GROUP 7

VLSI Project

Phase 1 Documentation

Group Members:

1. AbdelRahman Ahmed Amaar.
2. AbdelRahman Fawzy Mohamed.
3. AbdelRahman Nasr El-Shenawy.
4. Ayman Mohamed Abo El-Maaty.

24-Apr-12

Table of Contents

List of Tables

List of Figures

Abstract

This document describes the plan of phase 1 according to the requirements illustrated in the Project document, and what is done to complete these requirements.

Project Description

The requirement of the project is to build a 32 bit Processor with 3 bus architecture, having 1 ALU, and 1 Memory for data & instructions with 8 bits width, and working with 32 GRP each of 32 bit width as shown in the following figure.

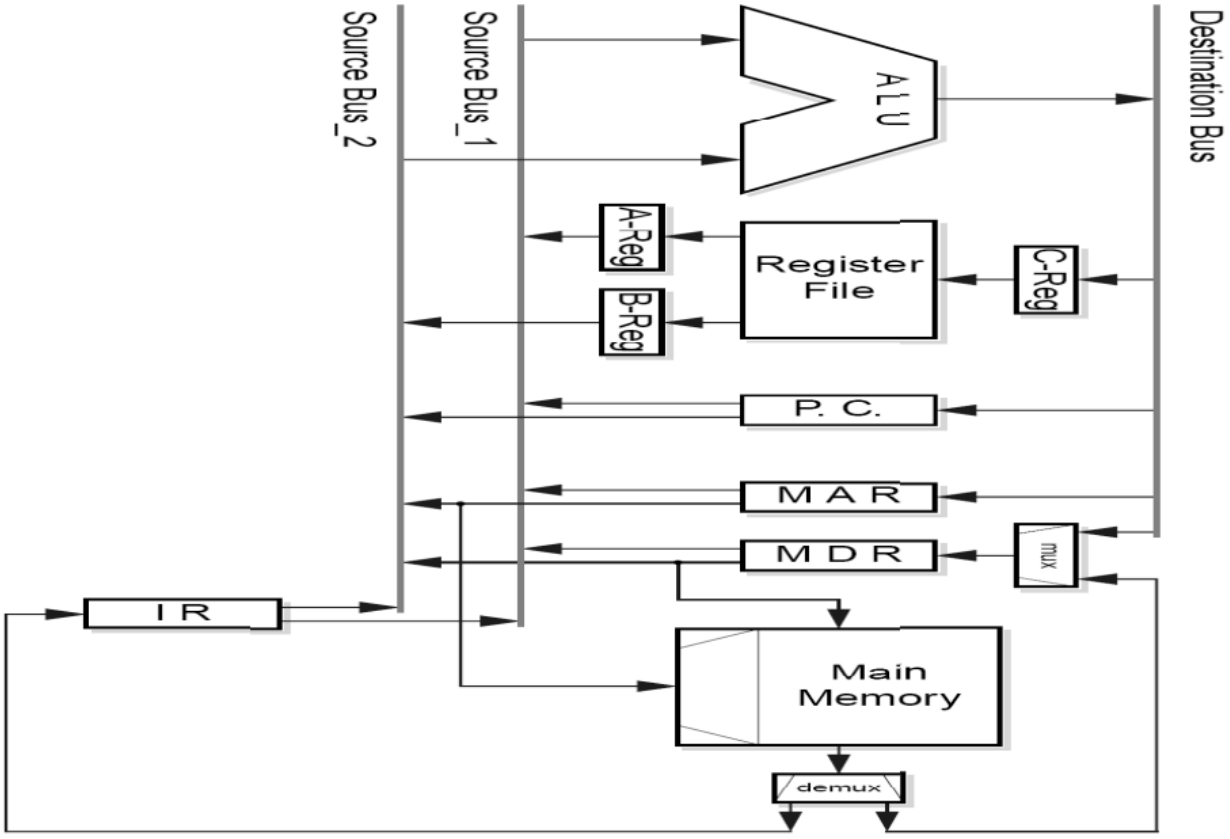

Fig 1 Processor Architecture

Project Plan

We divided the Project into small pieces as follows.

1. ALU Component.
2. RegFile Component.
3. Memory Component.
4. Controller Component.
5. Buses Connections.

- **ALU Component**

This component contains the arithmetic & logic operations required to execute the Instructions reviewed in the project description, the operations are concurrence as no need for the system clock in execution, and the operations done are as follows.

Type	Operation	Group (S3 & S2)	Selectors (S1 & S0)
Buffer	BUF 'A'	00	00
Arith	SUB		01
	ADD		10
	ADD4		11
Logic	AND	01	00
	XOR		01
	OR		10
	BUF 'B'		11
Set	SEQ	10	00
	SNE		01
	SLE		10
	SLT		11
Shift	SRA	11	00
	SRL		01
	SLL		10
	LHI		11

Table 1 ALU Operation

Each Operation requires 2 Operands 'A' & 'B' each on have 32 bit wide while the operation is determined by 4 bit Selector 'S' that is settled by the controller according to the instruction, and outs the 32 bit Output 'F', and Zero Flag 'Z' according to "set" component output.

The ALU consists of 4 Components that are divided according to the Group of Operations, i.e.: Shift component, Set component, Logic Component, and Arith Component.

Each one of these sub-components contains a RTL code that executes the operation according to S (1) and S (2) from the Selectors as shown in figure.

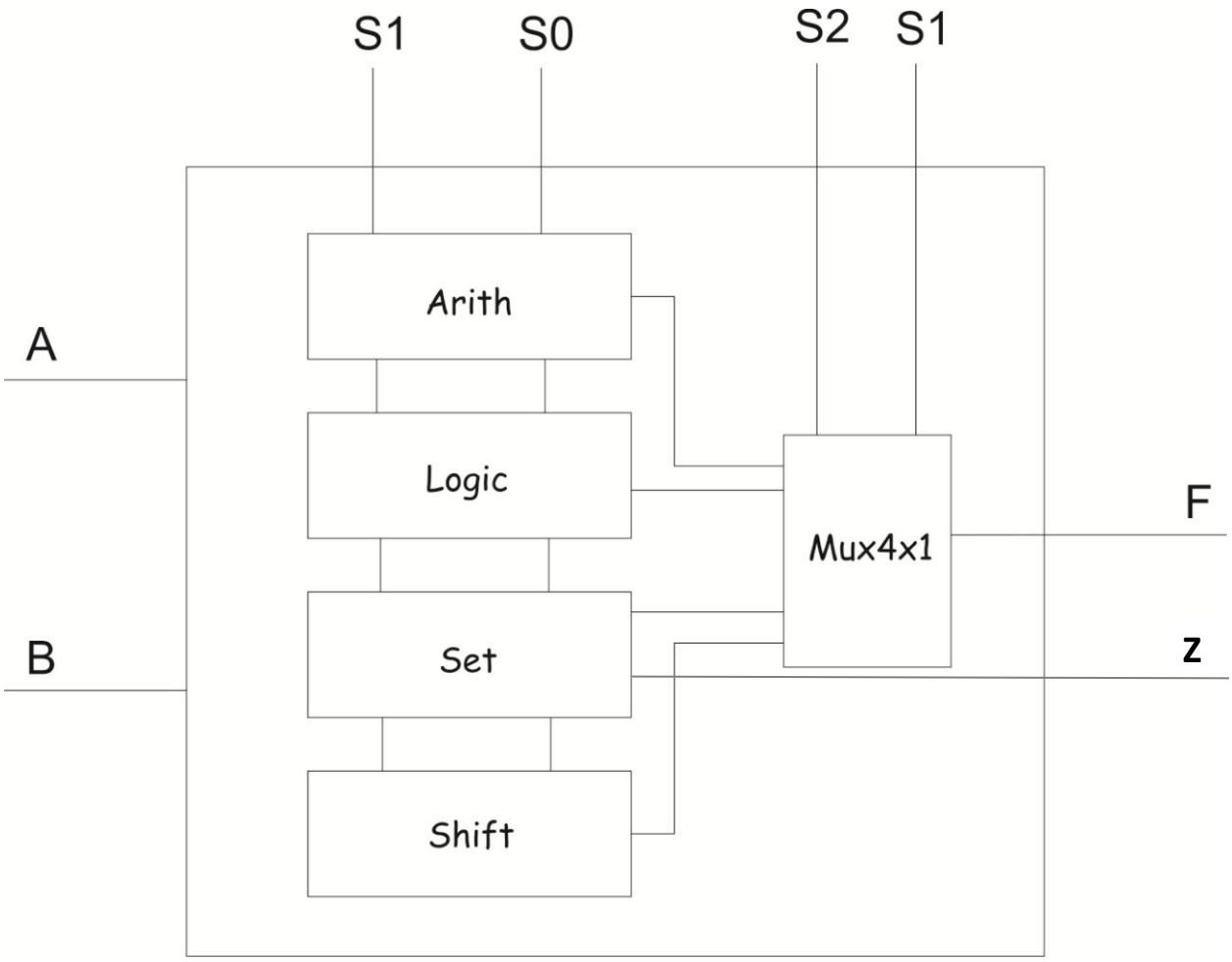

Fig 2 ALU Components

- **RegFile Component**

This component contains 32 GPR each with 32 bit width that can hold any data, except Register num 0 that doesn't change its value "0", the GPR interface with 2 Output Registers 'A' & 'B' and 1 Input Register 'C'.

It is allowed to out on 'A' and 'B' in the same clock cycle, this is done by separate the decoded control signals that out on 'A' and that out on 'B', as well as 'C' too, the component has 5x32 decoder to decode the control signal 'S' as shown in figure.

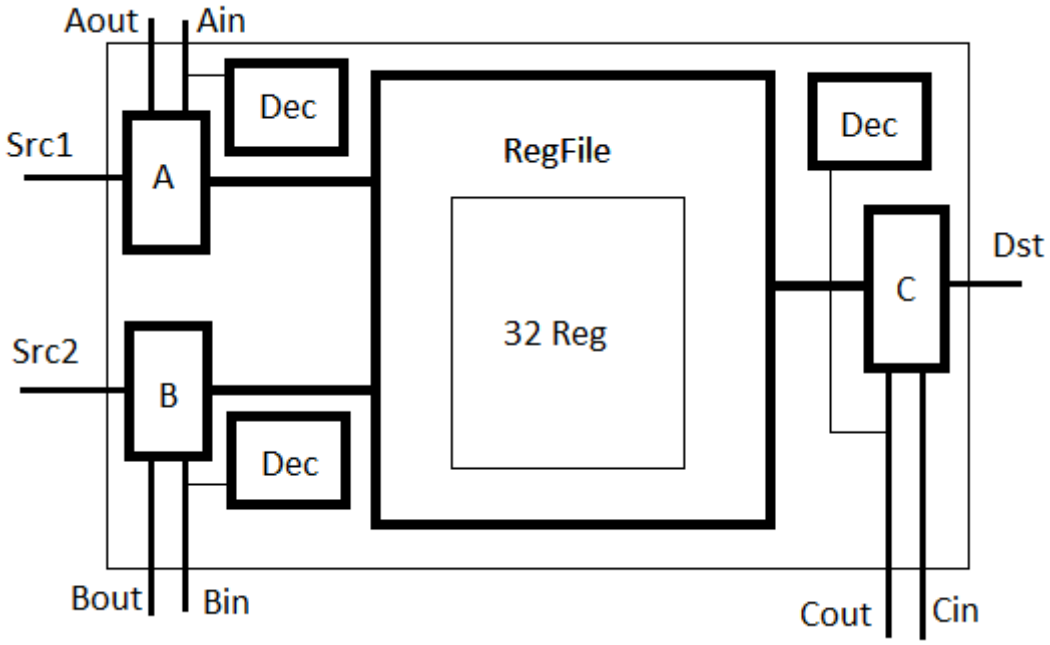

Fig 3 RegFile Component

- **Memory Component**

This component contains the memory interface with the MAR and MDR Registers that are 32 bit width each, the memory is controlled by ‘En’ signal that activate or deactivate the RAM, while ‘WE’ signal enables the write in RAM or Read from it, the RAM takes the data directly from MDR Register, while the address is directly taken from the MAR, and the output of the RAM is directed to IR bus or MDR according to ‘S1’ & ‘S2’ signals that control the Mux & DeMux components to set the data flow path as shown in the figure.

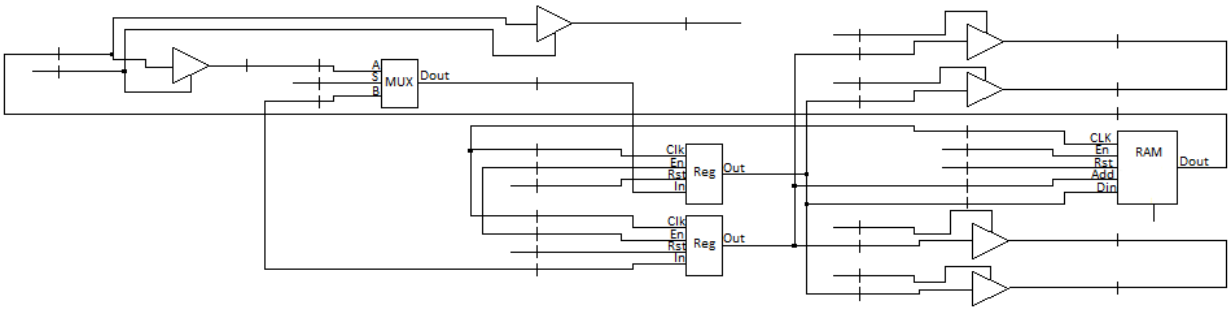

Fig 4 Memory Component

MDR and MAR Registers outs the data on Source1 bus and Source2 bus according to the tri-state buffers that control writing on these buses as in RegFile.

- **Controller Component**

This component is the Main component that synchronies the data flow between the rests of processor components to execute any Instruction according to the given Instruction set in the project description.

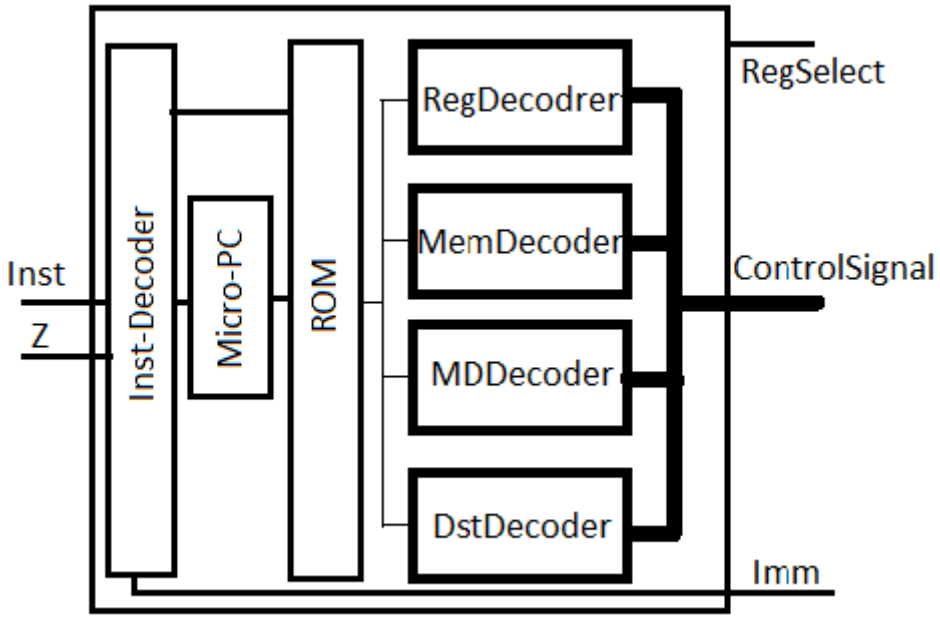

Fig 5 Controller Schematic & its components

The controller fetches the data from the Memory RAM to the IR register that is directly connected with the decoder with a logic circuit to forward the IR input in the same clock cycle that the IR save the Inst received from RAM.

The decoder processes the Instruction and jumps to the sequence of micro-Instructions in the Negative edge cycle that would be executed according to the Instruction from IR and “State” signals from ROM that indicate that the next micro-Inst will not be the next one, and set micro-MAR according to the jumping address saved in “MAddress” that is a memory that save the jumping addresses, the ROM that have all Micro-Instructions saved in it outs the micro-Inst according to the micro-MAR.

The Micro-Instruction is transferred to set of decoders to get the required control signal that is sent to the Processor Components, these decoders are as follows.

1. RegDecoder

This decoder gets the Register number from IR and get the required operation on it (Read, write to ‘A’, or write to ‘B’), and send the required control signal to RegFile Component.

2. MemDecoder

This decoder gets the required Memory Operation (Read or Write) and the flow of data from MAR & MDR to decode them to the set of required control signals for Memory Component.

3. DstDecoder

This decoder control the dataflow from the ALU on Destination bus to the RegFile through 'C' Register to be saved to one of 31 Register as R0's value always zero.

4. MDDecoder

This decoder control the dataflow from the Memory output to MDR or IR according to the 'S' selector that control the DeMux & Mux to connect IR of MDR to the Memory.

5. Extend

This component is connected to the IR register that determine which part of the Instruction will be extended when IR outs data on Source bus according to the Type of Instruction

The **Control signals** are 25 bit vector that are defined as following:

1. Bit 0: S (Mux & DeMux Selector)
2. Bit 1: En (Memory Enable)
3. Bit 2: WE (Write Enable/!Read)
4. Bit 3: MDRin (Save data in MDR)
5. Bit 4: IRin (save Inst in IR)
6. Bit 5: MARin (save add. in MAR)
7. Bit 6: Rdstin (save data in RegFile "Rd")
8. Bit 7: Cin (save data in 'C' Reg)
9. Bit 8: PCin (save address in PC)
10. Bit 9: Bout (out data from 'B' to Src2 bus)
11. Bit 10: PCout2 (out PC on Src2 bus)
12. Bit 11: IRout2 (out IR on Src2 bus)
13. Bit 12: MDRout2 (out MDR on Src2 bus)
14. Bit 13: MARout2 (out MAR on Src2 bus)
15. Bit 14: Rsrc2out (out RegFile on 'B' Reg)
16. Bit 15: Aout (out data from 'A' to Src1 bus)
17. Bit 16: PCout1 (out PC on Src1 bus)
18. Bit 17: IRout1 (out IR on Src1 bus)
19. Bit 18: MDRout1 (out MDR on Src1 bus)
20. Bit 19: MARout1 (out MAR on Src1 bus)
21. Bit 20: Rsrc1out (out RegFile on 'A' Reg)
22. Bits from 21 to 24: ALU Operation

- ## Buses Connections

This component shows the 3 Main Buses (Source1, Source2 and Destination) and there connections with the rest of processor components and the data flow between them as shown in the project description.