CO224 – Lab 5

Part 5 – Extended ISA

Group No: 02

Members: E/18/397 - Wijerathne E.S.G.

E/18/402 - Wimalasiri K.H.C.T.

- In this part, six instructions were added to our one clock cycle CPU. Instructions are,
  - mult (Multiplication)
  - sll (Logical shift left)
  - srl (Logical shift right)
  - sra (Arithmetic shift right)
  - ror (Rotate right)
  - bne (Branch not equal)

  Therefore, few things were added to the previous data path of the CPU. The new data path of the CPU is given below. New unit, which is Barrel Shifter, two control signals, 2 to 1 MUX, AND gate, NOT gate and OR gate were added.

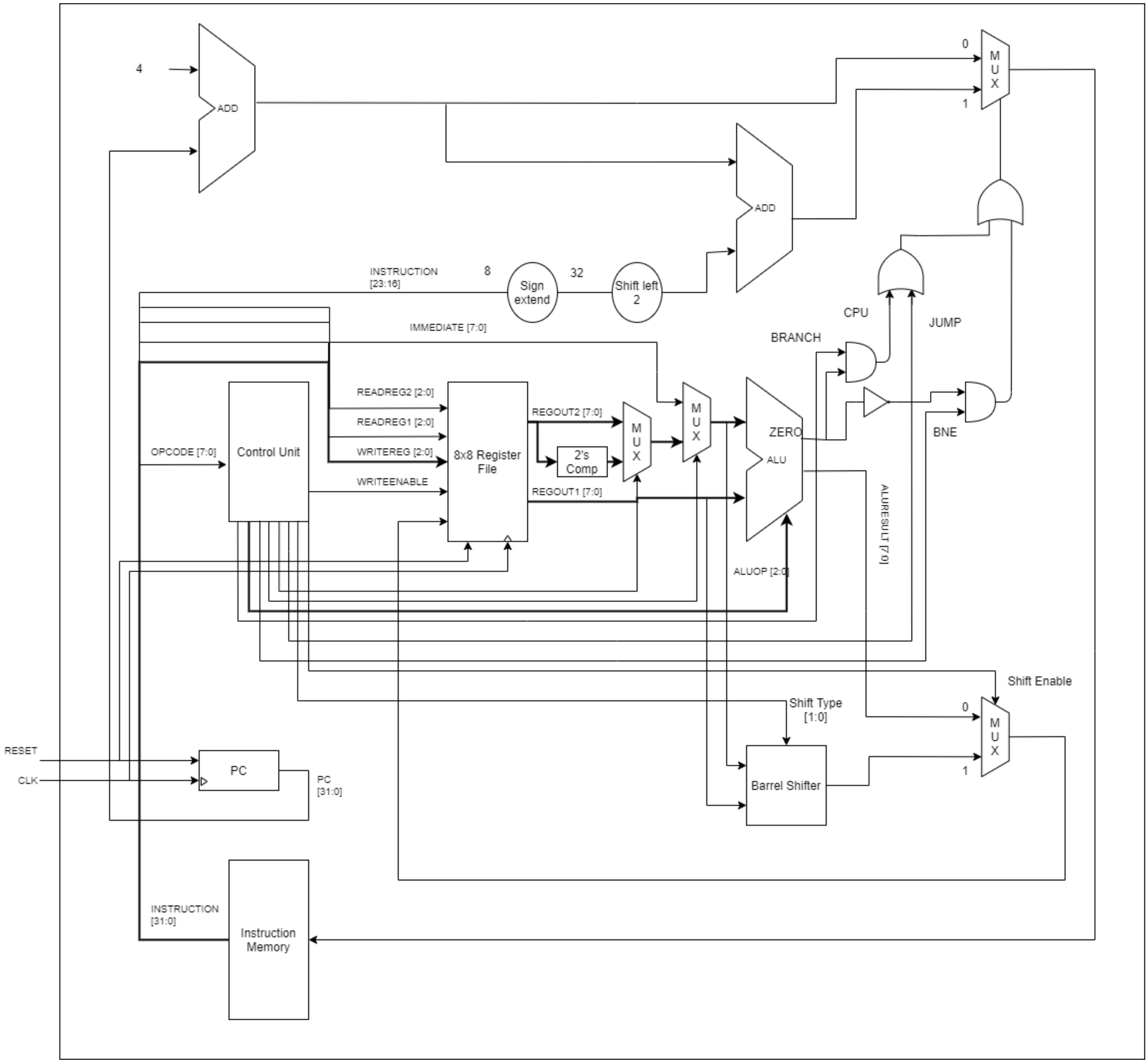


Figure 1: data path of the CPU

All the Opcode and their operations are given below,

Table 1

Operation	Code	Opcode (8-bits binary)
Load immediate	loadi	00000000
Move from one register to another	mov	00000001
Addition	add	00000010
Subtraction	sub	00000011
Bitwise AND	and	00000100
Bitwise OR	or	00000101
Jump	j	00000110
Brach equal	beq	00000111
Multiplication	mult	00001100
Logical shift left	sll	00001101
Logical shift right	srl	00001110
Arithmetic shift right	sra	00001111
Rotate right	ror	00010000
Branch not equal	bne	00010001

All the ALU Opcodes are given below,

Table 2

Operation	Code	Opcode (3-bits binary)
Forward two data	forward	000
Addition	add	001
Bitwise and	and	010
Bitwise or	or	011
Multiplication	mult	100

1. Barrel Shifter

- This unit can do logical shift left and right, arithmetic shift right, and rotate right. This unit has 3 inputs and 1 output. Three inputs are two values of corresponding register and Shift Opcode. The types of shifts and opcodes are given below.

Table 3

Operation	Code	Opcode (2-bits binary)
Logical shift right	srl	00
Arithmetic shift right	sra	01
Rotate right	ror	10
Logical shift left	sll	11

- Barrel shift unit is parallel to the ALU because CPU does only one operation at a time. We assume, this unit has 2 units delay. Barrel shifter was created using multiplexers which are 1-bit 2 to 1, 1-bit 4 to 1, 8-bits 2 to 1 and 8-bits 4 to 1. Therefore, we had to create another new 3 types of multiplexers because 8-bits 2 to 1 was created in previous part. New all the multiplexers and the barrel shifter were created using gate level modeling. The outputs of barrel shifter and ALU were joined to 2 to 1 multiplexer. Multiplexer gives output according to the shift signal which comes from control unit.

- The circuit of Logical left shift operation is given below,

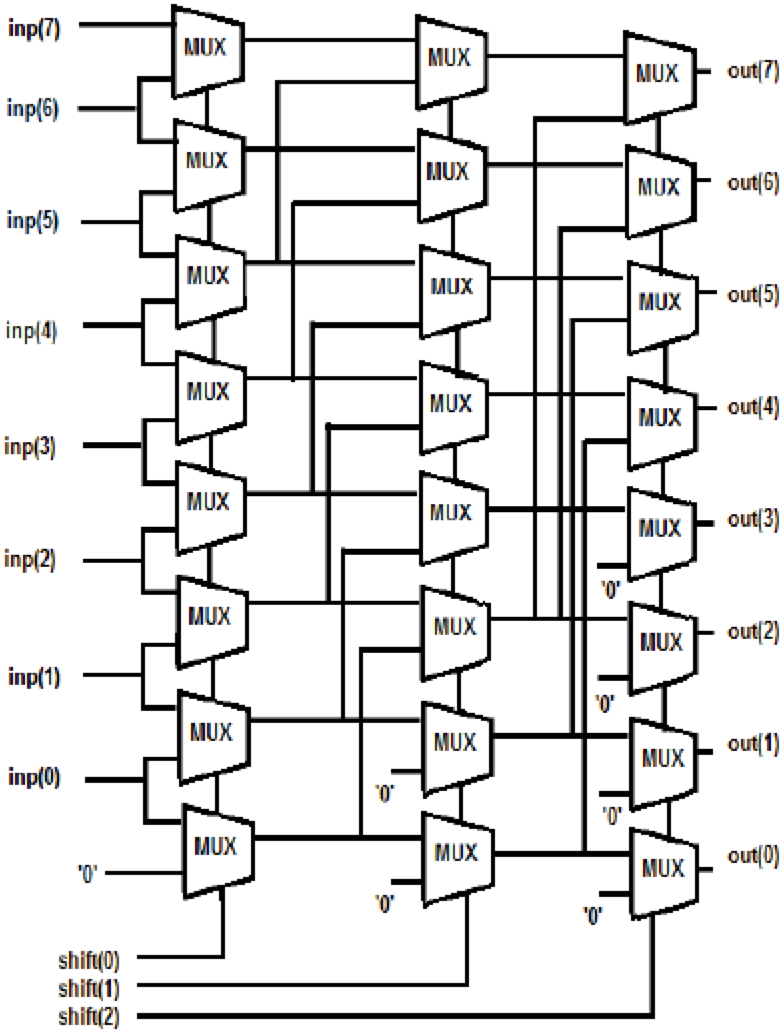


Figure 2: The circuit diagram of sll operation

- The circuit of all the other operation is given below,

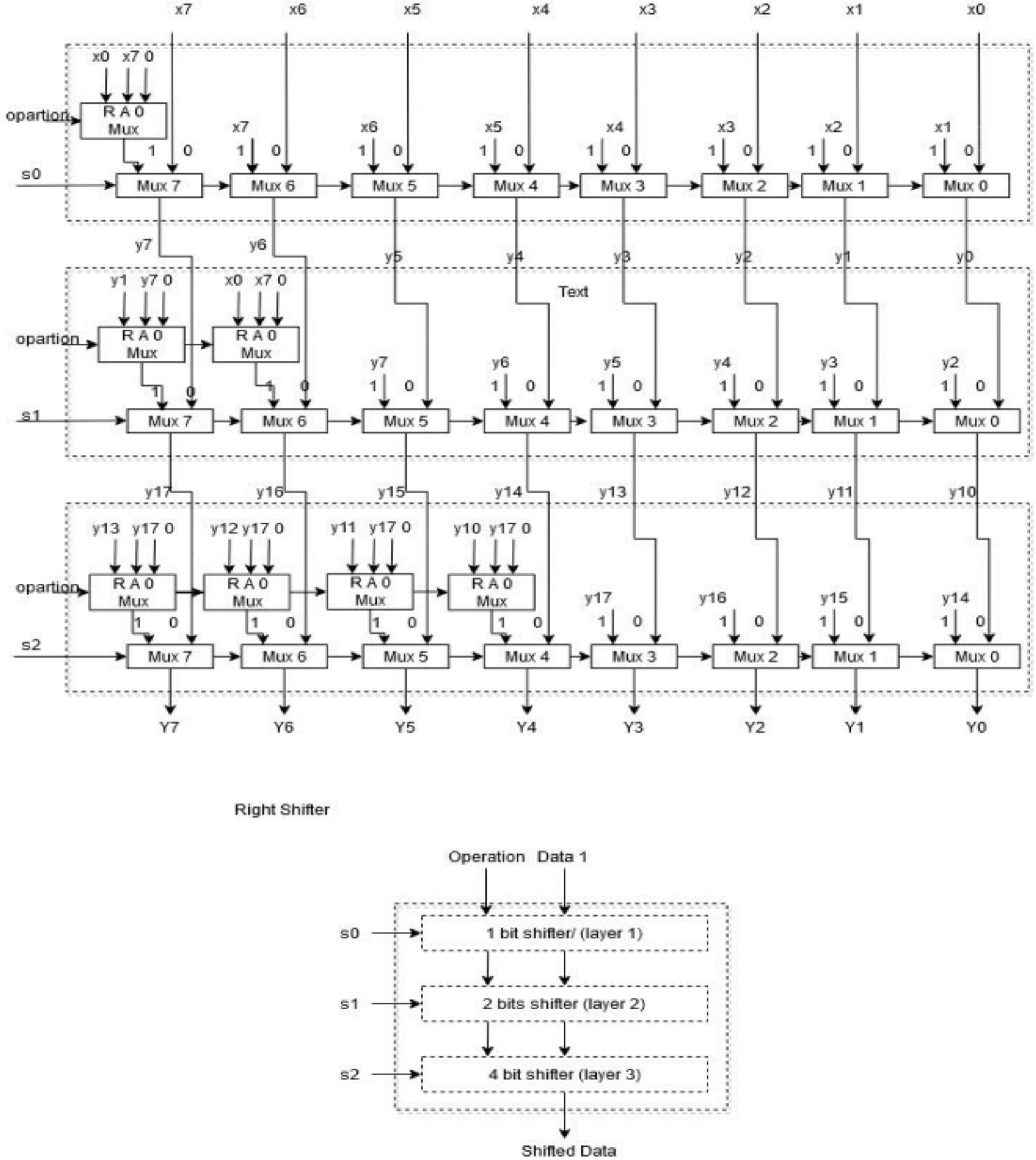
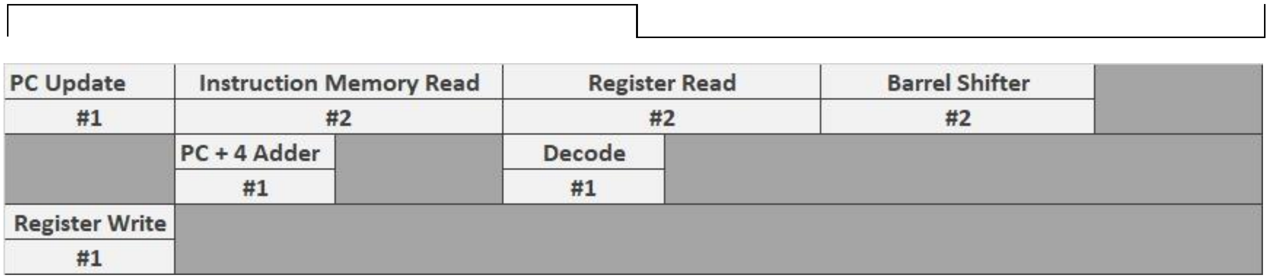


Figure 3: The circuit diagram of srl,sra and ror operations

- Therefore, the barrel shifter is the combination of those circuits.



PC Update #1 | Instruction Memory Read #2 | Register Read #2 | Barrel Shifter #2

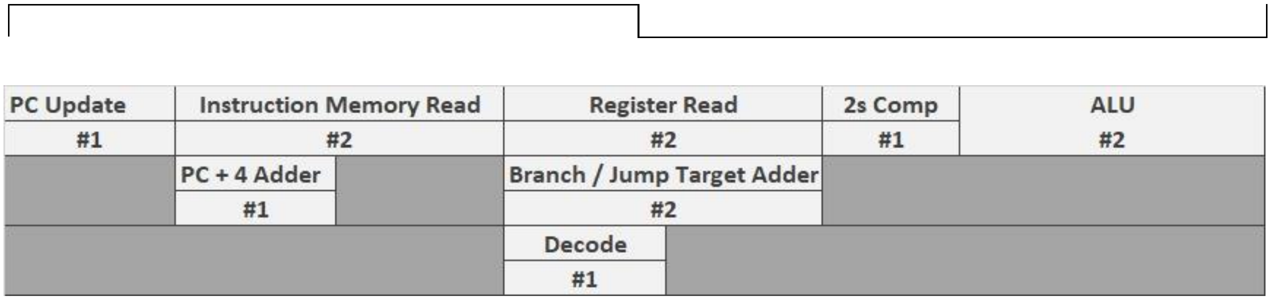
PC + 4 Adder #1 | Decode #1

Register Write #1

Figure 4: Timing diagram of sll, srl, sra and ror operations

2. BNE operation

- The data path of the BNE is given in the above figure. New BNE signal was created and joined to and gate. ALU zero signal was joined NOT gate and output is joined to the that AND gate. The output of the AND gate was joined to the OR gate and the output of the beq, jump signals were joined to the that OR gate. All the data paths are shown above figure.



PC Update	Instruction Memory Read	Register Read	2s Comp	ALU
#1	#2	#2	#1	#2
	PC + 4 Adder	Branch / Jump Target Adder		
	#1	#2		
		Decode		
		#1		

Figure 5: Timing diagram of bne operation

3. Multiplication

- Multiplication uno was implemented as separate module along with ALU. Since we are instructed to develop multiplication for signed numbers, we developed a special control unit for multiplier which help us to deal with multiply signed numbers. Our multiplier module was developed by using basic logic gates, Full adder and half adders which were also modeled by gate level.

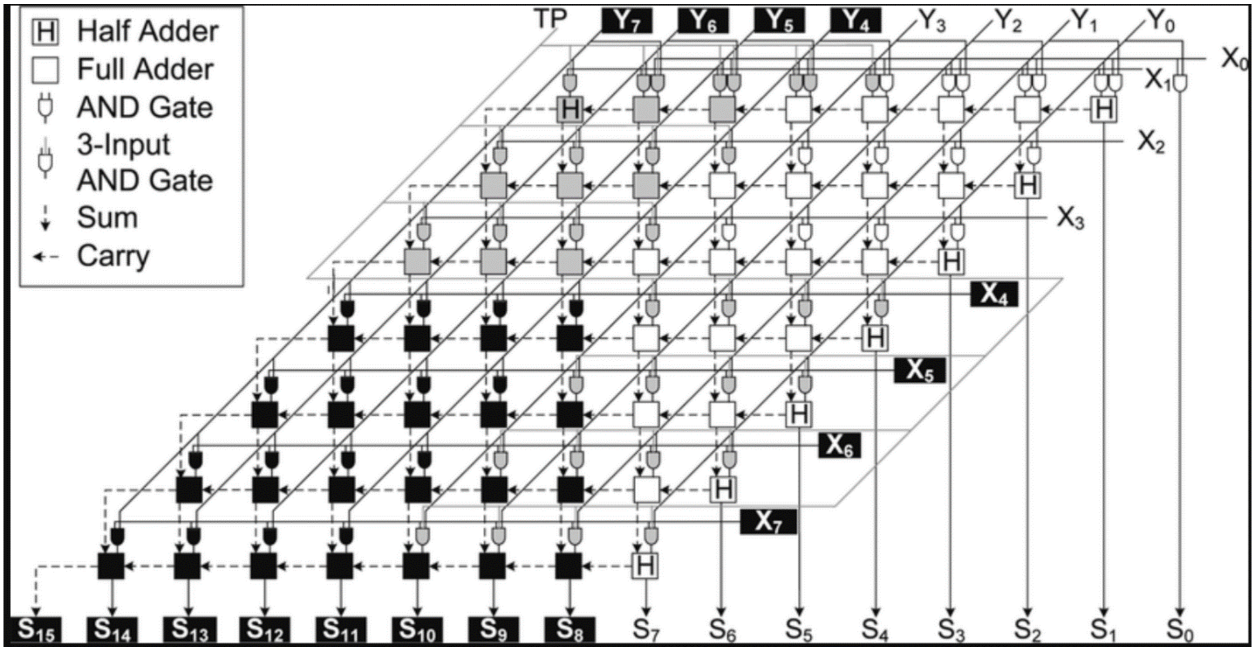
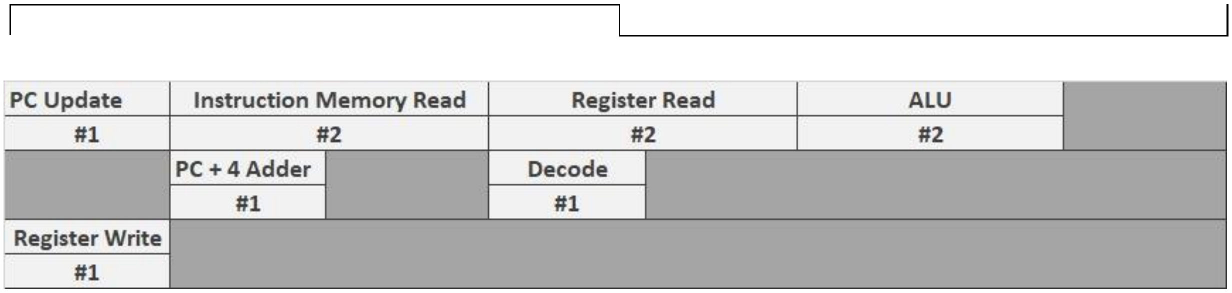


Figure 6: The circuit diagram of the mult operation

Since we are developing 8-bit processor we have small drawback. Since we deal with signed numbers, we must preserve single bit for the sign. After that we have 7 bits to display our values. Therefore, we can only get correct result for the multiplier result in range -128 to 127. If the result not in this range it will produce some incorrect value.

It is very important to add some fair delay to whole process since the process is Slightly larger and involved with much hardware components. For the sake of understanding we added a delay of two-time units to whole process Such that total instruction processing time was within a single clock cycle.



PC Update	Instruction Memory Read	Register Read	ALU	
#1	#2	#2	#2	
	PC + 4 Adder	Decode		
	#1	#1		
Register Write				
#1				

Figure 7: Timing diagram of mult operations