Robert Pierce

CS 2200

06/01/2016

**Project 1**

**ISA Micro-Instructions**

**ROM Description**

* 12 bit address width

|  |  |  |
| --- | --- | --- |
| OPCODE  (B11-B6) (6-bits) | $Z  (B5) (1-bit) | Next Sate (from state register)  (B04-B00) (5-bits) |

* 24 bit data width
  + **ROM DATA FORMAT**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **DRIVE SIGNALS**  (B25-B20) (6-bits) | | | | | | **LOAD SIGNALS**  (B19-B14) (6-bits) | | | | | | **WRITE**  **SIGNALS**  (B13-B12) (2-bits) | | **RESET** **SIGNALS**  (B11-B10) (2-bits) | | B09-B07  (3-bits) | B06-B05  (2-bits) | B04-B00  (5-bits) |
| PC | ALU | REG | MEM | IMM | TARGET | PC | A | B | MAR | IR | Z | MEM | REG |  |  | FUNC | REGSEL | Next State |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

* ROM is initialized at address 0

**FETCH INSTRUCTION**

* **FETCH** 
  + **FETCH-1 MAR ← $PC**

**A ← $PC**

* + - DrPC
    - LdMAR
    - LdA
  + **FETCH-2 $PC ← $PC + 1**
    - DrALU
    - LdPC
    - func 001
  + **FETCH-3 IR ← MEM[MAR]**
    - DrMEM
    - LdIR

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| FETCH |  | **DRIVE SIGNALS**  (B25-B20) (6-bits) | | | | | | **LOAD SIGNALS**  (B19-B14) (6-bits) | | | | | | **WRITE**  **SIGNALS**  (B13-B12) (2-bits) | | **RESET** **SIGNALS**  (B11-B10) (2-bits) | | (B09-B07)  (3-bits) | (B06-B05)  (2-bits) | (B04-B00)  (5-bits) |
| curr state | addr | PC | ALU | REG | MEM | IMM | TARGET | PC | A | B | MAR | IR | Z | MEM | REG | IR | Z | FUNC | REGSEL | Next State |
| FETCH-1 | 0000 0000 0000 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 00 | 0 0001 |
| FETCH-2 | 0000  0000  0001 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 001 | 00 | 0 0010 |
| FETCH-3 | 0000  0000  0010 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 000 | 00 | 0 0000 |

|  |  |  |
| --- | --- | --- |
| **curr state** | **addr** | **HEX INSTRUCTION** |
| FETCH-1 | 0x  000 | 0x2050001 |
| FETCH-2 | 0x  001 | 0x1080082 |
| FETCH-3 | 0x  002 | 0x0408000 |

**Arithmetic**

* **ADD** add $Rd, $Rb, $Rc OPCODE: 00 0001
  + **Add-1 A ← $Rb**
    - DrREG
    - LdA
    - REGSEL 00
  + **Add-2 B ← $Rc**
    - DrREG
    - LdB
    - REGSEL 01
  + **Add-3 $Rd ← A + B**
    - DrALU
    - WrREG
    - REGSEL 10
    - func 000
  + **Add-4 Reset IR**
    - ResIR

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ADD |  | **DRIVE SIGNALS**  (B25-B20) (6-bits) | | | | | | **LOAD SIGNALS**  (B19-B14) (6-bits) | | | | | | **WRITE**  **SIGNALS**  (B13-B12) (2-bits) | | **RESET SIGNALS**  (B11-B10)  (2-bits) | | B09-B07  (3-bits) | B06-B05  (2-bits) | B04-B00  (5-bits) |
| curr state | addr | PC | ALU | REG | MEM | IMM | TARGET | PC | A | B | MAR | IR | Z | MEM | REG | IR | Z | FUNC | REGSEL | Next State |
| ADD-1 | 0000 0100 0000 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 00 | 0 0001 |
| ADD-2 | 0000  0100  0001 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 01 | 0 0010 |
| ADD-3 | 0000  0100  0010 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 000 | 10 | 0 0011 |
| ADD-4 | 0000  0100  0011 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 000 | 00 | 0 0000 |

|  |  |  |
| --- | --- | --- |
| **curr state** | **addr** | **HEX INSTRUCTION** |
| ADD-1 | 0x 040 | 0x0840001 |
| ADD-2 | 0x  041 | 0x0820022 |
| ADD-3 | 0x  042 | 0x1001043 |
| ADD-4 | 0x  043 | 0x0000800 |

* **ADDi** addi $Rd, $Rb, immediateOPCODE: 00 0010
  + **Addi-1 A ← $Rb**
    - DrREG
    - LdA
    - REGSEL 00
  + **Addi-2 B ← immediate**
    - DrIMM
    - LdB
  + **Addi-3 $Rd ← A + B**
    - DrALU
    - WrREG
    - REGSEL 01
    - func 000
  + **Addi-4 Reset IR**
    - ResIR

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Addi |  | **DRIVE SIGNALS**  (B25-B20) (6-bits) | | | | | | **LOAD SIGNALS**  (B19-B14) (6-bits) | | | | | | **WRITE**  **SIGNALS**  (B13-B12) (2-bits) | | **RESET SIGNALS**  (B11-B10)  (2-bits) | | B09-B07  (3-bits) | B06-B05  (2-bits) | B04-B00  (5-bits) |
| curr state | addr | PC | ALU | REG | MEM | IMM | TARGET | PC | A | B | MAR | IR | Z | MEM | REG | IR | Z | FUNC | REGSEL | Next State |
| ADDi-1 | 0000 1000 0000 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 00 | 0 0001 |
| ADDi-2 | 0000  1000  0001 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 00 | 0 0010 |
| ADDi-3 | 0000  1000  0010 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 000 | 01 | 0 0011 |
| ADDi-4 | 0000  1000  0011 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 000 | 00 | 0 0000 |

|  |  |  |
| --- | --- | --- |
| **curr state** | **addr** | **HEX INSTRUCTION** |
| ADDi-1 | 0x 080 | 0x0840001 |
| ADDi-2 | 0x  081 | 0x0220022 |
| ADDi-3 | 0x  082 | 0x1001023 |
| ADDi-4 | 0x  083 | 0x0000800 |

**Logic**

* **AND** and $Rd, $Rb, $Rc OPCODE: 00 0110
  + **AND-1 A ← Rb**
    - DrREG
    - LdA
    - REGSEL 00
  + **AND-2 B ← Rc**
    - DrREG
    - LdB
    - REGSEL 01
  + **AND-3 Rd ← A and B**
    - DrALU
    - WrREG
    - func 110
    - REGSEL 10
  + **AND-4 Reset IR**
    - ResIR

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| AND |  | **DRIVE SIGNALS**  (B25-B20) (6-bits) | | | | | | **LOAD SIGNALS**  (B19-B14) (6-bits) | | | | | | **WRITE**  **SIGNALS**  (B13-B12) (2-bits) | | **RESET SIGNALS**  (B11-B10)  (2-bits) | | B09-B07  (3-bits) | B06-B05  (2-bits) | B04-B00  (5-bits) |
| curr state | addr | PC | ALU | REG | MEM | IMM | TARGET | PC | A | B | MAR | IR | Z | MEM | REG | IR | Z | FUNC | REGSEL | Next State |
| AND-1 | 0001 1000 0000 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 00 | 0 0001 |
| AND-2 | 0001  1000  0001 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 01 | 0 0010 |
| AND-3 | 0001  1000  0010 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 110 | 10 | 0 0011 |
| AND-4 | 0001  1000  0011 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 000 | 00 | 0 0000 |

|  |  |  |
| --- | --- | --- |
| **curr state** | **Addr** | **HEX INSTRUCTION** |
| AND-1 | 0x  180 | 0x0840001 |
| AND-2 | 0x  181 | 0x0820022 |
| AND-3 | 0x  182 | 0x1001343 |
| AND-4 | 0x  183 | 0x0000800 |

* **OR** or $Rd, $Rb, $Rc OPCODE: 00 0111
  + **OR-1 A ← Rb**
    - DrREG
    - LdA
    - REGSEL 00
  + **OR-2 B ← Rc**
    - DrREG
    - LdB
    - REGSEL 01
  + **OR-3 Rd ← A or B**
    - DrALU
    - WrREG
    - func 111
    - REGSEL 10
  + **OR-4 Reset IR**
    - ResIR

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| OR |  | **DRIVE SIGNALS**  (B25-B20) (6-bits) | | | | | | **LOAD SIGNALS**  (B19-B14) (6-bits) | | | | | | **WRITE**  **SIGNALS**  (B13-B12) (2-bits) | | **RESET SIGNALS**  (B11-B10)  (2-bits) | | B09-B07  (3-bits) | B06-B05  (2-bits) | B04-B00  (5-bits) |
| curr state | addr | PC | ALU | REG | MEM | IMM | TARGET | PC | A | B | MAR | IR | Z | MEM | REG | IR | Z | FUNC | REGSEL | Next State |
| OR-1 | 0001 1100 0000 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 00 | 0 0001 |
| OR-2 | 0001  1100  0001 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 01 | 0 0010 |
| OR-3 | 0001  1100  0010 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 111 | 10 | 0 0011 |
| OR-4 | 0001  1100  0011 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 000 | 00 | 0 0000 |

|  |  |  |
| --- | --- | --- |
| **curr state** | **Addr** | **HEX INSTRUCTION** |
| OR-1 | 0x  1c0 | 0x0840001 |
| OR-2 | 0x  1c1 | 0x0820022 |
| OR-3 | 0x  1c2 | 0x10013c3 |
| OR-4 | 0x  1c3 | 0x0000800 |

**Load/Store**

* **Load Word** lw $Rd, Offset($Rb) OPCODE: 00 1000
  + **LW-1 A ← Rb**
    - DrREG
    - LdA
    - REGSEL 00
  + **LW-2 B ← Offset**
    - DrIMM
    - LdB
  + **LW-3 MAR ← A + B**
    - DrALU
    - LdMAR
    - func 000
  + **LW-4 Rd ← MEM[MAR]**
    - DrMEM
    - WrREG
    - REGSEL 01
  + **LW-5 Reset IR**
    - ResIR

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| LW |  | **DRIVE SIGNALS**  (B25-B20) (6-bits) | | | | | | **LOAD SIGNALS**  (B19-B14) (6-bits) | | | | | | **WRITE**  **SIGNALS**  (B13-B12) (2-bits) | | **RESET SIGNALS**  (B11-B10)  (2-bits) | | B09-B07  (3-bits) | B06-B05  (2-bits) | B04-B00  (5-bits) |
| curr state | addr | PC | ALU | REG | MEM | IMM | TARGET | PC | A | B | MAR | IR | Z | MEM | REG | IR | Z | FUNC | REGSEL | Next State |
| LW-1 | 0010 0000 0000 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 00 | 0 0001 |
| LW-2 | 0010  0000  0001 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 00 | 0 0010 |
| LW-3 | 0010  0000  0010 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 00 | 0 0011 |
| LW-4 | 0010  0000  0011 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 000 | 01 | 0 0100 |
| LW-5 | 0010  0000  0100 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 000 | 00 | 0 0000 |

|  |  |  |
| --- | --- | --- |
| **curr state** | **Addr** | **HEX INSTRUCTION** |
| LW-1 | 0x  200 | 0x0840001 |
| LW-2 | 0x  201 | 0x0220002 |
| LW-3 | 0x  202 | 0x1010003 |
| LW-4 | 0x  203 | 0x0401024 |
| LW-5 | 0x  204 | 0x0000800 |

* **Store Word** sw $Rd, Offset($Rb) OPCODE: 00 1001
  + **SW-1 A ← Rb**
    - DrREG
    - LdA
    - REGSEL 00
  + **SW-2 B ← Offset**
    - DrIMM
    - LdB
  + **SW-3 MAR ← A + B**
    - DrALU
    - LdMAR
    - func 000
  + **SW-4 MEM[MAR] ← Rd**
    - DrREG
    - WrMEM
    - REGSEL 01
  + **SW-5 Reset IR**
    - ResIR

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| SW |  | **DRIVE SIGNALS**  (B25-B20) (6-bits) | | | | | | **LOAD SIGNALS**  (B19-B14) (6-bits) | | | | | | **WRITE**  **SIGNALS**  (B13-B12) (2-bits) | | **RESET SIGNALS**  (B11-B10)  (2-bits) | | B09-B07  (3-bits) | B06-B05  (2-bits) | B04-B00  (5-bits) |
| curr state | addr | PC | ALU | REG | MEM | IMM | TARGET | PC | A | B | MAR | IR | Z | MEM | REG | IR | Z | FUNC | REGSEL | Next State |
| SW-1 | 0010 0100 0000 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 00 | 0 0001 |
| SW-2 | 0010  0100  0001 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 00 | 0 0010 |
| SW-3 | 0010  0100  0010 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 00 | 0 0011 |
| SW-4 | 0010  0100  0011 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 000 | 01 | 0 0100 |
| SW-5 | 0010  0100  0100 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 000 | 00 | 0 0000 |

|  |  |  |
| --- | --- | --- |
| **curr state** | **Addr** | **HEX INSTRUCTION** |
| SW-1 | 0x  240 | 0x0840001 |
| SW-2 | 0x  241 | 0x0220002 |
| SW-3 | 0x  242 | 0x1010003 |
| SW-4 | 0x  243 | 0x0802024 |
| SW-5 | 0x  244 | 0x0000800 |

**Jumps**

* **JUMP** j Target Address OPCODE: 00 1010
  + **JUMP-1 PC ← TARGET**
    - DrTARGET
    - LdPC
  + **JUMP-2 Reset IR**
    - ResIR

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| JUMP |  | **DRIVE SIGNALS**  (B25-B20) (6-bits) | | | | | | **LOAD SIGNALS**  (B19-B14) (6-bits) | | | | | | **WRITE**  **SIGNALS**  (B13-B12) (2-bits) | | **RESET SIGNALS**  (B11-B10)  (2-bits) | | B09-B07  (3-bits) | B06-B05  (2-bits) | B04-B00  (5-bits) |
| curr state | addr | PC | ALU | REG | MEM | IMM | TARGET | PC | A | B | MAR | IR | Z | MEM | REG | IR | Z | FUNC | REGSEL | Next State |
| JUMP-1 | 0010 1000 0000 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 00 | 0 0001 |
| JUMP-2 | 0010  1000  0001 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 000 | 00 | 0 0000 |

|  |  |  |
| --- | --- | --- |
| **curr state** | **addr** | **HEX INSTRUCTION** |
| JUMP-1 | 0x 280 | 0x0180001 |
| JUMP-2 | 0x  281 | 0x0000800 |

* **JUMP REGISTER** jr $Rb OPCODE: 00 1011
  + **JR-1 PC ←** $Rb
    - DrREG
    - LdPC
    - REGSEL 00
  + **JR-2 Reset IR**
    - ResIR

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| JR |  | **DRIVE SIGNALS**  (B25-B20) (6-bits) | | | | | | **LOAD SIGNALS**  (B19-B14) (6-bits) | | | | | | **WRITE**  **SIGNALS**  (B13-B12) (2-bits) | | **RESET SIGNALS**  (B11-B10)  (2-bits) | | B09-B07  (3-bits) | B06-B05  (2-bits) | B04-B00  (5-bits) |
| curr state | addr | PC | ALU | REG | MEM | IMM | TARGET | PC | A | B | MAR | IR | Z | MEM | REG | IR | Z | FUNC | REGSEL | Next State |
| JUMP-1 | 0010 1100 0000 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 00 | 0 0001 |
| JUMP-2 | 0010  1100  0001 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 000 | 00 | 0 0000 |

|  |  |  |
| --- | --- | --- |
| **curr state** | **addr** | **HEX INSTRUCTION** |
| JUMP-1 | 0x 2c0 | 0x0880001 |
| JUMP-2 | 0x  2c1 | 0x0000800 |

* **JUMP AND LINK** jal Target Address OPCODE: 00 1100
  + **JAL-1 $Ra ← PC**
    - DrPC
    - WrREG
    - REGSEL 11

Note: $PC is incremented in the fetch instruction

* + **JAL-2 $PC ← Target Address**
    - DrTARGET
    - LdPC
  + **JR-3 Reset IR**
    - ResIR

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| JAL |  | **DRIVE SIGNALS**  (B25-B20) (6-bits) | | | | | | **LOAD SIGNALS**  (B19-B14) (6-bits) | | | | | | **WRITE**  **SIGNALS**  (B13-B12) (2-bits) | | **RESET SIGNALS**  (B11-B10)  (2-bits) | | B09-B07  (3-bits) | B06-B05  (2-bits) | B04-B00  (5-bits) |
| curr state | addr | PC | ALU | REG | MEM | IMM | TARGET | PC | A | B | MAR | IR | Z | MEM | REG | IR | Z | FUNC | REGSEL | Next State |
| JAL-1 | 0011 0000 0000 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 000 | 11 | 0 0001 |
| JAL-2 | 0011  0000  0001 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 00 | 0 0010 |
| JAL-3 | 0011  0000  0010 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 000 | 00 | 0 0000 |

|  |  |  |
| --- | --- | --- |
| **curr state** | **addr** | **HEX INSTRUCTION** |
| JAL-1 | 0x 300 | 0x2001061 |
| JAL-2 | 0x  301 | 0x0180002 |
| JAL-3 | 0x  302 | 0x0000800 |

**Branch**

* **BRANCH ON EQUAL** beq $Rb, $Rc, Signed Offset OPCODE: 00 1101
  + **BEQ-1 A ← $Rb**
    - DrREG
    - LdA
    - REGSEL 00
  + **BEQ-2 B ← $Rc**
    - DrREG
    - LdB
    - REGSEL 01
  + **BEQ-3 Z ← A - B**
    - DrALU
    - LdZ
    - func 010

**IF ($Rb == $Rc)**

* + **BEQ-4 A ← PC**
    - DrPC
    - LdA
  + **BEQ-5 B ← IMM**
    - DrIMM
    - LdB
  + **BEQ-6 PC ← A + B**
    - DrALU
    - LdPC
    - func 000

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| BEQ |  | **DRIVE SIGNALS**  (B25-B20) (6-bits) | | | | | | **LOAD SIGNALS**  (B19-B14) (6-bits) | | | | | | **WRITE**  **SIGNALS**  (B13-B12) (2-bits) | | **RESET SIGNALS**  (B11-B10)  (2-bits) | | B09-B07  (3-bits) | B06-B05  (2-bits) | B04-B00  (5-bits) |
| curr state | addr | PC | ALU | REG | MEM | IMM | TARGET | PC | A | B | MAR | IR | Z | MEM | REG | IR | Z | FUNC | REGSEL | Next State |
| BEQ-1 | 0011 0100 0000 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 00 | 0 0001 |
| BEQ-2 | 0011  0100  0001 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 01 | 0 0010 |
| BEQ-3 | 0011  0100  0010 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 010 | 00 | 0 0011 |
| **IF ($Rb == $Rc)**  **NOTE: Address changed b/c Z == 1** | | | | | | | | | | | | | | | | | | | | |
| BEQ-4 | 0011  0110  0011 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 00 | 0 0100 |
| BEQ-5 | 0011  0110  0100 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 00 | 0 0101 |
| BEQ-6 | 0011  0110  0101 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000 | 00 | 0 0110 |
| BEQ-7 | 0011  0110  0110 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 000 | 00 | 0 0000 |

**IF ($Rb != $Rc)**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| BEQ |  | **DRIVE SIGNALS**  (B25-B20) (6-bits) | | | | | | **LOAD SIGNALS**  (B19-B14) (6-bits) | | | | | | **WRITE**  **SIGNALS**  (B13-B12) (2-bits) | | **RESET SIGNALS**  (B11-B10)  (2-bits) | | B09-B07  (3-bits) | B06-B05  (2-bits) | B04-B00  (5-bits) |
| curr state | addr | PC | ALU | REG | MEM | IMM | TARGET | PC | A | B | MAR | IR | Z | MEM | REG | IR | Z | FUNC | REGSEL | Next State |
| BEQ-4 | 0011 0100 0011 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 000 | 00 | 0 0000 |

|  |  |  |
| --- | --- | --- |
| **curr state** | **addr** | **HEX INSTRUCTION** |
| BEQ-1 | 0x  340 | 0x0840001 |
| BEQ-2 | 0x  341 | 0x0820022 |
| BEQ-3 | 0x  342 | 0x1004103 |
| **IF ($Rb == $Rc)**  **NOTE: Address changed b/c Z == 1** | | |
| BEQ-4 | 0x  363 | 0x2040004 |
| BEQ-5 | 0x  364 | 0x0220005 |
| BEQ-6 | 0x  365 | 0x1080006 |
| BEQ-7 | 0x  366 | 0x0000C07 |

**IF ($Rb != $Rc)**

|  |  |  |
| --- | --- | --- |
| BEQ-5 | 0x  343 | 0x0000C07 |