CSE-316
Digital System Design Sessional

Group-18

Sec-B

Members

201814042 Md.Muhtasim Fuad
201814053 Saba Ahmed

Problem No: 05

Introduction

The Simple-As-Possible (SAP)-1 computer is a very basic model of a microprocessor explained by Albert Paul Malvino. The SAP-1 design contains the basic necessities for a functional Microprocessor. Its primary purpose is to develop a basic understanding of how a microprocessor works, interacts with memory and other parts of the system like input and output. The instruction set is very limited and is simple.

SAP (Simple-As-Possible)-1 is the first stage in the evolution toward modern computers. The type of computer is specially designed for the academic purpose and nothing has to do with the commercial use. The architecture is 8 bits and comprises of 16 X 8 memory i.e. 16 memory location with 8 bits in each location, therefore, need 4 address lines which either comes from the PC (Program Counter which may be called instruction pointer) during computer run phase or may come from the 4 address switches during the program phase. All instructions (5 only) get stored in this memory. It means SAP cannot store programs having more than 16 instructions.

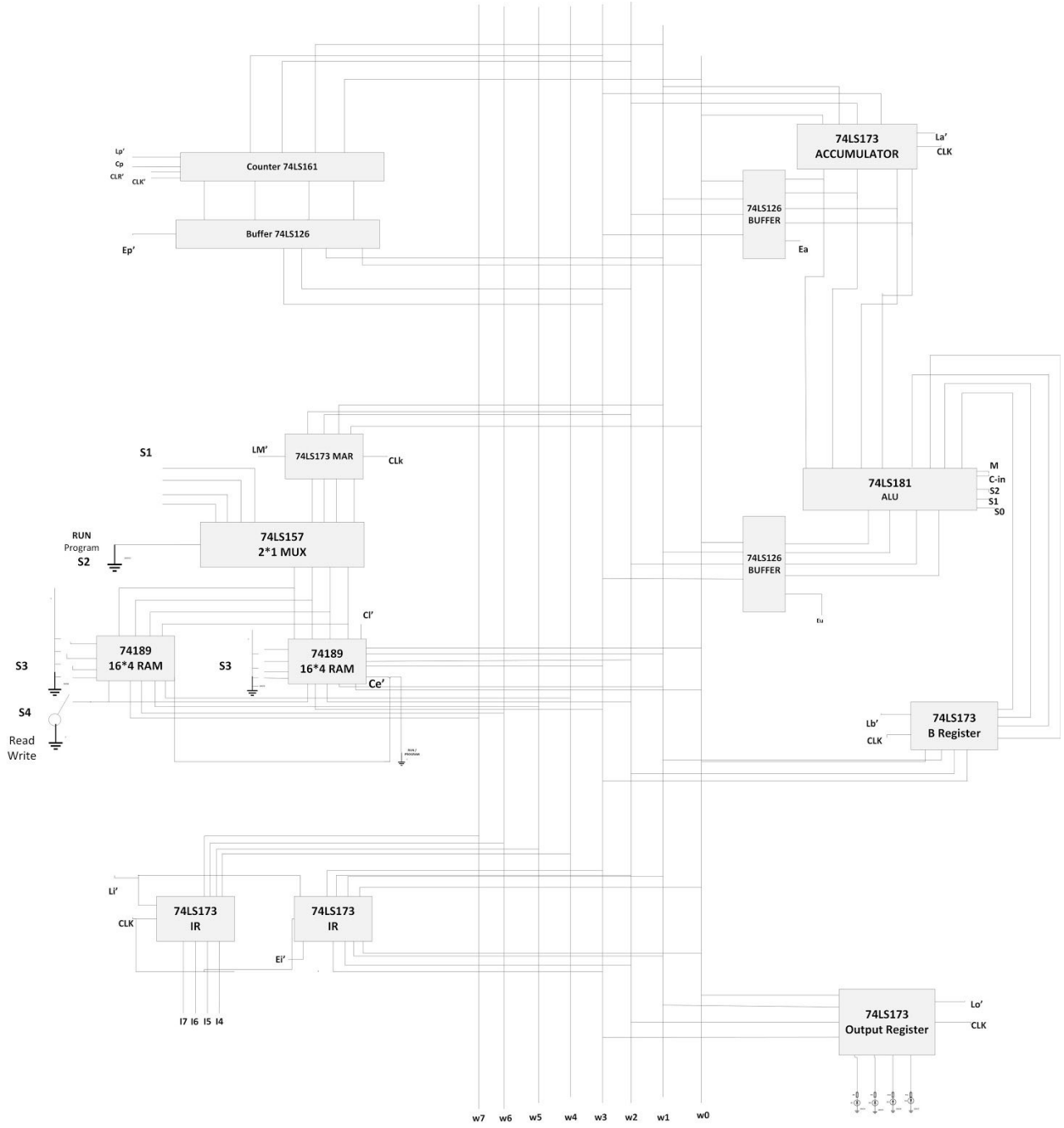
The problem of our project is to design a simple 4-Bit Microprocessor which have the following operations:

Sr.	Opcode	Instruction	Function
1	0000	LDA M	Load RAM data from memory address M to Accumulator
2	0001	LDB M	Load RAM data from memory address M to B Register
3	0011	ADD	Add data of B register with carry 1 with accumulator and store the result in the accumulator (A=A+B+1)
4	0110	INR A	Increment the content of Accumulator
5	0111	JMP M	Jump to memory address M
6	1001	OUT	Load accumulator data to output register
7	1010	HLT	Stop the program (No operation)

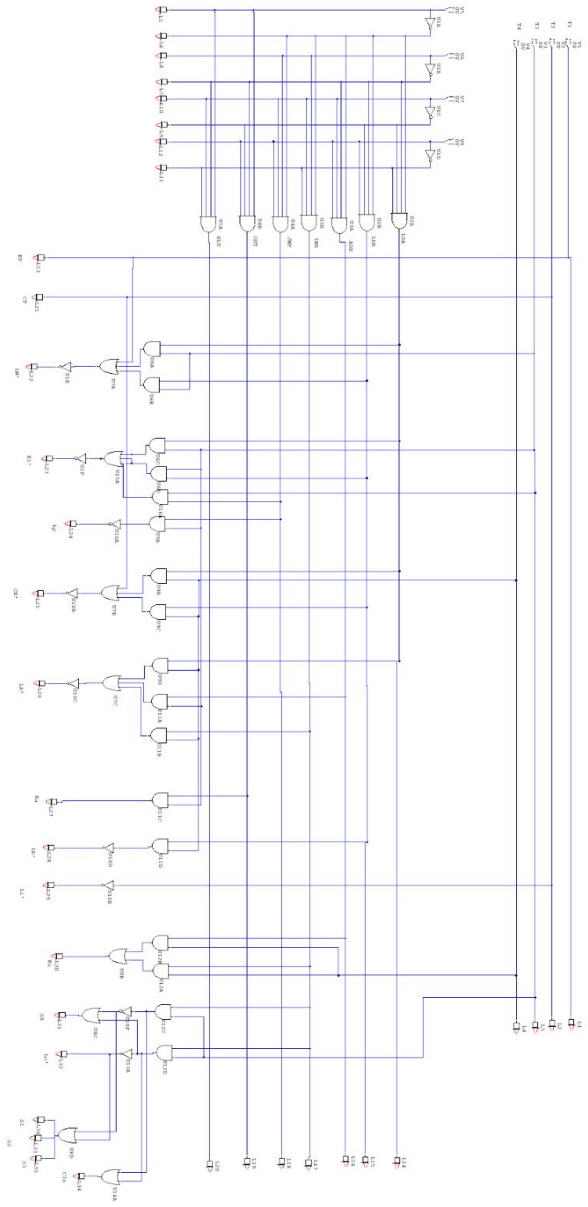
Used IC numbers

Components	Name	IC no.
Counter	Sync 4 bit counter	74LS161
	Buffer	74LS126
MAR & Input	D type register	74LS173
	2*1 MUX	74LS157
Instruction register	D type register	74LS173
RAM	RAM	74LS189
Accumulator (A register)	D type register	74LS173
	buffer	74LS126
B register	D type register	74LS173
	buffer	
ALU	ALU	74LS181
	Buffer	74LS126
Output Register	D type register	74LS173

Block diagram



Control Matrix



Description of used ICs

IC No.	Description
74LS181	74LS181 is a 4-Bit Arithmetic logic Unit which provides 16 arithmetic operations including addition, subtraction and also provides 16 logic operations like AND, OR, NAND etc.
74LS161	74LS161 is a high speed 4-Bit synchronous counter.They are edge-triggered, synchronously presettable, and cascadable MSI building blocks for counting, memory addressing, frequency division and other applications
74LS126	74LS126 is a Quad 3-STATE Buffer which contains four independent gates each of which performs a non-inverting buffer function.
74LS173	74LS173 is a 4-BIT D-TYPE REGISTER including D-type flip-flops featuring totem-pole 3-state outputs. Max voltage is 5.25V.
74LS157	74LS157 is a high speed Quad 2-Input Multiplexer. Four bits of data from two sources can be selected using the common Select and Enable inputs. The four buffered outputs present the selected data in the true (non-inverted) form
74LS189	74LS189 is a 64-BIT random access memory. It is organized as a 16- word by 4-bit array. Address inputs

	are buffered to minimize loading and are fully decoded on-chip. The outputs are 3-STATE and are in the high impedance state whenever the Chip Select (CS) input is HIGH

Description of Control Matrix (Eqn of each signals)

Ep	T1
Cp	T2
Lm’	T1 + I7’I6’I5’I4’T3 + I7’I6’I5’I4T3
Ei’	I7’I6’I5’I4’.T3 + I7’I6’I5’I4.T3 + I7’I6I5I4.T3
CE’	T2 + I7’I6’I5’I4’T4 + I7’I6’I5’I4T4
La’	I7’I6’I5’I4’T4 + I7’I6’I5I4T4 + I7’I6I5I4’T4
Lb’	I7’I6’I5’I4T4
Ea	I7I6’I5’I4.T3
Eu	I7’I6’I5I4.T3 + I7’I6I5I4’.T3
Lp’	I7’I6I5I4.T3
Lo’	I7I6’I5’I4.T3
Li’	T2
Cin	I7’I6’I5I4.T3 + I7’I6I5I4’.T3
S1, S2, S3	(I7’I6’I5I4.T3)’ + (I7’I6I5I4’.T3)’
S0	(I7’I6’I5I4.T3)’ + I7’I6I5I4’.T3

Table

Fetch Cycle

Operations	**T1**	**T2**
LDA M	Ep, Lm’	Cp, CE’, Li’
LDB M	Ep, Lm’	Cp, CE’, Li’
ADD	Ep, Lm’	Cp, CE’, Li’
INR A	Ep, Lm’	Cp, CE’, Li’
JMP M	Ep, Lm’	Cp, CE’, Li’
OUT	Ep, Lm’	Cp, CE’, Li’
HLT	Ep, Lm’	Cp, CE’, Li’

Execution Cycle

Operations	**T3**	**T4**
LDA M	Lm’, Ei’	La’, CE’
LDB M	Lm’, Ei’	CE’, Lb’
ADD	Cin, S0, S1, S2, S3, Eu, La’	NOP
INR A	Cin, S0, S1, S2, S3, Eu, La’	NOP
JMP M	Ei’, Lp’	NOP
OUT	Lo’, Ea	NOP
HLT	NOp	NOP

Discussion

1. In this project, a simple microprocessor is designed to perform some limited arithmetic operations.
2. The fetch cycle is completed in 2 stages instead of 3 stages and the execution step is done in 2 steps to reduce operation time and increase efficiency.
3. The external signals (S0, S1, S2, S3, Cin) are introduced to perform ADD and INR operation.
4. We found that the execution cycle of ADD and INR A can be done in only T3 stage as (Cin, S0, S1, S2, S3) don’t need any extra bus communication.