Course Project - Computer Architecture

180010030 *180010032* *180020006*

Multiply two numbers using Booth’s Algorithm in ARM

1.

> There are 23 Static instructions in the code.

> There are 552 Dynamic instructions in the code.

2.

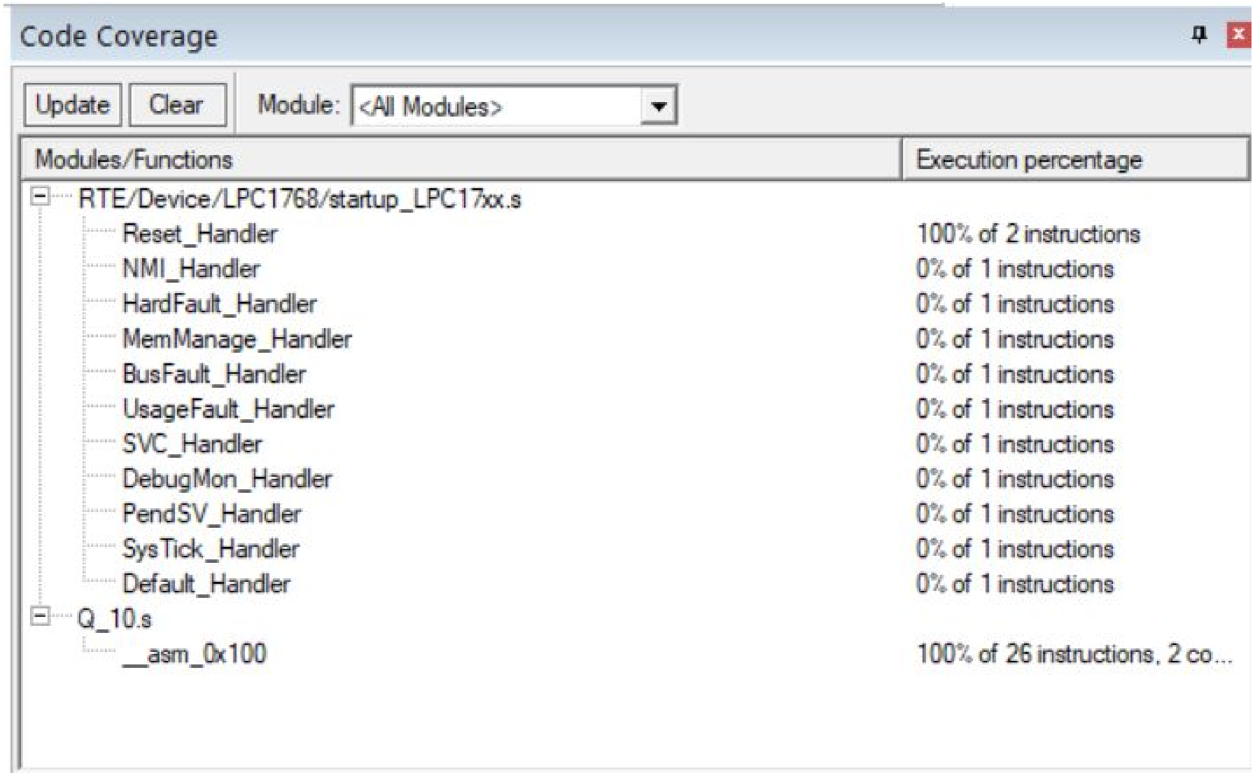
> There are 29 instructions if we don't include conditional execution.

> There are 23 instructions using conditional execution.

> There are 24 instructions if we write code in THUMB instructions.

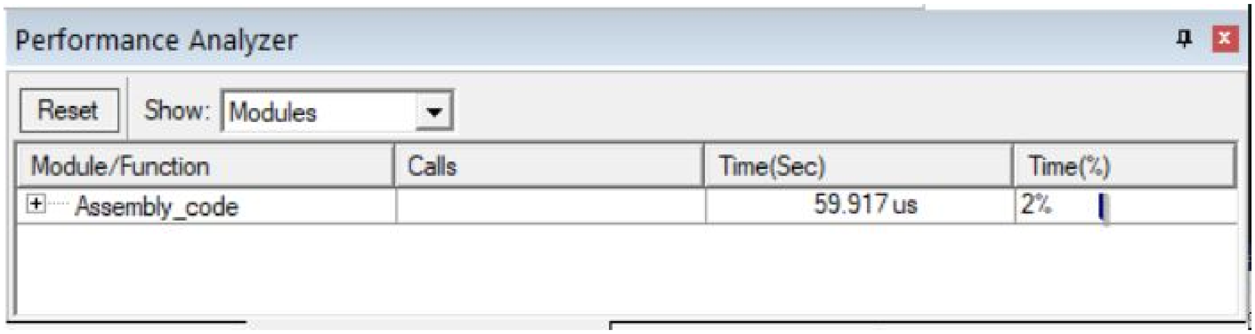
3.

>



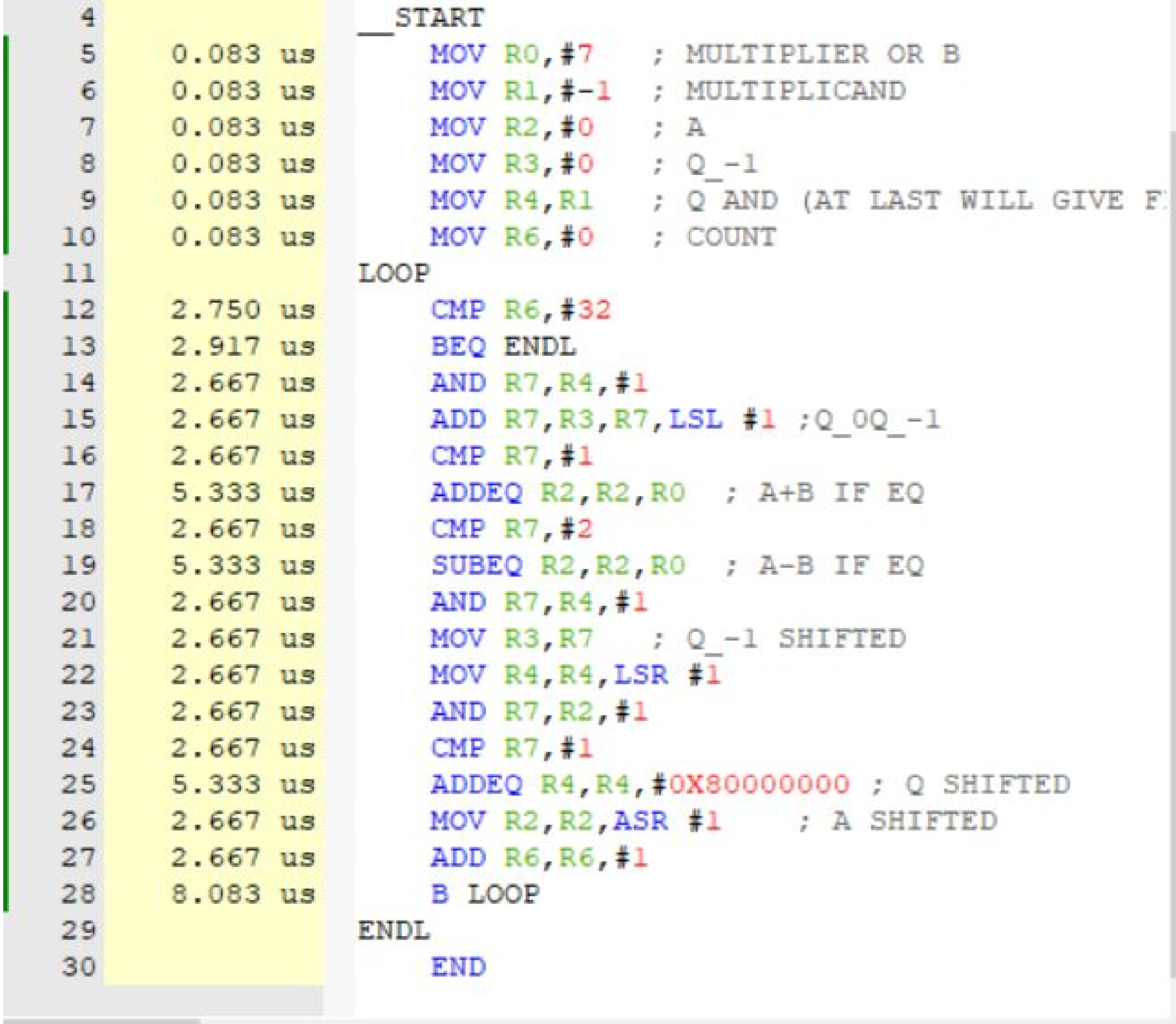
Modules/Functions	Execution percentage
RTE/Device/LPC1768/startup_LPC17xx.s	
Reset_Handler	100% of 2 instructions
NMI_Handler	0% of 1 instructions
HardFault_Handler	0% of 1 instructions
MemManage_Handler	0% of 1 instructions
BusFault_Handler	0% of 1 instructions
UsageFault_Handler	0% of 1 instructions
SVC_Handler	0% of 1 instructions
DebugMon_Handler	0% of 1 instructions
PendSV_Handler	0% of 1 instructions
SysTick_Handler	0% of 1 instructions
Default_Handler	0% of 1 instructions
Q_10.s	
__asm_0x100	100% of 26 instructions, 2 co...

> Total execution time :-



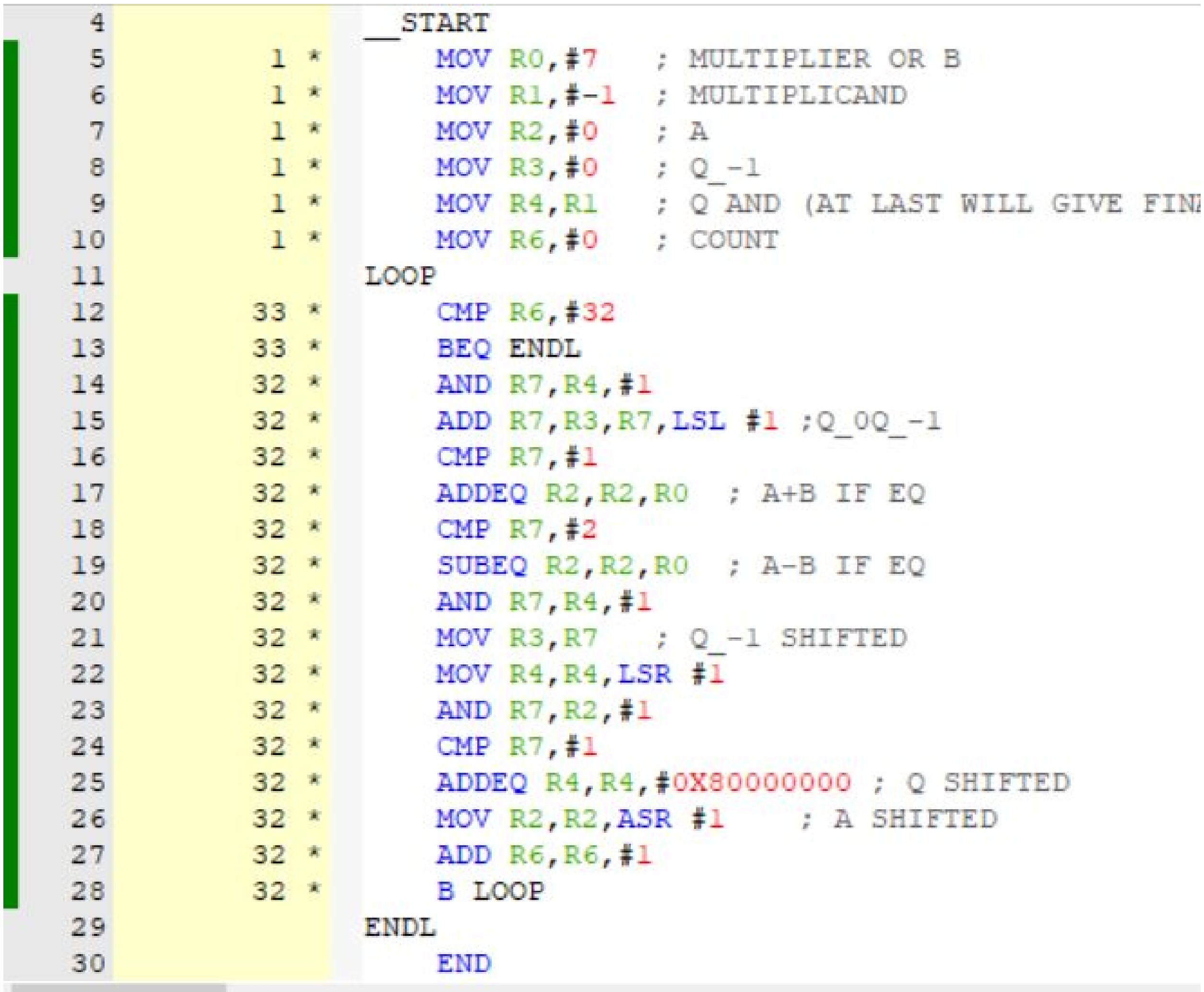
Show: Modules

Module/Function	Calls	Time(Sec)	Time(%)
⊞ Assembly_code		59.917 us	2%



```
              __START
  0.083 us        MOV R0,#7    ; MULTIPLIER OR B
  0.083 us        MOV R1,#-1   ; MULTIPLICAND
  0.083 us        MOV R2,#0    ; A
  0.083 us        MOV R3,#0    ; Q_-1
  0.083 us        MOV R4,R1    ; Q AND (AT LAST WILL GIVE F
  0.083 us        MOV R6,#0    ; COUNT
              LOOP
  2.750 us        CMP R6,#32
  2.917 us        BEQ ENDL
  2.667 us        AND R7,R4,#1
  2.667 us        ADD R7,R3,R7,LSL #1 ;Q_0Q_-1
  2.667 us        CMP R7,#1
  5.333 us        ADDEQ R2,R2,R0   ; A+B IF EQ
  2.667 us        CMP R7,#2
  5.333 us        SUBEQ R2,R2,R0   ; A-B IF EQ
  2.667 us        AND R7,R4,#1
  2.667 us        MOV R3,R7    ; Q_-1 SHIFTED
  2.667 us        MOV R4,R4,LSR #1
  2.667 us        AND R7,R2,#1
  2.667 us        CMP R7,#1
  5.333 us        ADDEQ R4,R4,#0X80000000 ; Q SHIFTED
  2.667 us        MOV R2,R2,ASR #1     ; A SHIFTED
  2.667 us        ADD R6,R6,#1
  8.083 us        B LOOP
              ENDL
                  END
```

>



```
                __START
     1 *            MOV R0,#7    ; MULTIPLIER OR B
     1 *            MOV R1,#-1   ; MULTIPLICAND
     1 *            MOV R2,#0    ; A
     1 *            MOV R3,#0    ; Q_-1
     1 *            MOV R4,R1    ; Q AND (AT LAST WILL GIVE FIN
     1 *            MOV R6,#0    ; COUNT
                LOOP
    33 *            CMP R6,#32
    33 *            BEQ ENDL
    32 *            AND R7,R4,#1
    32 *            ADD R7,R3,R7,LSL #1 ;Q_0Q_-1
    32 *            CMP R7,#1
    32 *            ADDEQ R2,R2,R0   ; A+B IF EQ
    32 *            CMP R7,#2
    32 *            SUBEQ R2,R2,R0   ; A-B IF EQ
    32 *            AND R7,R4,#1
    32 *            MOV R3,R7    ; Q_-1 SHIFTED
    32 *            MOV R4,R4,LSR #1
    32 *            AND R7,R2,#1
    32 *            CMP R7,#1
    32 *            ADDEQ R4,R4,#0X80000000 ; Q SHIFTED
    32 *            MOV R2,R2,ASR #1     ; A SHIFTED
    32 *            ADD R6,R6,#1
    32 *            B LOOP
                ENDL
                    END
```