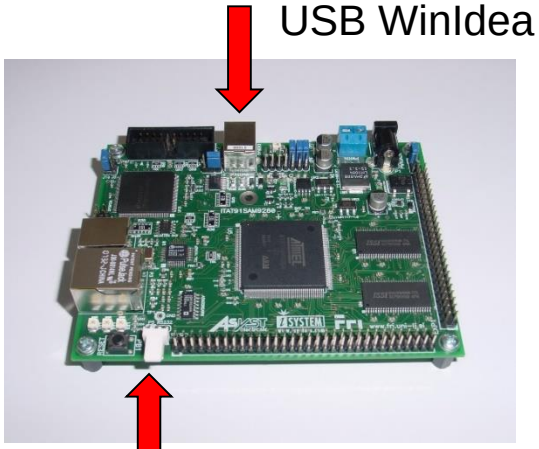
ARM

Vhodno / izhodne naprave

DMA krmilnik

Delo na FRI-SMS razvojnem sistemu



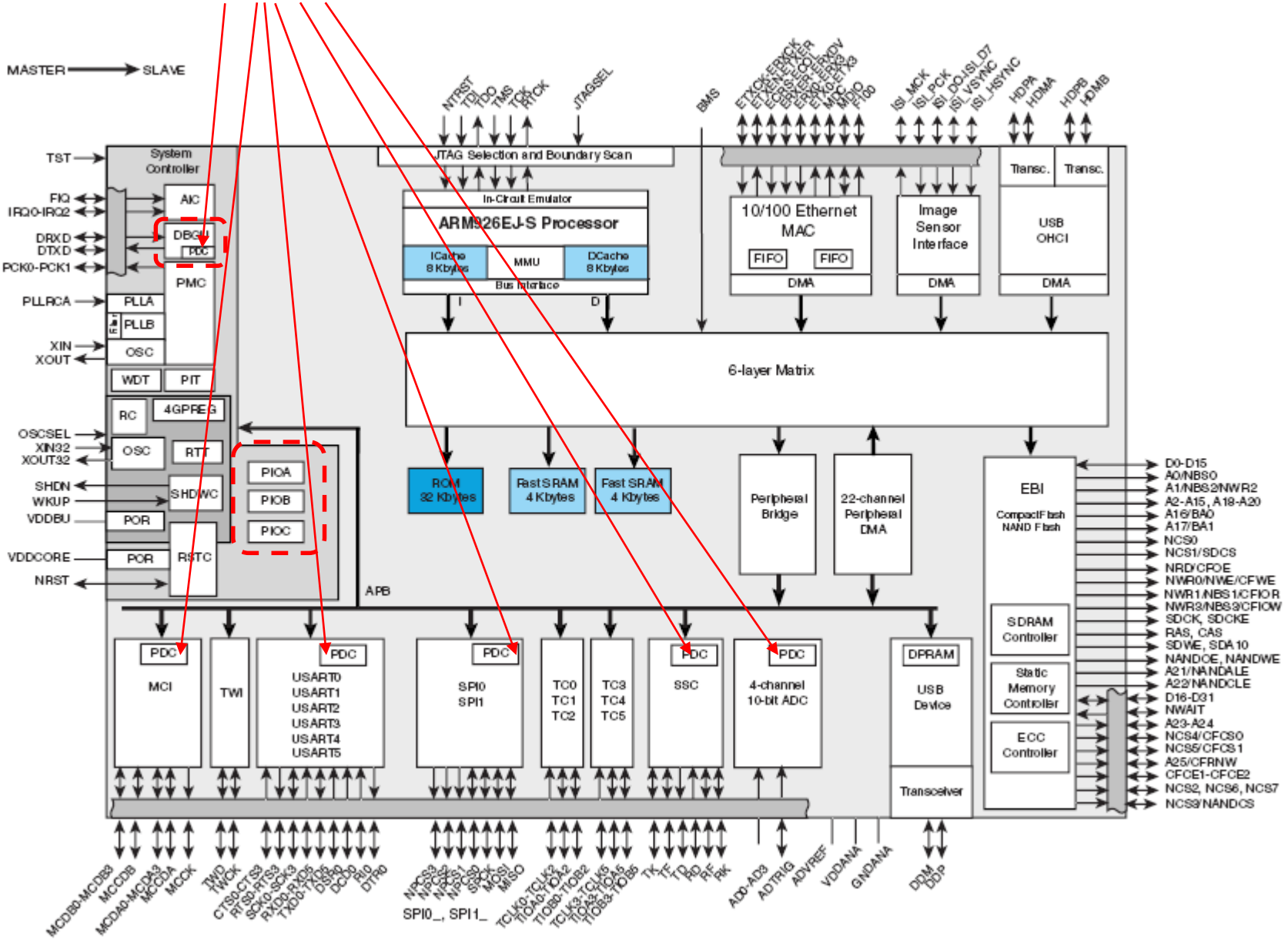
Priključitev :

- **USB** priklop na **daljši stranici**, sveti **zelena LED** dioda

Poseben projekt za FRI-SMS (e-učilnica) :

- ***dodatne nastavitve*** (informativno) :
  - frekvenca urinega signala (višja poveča porabo!)
  - vklop predpomnilnikov
  - inicializacija sklada oz. SP – kazalca na sklad
- ***dodajanje vsebine (start.s):***
  - *podatki/operandi:*
    - dodamo v `/*constants*/` ,končamo z .align
  - *program* :
    - dodamo v `/* enter your code here */`
    - na koncu programa je mrtva zanka
    - podprograme dodamo za mrtvo zanko

DMA Krmilnik (PDC)



DMA krmilnik

- **Za prenos podatkov, ki ne troši CPE časa**
- **22 DMA kanalov**
- **V/I naprave z DMA kanali**
  - DBGU, SPI, USART, SSC, MCI, EMAC, ISI, ADC
- **Hkratno dvosmerne V/I naprave (full duplex) imajo dva DMA kanala**
- **Enosmerne in izmenično dvosmerne (half duplex) V/I naprave imajo po en DMA kanal**
- **Preprosto programiranje. Potrebno je vpisati le:**
  - začetni naslov in
  - dolžino bloka za prenos
- **DMA krmilnik je dostopen preko naslovnega prostora vsake naprave posebej od odmika 0x100 dalje**

DMA krmilnik

Potrebni koraki za sprožitev DMA prenosa (branje iz V/I naprave):

1. vpiši začetni naslov bloka podatkov v RPR
2. vpiši število podatkovnih elementov v RCR (dolžina elementa: 1, 2 ali 4 bajte je samodejno določena glede na nastavitve V/I naprave)
3. omogoči DMA prenos s postavitvijo bita RXTEN v PTCR

Potrebni koraki za sprožitev DMA prenosa (pisanje v V/I napravo):

1. vpiši začetni naslov bloka podatkov v TPR
2. vpiši število podatkovnih elementov v TCR (dolžina elementa: 1, 2 ali 4 bajte je samodejno določena glede na nastavitve V/I naprave)
3. omogoči DMA prenos s postavitvijo bita TXTEN v PTCR

Branje in pisanje lahko sprožimo tudi hkrati

Informacijo o tem ali je DMA prenos že zaključen, dobimo v statusnih registrih posameznih V/I naprav

Primer:

ko je sprejemanje želenega števila znakov končano, se postavi zastavica ENDRX

OPOZORILO:

SRAM od naslova 0 dalje, ki ga uporabljamo za programe in podatke, je v resnici v naslovnem prostoru od naslova 0x200000 dalje, na naslovu 0 je le njegova kopija. DMA krmilnik te kopije ne 'vidi', zato je pri vpisovanju začetnih naslovov v ustrezne registre potrebno uporabljati naslove od 0x200000 dalje.

DMA krmilnik

Bazni naslov je naslov vsake V/I naprave posebej

Odmiki registrov od baznega naslova:

```
.equ DMA_RPR, 0x100 /* Receive Pointer Register */
.equ DMA_RCR, 0x104 /* Receive Counter Register */
.equ DMA_TPR, 0x108 /* Transmit Pointer Register */
.equ DMA_TCR, 0x10C /* Transmit Counter Register */
.equ DMA_RNPR, 0x110 /* Receive Next Pointer Register */
.equ DMA_RNCR, 0x114 /* Receive Next Counter Register */
.equ DMA_TNPR, 0x118 /* Transmit Next Pointer Register */
.equ DMA_TNCR, 0x11C /* Transmit Next Counter Register */
.equ DMA_PTCR, 0x120 /* Periph. Transfer Control Register */
.equ DMA_PTSR, 0x124 /* Periph. Transfer Status Register */
```

Informacijo o tem ali je DMA prenos že zaključen, dobimo v statusnih registrih posameznih V/I naprav (tam je tudi dokumentirana).

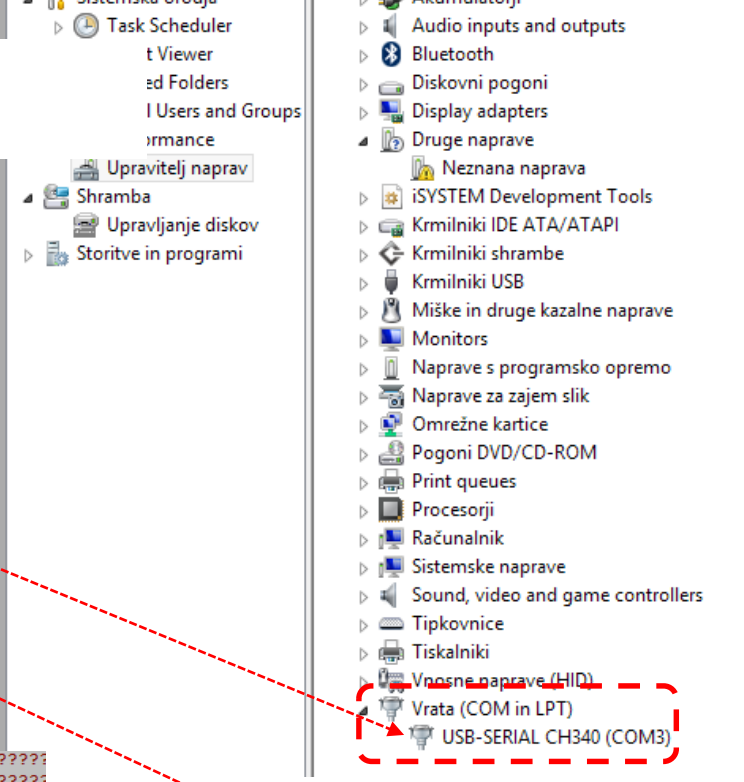
Za DBGU je statusni register DBGU_SR.

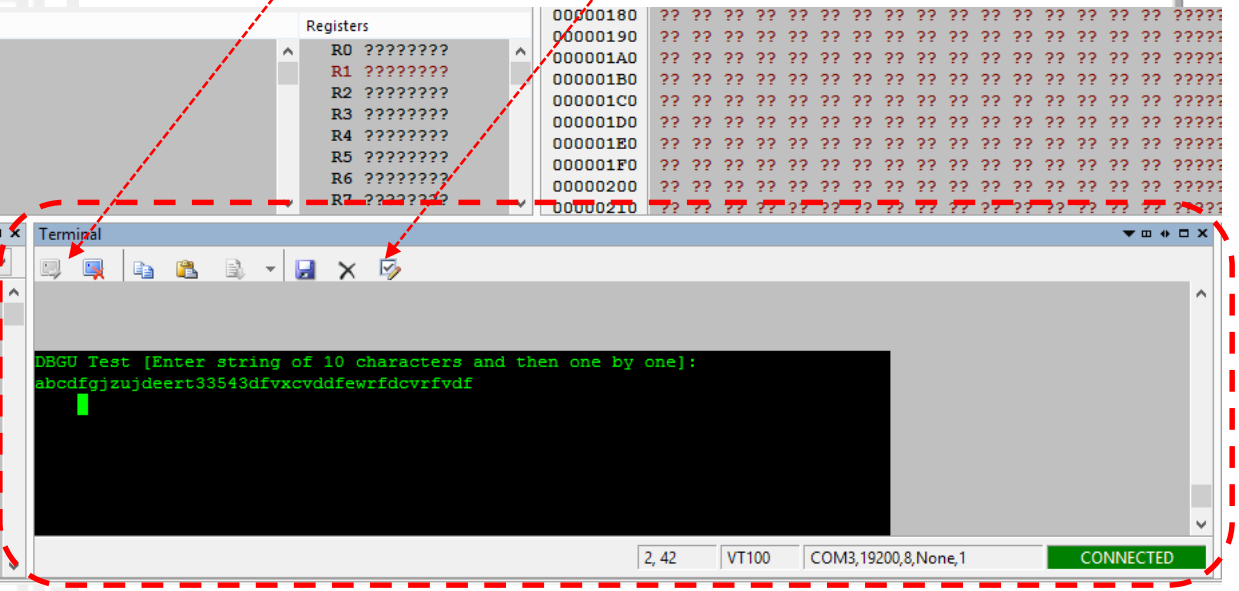
DMA-DBGU Priključitev na PC strani

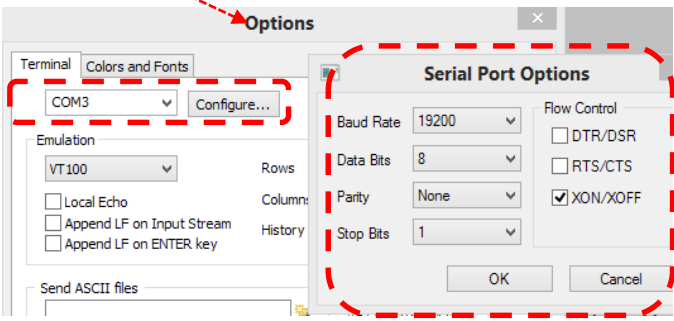
1. **Ugotovimo številko serijskega vmesnika (COM porta):**
   „Upravitelj naprav“

2. **Okno Terminal v WinIdea (PC stran):**
   - Nastavitve COM porta (Options)
   - Priklop/Odklop

   tukaj se kažejo znaki, ki jih pošlje FRI-SMS







Izseki iz podatkovne listine

Opis delovanja DMA krmilnika

24.1 Description

The Peripheral DMA Controller (PDC) transfers data between on-chip serial peripherals and the on- and/or off-chip memories. The link between the PDC and a serial peripheral is operated by the AHB to ABP bridge.

The PDC contains 22 channels. The full-duplex peripherals feature 21 mono directional channels used in pairs (transmit only or receive only). The half-duplex peripherals feature 1 bi-directional channels.

The user interface of each PDC channel is integrated into the user interface of the peripheral it serves. The user interface of mono directional channels (receive only or transmit only), contains two 32-bit memory pointers and two 16-bit counters, one set (pointer, counter) for current transfer and one set (pointer, counter) for next transfer. The bi-directional channel user interface contains four 32-bit memory pointers and four 16-bit counters. Each set (pointer, counter) is used by current transmit, next transmit, current receive and next receive.

Using the PDC removes processor overhead by reducing its intervention during the transfer. This significantly reduces the number of clock cycles required for a data transfer, which improves microcontroller performance.

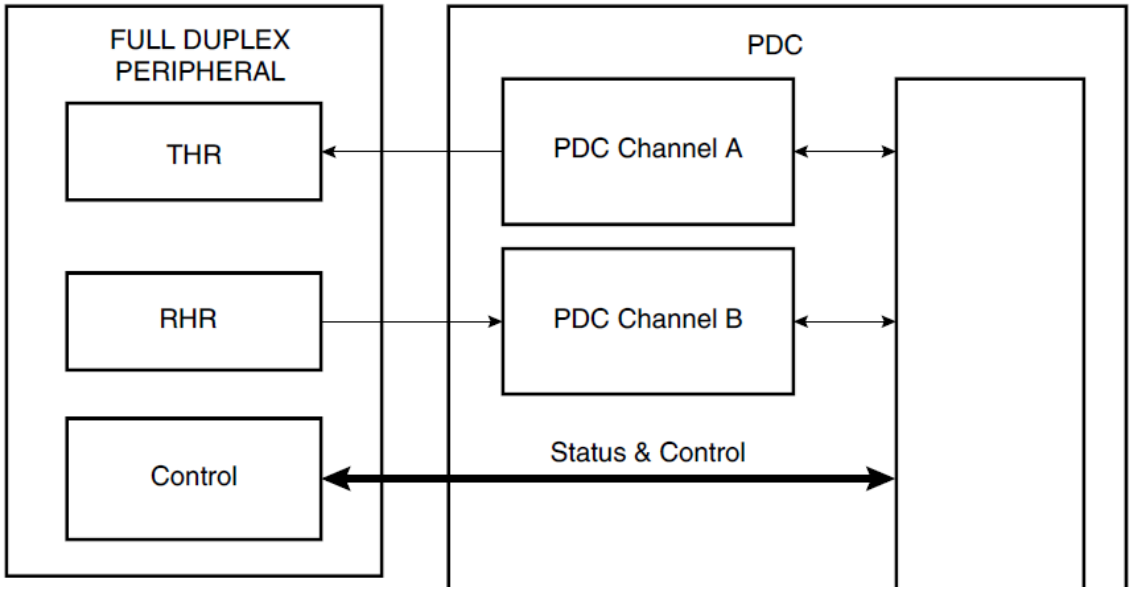
To launch a transfer, the peripheral triggers its associated PDC channels by using transmit and receive signals. When the programmed data is transferred, an end of transfer interrupt is generated by the peripheral itself.

Izseki iz podatkovne listine

Shema delovanja DMA krmilnika za DBGU napravo („full-duplex“)

24.2 Block Diagram

Figure 24-1. Block Diagram



Izseki iz podatkovne listine

Kazalci

24.3.2 Memory Pointers

Each full duplex peripheral is connected to the PDC by a receive channel and a transmit channel. Both channels have 32-bit memory pointers that point respectively to a receive area and to a transmit area in on- and/or off-chip memory.

Števci

24.3.3 Transfer Counters

Each channel has two 16-bit counters, one for current transfer and the other one for next transfer. These counters define the size of data to be transferred by the channel. The current transfer counter is decremented first as the data addressed by current memory pointer starts to be transferred. When the current transfer counter reaches zero, the channel checks its next transfer counter. If the value of next counter is zero, the channel stops transferring data and sets the appropriate flag. But if the next counter value is greater then zero, the values of the next pointer/next counter are copied into the current pointer/current counter and the channel resumes the transfer whereas next pointer/next counter get zero/zero as values. At the end of this transfer the PDC channel sets the appropriate flags in the Peripheral Status Register.

The following list gives an overview of how status register flags behave depending on the counters' values:

- ENDRX flag is set when the PERIPH_RCR register reaches zero.
- RXBUFF flag is set when both PERIPH_RCR and PERIPH_RNCR reach zero.
- ENDTX flag is set when the PERIPH_TCR register reaches zero.
- TXBUFE flag is set when both PERIPH_TCR and PERIPH_TNCR reach zero.

These status flags are described in the Peripheral Status Register.

Izseki iz podatkovne listine

Sprožitev DMA krmilnika

24.3.4 Data Transfers

The serial peripheral triggers its associated PDC channels' transfers using transmit enable (TXEN) and receive enable (RXEN) flags in the transfer control register integrated in the peripheral's user interface.

Izseki iz podatkovne listine

Označitev konca prenosa

24.3.5 PDC Flags and Peripheral Status Register

Each peripheral connected to the PDC sends out receive ready and transmit ready flags and the PDC sends back flags to the peripheral. All these flags are only visible in the Peripheral Status Register.

Depending on the type of peripheral, half or full duplex, the flags belong to either one single channel or two different channels.

24.3.5.1 Receive Transfer End

This flag is set when PERIPH_RCR register reaches zero and the last data has been transferred to memory.

It is reset by writing a non zero value in PERIPH_RCR or PERIPH_RNCR.

24.3.5.2 Transmit Transfer End

This flag is set when PERIPH_TCR register reaches zero and the last data has been written into peripheral THR.

It is reset by writing a non zero value in PERIPH_TCR or PERIPH_TNCR.