

**Naloga 1**

KOPIRAJ:	STMFD	SPI!, { R3-R4, LR }	@ shranimo registre na sklad
	MOVS	R3, R2, LSR #2	@ število 32-bitnih prenosov v R3 (R2/4)
	BEQ	LOOP2	@ če ni 32-bitnih prenosov gremo na LOOP2
LOOP1:	LDR	R4, [R0], #4	
	STR	R4, [R1], #4	@ prekopiramo 32-bitno
	SUBS	R3, R3, #1	@ zmanjšamo števec 32-bitnih prenosov
	BNE	LOOP1	@ ponovimo, če ni konec
	AND	R2, R2, #3	@ v R2 koliko nam ostane 8 bitnih prenosov
LOOP2:	LDRB	R4, [R0], #1	
	STRB	R4, [R1], #1	@ prekopiramo 8-bitno
	SUBS	R2, R2, #1	@ zmanjšamo števec 8-bitnih prenosov
	BNE	LOOP2	@ ponovimo, če ni konec
	LDMFD	SPI!, { R3-R4, PC }	@ obnovimo registre in nazaj v gl. program

**Naloga 2**

KTAB:	.BYTE	0x35, 0x06, 0x5D, 0x4F, 0x66, 0x6B, 0x5F, 0x07, 0x7F, 0x6F	
	...		
	BL	INIT_PIO	
	ADR	R0, KTAB	
	MOV	R1, #0	@ trenutno število
	LDR	R3, =PIOB_BASE	
LOOP:	LDRB	R2, [R0, R1]	@ r2 <- 7SEG koda[r1]
	STR	R2, [R3, #PIO_SODR]	@ vklopimo segmente
	MVN	R2, R2	@ negiramo
	STR	R2, [R3, #PIO_CODR]	@ izklopimo ostale segmente
	BL	DELAY	@ 500 ms
	ADD	R1, R1, #1	@ naslednje število
	CMP	R1, #10	
	MOVEQ	R1, #0	@ če je 10, zopet od 0
	B	LOOP	

**Naloga 3**

a)

$$1,2t_{ag} = 0,01 \cdot 2t_{ag} + t_{ag} + p \cdot 12\text{ms}$$

$$0,18 \cdot 50\text{ns} = p \cdot 12\text{ms}$$

$$p = 7,5 \cdot 10^{-7}$$

b)

biti 0-11 določajo naslov znotraj strani, biti 12-22 indeks v tabeli strani na nivoju 1 in biti 23-31 indeks v tabeli strani na nivoju 2.

Vsek deskriptor na nivoju 2 kaže na eno tabelo strani na nivoju 1. Zanima nas torej, do koliko različnih deskriptorjev na nivoju 2 se dostopa.

0x00070000 – 0x01FFFF. Biti 23-31 se spremenjajo od 00000000 do 00000011, torej imamo 4 različne deskriptorje v tabeli na nivoju 2.

## Dostopa se do 4 različnih tabel strani na nivoju 1.

c) dostopamo do vseh naslovov od 0x00070000 do 0x01FFFFFF. To pomeni, da dostopamo do strani od 0x00070 do 0x01FFF, torej do 8080 različnih strani. Ob prvem dostopu do strani se v TLB zapiše ustrezen deskriptor, pri nadalnjih 4095 dostopih v isto stran torej prihranimo dostop do dveh tabel strani oziroma dva dostopa glavnega pomnilnika. Skupno prihranimo:

$$8080 * 4095 * 50\text{ns} * 2 = 3308760000 \text{ ns} = 3,30876 \text{ s}$$

## Naloga 4

Razlika med izhodnimi in vhodnimi napetostnimi nivoji je:

- Visok nivo:  $3V - 2,7V = 0,3V$
- Nizek nivo:  $0,8V - 0,2V = 0,6V$

Odpornost proti šumu je torej **0,3V** (vzamemo manjšo razliko).

## Naloga 5

Pri navadnem protokolskem prenosu smo vezani na fronte centralne ure, ki pa zaradi časa, ki je potreben za potovanje signala, niso povsod istočasne. Pojavlja se urin zamik ( $t_{zamik}$ ).

Pri izvorno sinhronskem protokolskem prenosu sihronizacijski signal (uro) tvori naprava, ki podatke pošilja. Ker ta signal potuje po podobnih poteh kot podatki, je relativni zamik ( $\Delta t$ ) med podatki in sinhronizacijskim signalom zelo majhen.

$$\text{Namesto enačbe } t_{dostop} + t_{vzpostavitev} \leq T - t_{zamik} \text{ velja } t_{dostop} + t_{vzpostavitev} \leq T_s - \Delta t$$

Ker je  $\Delta t$  bistveno manjši od  $t_{zamik}$ , je lahko perioda  $T_s$  bistveno krajsa od  $T$ .