MAKE A COPY OF THIS SHEET	Instruction Metadata				ROM Input [1]		Control Signals								ROM Output [2
							addi is provided as an example								
Instruction	Type	Opcode	Funct3	Funct7	Binary	Decimal	**RegWEn**	**ImmSel**	**BrUn**	**ASel**	**BSel**	**ALUSel**	**MemRW**	**WBSel**	Hex
							1 binary digit	3 binary digits	1 binary digit	1 binary digit	1 binary digit	4 binary digits	1 binary digit	2 binary digits	
add rd, rs1, rs2	R	0b0110011	0b000	0b0000000	0b000000	0									0000
mul rd, rs1, rs2			0b000	0b0000001	0b000001	1									0000
sub rd, rs1, rs2			0b000	0b0100000	0b000010	2									0000
sll rd, rs1, rs2			0b001	0b0000000	0b000011	3									0000
mulh rd, rs1, rs2			0b001	0b0000001	0b000100	4									0000
mulhu rd, rs1, rs2			0b011	0b0000001	0b000101	5									0000
slt rd, rs1, rs2			0b010	0b0000000	0b000110	6									0000
xor rd, rs1, rs2			0b100	0b0000000	0b000111	7									0000
srl rd, rs1, rs2			0b101	0b0000000	0b001000	8									0000
sra rd, rs1, rs2			0b101	0b0100000	0b001001	9									0000
or rd, rs1, rs2			0b110	0b0000000	0b001010	10									0000
and rd, rs1, rs2			0b111	0b0000000	0b001011	11									0000
lb rd, offset(rs1)	I	0b0000011	0b000		0b001100	12									0000
lh rd, offset(rs1)			0b001		0b001101	13									0000
lw rd, offset(rs1)			0b010		0b001110	14									0000
addi rd, rs1, imm		0b0010011	0b000		0b001111	15	1	000 [3]	0 [4]	0 [5]	1 [6]	0000	0	01 [7]	1041
slli rd, rs1, imm			0b001	0b0000000	0b010000	16									0000
slti rd, rs1, imm			0b010		0b010001	17									0000
xori rd, rs1, imm			0b100		0b010010	18									0000
srli rd, rs1, imm			0b101	0b0000000	0b010011	19									0000
srai rd, rs1, imm			0b101	0b0100000	0b010100	20									0000
ori rd, rs1, imm			0b110		0b010101	21									0000
andi rd, rs1, imm			0b111		0b010110	22									0000
sb rs2, offset(rs1)	S	0b0100011	0b000		0b010111	23									0000
sh rs2, offset(rs1)			0b001		0b011000	24									0000
sw rs2, offset(rs1)			0b010		0b011001	25									0000
beq rs1, rs2, offset	B	0b1100011	0b000		0b011010	26									0000
bne rs1, rs2, offset			0b001		0b011011	27									0000
blt rs1, rs2, offset			0b100		0b011100	28									0000
bge rs1, rs2, offset			0b101		0b011101	29									0000
bltu rs1, rs2, offset			0b110		0b011110	30									0000
bgeu rs1, rs2, offset			0b111		0b011111	31									0000
auipc rd, offset	U	0b0010111			0b100000	32									0000
lui rd, offset		0b0110111			0b100001	33									0000
jal rd, imm	J	0b1101111			0b100010	34									0000
jalr rd, rs1, imm	I	0b1100111	0b000		0b100011	35									0000

[1] This is the value that will be passed into the ROM

[2] This is the value that will be outputted from the ROM. It's all the control signals concatenated together.

[3] This value is provided as an example. Based on your design for the immediate generator, you may need to modify this value to generate the correct immediate value

[4] This value actually doesn't matter because the addi instruction never uses the branch comparator. However, you must fill out every cell so the control bits line up properly

[5] This value is provided as an example. Based on your design for the A MUX, you may need to modify this value to generate the correct immediate value

[6] This value is provided as an example. Based on your design for the B MUX, you may need to modify this value to generate the correct immediate value

[7] This value is provided as an example. Based on your design for the Writeback MUX, you may need to modify this value to generate the correct immediate value