

Digitális technika 2 második laborgyakorlat


A foglalkozás célja:

A már megismert mikrokontrolleres fejlesztői környezetben egyszerű programozási feladatok készítése:


- Szoftveres késleltetés megvalósítása
- 7 szegmens kijelző kezelése: adat továbbítása bitsorosan a kijelző felé
- Számlálás megvalósítása a kijelzőn

A feladatok megvalósítása során előre elkészített projektet kell betölteni (2_Labor.X), amely tartalmazza a megvalósítandó feladatok vázát, valamint egy külön assembly állományban (init_send.s) a használandó hardver elemek (portok) kezdeti értékadását megvalósító szubrutint (**Disp_init**) és egy konverter rutint (**Disp_conv**) amely a paraméterként kapott bináris számhoz (0...99 közötti érték) előállítja a 7 szegmens kijelzőre kiküldendő bitmintát.


1. 1ms szoftver késleltetés


1. Töltse le a Villamoskari oktatási portálról (edu.vik.bme.hu) a „2_labor.X.zip” projektet, majd tömörítse ki az asztalra. Indítsa el az MPLAB X IDE v5.50 fejlesztői környezetet, majd „File→Open Project...” funkcióval tölts be ezt a projektet.
2. Nyissa meg a „main.s” forráskódot. **A projektben található init_send.s állomány tartalmát NE módosítsa!**
3. Tervezzon szoftveres várakozó ciklust, amely 1ms ideig fut. Helyezze el a kódot a **Delay1ms** szubrutin törzsében. Ügyeljen arra a szubrutinnal szemben támasztott követelményre, hogy a szubrutin nem ronthat el regiszter értékeket!
4. Ellenőrizze az algoritmus futási idejét a szubrutin meghívásától indulva a szubrutinból visszatérésig. Az utasítás végrehajtás órajelét állítsa 16MHz értékre a szimulátorban.
( Dash board→Conf→Simulator→Instruction Frequency (F_{CYC}))
Időmérés: „Window→Debugging→Stopwatch”
5. Ha kell, hangolja a szubrutint, hogy a futási idő (beleértve a szubrutin meghívását és a szubrutinból visszatérést) a lehető legjobban közelítse meg az 1ms értéket.
6. Dokumentálja a megoldását (Kódrészlet és Stopwatch ablak, melyben látszik a megvalósított késleltetés értéke).

2. N ms szoftver késleltetés

1. Írja meg a **DelayNms** szubrutint az előző feladatban megvalósított szubrutin felhasználásával.
2. Ellenőrizze az algoritmus idejét a szubrutin meghívásától indulva a szubrutinból visszatérésig. Az utasítás végrehajtás órajelét állítsa 16MHz értékre.
( Dash board→Conf→Simulator→Instruction Frequency (F_{CYC}))
Időmérés: „Window→Debugging→Stopwatch”
3. Dokumentálja a megoldását. (Kódrészlet és Stopwatch ablak)
4. Adja meg, hogy a megvalósítása mekkora eltérést jelent 1sec (N=1000) késleltetés esetén.

3. Pontosan 1 sec késleltetés megvalósítása

1. Írja meg a **Delay1sec** szubrutint, amely pontosan 1 másodpercet késleltet.
2. Ellenőrizze az algoritmus idejét a szubrutin meghívásától indulva a szubrutinból visszatérésig. Az utasítás végrehajtás órajelét állítsa 16MHz értékre.
( Dash board → Conf → Simulator → Instruction Frequency (F_{CYC}))
Időmérés: „Window → Debugging → Stopwatch”
3. Írjon kódot, amely végtelen ciklusban minden szubrutin hívás után ellenkezőjére váltja a LED1 és LED2 kijelzők értékét. Kezdetben a LED1 kijelző világítson (a kimeneti bit értéke 1), a LED2 kijelző legyen sötét (a kimeneti bit értéke 0). A LED kijelzők az A port 8. (LED1) és 9. (LED2) bitjén helyezkednek el.
4. Csatlakoztassa a fejlesztőpanelt a számítógép USB portjához.
5. Állítsa át a konfigurációt szimulátorról hardver eszközre.

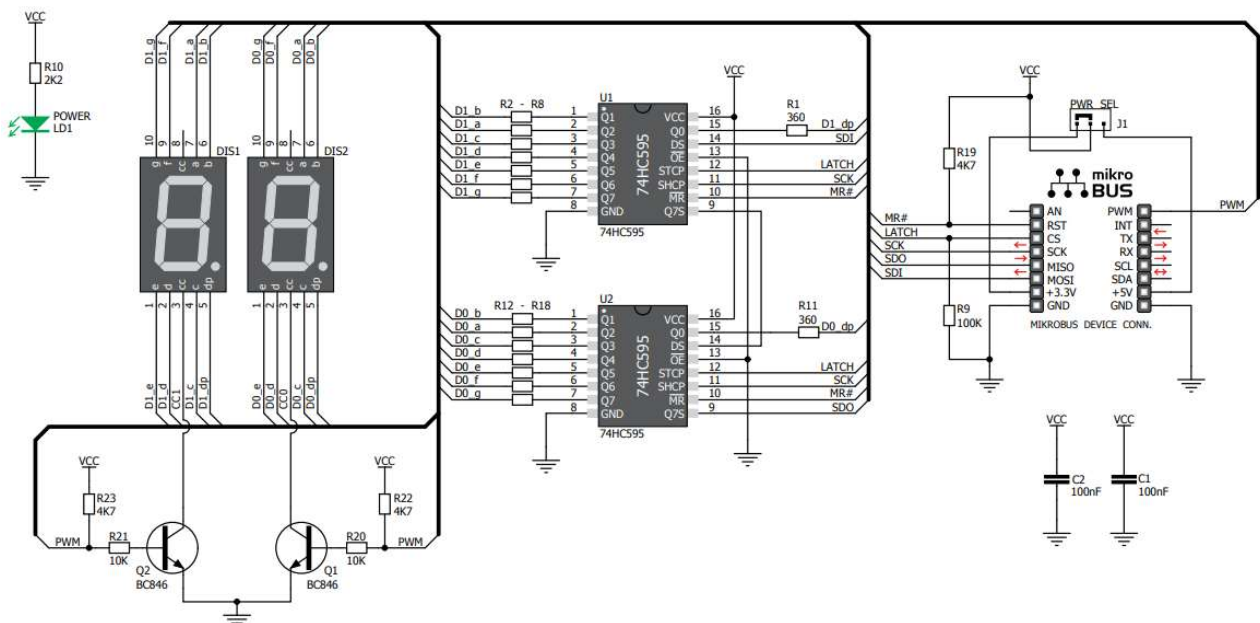
(Dashboard ablak – Properties funkció )

Connected Hardware Tool: Starter Kits (PKOB)-SN: BUR.....)

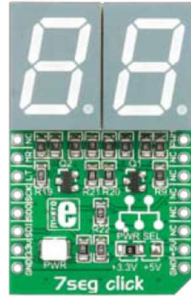
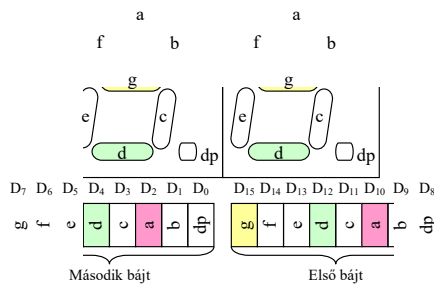
6. Fordítsa le és töltsse be az eszközbe a programot.
7. Dokumentálja a megoldását. (Kódrészlet)

4. 7 szegmens kijelző kezelése

A 7 szegmens kijelző hardver megvalósítása a következő ábrán látható:



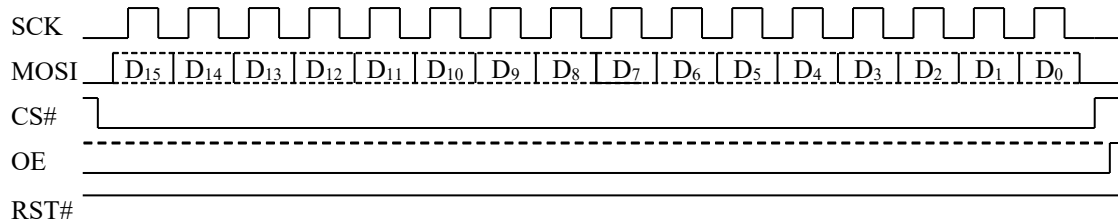
Figyeljük meg, hogy a kijelzendő információt két 8 bites léptető regiszter tárolja. A méréshez használt fejlesztő panelen két „Mikrobus” csatlakozó található. A feladat elvégzéséhez az „A” jelű csatlakozóhoz kapcsolódik a kijelző modul. A következő táblázatban megadjuk mindkét csatlakozó kapcsolódását a mikrokontroller megfelelő portjaihoz.



Mikrobus csatlakozópontok:

Jelnév	A	B	Irány
RST#	RA13	RA14	O
CS#	RC9	RA10	O
SCK	RB15	RC6	O
MISO	RB13	RB9	I
MOSI	RB14	RC2	O
PWM	RC3	RC4	O

A léptető regiszterekbe a következő protokollal lehet információt juttatni:



Megjegyzések:

A shiftregiszterek törlés bemenetét az RST# jel vezérli, ezt magas értéken kell tartani.

A PWM (OE) kimenet magas értéke esetén világítanak a kijelölt szegmensek.

A MISO jel segítségével visszaolvasható a shiftregiszterek tartalma. Fontos, hogy bemenetre legyen állítva ez a portbit.

A CS# jel felfutó élére történik a kijelző látható tartalmának átírása.

A léptető regiszter adatlapja a következő időzítéseket tartalmazza. Az ábrán bekeretezve megadjuk a fenti csatlakozón található elnevezéseket is a jelek megfeleltetése érdekében.

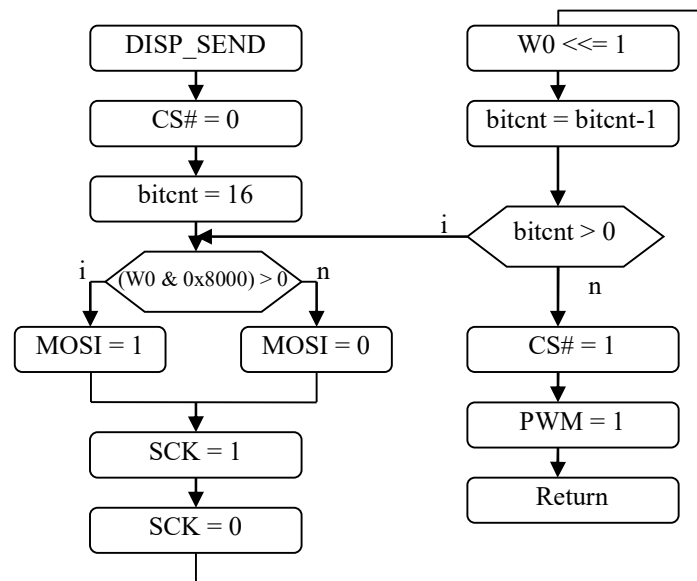
6.6 Timing Requirements

over operating free-air temperature range (unless otherwise noted)

		V _{CC}	T _A = 25°C		SN54HC595		SN74HC595		UNIT
			MIN	MAX	MIN	MAX	MIN	MAX	
f _{clock}	Clock frequency	2 V		6	4.2		5		MHz
		4.5 V		31	21		25		
		6 V		36	25		29		
t _w	SRCLK or RCLK high or low	2 V	80		120		100		ns
		4.5 V	16		24		20		
		6 V	14		20		17		
	SRCLR low	2 V	80		120		100		
		4.5 V	16		24		20		
		6 V	14		20		17		
t _{su}	SER before SRCLK↑	2 V	100		150		125		ns
		4.5 V	20		30		25		
		6 V	17		25		21		
	SRCLK↑ before RCLK↑ ⁽¹⁾	2 V	75		113		94		
		4.5 V	15		23		19		
		6 V	13		19		16		
	SRCLR low before RCLK↑	2 V	50		75		65		
		4.5 V	10		15		13		
		6 V	9		13		11		
	SRCLR high (inactive) before SRCLK↑	2 V	50		75		60		
		4.5 V	10		15		12		
		6 V	9		13		11		
t _h	Hold time, SER after SRCLK↑	2 V	0		0		0		ns
		4.5 V	0		0		0		
		6 V	0		0		0		

A panel 3.3V feszültséggel működik, ezért ha a 2V feszültségre előírt értékeket betartjuk, biztosan helyesen fog működni.

A protokollt a következő folyamatábra alapján célszerű megvalósítani:



1. A folyamatábra alapján készítse el a **Disp_send** szubrutint.
2. Állítsa át konfigurációt hardver eszközről szimulátorra.

(Dashboard ablak – Properties funkció )

Connected Hardware Tool: Simulator)

3. Ellenőrizze a működést a Logic Analyzer ablakban.
(Tipp: Az időzítések ellenőrzéséhez állítsa át a megjelenítést órajel ciklusról idő alapúra)
4. Ha kell, hangolja a kódot a Setup időzítések teljesítéséhez.
5. Dokumentálja az elvégzett feladatot (Kódrészlet, Logic Analyzer ablak)

5. Másodperc számláló készítése

1. Az előző feladatokban megvalósított szubrutin felhasználásával készítsen programot, amely a kijelzőn a mérésvezető által megadott számtartományban (legfeljebb 00 és 99 között), a mérésvezető által megadott irányban (felfelé, lefelé, két irányban) másodpercenként ciklikusan számlál.
2. Ellenőrizze a számláló algoritmusát szimulátorban. A **Disp_send** és a **Delay1sec** rutinok elejére egyelőre tegyen egy **return** utasítást, hogy ne kelljen azok idejét is kivárni.
3. Ha a számlálási ciklusa helyesen működik, távolítsa az előbb elhelyezett **return** utasításokat.
4. Állítsa át konfigurációt szimulátorról hardver eszközre.

(Dashboard ablak – Properties funkció )

Connected Hardware Tool: Starter Kits (PKOB)-SN: BUR.....)

5. Fordítsa le és töltse be az eszközbe a programot

(Production – Make Program Device )

6. Mutassa be a mérésvezetőnek a működést
7. Dokumentálja a programot a jegyzőkönyvben. (Kódrészlet)

6. Számláló léptetése gomb megnyomására

1. Módosítsa az előző feladatot úgy, hogy a számlálója a BTN1 gomb megnyomásának hatására lépjen, és a mérésvezető által megadott tartományon számláljon ciklikusan (legfeljebb 0...99). A BTN1 gomb aktuális állapota az RA11 porton (PORTA,#11) olvasható le. A 0 érték jelenti gomb megnyomott állapotát.
2. Először ellenőrizze a programot a szimulátorban. Ehhez állítsa át a konfigurációt hardver eszköztől szimulátorra.

(Dashboard ablak – Properties funkció



Connected Hardware Tool: Simulator)

A gomb megnyomását és elengedését a Stimulus funkcióval szimulálja.

(Window – Simulator – Stimulus)

3. Ha a program a szimulátorban helyesen működik, állítsa át a konfigurációt szimulátorról hardver eszközre.

(Dashboard ablak – Properties funkció



Connected Hardware Tool: Starter Kits (PKOB)-SN: BUR.....)

4. Fordítsa le és töltse be az eszközbe a programot

(Production – Make Program Device



)

5. Próbálja ki a működést.
6. Mutassa be a mérésvezetőnek a működést.
7. Dokumentálja a programot a jegyzőkönyvben. (Kódrészlet)
8. Zárja be a projektet a „File→Close All Projects” funkcióval.

A feladatok során használt assembly forrás:

```
.global __reset ; kötelezően exportálandó címke, ettől kerül a reset
                  ; vektor helyére a kódunk és nem lesz semmi más a
                  ; memóriában
.extern Disp_init ; Máshol megírt szubrutin a hardver inicializálására
.extern Disp_conv ; Máshol megírt szubrutin
                  ; Be: w0 - 0 ... 99 közötti érték
                  ; Ki: w0 - a 7 seg kijelzőre kiviendő bitminta

.bss
    szam: .space 2

.text
__reset:          ; kötelezo startup kód,ha stack hivatkozás
                  ; történik ezelott, az reset
    mov #__SP_init,w15 ; Stack pointer inicializalas
    mov #__SPLIM_init,w0 ;
    mov w0,_SPLIM      ; stack limit inicializalas

    call Disp_init      ; a kijelzőhöz és a LED-ekhez szükséges
                        ; portlábak megfelelő állapotának beállítása

main:
;-----
;1. feladat
;-----
; 1ms késleltetés kipróbálása, ellenőrzése
    call Delay1ms
    bra main

;-----
;2. feladat
;-----
; N ms késleltetés kipróbálása
    mov #100,w0
    call DelayNms
    bra main

;-----
;3. feladat
;-----
; Pontosán 1sec késleltetés kipróbálása a fejlesztő panelen
; A próba előtt ne felejtse el a konfigurációt
; szimulátorról panelre átállítani
    call Delay1sec
; LED1 (LATA,#8) - LED2 (LATA,#9) felváltva le-fel kapcsol

    bra main
```

```
;-----  
;4. feladat  
;-----  
;  
;   Disp_send elkészítése és kipróbálása  
;   A jelalak megjelenítése a Logic Analyzer ablakban  
;   Ne felejtse el a konfigurációt panelről szimulátorra visszaállítani  
mov   #0x5AC5,w0  
call  Disp_send  
bra   main  
  
;-----  
;5. feladat  
;-----  
;  
;   0 ... 99 közötti számláló - szimuláció után kipróbálás a panelen  
;   A kijelzendő értékhez használja a szam változót  
mov   szam,w0  
call  Disp_conv   ; A 0 és 99 közti számokat  
                        ; kijelzőre kiírható formátumra hozza  
call  Disp_send   ; Elküldi a kijelzőre a w0-ban található értéket  
  
call  Delay1ms    ; vagy call Delay1sec  
;  
; ....  
; Számláló értékének megváltoztatása  
;  
;  
bra   main  
  
;-----  
;6. feladat  
;-----  
;  
;   N ... M közötti számláló  
;   A számláló a BTN1 gomb megnyomásának hatására lép  
mov   szam,w0  
call  Disp_conv   ; A 0 és 99 közti számokat  
                        ; kijelzőre kiírható formátumra hozza  
call  Disp_send   ; Elküldi a kijelzőre a w0-ban található értéket  
  
;  
; ....  
; Gomb figyelése, számláló értékének megváltoztatása  
;  
;  
bra   main
```

```
;A mérésen készítendő szubrutinok helye:

;-----
;1. feladat
;-----
; 1 ms késleltetés
; Be: -
; Ki: -
; Ront: -
Delay1ms:
; ....
; ....
    return

;-----
;2. feladat
;-----
; N msec késleltetés
; Be: w0 - N
; Ki: -
; Ront: w0
DelayNms:
; ....
; ....
    return

;-----
;3. feladat
;-----
; Pontosán 1 sec késleltetés
; Be: -
; Ki: -
; Ront: -
Delay1sec:
; ....
; ....
    return

;-----
;4. feladat
;-----
; 16 bites érték kiküldése a kijelzőre
; Be: w0 - a kiküldendő érték
; Ki: -
; Ront: w0
Disp_send:
; ....
; ....
    return

;-----
```


1. Ellenőrző kérdések

Az alábbi ellenőrző kérdések megválaszolásához nézze át az előadáson elhangzott anyagokat, tanulmányozza a mérési útmutatót és a méréshez tartozó kiegészítő tananyagot.

- Hogyan lehet szoftveres késleltetést megvalósítani?
- A szimulátor melyik funkciójával lehet egy szoftveres késleltetés idejét meghatározni?
- Rajzolja fel a következő utasítás sorozat hatására az A port 9. bitjén előálló kimeneti jelalakot, ha a kimenet értéke kezdetben 0.

NOF

BSET LATA, #9

NOF

BCLR LATA, #9

NOF

- Hány órajelciklust késleltet az alábbi kódrészlet?

NOF

MOV W0, W0

NOF

- Mire szolgál a kijelző MOSI kivezetése?
- Mire szolgál a kijelző MISO kivezetése?
- Mire szolgál a kijelző SCK kivezetése?
- Mire szolgál a kijelző CS# kivezetése?
- A kijelzőre a 0xFEDC adatot írtuk. Mi látható rajta?