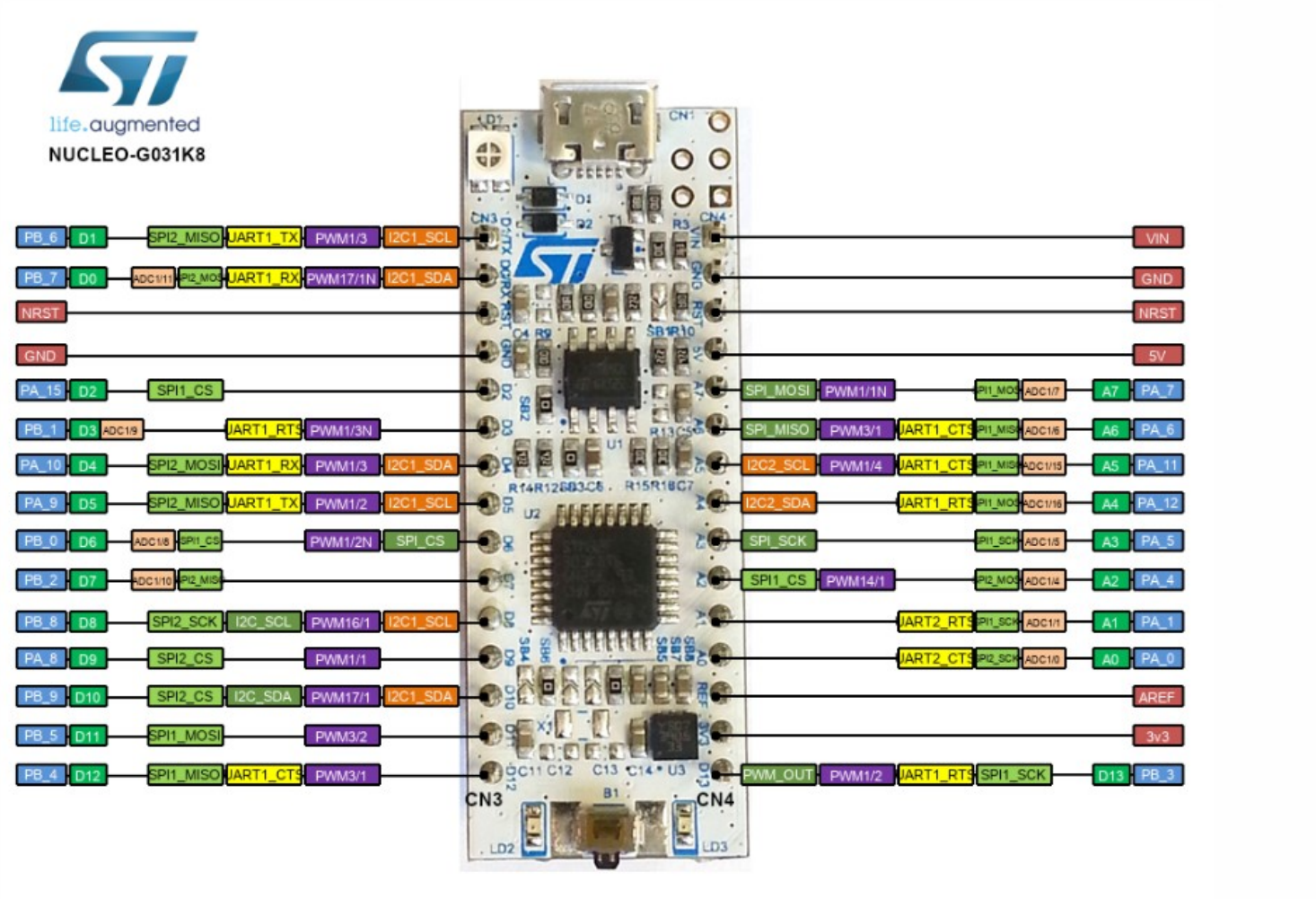
GEBZE TECHNİCAL UNIVERTİSY

ELEC 335

LAB 1 REPORT

BERKAY TÜRK

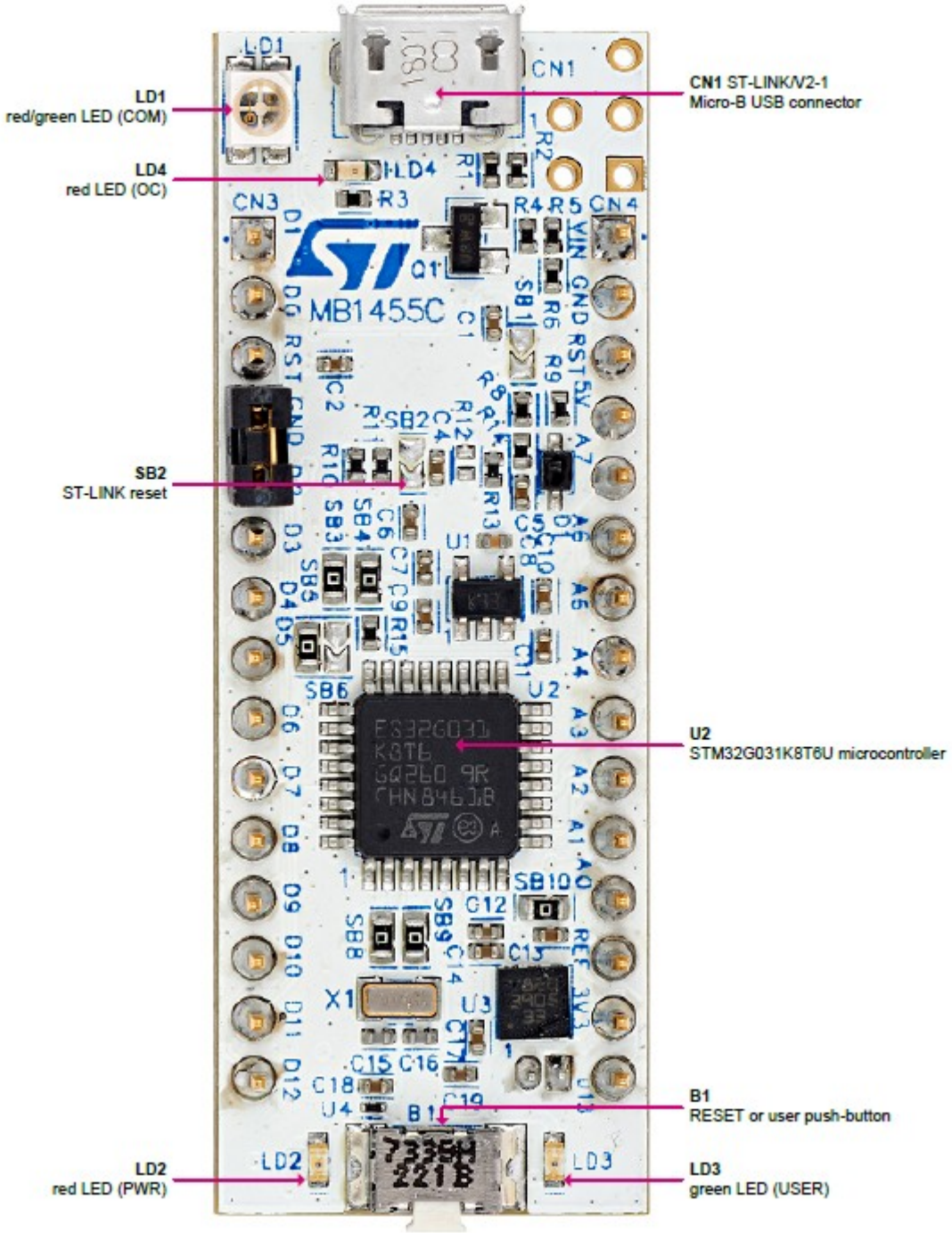
171024024



• STM32G031K8T6U microcontroller (Arm® Cortex®-M0+ at 64 MHz) in LQFP32 package, featuring 64 Kbytes of Flash memory and 8 Kbytes of SRAM
• 1 user LED
• 1 RESET or user push-button
• Board connectors:
– Arduino™ Nano V3 expansion connector
– USB with Micro-B
• Flexible power-supply options: ST-LINK USB VBUS or external sources
• On-board ST-LINK/V2-1 debugger/programmer with USB re-enumeration capability: mass storage, Virtual COM port, and debug port

ICs:

Microcontroller (CPU): The STM32G031x4/x6/x8 mainstream microcontrollers are based on high-performance Arm® Cortex®-M0+ 32-bit RISC core operating at up to 64 MHz frequency. Offering a high level of integration, they are suitable for a wide range of applications in consumer, industrial and appliance domains and ready for the Internet of Things (IoT) solutions. The devices incorporate a memory protection unit (MPU), high-speed embedded memories (8 Kbytes of SRAM and up to 64 Kbytes of Flash program memory with read protection, write protection, proprietary code protection, and securable area), DMA, an extensive range of system functions, enhanced I/Os, and peripherals.


LD1
red/green LED (COM)
LD4
red LED (OC)
SB2
ST-LINK reset
LD2
red LED (PWR)
CN1 ST-LINK/V2-1
Micro-B USB connector
U2
STM32G031K8T6U microcontroller
B1
RESET or user push-button
LD3
green LED (USER)
MB1455C

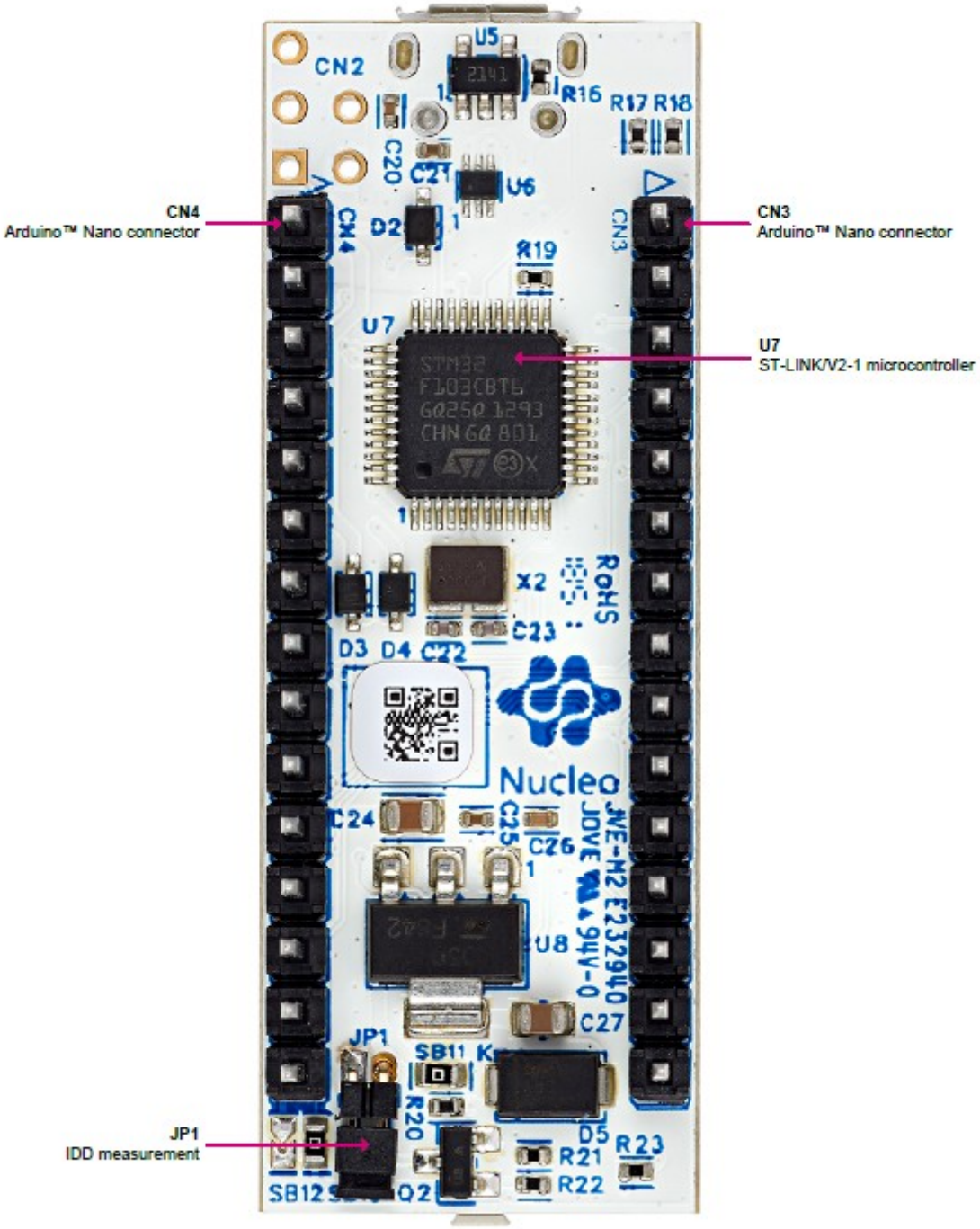


Table 6. External power sources

Input power name	Connector pin	Voltage range	Max current	Limitation
VIN	CN4 pin 1	7 V to 12 V	800mA	From 7 V to 12 V only and input current capability is linked to input voltage: • 800 mA input current when VIN = 7 V • 450 mA input current when 7 V < VIN < 9 V • 300 mA input current when 10 V > VIN > 9 V • less than 300 mA input current when VIN > 10 V
+5V	CN4 pin 4	4.75 V to 5.25 V	500mA	-
+3V3	CN4 pin 14	3 V to 3.6 V	-	ST-LINK not powered and SB2 and SB3 must be OFF.

Pin Connections:

VIN: Input voltage on any other pin. Min=-0.3V,Max=4.0V

GND: (GROUND)

NRST: It resets chip in other words.İn short, In short, NRST pin is used to activate bootloader when you need to flash new program using one of the communication channels and to reset MCU to default state when you want to flash new program using SWD interface and SWD pins are not available.

5V: It provides the system's power.

It manages the Flash memory reprogramming through USART on pins PA9/PA10 or PA2/PA3, or through I2C-bus on pins PB6/PB7 or PB10/PB11.

+3V3 power supply
Using the +3V3 (CN4 pin 14) directly as power input is interesting, for instance if the 3.3 V is provided by a shield board. In this case, the ST-LINK is not powered, thus programming and debugging features are not available.
When the board is powered by +3V3 (CN4 pin 14), solder bridges SB2 (NRST) and SB3 must be OFF.

LD1 ST-LINK COM LED
The bicolor LED LD1 (green, red) provides information about ST-LINK communication status. LD1 default color is red. LD1 turns to green to indicate that communication is in progress between the PC and the ST-LINK, with the following setup:
• Blinking red: the first USB enumeration with the PC is taking place
• Red on: the initialization between the PC and ST-LINK is complete
• Blinking red or green: programming and debugging with target
• Green on: communication finished and successful
• Orange on: communication failure

LD2 PWR
The red LED indicates that the STM32G0 part is powered and 5 V power is available on CN4 pin 4.

LD3 USER
The LD3 USER green LED is connected to the following STM32G031K8T6 I/O:
• PB3, if the configuration is SB12 ON, and SB13 OFF
• PC6, if the configuration is SB12 OFF, and SB13 ON (default configuration)
It is also connected to the Arduino™ D13 signal when SB12 is ON. To light this LED, a high-logic state "1" must be written in the corresponding GPIO PB3 or PC6. A transistor drives the LED, therefore its consumption does not affect the STM32G0 power measurement.

LD4 USB power fault (OC, overcurrent)
The LD4 red LED indicates that the board power consumption on USB ST-LINK exceeds 500 mA. Therefore, the user must check the root cause of the overconsumption, and consequently power the STM32G0 Nucleo-32 board with an external power supply if needed.

B1 RESET/USER
This push-button is connected to NRST (PF2-NRST) and is used to reset the STM32G0 microcontroller or to generate a USER event.

Virtual COM port (VCP): USART
The STM32G0 Nucleo-32 board offers the possibility to connect a USART interface to the ST-LINK/V2-1.

Figure 1. Block diagram

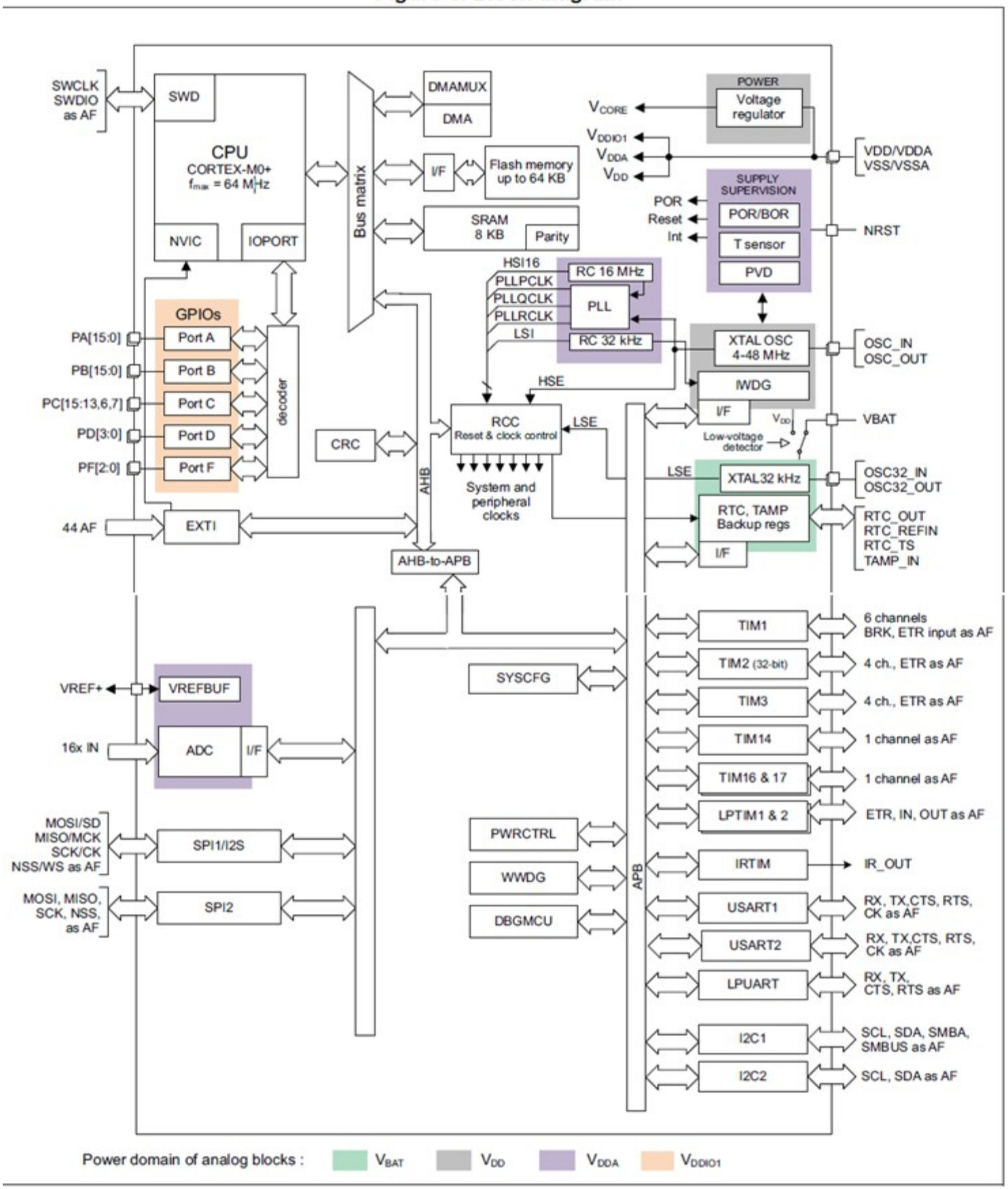


Figure 2. Memory map

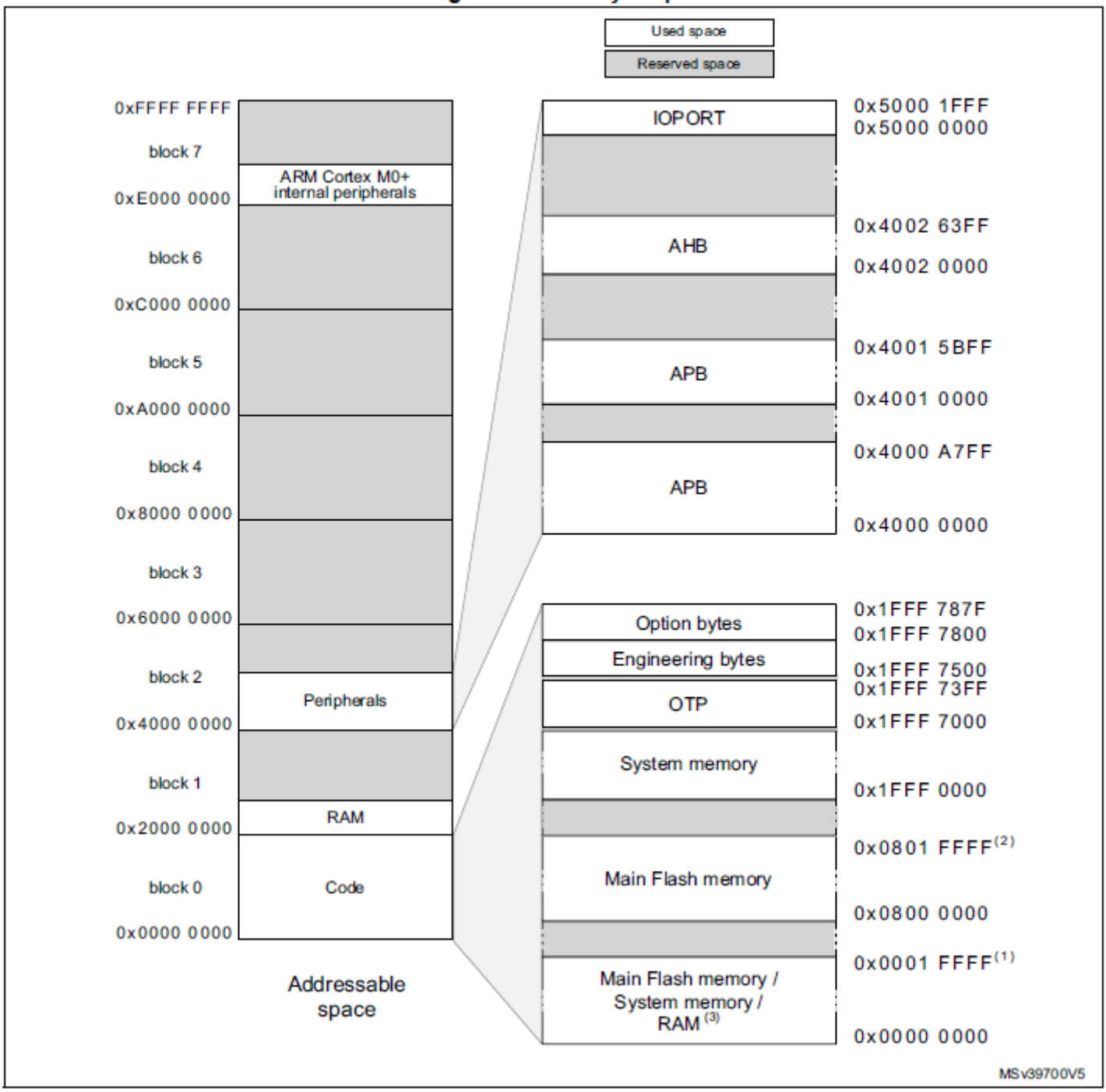


Table 4. STM32G0x1 peripheral register boundary addresses

Bus	Boundary address	Size	Peripheral	Peripheral register map
-	0xE000 0000 - 0xE00F FFFF	1MB	Cortex®-M0+ internal peripherals	-
IOPORT	0x5000 1800 - 0x5FFF FFFF	~256 MB	Reserved	-
	0x5000 1400 - 0x5000 17FF	1 KB	GPIOF	*Section 6.4.12 on page 211*
	0x5000 1000 - 0x5000 13FF	1 KB	Reserved	-
	0x5000 0C00 - 0x5000 0FFF	1 KB	GPIOD	*Section 6.4.12 on page 211*
	0x5000 0800 - 0x5000 0BFF	1 KB	GPIOC	*Section 6.4.12 on page 211*
	0x5000 0400 - 0x5000 07FF	1 KB	GPIOB	*Section 6.4.12 on page 211*
	0x5000 0000 - 0x5000 03FF	1 KB	GPIOA	*Section 6.4.12 on page 211*

These bits are read-only. They contain the input value of the corresponding I/O port.

6.4.6 GPIO port output data register (GPIOx_ODR) (x = A to D, F)

Address offset: 0x14

Reset value: 0x0000 0000

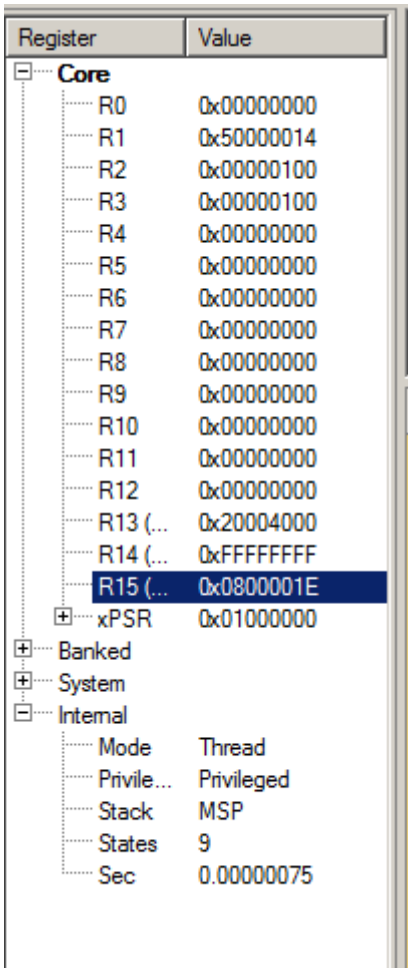
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OD15	OD14	OD13	OD12	OD11	OD10	OD9	OD8	OD7	OD6	OD5	OD4	OD3	OD2	OD1	OD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 Reserved, must be kept at reset value.

Bits 15:0 **OD[15:0]:** Port output data I/O pin y (y = 15 to 0)
These bits can be read and written by software.
Note: For atomic bit set/reset, the OD bits can be individually set and/or reset by writing to the GPIOx_BSRR register (x = A..D, F).

CODE 1:

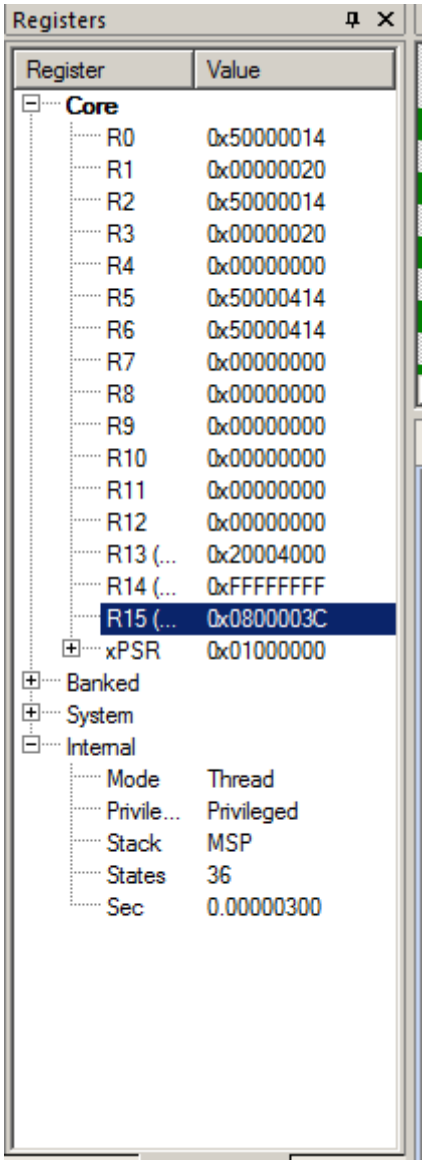
```
Stack_Size      EQU     0x00004000

                AREA    STACK, NOINIT, READWRITE, ALIGN=3
Stack_Mem       SPACE   Stack_Size
__initial_sp

    THUMB

    AREA    RESET, DATA, READONLY
    EXPORT  __Vectors

__Vectors
    DCD     __initial_sp                    ; Top of Stack
    DCD     Reset_Handler                   ; Reset Handler
    DCD     NMI_Handler                     ; NMI Handler
    DCD     HardFault_Handler               ; Hard Fault Handler

    AREA    |.text|, CODE, READONLY
```




```
; nmi handler
NMI_Handler    PROC
    EXPORT  NMI_Handler
    B .
    ENDP



; hardfault handler
HardFault_Handler    PROC
    EXPORT  HardFault_Handler
    B .
    ENDP



; entry function
Reset_Handler    PROC
    EXPORT  Reset_Handler
    ; Edit below this line
     GPIOA_ODR       equ 0x50000014

    LDR r1 ,=GPIOA_ODR
     LDR r3,[r1]
     LDR r2 ,=0x100
     ORRS r3,r3 ,r2
     STR r3,[r1]
    ; Edit above this line
    B .
    ENDP

    END
```

CODE 2:

```
Stack_Size      EQU     0x00004000

                AREA    STACK, NOINIT, READWRITE, ALIGN=3
Stack_Mem       SPACE   Stack_Size
__initial_sp


    THUMB


    AREA    RESET, DATA, READONLY
    EXPORT  __Vectors
__Vectors
    DCD     __initial_sp                    ; Top of Stack
    DCD     Reset_Handler                   ; Reset Handler
    DCD     NMI_Handler                     ; NMI Handler
    DCD     HardFault_Handler               ; Hard Fault Handler
    AREA    |.text|, CODE, READONLY
; nmi handler
NMI_Handler    PROC
    EXPORT  NMI_Handler
    B .
    ENDP
; hardfault handler
HardFault_Handler    PROC
    EXPORT  HardFault_Handler
    B .
    ENDP
; entry function
Reset_Handler    PROC
    EXPORT  Reset_Handler
    ; Edit below this line
```




```
GPIOA_ODR       equ 0x50000014
GPIOB_ODR       equ 0x50000414

    LDR r0 ,=GPIOA_ODR
     LDR r3,[r0]
     LDR r1 ,=0x800
     ORRS r3,r3 ,r1
     STR r3,[r0]

     LDR r2 ,=GPIOA_ODR
     LDR r3,[r2]
     LDR r1 ,=0x1000
     ORRS r3,r3 ,r1
     STR r3,[r2]

     LDR r5 ,=GPIOB_ODR
     LDR r3,[r5]
     LDR r1 ,=0x10
     ORRS r3,r3 ,r1
     STR r3,[r5]

     LDR r6 ,=GPIOB_ODR
     LDR r3,[r6]
     LDR r1 ,=0x20
     ORRS r3,r3 ,r1
     STR r3,[r6]

    ; Edit above this line
    B .
    ENDP

    END
```

CODE 3:

```
Stack_Size      EQU     0x00004000

                AREA    STACK, NOINIT, READWRITE, ALIGN=3
Stack_Mem       SPACE   Stack_Size
__initial_sp


    THUMB


    AREA    RESET, DATA, READONLY
    EXPORT  __Vectors
__Vectors
    DCD     __initial_sp                ; Top of Stack
    DCD     Reset_Handler               ; Reset Handler
    DCD     NMI_Handler                 ; NMI Handler
    DCD     HardFault_Handler           ; Hard Fault Handler
    AREA    |.text|, CODE, READONLY
; nmi handler
NMI_Handler    PROC
    EXPORT  NMI_Handler
    B .
    ENDP
; hardfault handler
HardFault_Handler    PROC
    EXPORT  HardFault_Handler
    B .
    ENDP
; entry function
Reset_Handler    PROC
    EXPORT  Reset_Handler
    ; Edit below this line
```

```
GPIOA_ODR       equ (0x50000014)


loop1


    LDR r1 ,=GPIOA_ODR
     LDR r3,[r1]


     LDR r2 ,=0x100


     ORRS r3,r3 ,r2
     STR r3,[r1]
     LDR r1 ,=GPIOA_ODR
     LDR r3,[r1]
     LDR r2 ,=0x0
    ANDS r3,r3,r2       // RESET VALUE
     STR r3,[r1]
     ;delay:
     MOVS r0,#100
loop2
     SUBS r0,#1
     BNE loop2
    MOVS r4,#10     ;LED 10 KERE 1 ER SANIYE ARALIKLARLA YANACAKTIR.
     SUBS r4,#1
    BNE loop1


    ; Edit above this line
    B .
    ENDP


    END
```