

ICS Lab2 实验报告

Name:张展翔

Student Number: PB20111669

L程序

程序设计

设计思路如下:

本程序利用R1, R2, R3三个寄存器来分别存储F (n-3) , F (n-1) 和F (n) 三个的数值。

先对程序进行初始化, R1=F (0) =1, R2=F (1) =1, R3=F (2) =2, R0=F (0) =n, R5=1023

R0-=2即为程序需要进行循环的次数

如果此时R (0) >0,说明n>2需要进行计算

如果此时R (0) =0, 或R (0) <0,则不需要进行循环计算

- n>2时

由于 $F(n) = (F(n-1) + 2 * F(n-3)) \bmod 1024$

故先将R1翻倍, 再加上R3即可

由于此运算结果是最新的, 故将其存入R3中, 原R3数值转存R2, 原R2数值转存R1

判断R0的值是否为0, 若为0, 结束循环, 若不为0, 则继续进行循环

结束循环后, F (n) 的值即为最新更新的R3中储存的值, 再将该值和R5 (1023) 进行与运算, 即求 $\bmod (1024)$ 即可

- n=2时

n=2时, 即求F (2) 的值, 注意到此时若不进入循环, 得到的结果即为R3寄存器中的值 (2) , 故n=2时可以和n>2的情况合并

- n<2时, 即求F (0) 或F (1) ,注意到二者都等于1, 故可以进行特殊判断

综上所述, 可写出如下代码:

```
.ORIG x3000
ADD R1,R1,#1
ADD R2,R2,#1
ADD R3,R3,#2
ADD R0,R0,#-2
BRZ OUTZ
BRn OUTN
LOOP ADD R1,R1,R1
ADD R4,R1,R3
ADD R1,R2,#0
ADD R2,R3,#0
ADD R3,R4,#0
ADD R0,R0,#-1
BRp LOOP
OUTZ LD R5,MOD
AND R7,R5,R3
```

```
TRAP x25
OUTN ADD R7,R7,#1
TRAP x25
MOD .FILL x03FF
F20 .FILL #930
F11 .FILL #246
F16 .FILL #386
F69 .FILL #454
.END
```

程序分析

总代码**共25行**，去除结尾和起始地址代码，**共23行**，去除学号运算数据后，纯计算代码**共19行**

部分测试数据如下：

```
测试数据 F(1) = 1 你的回答正确，指令数 7
测试数据 F(2) = 2 你的回答正确，指令数 7
测试数据 F(3) = 4 你的回答正确，指令数 15
测试数据 F(24) = 706 你的回答正确，指令数 162
测试数据 F(144) = 642 你的回答正确，指令数 1002
测试数据 F(456) = 66 你的回答正确，指令数 3186
测试数据 F(1088) = 2 你的回答正确，指令数 7610
测试数据 F(1092) = 290 你的回答正确，指令数 7638
测试数据 F(2096) = 898 你的回答正确，指令数 14666
测试数据 F(4200) = 322 你的回答正确，指令数 29394
测试数据 F(8192) = 514 你的回答正确，指令数 57338
测试数据 F(12000) = 258 你的回答正确，指令数 83994
测试数据 F(14000) = 898 你的回答正确，指令数 97994
测试数据 F(20) = 930 你的回答正确，指令数 134
测试数据 F(11) = 246 你的回答正确，指令数 71
测试数据 F(16) = 386 你的回答正确，指令数 106
测试数据 F(69) = 454 你的回答正确，指令数 477
平均指令数 17870.647058823528
```