Group 111 CPU - User Manual

Table of Contents

Group 111 CPU - User Manual 1

Introduction 2

Features 2

ROM 2

Accumulator 2

Program Counter 2

Clock 2

Instructions - opcodes 2

LOAD [00] 2

ADD [01] 2

LSL [10] 2

JMP [11] 3

Configuration 3

Examples 4

Example 1: Adding 3 and 5 4

Example 2: Calculate 23 4

Example 3: Count in odd multiples of 2 5

Introduction

The designed CPU is a processor able to compute the output to a combination of 4 input commands and ROM values. The instructions are executed in 2 clock ticks, triggered from a button operated by the user.

Features

ROM

The ROM is represented by 4 DIP-switch banks, each with 6 switches. For each bank the first 2 switches represent op code, while the following 4 represent the data from least to most significant bit.

Accumulator

The accumulator is a 4-bit register that contains loaded data, and allows it to be acted upon by other opcodes. The accumulator value is presented with a 7-segment display, showing values from 0x0 [0b0] to 0xF [0b1111].

Program Counter

The program counter is 2-bit, and modified by either the manual clock, or when a jmp command is run. The current value in the program counter can be viewed from 2 connected LEDs, indicating the binary value stored.

Clock

As mentioned above, the clock is a push-button that can be used to step though the CPUs operation. The clocked is fitted with a de-bounce circuit, ensuring only 1 ‘tick’ is recorded with every push.

There is also a reset button that can be used between instruction sets to clear the program counter.

Instructions - opcodes

LOAD [00]

Load takes the value in the 4-bit ROM entries and puts them in the accumulator

ADD [01]

Add will take the value in the 4-bit ROM entries and sum them against the value already stored in the accumulator.

LSL [10]

LSL is Logically Shift Left. The binary bit values in the accumulator are all shifted left, with the most significant one being dropped off, and the new least significant one being assigned 0

JMP [11]

The program counter is set to the value in the 4-bit data portion of the ROM entry. This portion is stored in an address register for the jmp command to access and set the program counter to.

Configuration

The ROM is configured as follows:

Word 1						Word 2						Word 3						Word 4					
C	C	D_L	D	D	D_H	C	C	D_L	D	D	D_H	C	C	D_L	D	D	D_H	C	C	D_L	D	D	D_H

There are 4 words, and for each word the first 2 bits represent the op code, while the remain 4 are the data.

The opcodes are as follows:

Bit-Value	**Code**
00	LOAD
01	ADD
10	LSL
11	JMP

Sample values can be inputted and tested using our online simulator, found here:

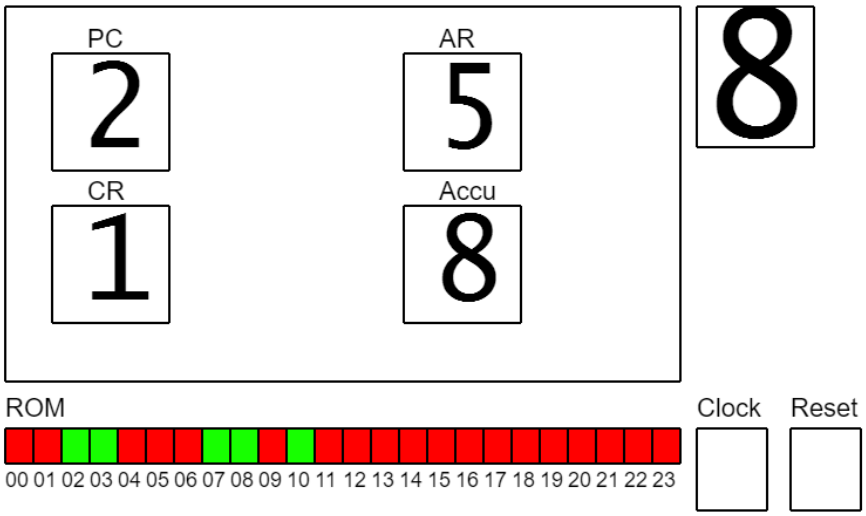
http://elec4403-group111.tk/

Examples

Example 1: Adding 3 and 5

ROM setting: 001100 011010 000000 000000

Output: 8 in accumulator after 2 clock cycles

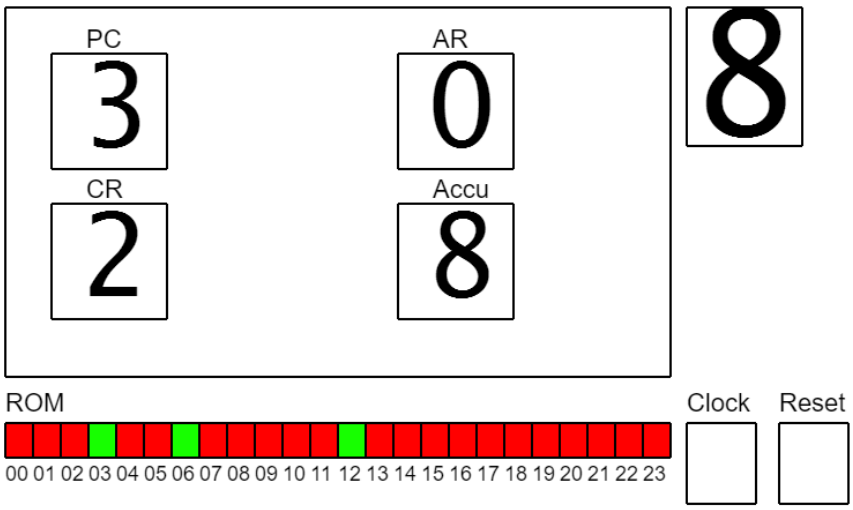



```
LOAD 3
ADD 5
LOAD 0
LOAD 0
```

Example 2: Calculate 2^3

ROM Setting: 000100 100000 100000 000000

Output: 8 after 3 clock cycles

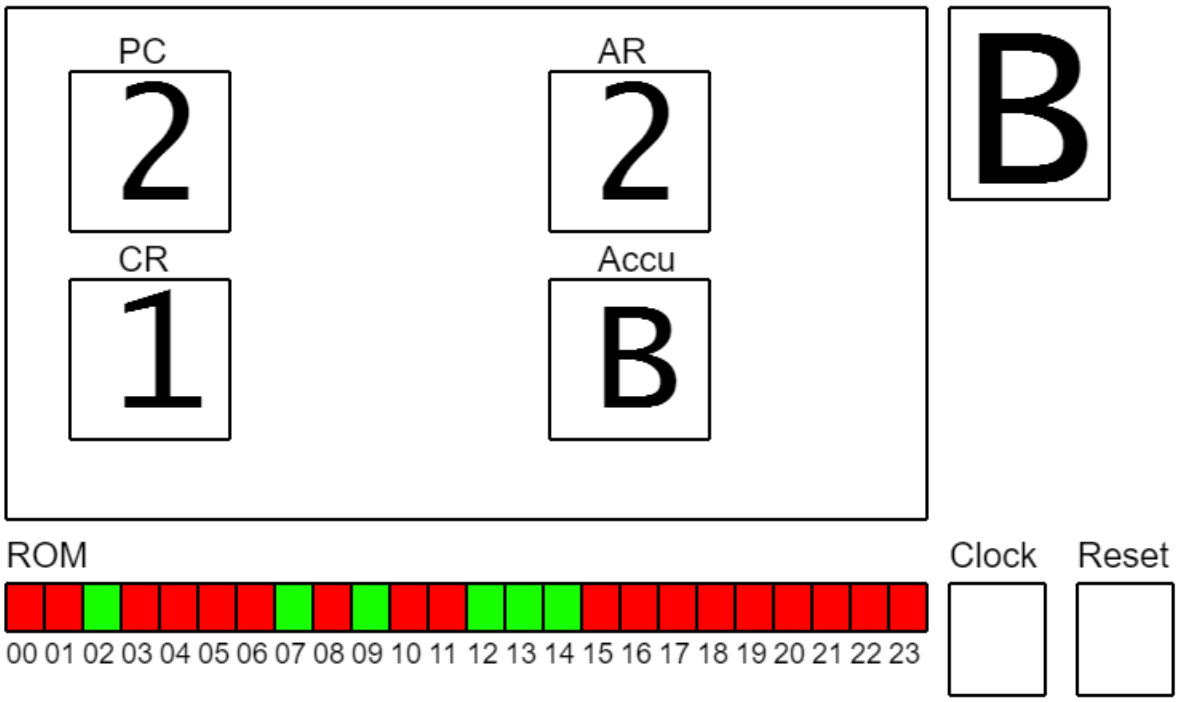



```
LOAD 2
LSL
LSL
LOAD 0
```

Example 3: Count in odd multiples of 2

ROM: 00100 010100 111000 000000

Output: 1; 3; 5; 7; 9; B; D; F; 1; etc.. with values updated every 2 clock cycles




```
LOAD 1
ADD 2
JMP 1
LOAD 0
```