1st initializing the memory with the following values :

* M[0] = 0001
* M[1]= 0001
* M[2]= 000a
* M[60]= 430a
* M[61]= 7342

The initializing code :

|  |  |  |
| --- | --- | --- |
| **Instruction** | **Hex Code** | **Expected Value** |
| LUI R1, 0 | 9000 | R1 = 0 |
| ORI R1, R1, 1 | 2849 | R1 = 1 |
| sw $1 , 0($0) | 6801 | Mem[0] = 1 |
| LUI R1, 0 | 9000 | R1 = 0 |
| ORI R2, R1, 1 | 284A | R2 = 1 |
| ADD $1, $0, $0 | 0840 | R1 = 0 |
| sw $2 , 1($0) | 6842 | Mem[1] = 1 |
| LUI R1, 0 | 9000 | R1 = 0 |
| ORI R3, R1, 10 | 2A8B | R3 = 10 |
| ADD $1, $0, $0 | 0840 | R1 = 0 |
| sw $3 , 2($0) | 6883 | Mem[2] = 10 |
| LUI R1, 1 | 9001 | R1 = 32 |
| ORI R3, R1, 28 | 2F0B | R3 = 60 |
| ADD $1, $0, $0 | 0840 | R1 = 0 |
| LUI R1, 536 | 9218 | R1 = 17152 |
| ORI R1, R1, 10 | 2A89 | R1 = 17162 |
| sw $1 , 0($3) | 6819 | Mem[60] = 17162 = 0x430A |
| LUI R1, 922 | 939A | R1 = 29504 |
| ORI R2, R1, 2 | 288A | R2 = 29506 |
| ADD $1, $0, $0 | 0840 | R1 = 0 |
| sw $2 , 1($3) | 685A | Mem[61] = 29506 = 0x7342 |
| add $2 , $0 ,$0 | 0880 | R2 = 0 |
| add $3 , $0 ,$0 | 08C0 | R3 = 0 |

**The main code :**

|  |  |  |
| --- | --- | --- |
| **Instruction** | **Hex Code** | **Expected Value** |
| Lui 900 | 9384 | R1 = 28800 |
| Addi R5, R1,13 | 3B4D | R5 = 28813 |
| Xor R3, R1, R5 | 04CD | R3 = 13 |
| Lw R1, 0(R0) | 6001 | R1 = 1 |
| Lw R2, 1(R0) | 6042 | R2 = 1 |
| Lw R3, 2(R0) | 6083 | R3 = 10 |
| Addi R4, R4, 10 | 3AA4 | R4 = 10 |
| Sub R4, R4, R4 | 0B24 | R4 = 0 |
| Add R4, R2, R4 | 0914 | R4 = 55 (Last value) |
| Slt R6, R2, R3 | 0D93 | R6 = 0 (Last value) |
| Beq R6, R0, L1 | 70F0 | PC = 36 (Last value) |
| Add R2, R1, R2 | 088A | R2 = 10 (Last value) |
| Beq R0, R0, L2 | 7700 | PC = 31 |
| Sw R4, 0(R0) | 6804 | Mem[0] = 55 |
| Jal func | F804 | PC = 41 , R7 = 38 |
| Sll R3, R2, 6 | 4193 | R3 = -15744 = 0XC280 |
| ROR R6, R3, 3 | 58DE | R6 = 6224 = 0x1850 |
| beq r0,r0,0 | 7000 | PC = 40 (always) |
| or R5, R2, R3 | 0353 | R5 = 10 |
| Lw R1, 0(R0) | 6001 | R1 = 55 |
| Lw R2, 5(R1) | 614A | R2 = 17162 = 0x430A |
| Lw R3 ,6(R1) | 618B | R3 = 29506 = 0x7342 |
| And R4, R2, R3 | 0113 | R4 = 17154 = 0x4302 |
| Sw R4, 0(R0) | 6804 | Mem[0] = 17154 = 0x4302 |
| Jr R7 | 1038 | PC = 38 |

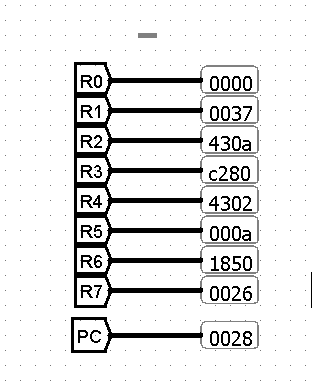
**An error faced :**

For the addressing mode in branch instructions the next PC should be PC = PC + sign-extend (Imm5)

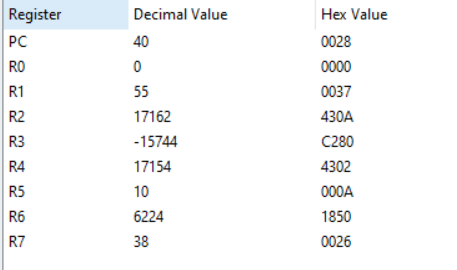
So in order to stay at the same instruction the offset should be zero so that the next PC will be the same as the current PC 🡪 PC = PC + 0

So we had to change the instruction ( beq r0,r0,-1 ) to ( beq r0,r0,0 ) to keep locking back to the same instruction and the program would be over.

**Actual values :**

****

**Final expected values :**

****