

- Main.o sections

```

MINGW64:/d/Courses/programming/NOW/EmbeddedSystems/Now/DIPLOMAS/KSESDiplo...
AS/KSESDiploma/5.Assignments/Unit 3 (Embedded C)/Lecture 3/Assignment 2 (toggle
led using startup.s)/source code (master)
$ arm-none-eabi-objdump.exe -h main.o

main.o:      file format elf32-littlearm

Sections:
Idx Name          Size      VMA           LMA           File off  Algn
  0 .text          00000007c  00000000  00000000  00000034  2**2
    CONTENTS, ALLOC, LOAD, READONLY, CODE
  1 .data          00000004  00000000  00000000  000000b0  2**2
    CONTENTS, ALLOC, LOAD, DATA
  2 .bss           00000000  00000000  00000000  000000b4  2**0
    ALLOC
  3 .rodata        00000004  00000000  00000000  000000b4  2**2
    CONTENTS, ALLOC, LOAD, READONLY, DATA
  4 .debug_info    000000c5  00000000  00000000  000000b8  2**0
    CONTENTS, RELOC, READONLY, DEBUGGING
  5 .debug_abbrev  000000a0  00000000  00000000  0000017d  2**0
    CONTENTS, READONLY, DEBUGGING
  6 .debug_loc     00000038  00000000  00000000  0000021d  2**0
    CONTENTS, READONLY, DEBUGGING
  7 .debug_aranges 00000020  00000000  00000000  00000255  2**0
    CONTENTS, RELOC, READONLY, DEBUGGING
  8 .debug_line    00000056  00000000  00000000  00000275  2**0
    CONTENTS, RELOC, READONLY, DEBUGGING
  9 .debug_str     00000140  00000000  00000000  000002cb  2**0
    CONTENTS, READONLY, DEBUGGING
10 .comment       0000007f  00000000  00000000  0000040b  2**0
    CONTENTS, READONLY
11 .debug_frame   0000002c  00000000  00000000  0000048c  2**2
    CONTENTS, RELOC, READONLY, DEBUGGING
12 .ARM.attributes 00000033  00000000  00000000  000004b8  2**0
    CONTENTS, READONLY

```

- Startup.o sections

```

MINGW64:/d/Courses/programming/NOW/EmbeddedSystems/Now/DIPLOMAS/KSESDiplo...
bdu1@Administrator MINGW64 /d/Courses/programming/NOW/EmbeddedSystems/Now/DIPLO
AS/KSESDiploma/5.Assignments/Unit 3 (Embedded C)/Lecture 3/Assignment 2 (toggle
led using startup.s)/source code (master)
$ arm-none-eabi-objdump.exe -h startup.o

startup.o:   file format elf32-littlearm

Sections:
Idx Name          Size      VMA           LMA           File off  Algn
  0 .text          00000008  00000000  00000000  00000034  2**1
    CONTENTS, ALLOC, LOAD, RELOC, READONLY, CODE
  1 .data          00000000  00000000  00000000  0000003c  2**0
    CONTENTS, ALLOC, LOAD, DATA
  2 .bss           00000000  00000000  00000000  0000003c  2**0
    ALLOC
  3 .vectors       00000050  00000000  00000000  0000003c  2**0
    CONTENTS, RELOC, READONLY
  4 .debug_line    0000003b  00000000  00000000  0000008c  2**0
    CONTENTS, RELOC, READONLY, DEBUGGING
  5 .debug_info    00000026  00000000  00000000  000000c7  2**0
    CONTENTS, RELOC, READONLY, DEBUGGING
  6 .debug_abbrev  00000014  00000000  00000000  000000ed  2**0
    CONTENTS, READONLY, DEBUGGING
  7 .debug_aranges 00000020  00000000  00000000  00000108  2**3
    CONTENTS, RELOC, READONLY, DEBUGGING
  8 .debug_str     000000bf  00000000  00000000  00000128  2**0
    CONTENTS, READONLY, DEBUGGING
  9 .ARM.attributes 00000021  00000000  00000000  000001e7  2**0
    CONTENTS, READONLY

```

- application.elf sections

```

MINGW64/d/Courses/programming/NOW/EmbeddedSystems/Now/DIPLOMAS/KSES Diplo...
abdu1@Administrator MINGW64 /d/Courses/programming/NOW/EmbeddedSystems/Now/DIPLO
MAS/KSES Diploma/5.Assignments/Unit 3 (Embedded C)/Lecture 3/Assignment 2 (toggle
led using startup.s)/source code (master)
$ arm-none-eabi-objdump.exe -h toggleLed.elf

toggleLed.elf:      file format elf32-littlearm

Sections:
Idx Name              Size      VMA               LMA               File-off          Align
0 .text               000000d8  08000000  08000000  00010000  2**2
CONTENTS, ALLOC, LOAD, READONLY, CODE
1 .data               00000004  080000d8  080000d8  000100d8  2**2
CONTENTS, ALLOC, LOAD, DATA
2 .bss                00000004  20000000  20000000  00020000  2**2
ALLOC
3 .debug_info         000000eb  00000000  00000000  000100dc  2**0
CONTENTS, READONLY, DEBUGGING
4 .debug_abbrev       000000b4  00000000  00000000  000101c7  2**0
CONTENTS, READONLY, DEBUGGING
5 .debug_loc          00000038  00000000  00000000  0001027b  2**0
CONTENTS, READONLY, DEBUGGING
6 .debug_aranges     00000040  00000000  00000000  000102b8  2**3
CONTENTS, READONLY, DEBUGGING
7 .debug_line         00000091  00000000  00000000  000102f8  2**0
CONTENTS, READONLY, DEBUGGING
8 .debug_str          00000154  00000000  00000000  00010389  2**0
CONTENTS, READONLY, DEBUGGING
9 .comment            0000007e  00000000  00000000  000104dd  2**0
CONTENTS, READONLY
10 .ARM.attributes    00000031  00000000  00000000  0001055b  2**0
CONTENTS, READONLY
11 .debug_frame        0000002c  00000000  00000000  0001058c  2**2
CONTENTS, READONLY, DEBUGGING

```

- Symbols

```

MINGW64/d/Courses/programming/NOW/EmbeddedSystems/Now/DIPLOMAS/KSES Diplo...
abdu1@Administrator MINGW64 /d/Courses/programming/NOW/EmbeddedSystems/Now/DIPLO
MAS/KSES Diploma/5.Assignments/Unit 3 (Embedded C)/Lecture 3/Assignment 2 (toggle
led using startup.s)/source code (master)
$ arm-none-eabi-nm.exe main.o
00000004 C bss
00000000 D data
00000000 T main
00000000 R rodata

abdu1@Administrator MINGW64 /d/Courses/programming/NOW/EmbeddedSystems/Now/DIPLO
MAS/KSES Diploma/5.Assignments/Unit 3 (Embedded C)/Lecture 3/Assignment 2 (toggle
led using startup.s)/source code (master)
$ arm-none-eabi-nm.exe startup.o
U main
00000000 t reset
00000006 t vector_handler

abdu1@Administrator MINGW64 /d/Courses/programming/NOW/EmbeddedSystems/Now/DIPLO
MAS/KSES Diploma/5.Assignments/Unit 3 (Embedded C)/Lecture 3/Assignment 2 (toggle
led using startup.s)/source code (master)
$ arm-none-eabi-nm.exe toggleLed.elf
20000000 B bss
080000d8 D data
08000050 T main
080000cc t reset
080000d4 T rodata
080000d2 t vector_handler

abdu1@Administrator MINGW64 /d/Courses/programming/NOW/EmbeddedSystems/Now/DIPLO
MAS/KSES Diploma/5.Assignments/Unit 3 (Embedded C)/Lecture 3/Assignment 2 (toggle
led using startup.s)/source code (master)
$

```

# • Simulation

leLed - Proteus 8 Professional - Schematic Capture

it View Tool Design Graph Debug Library Template System Help

Base Design

Schematic Capture

DEVICES

- LED-RED
- STM32
- STM32F103C6

U1

PA0-WKUP  
PA1  
PA2  
PA3  
PA4  
PA5  
PA6  
PA7  
PA8  
PA9  
PA10  
PA11  
PA12  
PA13  
PA14  
PA15  
PB0  
PB1  
PB2  
PB3  
PB4  
PB5  
PB6  
PB7  
PB8  
PB9  
PB10  
PB11  
PB12  
PB13  
PB14  
PB15

NRST

PC13\_RTC  
PC14-OSC32\_IN  
PC15-OSC32\_OUT  
OSCIN\_Pb0  
OSCCOUT\_Pb1  
VBAT  
BOOT0

D2  
LED-RED

STM32F103C6

CM3 Source Code - U1

```
----- typedef volatile unsigned int vuInt32_t;
----- #define RCC_BASE      0x40021000
----- #define PORTA          0x40010800
----- #define APB2ENR        *((vuInt32_t*)(RCC_BASE + 0x18))
----- #define CRH              *((vuInt32_t*)(PORTA + 0x04))
----- #define ODR              *((vuInt32_t*)(PORTA + 0x0C))
----- #define IOPAEN2         2
----- #define GPIOA13         13
----- int data = 10;
----- int const rodota = 10;
----- int bss;

void main()
{
    APB2ENR |= (1<<IOPAEN2);
    CRH &= 0xFF0FFFFF;
    CRH |= 0x00200000;
    /* Loop forever */
    while(1){
        ODR |= (1<<GPIOA13);
        for(int i = 0; i<50000; i++);
        ODR &= ~(1<<GPIOA13);
        for(int i = 0; i<50000; i++);
    }
}
```

CM3 FLASH at 0x08000000 - U1

08000000	00 10 00 20 CC 00 00 08	.....
08000008	D3 00 00 08 D3 00 00 08	.....
08000010	D3 00 00 08 D3 00 00 08	.....
08000018	D3 00 00 08 D3 00 00 08	.....
08000020	D3 00 00 08 D3 00 00 08	.....
08000028	D3 00 00 08 D3 00 00 08	.....
08000030	D3 00 00 08 D3 00 00 08	.....
08000038	D2 00 00 08 D3 00 00 08	.....
08000040	D3 00 00 08 D3 00 00 08	.....
08000048	D3 00 00 08 D3 00 00 08	.....
08000050	80 54 63 80 00 AF 1A 48	.....K
08000058	1B 68 19 4A 43 F0 04 03	.....h.JC...
08000060	13 60 18 48 1B 68 17 4A	.....K.h.J
08000068	23 F4 70 03 13 60 15 48	.....h.JC...
08000070	1B 68 14 4A 43 F4 00 13	.....h.JC...
08000078	13 60 13 48 1B 68 12 4A	.....K.h.J
08000080	43 F4 00 53 13 60 00 23	.....C..S..#
08000088	7B 60 02 E0 7B 68 01 33	.....[.h.3
08000090	7B 60 7B 68 4C F2 4F 32	.....[.hL.02
08000098	93 42 F7 00 0A 4B 18 68	......B..K.h
080000A0	09 4A 23 F4 00 53 13 60	......3#..S.
080000A8	00 23 3B 60 02 E0 3B 68	.....#...2h
080000B0	01 33 3B 60 3B 68 4C F2	......3 !hL
080000B8	4F 32 93 42 F7 00 DC E7	.....02.B...
080000C0	1B 10 02 40 04 08 01 40	........@...6
080000C8	0C 08 01 40 FF F7 C0 FF	........@...
080000D0	FE E7 F8 E7 0A 00 00 00	........
080000D8	0A 00 00 00 00 00 00 00	........
080000E0	00 00 00 00 00 00 00 00	........
080000E8	00 00 00 00 00 00 00 00	........
080000F0	00 00 00 00 00 00 00 00	........
080000F8	00 00 00 00 00 00 00 00	........

CM3 RAM at 0x20000000 - U1

20000F40	00 00 00 00 00 00 00 00	.....
20000F48	00 00 00 00 00 00 00 00	.....
20000F50	00 00 00 00 00 00 00 00	.....
20000F58	00 00 00 00 00 00 00 00	.....
20000F60	00 00 00 00 00 00 00 00	.....
20000F68	00 00 00 00 00 00 00 00	.....
20000F70	00 00 00 00 00 00 00 00	.....
20000F78	00 00 00 00 00 00 00 00	.....
20000F80	00 00 00 00 00 00 00 00	.....
20000F88	00 00 00 00 00 00 00 00	.....
20000F90	00 00 00 00 00 00 00 00	.....
20000F98	00 00 00 00 00 00 00 00	.....
20000FA0	00 00 00 00 00 00 00 00	.....
20000FA8	00 00 00 00 00 00 00 00	.....
20000FB0	00 00 00 00 00 00 00 00	.....
20000FB8	00 00 00 00 00 00 00 00	.....
20000FC0	00 00 00 00 00 00 00 00	.....
20000FC8	00 00 00 00 00 00 00 00	.....
20000FD0	50 C3 00 00 50 C3 00 00	.....P...P...
20000FD8	00 00 00 00 00 00 00 00	.....
20000FE0	00 00 00 00 00 00 00 00	.....
20000FE8	00 00 00 00 00 00 00 00	.....
20000FF0	00 00 00 FF FF FF FF FF	.....
20000FF8	00 00 00 08 00 00 00 00	.....
20010000	00 00 00 00 00 00 00 00	.....
20010008	00 00 00 00 00 00 00 00	.....
20010010	00 00 00 00 00 00 00 00	.....
20010018	00 00 00 00 00 00 00 00	.....
20010020	00 00 00 00 00 00 00 00	.....

3 Message(s) Selects current assembly variant.