

**Naloga 1**

a)

```

PSP:      sub   lr,lr,#4
          stmfd sp!,{r0-r2,lr}      @ vstopna točka v PSP
          ldr   r0,=USART_BASE
          ldr   r1,[r0,#USART_CSR]  @ preberemo USART statusni register
          tst   r1,#1 << 1
          beq   KONEC                @ če bit TXRDY ni 1, gremo na konec

          ldrb  r1,INDEX              @ preberemo vsebino spr. INDEX
          cmp   r1,#8
          bhs   KONEC                @ če je že 8, gremo na konec

          ldr   r2,=BUFFER
          ldrb  r2,[r2,r1]            @ preberemo podatek iz spr. BUFFER+INDEX
          str   r2,[r0,#USART_THR]   @ zapišemo v USART_THR - oddamo podatek
          add   r1,r1,#1
          strb  r1,INDEX              @ povečamo INDEX za 1

KONEC:    ldr   r0,=AIC_BASE
          str   r0,[r0,#AIC_EOICR]   @ slepo pisanje v prek. krmilnik
          ldmfd sp!,(r0-r2,pc)^      @ izstopna točka iz PSP

```

b)

```

PODP:     stmfd sp!,{r0-r1,lr}      @ registri in povratni naslov na sklad
          ldr   r0,=USART_BASE
CAKAJ:    ldr   r1,[r0,#USART_CSR]   @ preberemo USART statusni register
          tst   r1,#1 << 1
          beq   CAKAJ                @ ponavljaj dokler TXRDY ni enak 1
          ldrb  r1,BUFFER            @ preberemo prvi podatek iz BUFFER
          str   r1,[r0,#USART_THR]   @ zapišemo v USART_THR - oddamo podatek
          mov   r1,#1
          strb  r1,INDEX              @ spremenljivko INDEX nastavimo na 1
          ldmfd sp!,{r0-r1,pc}      @ obnovimo registre in se vrnemo iz podp.

```

**Naloga 2 (ORS)**

a)

0x0000E00 - 0x0000FFF - 0x00001000 - 0x00001100

Dostopa se do strani 0 in 1 (poudarjeno), za preslikavo se uporabita deskriptor 0 in deskriptor 1 iz tabele deskriptorjev: 0x00002003, 0x00006003. Biti P in V sta pri obeh postavljena. Poudarjeni sta številki okvirja, ki skupaj z naslovom znotraj strani tvorita fizični naslov. Stran 0 je v okvirju 2, stran 1 pa v okvirju 6.

Program je torej na fizičnih naslovih:

0x00002E00 - 0x00002FFF in 0x00006000 - 0x00006100

b)

0x3C00 (0x0000C003 - ni napake strani)  
 0x3C04 (0x0000C003 - ni napake strani)  
 0x3C08 (0x0000C003 - ni napake strani)  
 0x2000 (0x00000000 - napaka strani, dodeli se okvir 256 -> na naslov 0xFFFF0008 se zapiše 0x00100003)  
 0x2004 (0x0000F003 - ni napake strani)  
 0x2008 (0x0000F003 - ni napake strani)  
 0x4000 (0x00000000 - napaka strani, dodeli se okvir 257 -> na naslov 0xFFFF0010 se zapiše 0x00101003)  
 0x4004 (0x0000F003 - ni napake strani)  
 0x3000 (0x0000D003 - ni napake strani)  
 0x3004 (0x0000D003 - ni napake strani)

## Naloga 2 (ARS II)

Urina perioda (up) traja 2,5 ns.

(številke pred dvopičjem pomenijo urino periodo)

- 1 : izstavimo ukaz za odpiranje vrstice. Moramo počakati  $t_{RCD} = 9$  ns tj. 4 up. Preden izstavimo ukaz za zapiranje, mora poteči  $t_{RAS} = 30$  ns, tj. 12 up. Ukaz za zapiranje zato lahko izstavimo v 13 up, vendar mora pred ponovnim odpiranjem poteči najmanj  $t_{RP} = 9$  ns tj. 4 up.
- 5 : izstavimo ukaz za branje. Čakamo  $t_{CL} = 4$  up pred branjem podatkov.
- 9: lahko beremo 1. in 2. besedo.
- 10: beremo 3. in 4. besedo.
- 13: izstavimo ukaz za zapiranje vrstice.

## Naloga 3

