Report - Lab 1. Lucky111

Name: 王晨雨

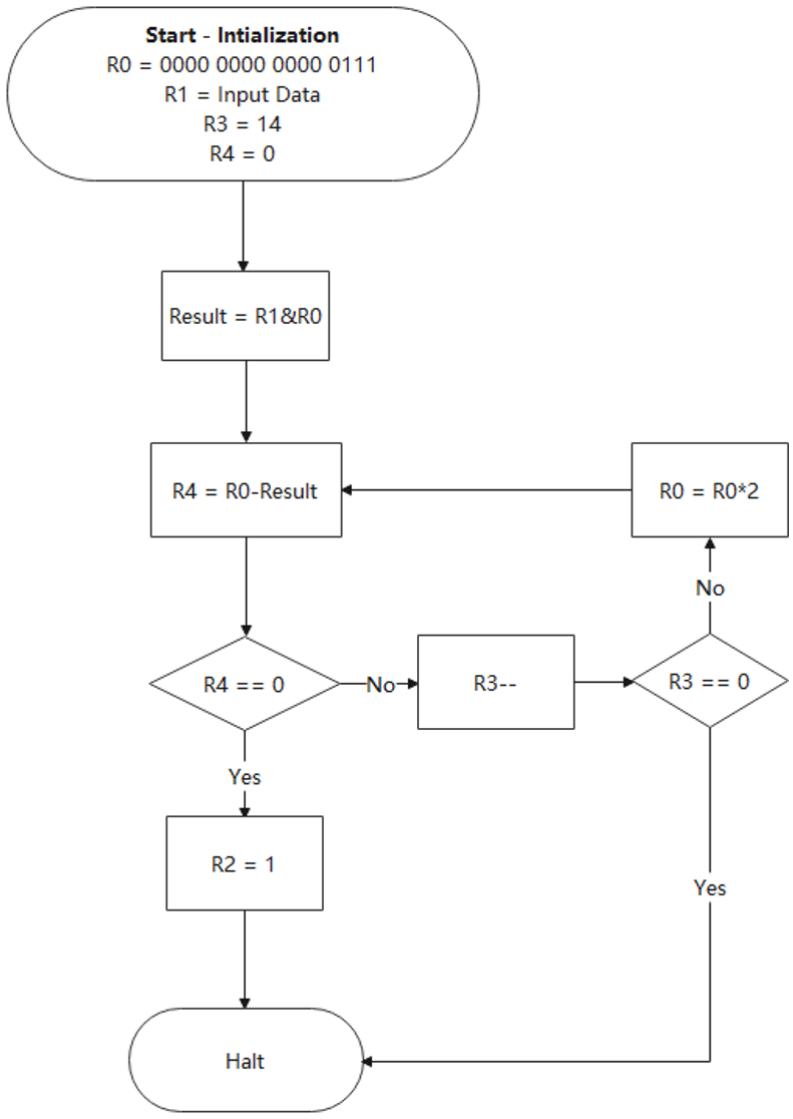
Student ID: 3200102324

TA Check Date: 2021.07.17(Saturday)

Report Submission Date: 2021.07.17(Saturday)

1 Algorithm

The following chart shows the algorithm used in the program.



Explanation:

- **Start**: Intialize the registers, load necessary data.
- **Two jugements**:
  - `If R4 == 0`
    - If yes, then this data contains three consecutive 1, let R2 = 1.
    - If no, means the three bits that are being judged do not meet the satisfaction.
  - `If R3 == 0`
    - If yes, means all bits have been tested, then end the loop, jump to `HALT`
    - If no, means there have been bits not tested, then continue the loop.

2 Code

```
0011 0000 0000 0000 ; .ORIG x3000
; Initialization
0101 000 000 1 00000 ; R0 <- 0
0101 001 001 1 00000 ; R1 <- 0
0101 010 010 1 00000 ; R2 <- 0
0010 001 0 1111 1100 ; R1 <- M[x3100]
0101 011 011 1 00000 ; R3 <- 0 R3 is the counter
0001 011 011 1 01110 ; R3 <- 14
0101 100 100 1 00000 ; R4 <- 0
0001 000 000 1 00111 ; R0 <- 0000 0000 0000 0111
; Loop
0000 111 000000011   ; Jump to line 16
0001 011 011 1 11111 ; R3--
0000 010 000000111   ; if R3==0,end the loop
0001 000 000 000 000 ; ADD R0,R0,R0 ; R0=2*R0
0101 100 000 000 001 ; AND R4,R0,R1
```

```
1001 100 100 111111  ; R4 <- NOT(result)
0001 100 100 1 00001 ; R4 <- -result
0001 100 100 000 000 ; R4 = R0 - result ,if R4==0, then result == R0
0000 101 111111000   ; if R4!= 0,then Jump to line 13,else end the recurrence
0001 010 010 1 00001 ; if R4 == 0, then R2 = 1
; HALT
1111 0000 0010 0101  ; HALT
```

3 Check process

TA: What's your general thought of this program?

Answer: The general idea of this program is testing every constant three bits of binary. For example, firstly, test if the smallest three bits of binary is '1', if yes, then change R2 to 1; If not , then test higher binary bits;

There are two loop exits, one is when we found three consecutive 1, the other is when we test [15:13] but haven't found three consecutive 1 yet, then PC jump to `HALT` to stop the program.