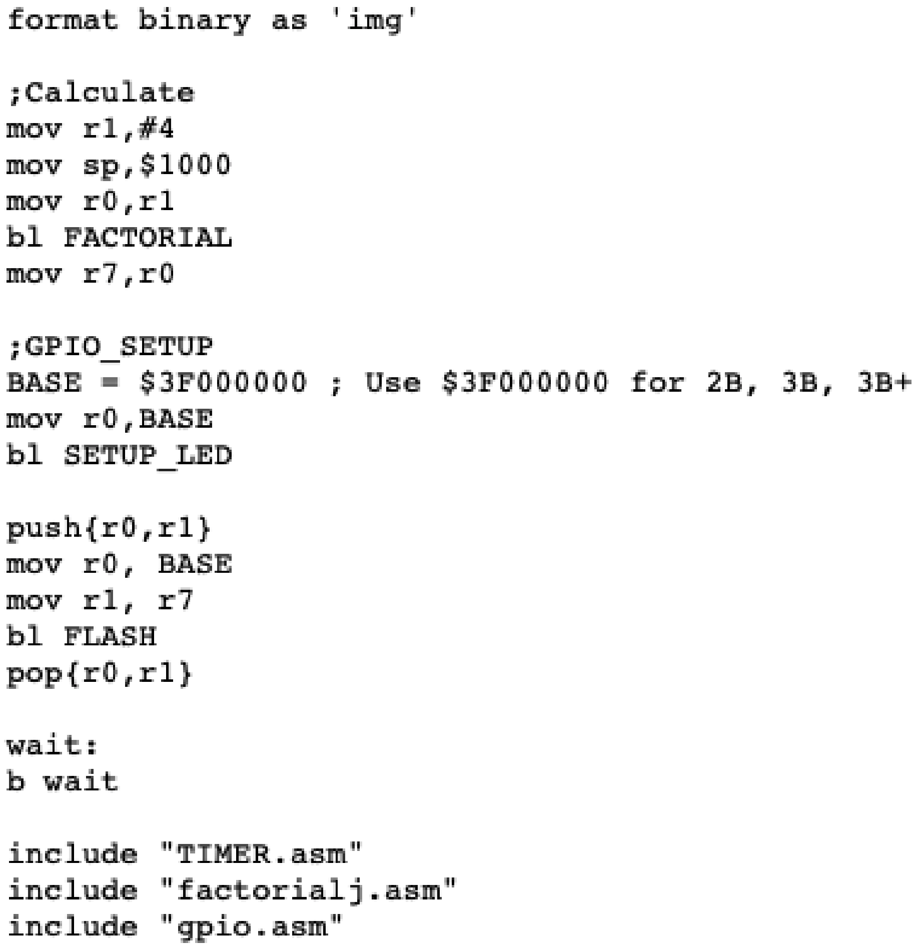
Kernel7	Kernel 7 is the file that brings together functions from all the files and execute them
GPIO	This file is the file used to send commands to the led light. GPIO.asm brings the FLASH function on the stack and removes it from the stack after 24 flashes are executed.
Timer	This file acts as a dumb timer to have pauses between the flashes
Factorial	This file turns the led flashing into a factorial function, in this case it being 4 and the led flashes 24 times. His file brings r1 to 1 and pop it off after that.

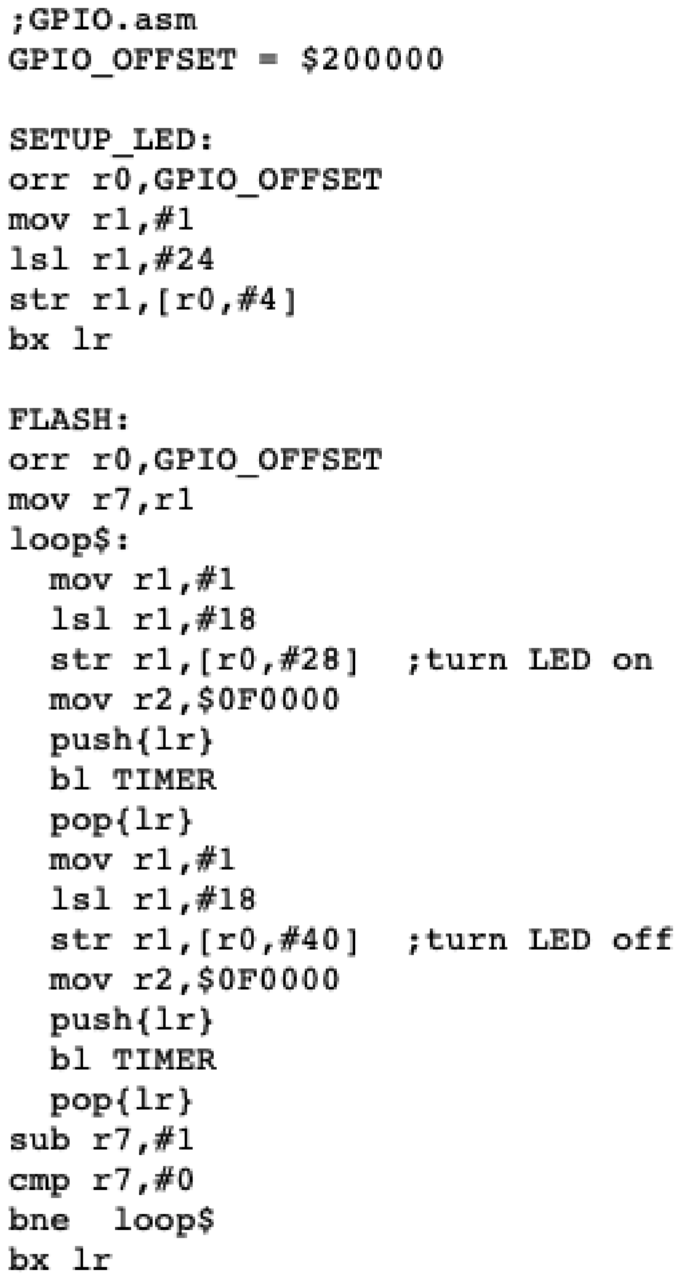
2. r1 holds the value of the program, in this case the value is 4 which will be used in factorial file to work out the number of flashes before the stack is popped out.

KERNEL7



```
format binary as 'img'

;Calculate
mov r1,#4
mov sp,$1000
mov r0,r1
bl FACTORIAL
mov r7,r0

;GPIO_SETUP
BASE = $3F000000 ; Use $3F000000 for 2B, 3B, 3B+
mov r0,BASE
bl SETUP_LED

push{r0,r1}
mov r0, BASE
mov r1, r7
bl FLASH
pop{r0,r1}

wait:
b wait

include "TIMER.asm"
include "factorialj.asm"
include "gpio.asm"
```

GPIO



```
;GPIO.asm
GPIO_OFFSET = $200000

SETUP_LED:
orr r0,GPIO_OFFSET
mov r1,#1
lsl r1,#24
str r1,[r0,#4]
bx lr

FLASH:
orr r0,GPIO_OFFSET
mov r7,r1
loop$:
  mov r1,#1
  lsl r1,#18
  str r1,[r0,#28]  ;turn LED on
  mov r2,$0F0000
  push{lr}
  bl TIMER
  pop{lr}
  mov r1,#1
  lsl r1,#18
  str r1,[r0,#40]  ;turn LED off
  mov r2,$0F0000
  push{lr}
  bl TIMER
  pop{lr}
sub r7,#1
cmp r7,#0
bne  loop$
bx lr
```