Faculty of Engineering, Cairo University

Computer Engineering Department

Computer Architecture Course Project

Orthrus - Phase 1 Report

Ahmad Khaled	Section 1 BN. 03
Maryam Shalaby	Section 2 BN. 21
Zeinab Rabie	Section 1 BN. 22
Omnia Zakaria	Section 1 BN. 23

1 Introduction

Orthrus is a pipelined static dual-issue microprocessor implementing a RISC ISA similar to the MIPS ISA. In the following sections we outline the instruction format, the general design of the processor, as well as the pipeline stages design.

2 Instruction Set Architecture

The structure for the IR for two-operand instructions is given in Figure 1. The structure for one-operand instructions is given in Figure 2. The structure for branching instructions is given in Figure 3. The structure for the rest of the instructions is given in Figure 4.

Direct	Register	Auto-increment	Auto-decrement	Indexed
Code	000	001	010	011
Indirect	Register	Auto-increment	Auto-decrement	Indexed
Code	100	101	110	111

Table 1: Addressing Mode Codes

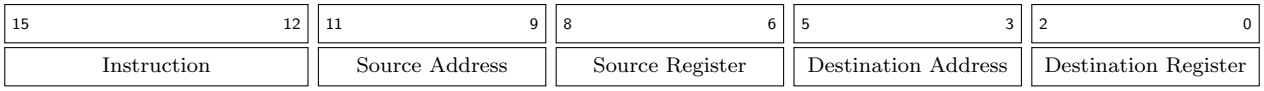
	R_0	R_1	R_2	R_3
Code	000	001	010	011
	R_4	R_5	R_6?	R_7
Code	100	101	110	111

Table 2: Register Addressing Codes

Instruction	MOV	ADD	ADC	SUB	SBC	AND	OR	XNOR	CMP
OP Code	1111	1110	1101	1100	1011	1010	1001	1000	0111

Table 3: Two-Operand Instruction Codes

Figure 1: IR Structure For Two-Operand Instructions



Instruction specifies the instruction to execute. Possible codes given in Table 3.

Source Address specifies the addressing mode to choose for the source. Possible codes given in Table 1.

Source Register specifies the source register. Possible codes given in Table 2.

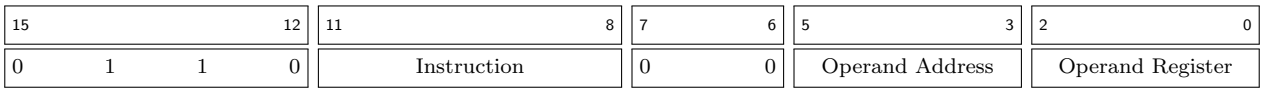
Destination Address specifies the addressing mode to choose for the destination. Possible codes given in Table 1.

Destination Register specifies the destination register. Possible codes given in Table 2.

Instruction	INC	DEC	CLR	INV	LSR	ROR	RRC	ASR	LSL	ROL	RLC
OP Code	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010

Table 4: Single Operand Instruction Codes

Figure 2: IR Structure For Single-Operand Instructions



Instruction specifies the instruction to execute. Possible codes given in Table 4.

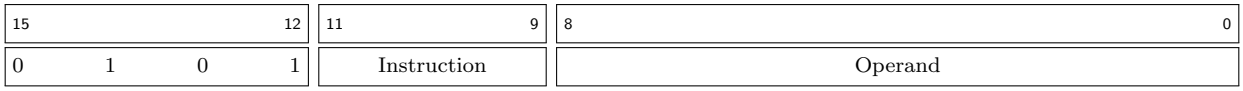
Operand Address specifies the addressing mode to choose for the operand. Possible codes given in Table 1.

Operand Register specifies the operand register. Possible codes given in Table 2.

Instruction	BR	BEQ	BNE	BLO	BLS	BHI	BHS
OP Code	000	001	010	011	100	101	110

Table 5: Branching Instruction Codes

Figure 3: IR Structure For Branching Instructions



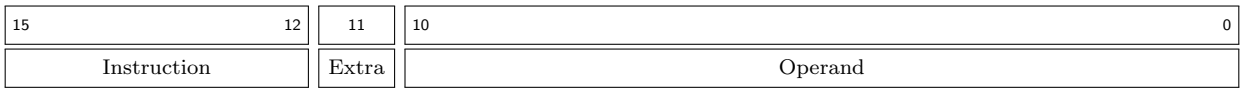
Instruction specifies the instruction to execute. Possible codes given in Table 5.

Operand specifies the branching offset.

Instruction	JSR	RTS	HITR	IRET	HLT	NOP
OP Code	0100	0011	0011	0010	0001	0000

Table 6: Miscellaneous Instruction Codes

Figure 4: IR Structure For Miscellaneous Instructions



Instruction specifies the instruction to execute (except for differentiating RTS and HITR, done using Extra bit). Possible codes given in Table 6.

Extra useful for JSR and HITR only. specifies whether the Operand is given directly or using a register/addressing mode for JSR. Indicates HITR when instruction is 0011 and Extra= 1, indicates RTS if instruction is 0011 and Extra= 0.

Operand useful for JSR only. specifies the memory location to jump to.

3 Schematics

Figures 5 and 6 include the schematic for the design.

4 Pipeline

4.1 Stages

4.2 Hazards

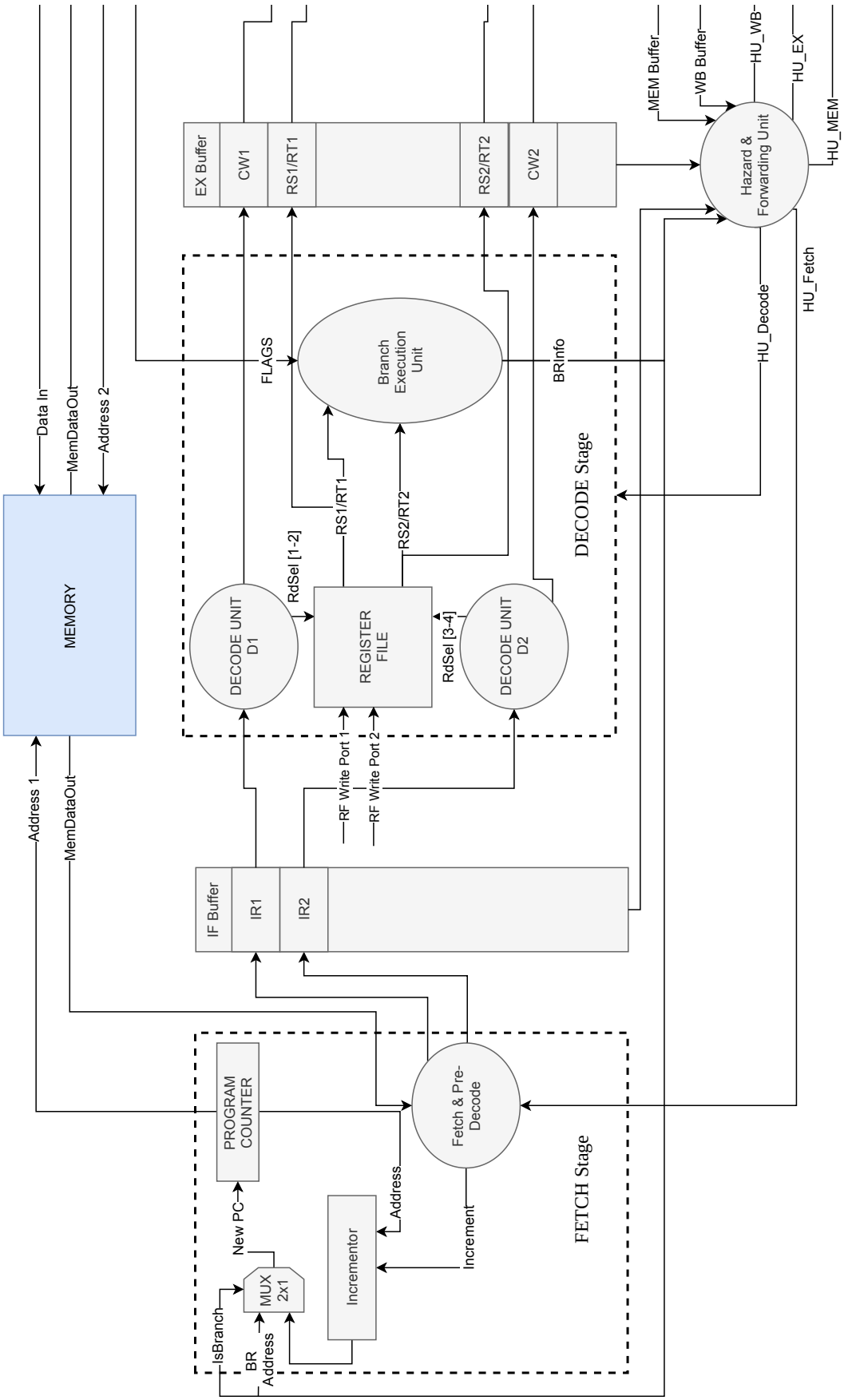


Figure 5: Overall Schematic Part 1

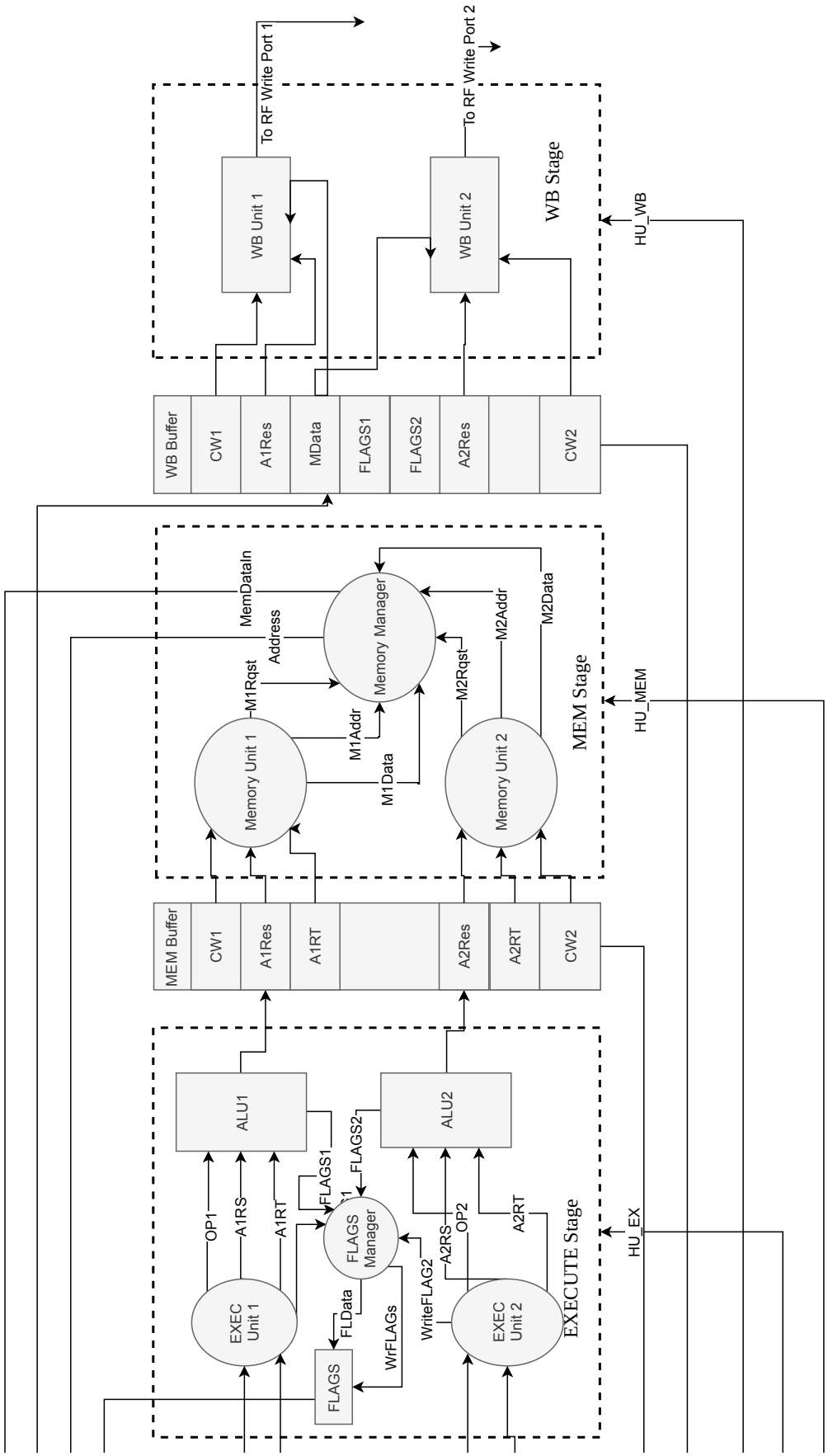


Figure 6: Overall Schematic Part 2