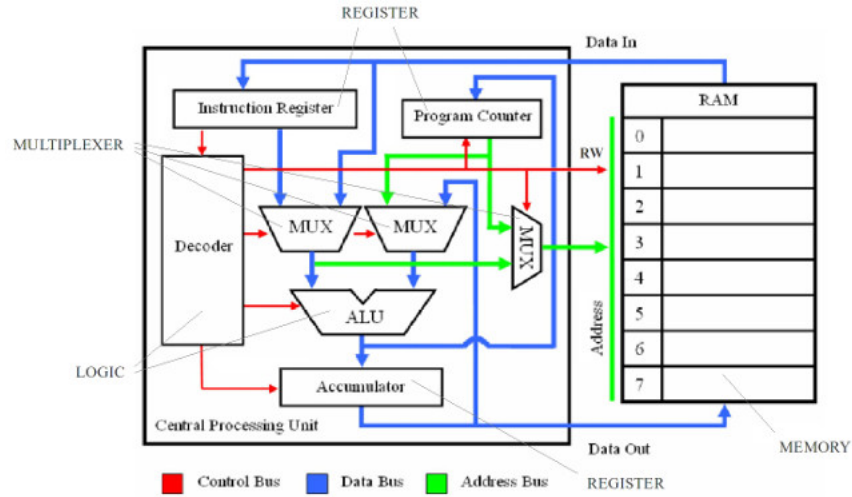
Omkar Girish Kamath

November 24, 2022

## Contents

<b>1</b>	<b>Processor</b>	<b>2</b>
1.1	Major Modules and Their Functions . . . . .	2
1.1.1	Program Counter . . . . .	2
1.1.2	Instruction Register . . . . .	2
1.1.3	Accumulator . . . . .	2
1.1.4	Decoder . . . . .	2
1.1.5	ALU . . . . .	3
1.2	Instruction Set . . . . .	3
1.3	Functioning of the processor . . . . .	4
1.4	Input Output Interface . . . . .	5
1.5	Timing Diagrams . . . . .	5
<b>2</b>	<b>Memory</b>	<b>6</b>
2.1	Description . . . . .	6
2.2	I/O of the Memory device . . . . .	6

Figure 1:



## 1 Processor

### 1.1 Major Modules and Their Functions

#### 1.1.1 Program Counter

Program Counter (PC) : 8 bit register used to store the address in memory of the current instruction being executed.

#### 1.1.2 Instruction Register

Instruction Register (IR) : 16 bit register, updated at the end of the fetch phase with the instruction to be processed (decoded and executed).

#### 1.1.3 Accumulator

8 bit register, a general purpose data register, providing data (operand) to be processed by the ALU and used to store any result produced. Note, we can only store one 8 bit value at a time on the processor, other data values will need to be buffered in external memory.

#### 1.1.4 Decoder

It decodes the 16 bit instruction received, tracks the state of the processor and generates control signals based on the first two.

### 1.1.5 ALU

It is the arithmetic and logic unit , which can take in 1 or 2 8bit inputs and perform arithmetic and logic based operations on the operand.

## 1.2 Instruction Set

In this instruction syntax X=Not used, K=Constant, A=Instruction Address, P=Data Address

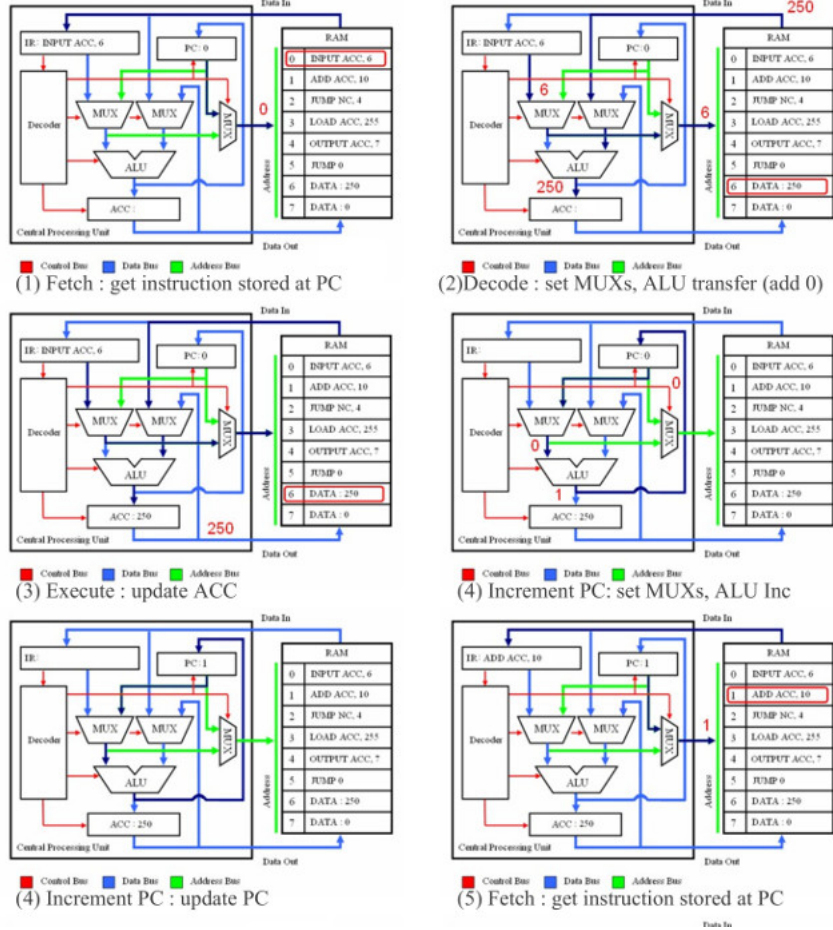
Table 1: Instruction Set of the Simple CPU

Opcode	Instruction	RTL
Load ACC kk	0000 XXXX KKKKKKKK	ACC <- KK
Add ACC kk	0100 XXXX KKKKKKKK	ACC <- ACC + KK
And ACC kk	0001 XXXX KKKKKKKK	ACC <- ACC & KK
Sub ACC kk	0110 XXXX KKKKKKKK	ACC <- ACC - KK
Input ACC pp	1010 XXXX PPPPPPPP	ACC <- M[PP]
Output ACC pp	1110 XXXX PPPPPPPP	M[PP] <- ACC
Jump U aa	1000 XXXX AAAAAAAAAA	PC <- AA
Jump Z aa	1001 00XX AAAAAAAAAA	IF Z=1 PC <- AA ELSE PC <- PC + 1
Jump C aa	1001 10XX AAAAAAAAAA	IF C=1 PC <- AA ELSE PC <- PC + 1
Jump NZ aa	1001 01XX AAAAAAAAAA	IF Z=0 PC <- AA ELSE PC <- PC + 1
Jump NC aa	1001 11XX AAAAAAAAAA	IF C=0 PC <- AA ELSE PC <- PC + 1

Here '>' indicates updated with .

The processor has an extra cycle to save on hardware which would have been required for incrementing the PC . So the processor follows **fetch-decode-execute-increment** cycle . The processor uses a 1 GHz frequency clock.

### 1.3 Functioning of the processor



**Step 1** : reset, clear line pulsed to reset all flip-flop, initialising all registers to their default values. First fetch phases initiated, ADDR MUX selects PC i.e. the value 0, memory address 0 read, first instruction (INPUT) stored in IR.

**Step 2** : decode, opcode field decoded, instruction identified as an INPUT instruction, control-logic configures data paths to route absolute address (6) stored in IR to address bus using ADDR MUX. Memory read, data at address 6 accessed (250), data driven onto data-in bus, routed through DATA MUX to input of ACC.

**Step 3** : execute, ACC updated with accessed data, at the end of this phase

ACC = 250.

**Step 4 :** increment, instruction completed, processor needs to increment PC to the address of the next instruction. Control-logic configures ALU to perform increment function, PC routed to ALU, value incremented and stored back into PC.

**Step 5 :** fetch, instruction at address 1 read (ADD) and stored in IR.

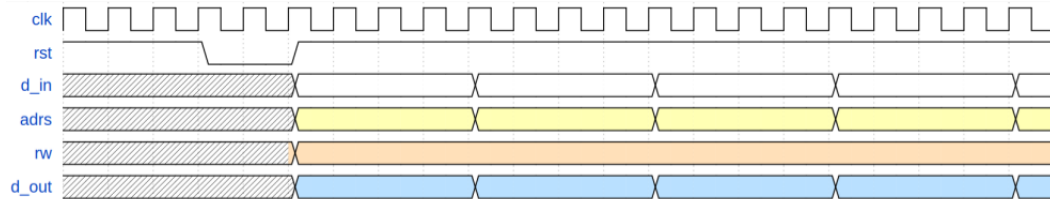
## 1.4 Input Output Interface

Table 2: I/O interface of the processor

Signals	Type	Size	Active	Description
clk	input	1 bit	-	square wave used to maintain synchronosity in the device
rst	input	1 bit	Low	resets the chip to a pre decided state
[15:0] d_in	input	16 bits	-	the instruction sent from memory
[7:0] adrs	output	8 bits	-	the address of the required instruction sent to memory
rw	output	1 bit	-	read write control signal sent to memory
[7:0] d_out	output	8 bits	-	output data from the processor

## 1.5 Timing Diagrams

Figure 2: Timing Diagrams of the processor



## 2 Memory

### 2.1 Description

**Size of RAM -> 4 Kilobytes**

RAM used is Volatile BJT type Synchronous Static RAM .

Instruction length is *16 bit* , maximum number of instructions and data than can be stored is **256** (address 0 to 255). Address length required is 8 bits.

### 2.2 I/O of the Memory device

Table 3: I/O of the Memory device

Signals	Type	Size	Active	Description
clk	input	1 bit	-	square wave used to maintain synchronosity in the device
rst	input	1 bit	Low	resets the memory to a pre decided state
[15:0] mem_in	input	16 bits	-	the data sent by chip
[7:0] adrs	input	8 bits	-	the address of the required instruction
rw	input	1 bit	-	read write control signal sent to memory
[15:0] mem_out	output	16 bits	-	output data/instruction from the memory