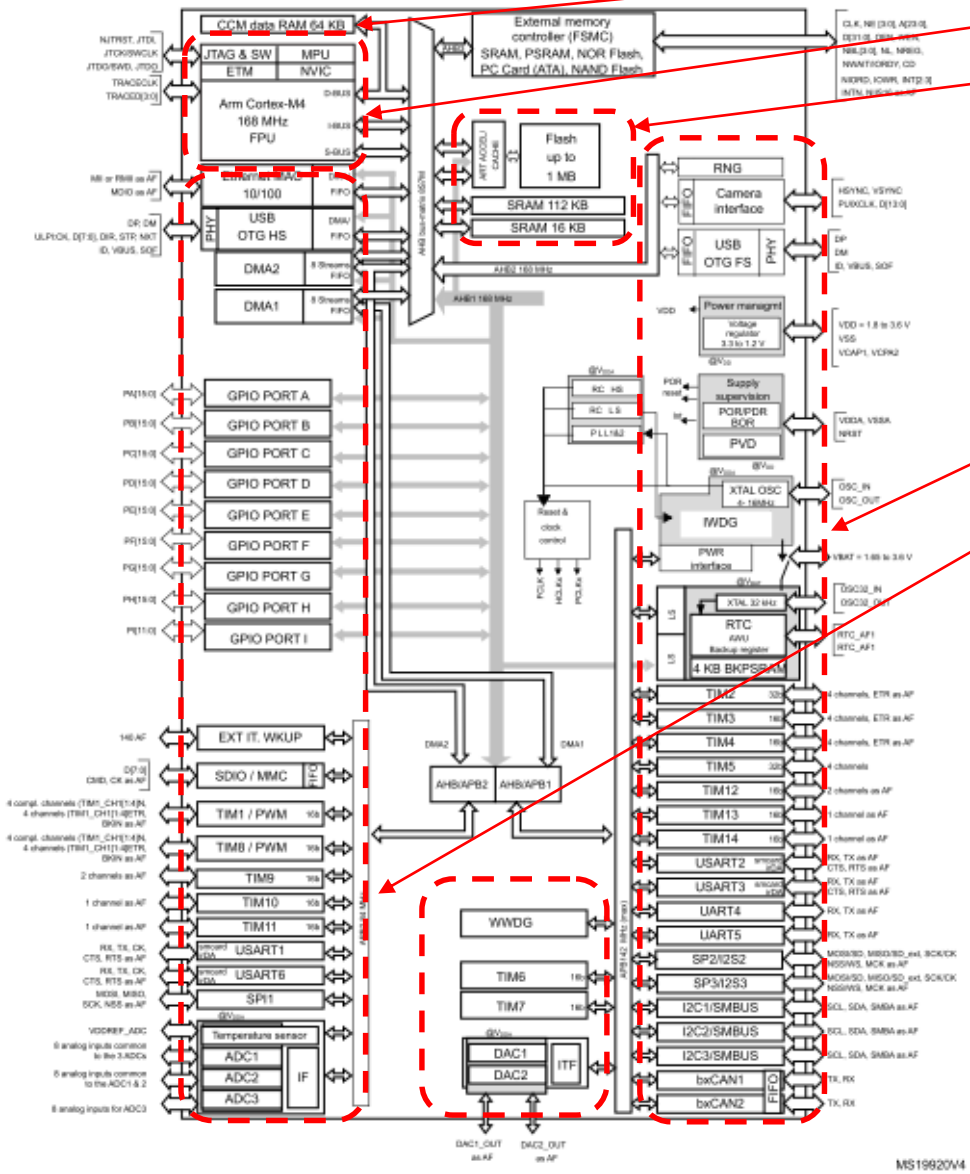
STM32F407 Discovery

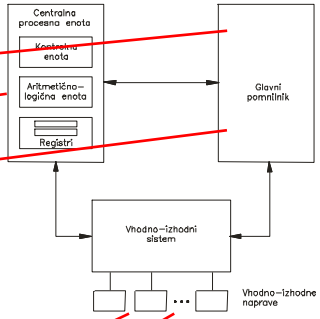
Vhodno / izhodne naprave

USART+DMA Serijska komunikacija

z uporabo DMA krmilnika

STM32F407VG





Delo na STM32F4 razvojnem sistemu

Priključitev :

- **Mini USB** priklop na **krajši stranici**, svetita rdeči **LED** diodi

Poseben začetni projekt za *STM32F4* (e-učilnica) :

- ***dodajanje vsebine (template.s):***

'template.s - STM32CubeIDE

avigate Search Project Run Window Help

template.s

```
54
55 _start:
56     // Enable GPIOD Peripheral Clock (bit 3 in AHB1ENR register)
57     ldr r6, = RCC_AHB1ENR       // Load peripheral clock reg address to r6
58     ldr r5, [r6]                // Read its content to r5
59     orr r5, #0x00000008          // Set bit 3 to enable GPIOD clock
60     str r5, [r6]                // Store result in peripheral clock register
61
62     // Make GPIOD Pin12 as output pin (bits 25:24 in MODER register)
63     ldr r6, = GPIOD_MODER       // Load GPIOD MODER register address to r6
64     ldr r5, [r6]                // Read its content to r5
65     bic r5, #0x3000000          // Clear bits 24, 25 for P12
66     orr r5, #0x01000000         // Write 01 to bits 24, 25 for P12
67     str r5, [r6]                // Store result in GPIOD MODER register
68
69     // Set GPIOD Pin12 to 1 (bit 12 in ODR register)
70     ldr r6, = GPIOD_ODR         // Load GPIOD output data register
71     ldr r5, [r6]                // Read its content to r5
72     orr r5, #0x1000              // write 1 to pin 12
73     str r5, [r6]                // Store result in GPIOD output data register
74
75     // Set GPIOD Pin12 to 0 (bit 12 in ODR register)
76     ldr r6, = GPIOD_ODR         // Load GPIOD output data register
77     ldr r5, [r6]                // Read its content to r5
78     bic r5, #0x1000             // write 0 to pin 12
79     str r5, [r6]                // Store result in GPIOD output data register
80
81 loop:
82     nop                         // No operation. Do nothing.
83     b loop                      // Jump to loop
84
```

STM32 CubeIDE, STM32F4 (izbrana dokumentacij

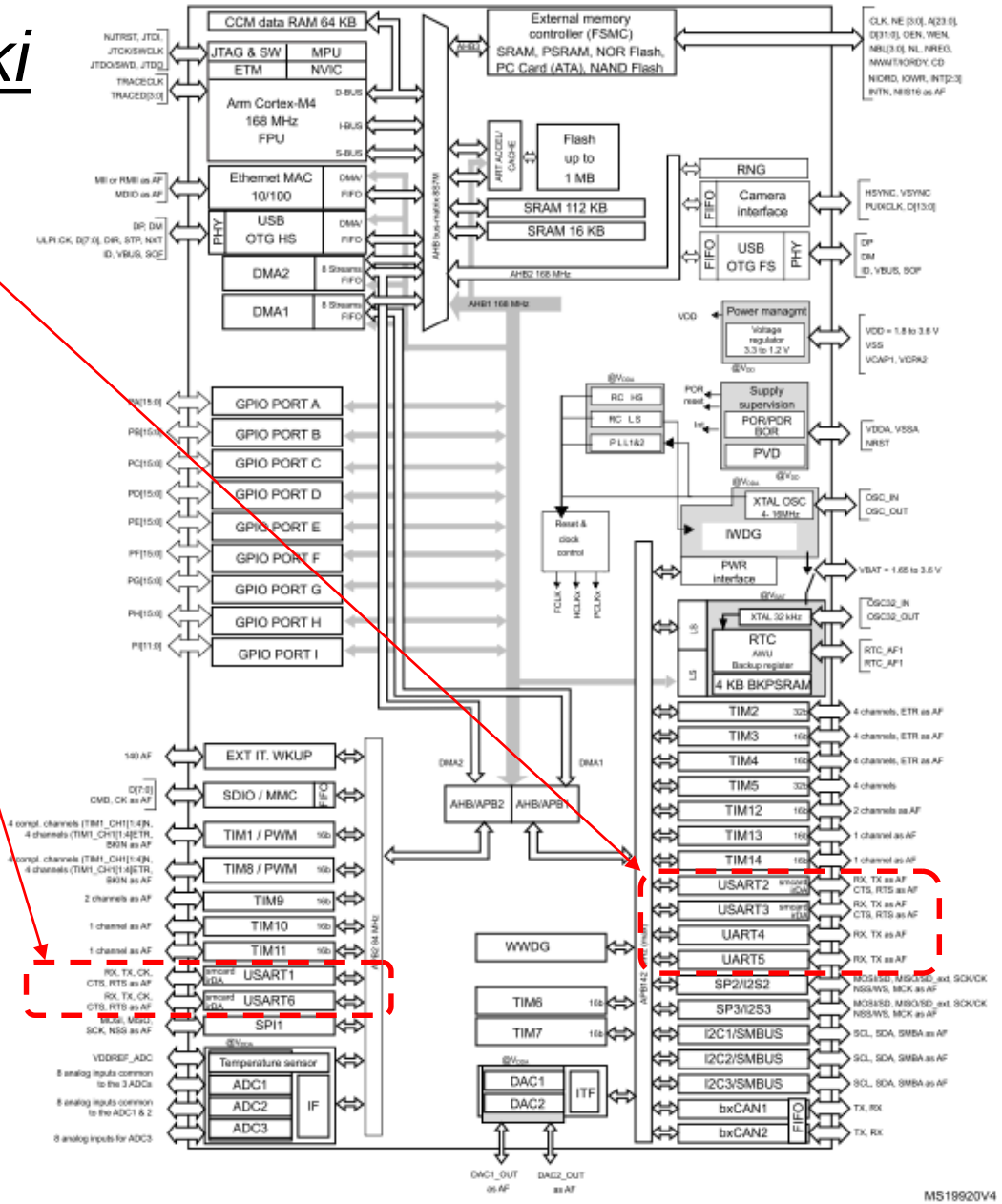
--------- Razvojni sistem ----------

- STM32 CubeIDE
- ORLab-STM32 - GitHub repozitorij
- User Manual Discovery kit stm32f407vg Uploaded 8/11/21, 12.58
- DataSheet_stm32f407vg Uploaded 8/11/21, 12.56
- Reference Manual rm0090-stm32f407417 Uploaded 8/11/21, 12.57
- Programming_Manual_pm0214-stm32-cortexm4-mcus-and-mpu
- Arm Cortex-M4 Processor Datasheet Short Uploaded 29/10/21, 15.00

--------- Cortex-M arhitektura, zbirnik ----------

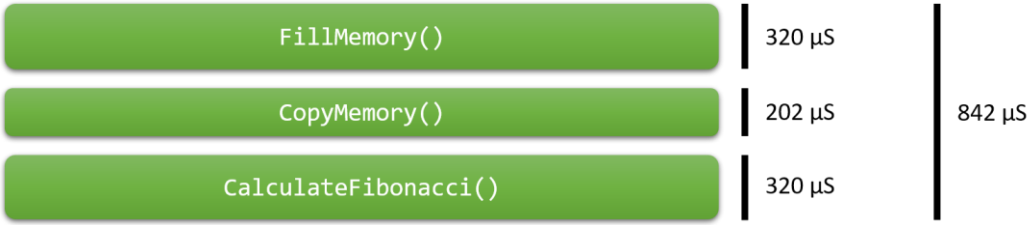
- ARM Cortex-M for Beginners ARM 2017 Uploaded 29/10/21, 14.50

Serijski vmesniki

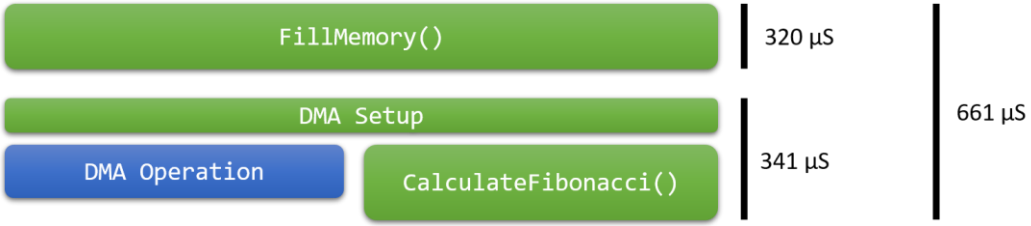


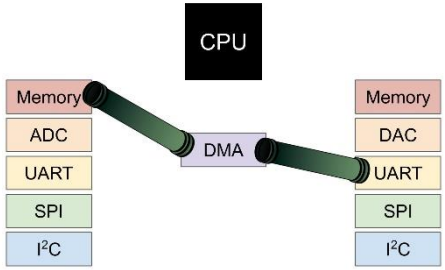
DMA- Splošno

In this example, filling the first and the third buffer took the exactly the same time, while copying the first buffer to the second one took slightly less time:



While the DMA cannot be used to compute Fibonacci numbers, or initialize arrays with non-constant values, it can be used for copying data between 2 memory locations.
Now the DMA operation ran in parallel with the CalculateFibonacci() function, reducing the overall program time by 21%:





The results are in:

80 DMA CRCs per second.
63 manual CRCs per second

On my processor, **DMA gives a 27% advantage over iterative memory assignment**. I think it is because everything is done with a hardware mover that doesn't have to increment, involve registers, gotos, branch less than, and so on.

DMA - STM32

Figure 7. STM32F405/415 and STM32F407/417 system architecture

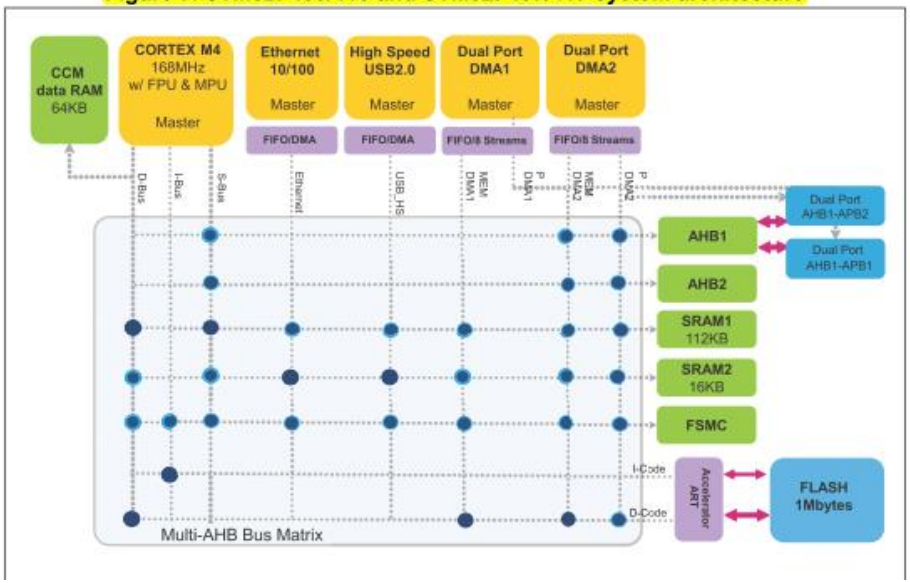
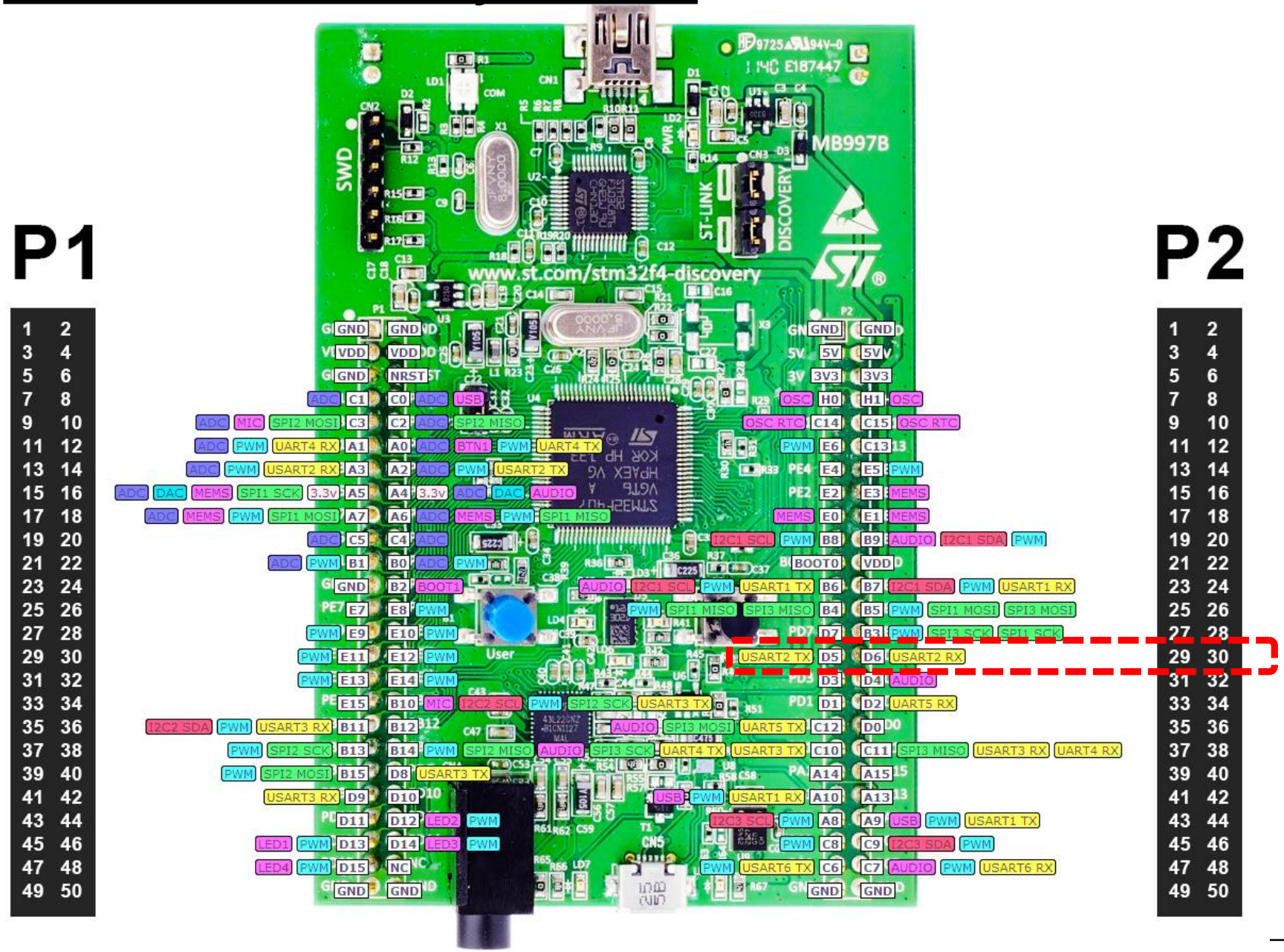


Table 42. DMA1 request mapping

Peripheral requests	Stream 0	Stream 1	Stream 2	Stream 3	Stream 4	Stream 5	Stream 6	Stream 7
Channel 0	SPI3_RX	-	SPI3_RX	SPI2_RX	SPI2_TX	SPI3_TX	-	SPI3_TX
Channel 1	I2C1_RX	-	TIM7_UP	-	TIM7_UP	I2C1_RX	I2C1_TX	I2C1_TX
Channel 2	TIM4_CH1	-	I2S3_EXT_RX	TIM4_CH2	I2S2_EXT_TX	I2S3_EXT_TX	TIM4_UP	TIM4_CH3
Channel 3	I2S3_EXT_RX	TIM2_UP TIM2_CH3	I2C3_RX	I2S2_EXT_RX	I2C3_TX	TIM2_CH1	TIM2_CH2 TIM2_CH4	TIM2_UP TIM2_CH4
Channel 4	UART5_RX	USART3_RX	UART4_RX	USART3_TX	UART4_TX	USART2_RX	USART2_TX	UART5_TX
Channel 5	UART8_TX(1)	UART7_TX(1)	TIM3_CH4 TIM3_UP	UART7_RX(1)	TIM3_CH1 TIM3_TRIG	TIM3_CH2	UART8_RX(1)	TIM3_CH3

STM32F4 Discovery -Pinout



Vira: Reference manual & Datasheet

life.augmented

RM0090
Reference manual

STM32F405/415, STM32F407/417, STM32F427/437 and
STM32F429/439 advanced Arm®-based 32-bit MCUs

RM0090 Universal synchronous asynchronous receiver transmitter (USART)

30 Universal synchronous asynchronous receiver transmitter (USART)

30.3.13 Continuous communication using DMA

DMA controller (DMA) RM0090

10 DMA controller (DMA)

DMA *(Registri za nastavitve delovanja)*

6.3.10 RCC AHB1 peripheral clock register (RCC_AHB1ENR)

Address offset: 0x30

Reset value: 0x0010 0000

Access: no wait state, word, half-word and byte access.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reser-ved	OTGHS ULPIEN	OTGHSEN	ETHMACPTPEN	ETHMACRXEN	ETHMACTXEN	ETHMACEN	Res.	**DMA2DEN**	DMA2EN	DMA1EN	CCMDATARAMEN	Res.	BKPSRAMEN	Reserved	
	rw	rw	rw	rw	rw	rw		rw	rw	rw			rw		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			CRCEN	Res.	GPIOKEN	GPIOJEN	GPIOIEN	GPIOHEN	GPIOGEN	GPIOFEN	GPIOEEN	GPIODEN	GPIOCEN	GPIOBEN	GPIOAEN
			rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit 22 **DMA2EN:** DMA2 clock enable
This bit is set and cleared by software.
0: DMA2 clock disabled
1: DMA2 clock enabled

Bit 21 **DMA1EN:** DMA1 clock enable
This bit is set and cleared by software.
0: DMA1 clock disabled
1: DMA1 clock enabled

DMA *(Registri za nastavitve delovanja)*

30.6.6 Control register 3 (USART_CR3)

Address offset: 0x14

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				ONEBIT	CTSIE	CTSE	RTSE	DMAT	DMAR	SCEN	NACK	HDSEL	IRLP	IREN	EIE
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit 7 **DMAT**: DMA enable transmitter

This bit is set/reset by software

1: DMA mode is enabled for transmission.

0: DMA mode is disabled for transmission.

Bit 6 **DMAR**: DMA enable receiver

This bit is set/reset by software

1: DMA mode is enabled for reception

0: DMA mode is disabled for reception

DMA (Registri za nastavitve delovanja)

Table 1. STM32F4xx register boundary addresses (continued)

Boundary address	Peripheral	Bus	Register map
0x5006 0800 - 0x5006 0BFF	RNG	AHB2	Section 24.4.4: RNG register map on page 771
0x5006 0400 - 0x5006 07FF	HASH		Section 25.4.9: HASH register map on page 795
0x5006 0000 - 0x5006 03FF	CRYP		Section 23.6.13: CRYP register map on page 763
0x5005 0000 - 0x5005 03FF	DCMI		Section 15.8.12: DCMI register map on page 478
0x5000 0000 - 0x5003 FFFF	USB OTG FS		Section 34.16.6: OTG_FS register map on page 1326
0x4004 0000 - 0x4007 FFFF	USB OTG HS		Section 35.12.6: OTG_HS register map on page 1472
0x4002 B000 - 0x4002 BBFF	DMA2D		Section 11.5: DMA2D registers on page 352
0x4002 8000 - 0x4002 93FF	ETHERNET MAC		Section 33.8.5: Ethernet register maps on page 1236
0x4002 6400 - 0x4002 67FF	DMA2		Section 10.5.11: DMA register map on page 335
0x4002 6000 - 0x4002 63FF	DMA1		

Table 42. DMA1 request mapping

Peripheral requests	Stream 0	Stream 1	Stream 2	Stream 3	Stream 4	Stream 5	Stream 6	Stream 7
Channel 0	SPI3_RX	-	SPI3_RX	SPI2_RX	SPI2_TX	SPI3_TX	-	SPI3_TX
Channel 1	I2C1_RX	-	TIM7_UP	-	TIM7_UP	I2C1_RX	I2C1_TX	I2C1_TX
Channel 2	TIM4_CH1	-	I2S3_EXT_RX	TIM4_CH2	I2S2_EXT_TX	I2S3_EXT_TX	TIM4_UP	TIM4_CH3
Channel 3	I2S3_EXT_RX	TIM2_UP TIM2_CH3	I2C3_RX	I2S2_EXT_RX	I2C3_TX	TIM2_CH1	TIM2_CH2 TIM2_CH4	TIM2_UP TIM2_CH4
Channel 4	UART5_RX	USART3_RX	UART4_RX	USART3_TX	UART4_TX	USART2_RX	USART2_TX	UART5_TX
Channel 5	UART8_TX(1)	UART7_TX(1)	TIM3_CH4 TIM3_UP	UART7_RX(1)	TIM3_CH1 TIM3_TRIG	TIM3_CH2	UART8_RX(1)	TIM3_CH3

DMA *(Registri za nastavitve delovanja)*

10.5.7 DMA stream x peripheral address register (DMA_SxPAR) (x = 0..7)

Address offset: 0x18 + 0x18 × *stream number*

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PAR[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PAR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:0 **PAR[31:0]**: Peripheral address

Base address of the peripheral data register from/to which the data will be read/written.

These bits are write-protected and can be written only when bit EN = '0' in the DMA_SxCR register.

0x04	**USART_DR**
	Reset value

10.5.8 DMA stream x memory 0 address register (DMA_SxM0AR) (x = 0..7)

Address offset: 0x1C + 0x18 × *stream number*

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
M0A[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
M0A[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:0 **M0A[31:0]**: Memory 0 address

Base address of Memory area 0 from/to which the data will be read/written.

These bits are write-protected. They can be written only if:

- the stream is disabled (bit EN= '0' in the DMA_SxCR register) or
- the stream is enabled (EN='1' in DMA_SxCR register) and bit CT = '1' in the DMA_SxCR register (in Double buffer mode).

Naslov v pomnilniku

DMA (Registri za nastavitve delovanja)

10.5.5 DMA stream x configuration register (DMA_SxCR) (x = 0..7)

This register is used to configure the concerned stream.

Address offset: 0x10 + 0x18 × *stream number*

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				CHSEL[2:0]			MBURST [1:0]		PBURST[1:0]		Reser-ved	CT	DBM	PL[1:0]	
				rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PINCOS	MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR[1:0]		PFCTRL	TCIE	HTIE	TEIE	DMEIE	EN
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 7:6 **DIR[1:0]**: Data transfer direction
These bits are set and cleared by software.
00: Peripheral-to-memory
01: Memory-to-peripheral
10: Memory-to-memory
11: reserved
These bits are protected and can be written only if EN is '0'.

it 0 **EN**: Stream enable / flag stream ready when read low
This bit is set and cleared by software.
0: Stream disabled
1: Stream enabled
This bit may be cleared by hardware:
- on a DMA end of transfer (stream ready to be configured)
- if a transfer error occurs on the AHB master buses
- when the FIFO threshold on memory AHB port is not compatible with the size of the burst

When this bit is read as 0, the software is allowed to program the Configuration and FIFO bits registers. It is forbidden to write these registers when the EN bit is read as 1.

Bits 27:25 **CHSEL[2:0]**: Channel selection
These bits are set and cleared by software.
000: channel 0 selected
001: channel 1 selected
010: channel 2 selected
011: channel 3 selected
100: channel 4 selected
101: channel 5 selected
110: channel 6 selected
111: channel 7 selected
These bits are protected and can be written only if EN is '0'

Bits 24:23 **MBURST**: Memory burst transfer configuration
These bits are set and cleared by software.
00: single transfer
01: INCR4 (incremental burst of 4 beats)
10: INCR8 (incremental burst of 8 beats)
11: INCR16 (incremental burst of 16 beats)
These bits are protected and can be written only if EN is '0'
In direct mode, these bits are forced to 0x0 by hardware as soon as bit EN= '1'

Bits 22:21 **PBURST[1:0]**: Peripheral burst transfer configuration
These bits are set and cleared by software.
00: single transfer
01: INCR4 (incremental burst of 4 beats)
10: INCR8 (incremental burst of 8 beats)
11: INCR16 (incremental burst of 16 beats)
These bits are protected and can be written only if EN is '0'
In direct mode, these bits are forced to 0x0 by hardware.

Bit 20 Reserved, must be kept at reset value.

Bits 14:13 **MSIZE[1:0]**: Memory data size
These bits are set and cleared by software.
00: byte (8-bit)
01: half-word (16-bit)
10: word (32-bit)
11: reserved
These bits are protected and can be written only if EN is '0'.
In direct mode, MSIZE is forced by hardware to the same value as PSIZE as soon as bit EN = '1'.

Bits 12:11 **PSIZE[1:0]**: Peripheral data size
These bits are set and cleared by software.
00: Byte (8-bit)
01: Half-word (16-bit)
10: Word (32-bit)
11: reserved
These bits are protected and can be written only if EN is '0'

Bit 10 **MINC**: Memory increment mode
This bit is set and cleared by software.
0: Memory address pointer is fixed
1: Memory address pointer is incremented after each data transfer (increment is done according to MSIZE)
This bit is protected and can be written only if EN is '0'.

Bit 9 **PINC**: Peripheral increment mode
This bit is set and cleared by software.
0: Peripheral address pointer is fixed
1: Peripheral address pointer is incremented after each data transfer (increment is done according to PSIZE)
This bit is protected and can be written only if EN is '0'.

DMA *(Registri za nastavitve delovanja)*

10.5.2 DMA high interrupt status register (DMA_HISR)

Address offset: 0x04

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				TCIF7	HTIF7	TEIF7	DMEIF7	Reserved	FEIF7	TCIF6	HTIF6	TEIF6	DMEIF6	Reserved	FEIF6
				r	r	r	r		r	r	r	r	r		r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				TCIF5	HTIF5	TEIF5	DMEIF5	Reserved	FEIF5	TCIF4	HTIF4	TEIF4	DMEIF4	Reserved	FEIF4
				r	r	r	r		r	r	r	r	r		r

Bits 31:28, 15:12 Reserved, must be kept at reset value.

Bits 27, 21, 11, 5 **TCIFx**: Stream x transfer complete interrupt flag (x=7..4)

This bit is set by hardware. It is cleared by software writing 1 to the corresponding bit in the DMA_HIFCR register.

0: No transfer complete event on stream x

1: A transfer complete event occurred on stream x

Stream 5	Stream 6
SPI3_TX	-
I2C1_RX	I2C1_TX
I2S3_EXT_TX	TIM4_UP
TIM2_CH1	TIM2_CH2 TIM2_CH4
USART2_RX	USART2_TX

10.5.4 DMA high interrupt flag clear register (DMA_HIFCR)

Address offset: 0x0C

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				CTCIF7	CHTIF7	CTEIF7	CDMEIF7	Reserved	CFEIF7	CTCIF6	CHTIF6	CTEIF6	CDMEIF6	Reserved	CFEIF6
				w	w	w	w		w	w	w	w	w		w

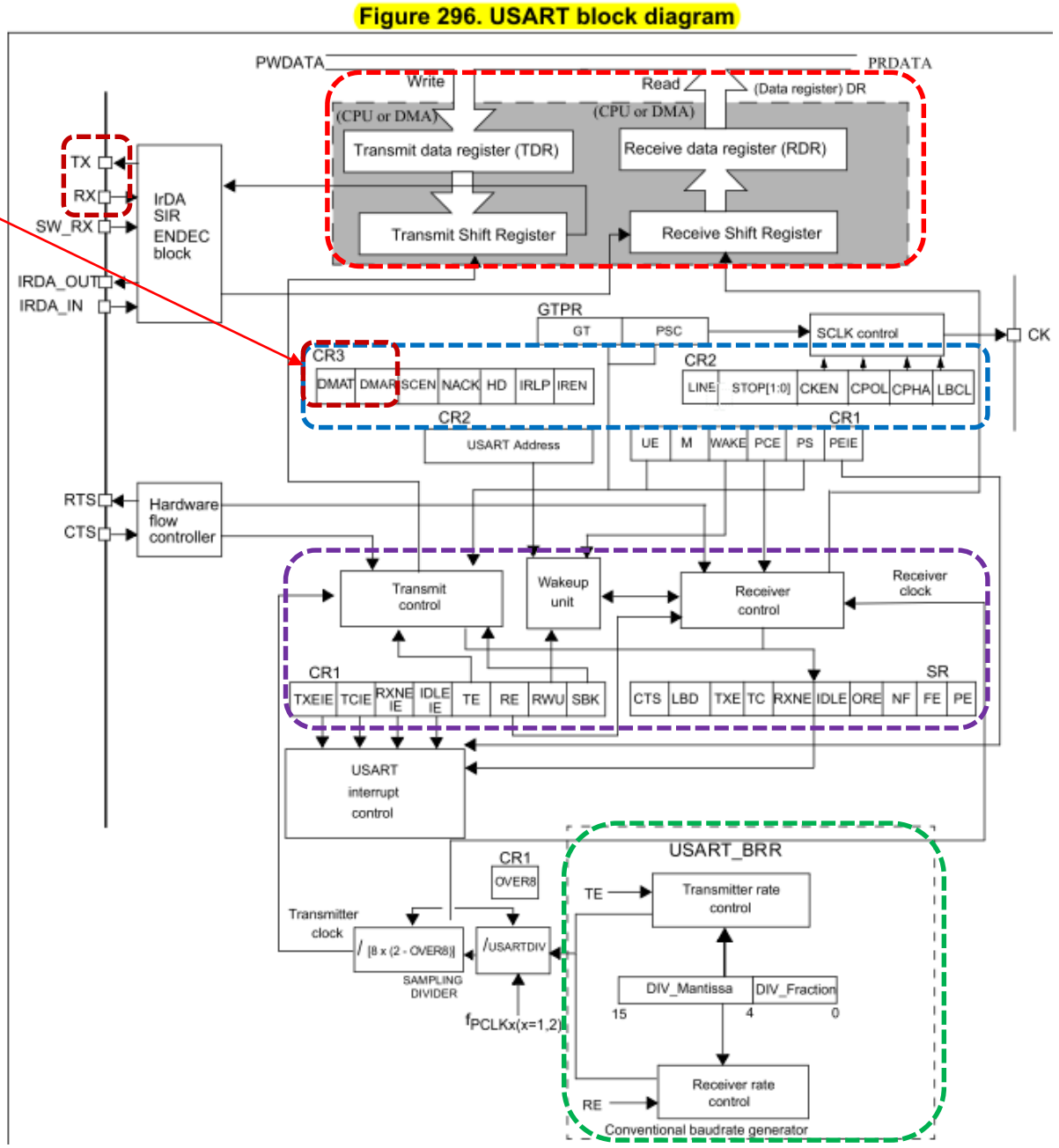
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				CTCIF5	CHTIF5	CTEIF5	CDMEIF5	Reserved	CFEIF5	CTCIF4	CHTIF4	CTEIF4	CDMEIF4	Reserved	CFEIF4
				w	w	w	w		w	w	w	w	w		w

Bits 31:28, 15:12 Reserved, must be kept at reset value.

Bits 27, 21, 11, 5 **CTCIFx**: Stream x clear transfer complete interrupt flag (x = 7..4)

Writing 1 to this bit clears the corresponding TCIFx flag in the DMA_HISR register

USART+DMA shema



USART – stanje , nastavitve

Table 149. USART register map and reset values

Offset	Register	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x14	USART_CR3	Ctrl3 Reserved reg.																				ONEBIT	CTSIE	CTSE	RTSE	DMAT	DMAR	SCEN	NACK	HDSEL	IRLP	IREN	EIE
	Reset value																					0	0	0	0	0	0	0	0	0	0	0	0

Dodatni register za delovanje USART+DMA naprav:

USART_CR3 : Control Register 3

ENABLE USART DMA Transmission (DMAT) and Reception (DMAR)

DMA – stanje , nastavitve

Table 51. DMA register map and reset values

Offset	Register	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	DMA_LISR	Reserved				TCIF3	HTIF3	TEIF3	DMEIF3	Reserved	FEIF3	TCIF2	HTIF2	TEIF2	DMEIF2	Reserved	FEIF2	Reserved				TCIF1	HTIF1	TEIF1	DMEIF1	Reserved	FEIF1	TCIF0	HTIF0	TEIF0	DMEIF0	Reserved	FEIF0
	Reset value					0	0	0	0		0	0	0	0	0		0					0	0	0	0		0	0	0	0	0		0
0x0004	DMA_HISR	Reserved				TCIF7	HTIF7	TEIF7	DMEIF7	Reserved	FEIF7	TCIF6	HTIF6	TEIF6	DMEIF6	Reserved	FEIF6	Reserved				TCIF5	HTIF5	TEIF5	DMEIF5	Reserved	FEIF5	TCIF4	HTIF4	TEIF4	DMEIF4	Reserved	FEIF4
	Reset value					0	0	0	0		0	0	0	0	0		0					0	0	0	0		0	0	0	0	0		0
0x0008	DMA_LIFCR	Reserved				CTCIF3	CHTIF3	TEIF3	CDMEIF3	Reserved	CFEIF3	CTCIF2	CHTIF2	CTEIF2	CDMEIF2	Reserved	CFEIF2	Reserved				CTCIF1	CHTIF1	CTEIF1	CDMEIF1	Reserved	CFEIF1	CTCIF0	CHTIF0	CTEIF0	CDMEIF0	Reserved	CFEIF0
	Reset value					0	0	0	0		0	0	0	0	0		0					0	0	0	0		0	0	0	0	0		0
0x000C	DMA_HIFCR	Reserved				CTCIF7	CHTIF7	CTEIF7	CDMEIF7	Reserved	CFEIF7	CTCIF6	CHTIF6	CTEIF6	CDMEIF6	-	CFEIF6	Reserved				CTCIF5	CHTIF5	CTEIF5	CDMEIF5	-	CFEIF5	CTCIF4	CHTIF4	CTEIF4	CDMEIF4	-	CFEIF4
	Reset value					0	0	0	0		0	0	0	0	0	-	0					0	0	0	0	-	0	0	0	0	0	-	0
0x0010	DMA_S0CR	Reserved				CHSEL[2:0]			MBURST[1:0]		PBURST[1:0]		Reserved	CT	DBM	PL[1:0]		PINCOS	MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR[1:0]		PFCTRL	TCIE	HTIE	TEIE	DMEIE	EN
	Reset value					0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0014	DMA_S0NDTR	Reserved																NDT[15:.]															
	Reset value																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0018	DMA_S0PAR	PA[31:0]																															
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x001C	DMA_S0M0AR	M0A[31:0]																															
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Status reg.

Clear flags reg.

Ctrl reg.

Števec

Naslov DReg

Naslov Pomn

DMA – stanje , nastavitve

Osnovni registri za delovanje DMA USART naprave:

DMA_HISR

TCIF5 .. Oddaja, TCIF6 .. sprejem

DMA_HIFCR

CTCIF5 .. zbriši TCIF5, CTCIF6 .. zbriši TCIF6

DMA_SxCR

- CHSEL — številka kanala = 4
- MINC — memory increment by 1
- DIR — Periph2Memory (0)
- EN — Enable channel (1)

DMA_SxNDTR — števec

DMA_SxPAR_RX — naslov podatk. registra naprave

DMA_SxM0AR_RX — naslov v pomnilniku

Od prej še :

RCC_AHB1ENR — **vklop/izklop DMA**

USART_CR3 — **vklop/izklop DMA prenosa**

DMA – krmiljenje (INIT)

Potrebni koraki za krmiljenje DMA naprave (INIT_DMA):

1. **Vklop DMA1 naprave**
   - **RCC_AHB1ENR : b_{21}=1 (DMA1 Enable Clock)**
2. **Vklop DMA sprejema in oddaje za USART 2**
   - **USART2_CR3 : b_6=1 (DMAR bit = 1)**
   - **USART2_CR3 : b_7=1 (DMAT bit = 1)**

Delovanje:

- **sprejem niza znakov** **RCV_DMA**
- **oddaja niza znakov** **SND_DMA**

DMA – sprejem

Potrebni koraki za krmiljenje DMA naprave (RCV_DMA):

1. **Počakaj** EN bit **= 0 v DMA_SxCR** registru (zaključek prejšnjega prenosa)
   - **DMA_SxCR_RX** **b_1=0 (Stream disabled)**
2. **Nastavitve naslovov**
   - **DMA_SxPAR_RX** **naslov DR registra (**USART2_BASE+USART2_DR)
   - **DMA_SxM0AR_RX** **naslov v pomnilniku**
   - **DMA_SxSNDTR_RX** **število znakov**
3. **Nastavitev lastnosti (skupine bitov) :**
   - CHSEL številka kanala = 4
   - MINC memory increment by 1
   - DIR Periph2Memory (0)
   - EN Enable channel (1)

Delovanje :

- **Konec prenosa :**
  - DMA_HISR **:** TCIF5 = 1 **(DMA označi konec prenosa)**
- **Brisanje zastavice:**
  - DMA_HICFR **:** CTCIF5 = 1 **(zbriši zastavico TCIF5)**

DMA – oddaja

Potrebni koraki za krmiljenje DMA naprave (SND_DMA):

1. **Počakaj EN bit = 0 v DMA_SxCR** registru (zaključek prejšnjega prenosa)
   - **DMA_SxCR_TX** **b_1=0 (Stream disabled)**
2. **Nastavitve naslovov**
   - **DMA_SxPAR_TX** **naslov DR registra (**USART2_BASE+USART2_DR)
   - **DMA_SxM0AR_TX** **naslov v pomnilniku**
   - **DMA_SxSNDTR_TX** **število znakov**
3. **Nastavitev lastnosti (skupine bitov) :**
   - CHSEL številka kanala = 4
   - MINC memory increment by 1
   - DIR Memory2Periph (1)
   - EN Enable channel (1)

Delovanje :

- **Konec prenosa :**
  - **DMA_HISR** TCIF6 = 1 **(DMA označi konec prenosa)**
- **Brisanje zastavice:**
  - **DMA_HICFR** CTCIF6 = 1 **(zbriši zastavico TCIF6)**

DMA – krmiljenje

Naslovi registrov:

```
// DMA Registers definitions
.equ     DMA1_BASE, 0x40026000  //RM, page 65
.equ     DMA_LISR, 0x00
.equ     DMA_HISR, 0x04
.equ     DMA_LIFCR, 0x08
.equ     DMA_HIFCR, 0x0C

.equ     DMA_USART2_RX_STREAM, 5    //Channel 4 on DMA1 stream 5 is USART_RX
.equ     DMA_USART2_TX_STREAM, 6    //Channel 4 on DMA1 stream 6 is USART_TX

// Registers differ on address determined from stream number
.equ     DMA_SxCR_RX,0x10 + 0x18 * DMA_USART2_RX_STREAM
.equ     DMA_SxSNDTR_RX,0x14 + 0x18 * DMA_USART2_RX_STREAM
.equ     DMA_SxPAR_RX,0x18 + 0x18 * DMA_USART2_RX_STREAM
.equ     DMA_SxM0AR_RX,0x1C + 0x18 * DMA_USART2_RX_STREAM

.equ     DMA_SxCR_TX,0x10 + 0x18 * DMA_USART2_TX_STREAM
.equ     DMA_SxSNDTR_TX,0x14 + 0x18 * DMA_USART2_TX_STREAM
.equ     DMA_SxPAR_TX,0x18 + 0x18 * DMA_USART2_TX_STREAM
.equ     DMA_SxM0AR_TX,0x1C + 0x18 * DMA_USART2_TX_STREAM
```

CubeIDE – Registers okno

Name	Value
General Registers	
r0	0x0
r1	0x0
r2	0x0
r3	0x0
r4	0x0
r5	0x1000
r6	0x40020c14
r7	0x0
r8	0x0
r9	0x0
r10	0x0
r11	0x0
r12	0x0
sp	0x20020000
lr	0xffffffff
pc	0x800002a
xpsr	0x41000000

CubeIDE – SFR okno