100

Cody Cziesler and Nick Desaulniers
Computer Organization
Project 1
5/4/10

OpCodes:

	OpCode [15]	OpCode [14]	OpCode [13]	OpCode [12]	RegDst	Branch	MemRead	MemToReg	MemWrite	ALUSrc	RegWrite	Jump	ALUOp [1]	ALUOp [2]	ALUOp [3]	ALUOp [4]
tfr	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1
add	0	0	0	1	1	0	0	0	0	0	1	0	0	0	0	1
sub	0	0	1	0	1	0	0	0	0	0	1	0	0	0	1	0
and	0	0	1	1	1	0	0	0	0	0	1	0	0	0	1	1
or	0	1	0	0	1	0	0	0	0	0	1	0	0	1	0	0
not	0	1	0	1	1	0	0	0	0	0	1	0	0	1	0	1
mul	0	1	1	0	1	0	0	0	0	0	1	0	0	1	1	0
div	0	1	1	1	1	0	0	0	0	0	1	0	0	1	1	1
j	1	0	0	0	X	X	0	X	0	X	0	1	X	X	X	X
sll	1	0	0	1	1	0	0	0	0	0	1	0	1	0	0	0
slr	1	0	1	0	1	0	0	0	0	0	1	0	1	0	0	1
beq	1	0	1	1	X	1	0	X	0	0	0	0	0	0	1	0
bne	1	1	0	0	X	1	0	X	0	0	0	0	0	0	1	0
load	1	1	0	1	0	0	1	1	0	1	1	0	0	0	0	1
store	1	1	1	0	X	0	0	X	1	1	0	0	0	0	0	1
nop	1	1	1	1	X	0	0	0	0	0	0	0	X	X	X	X

Control Logic:

RegDst =	C + A'B + A'D
Branch =	ABC'D' + AB'CD
MemRead =	ABC'D
MemToReg =	ABC'D
MemWrite =	ABCD'
ALUSrc =	ABC'D + ABCD'
RegWrite =	A' + C'D + B'CD'
Jump =	AB'C'D'
ALUOp[1] =	AB'C' + AB'D'
ALUOp[2] =	A'B
ALUOp[3] =	A'C
ALUOp[4] =	A'D + BD + ACD' + A'B'D

OpCode Types:

There are three types of encoding styles. Each one will tell the data path how to read the data.

Math Type

Opcode [15-12]	Source 1 [11-9]	Source 2 [8-6]	Destination [5-3]	Not Used [2-0]

This type is used for adding together two registers and placing the result in the destination. Some commonly used commands are add, sub, and, or, etc.