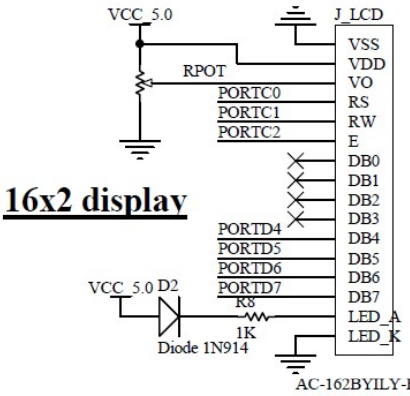
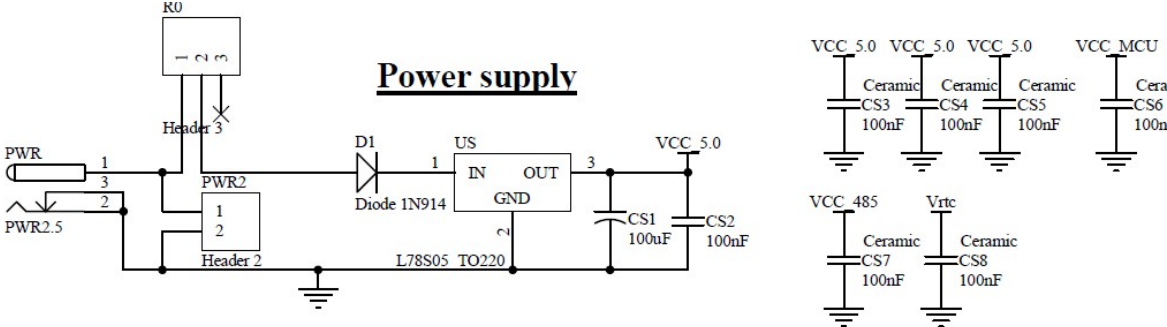


Računske vežbe iz
Projektovanja Elektronskih
Sistema
cas 8
Doc.dr Borisav Jovanović

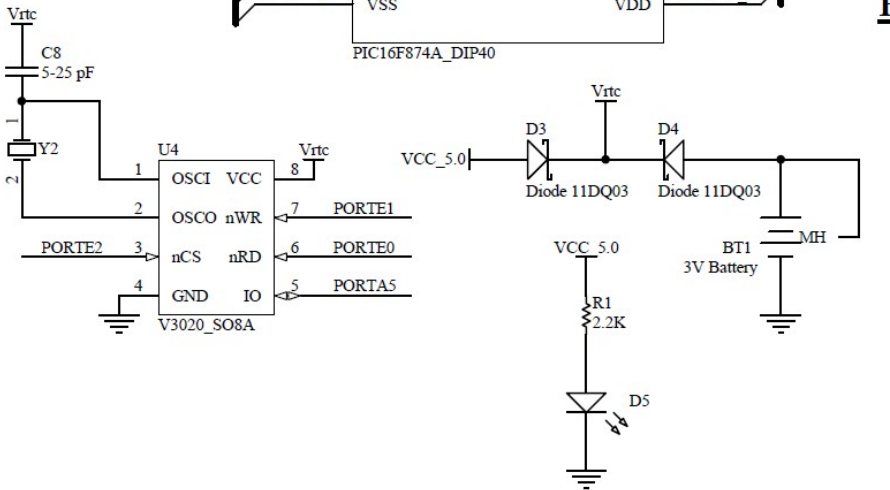
Sadržaj:

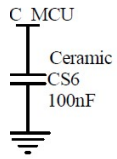
- opis komunikacionih funkcija za rad sa UART-om,
- funkcija UpdateLCD(),
- opis ***interrupt()*** funkcije
- opis funkcije **IncrementTime**
- i ConvertTime.

Detailan opis hardvera

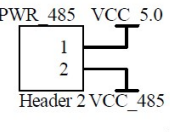
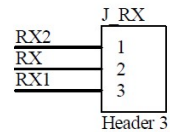
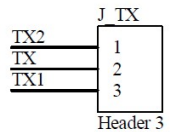
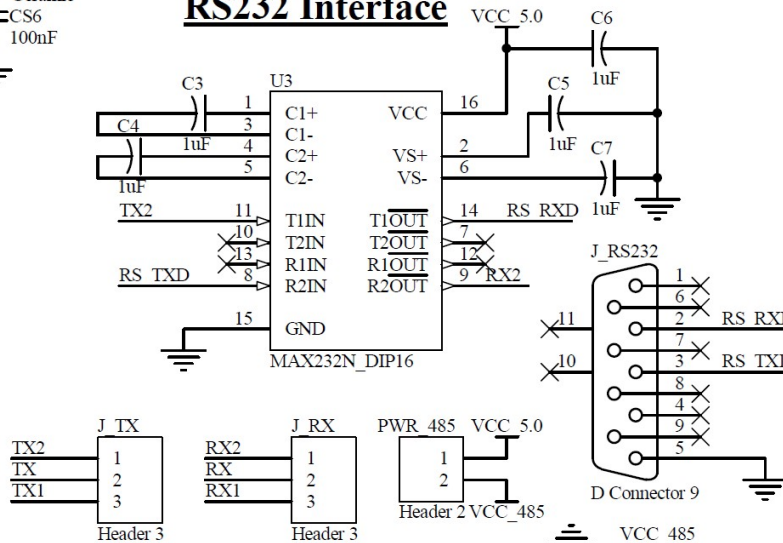


PIC16F877A

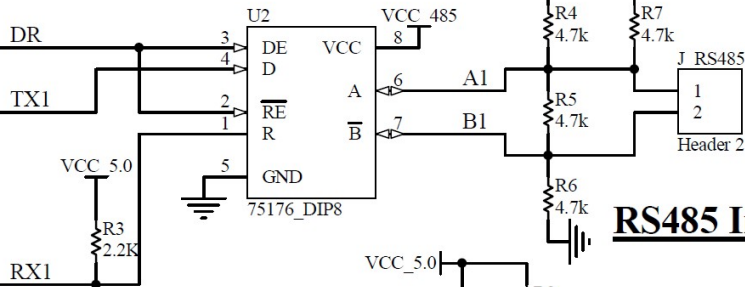
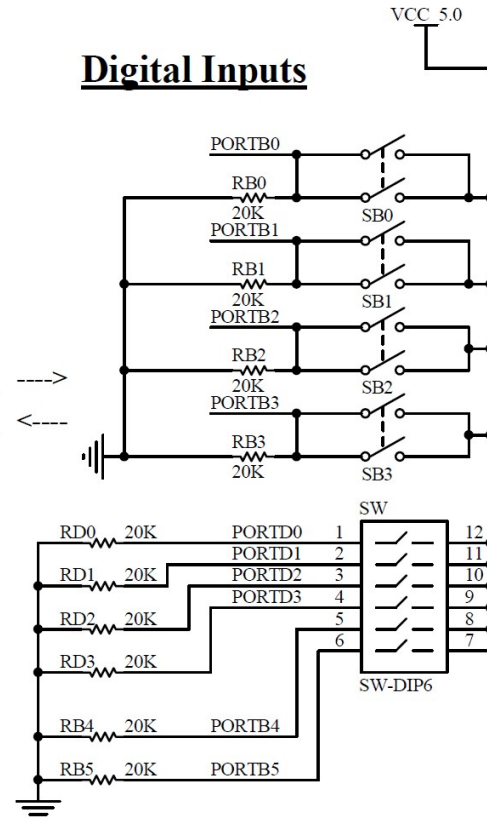




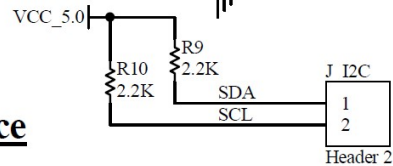
RS232 Interface



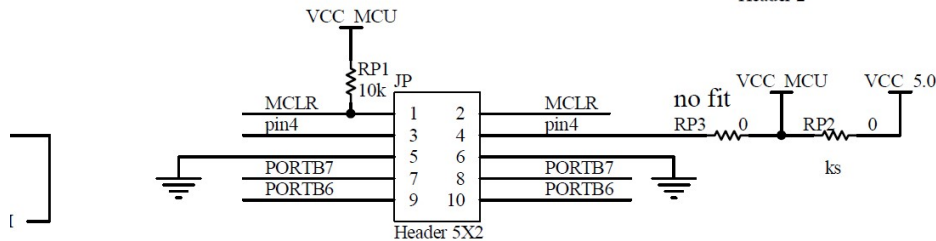
Digital Inputs



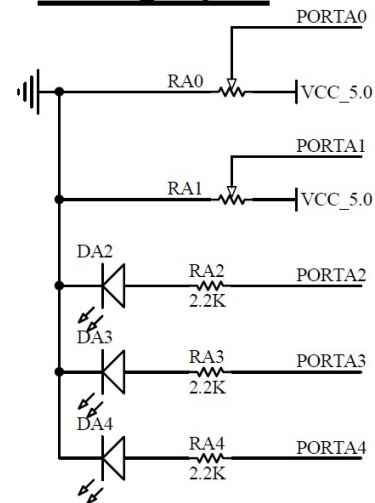
RS485 Interface

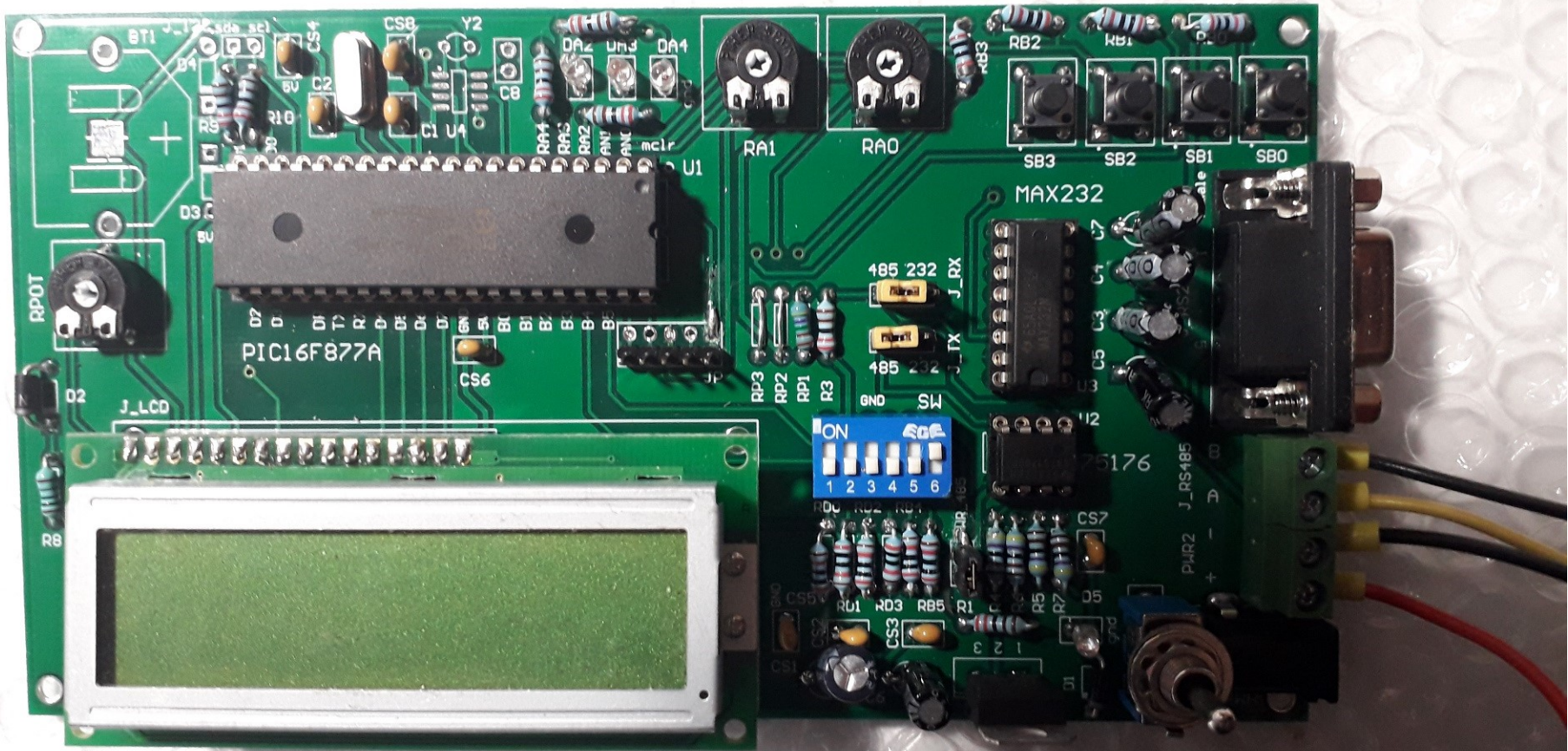


Programmer Interface



Analog Inputs





Raspored iskorišćenih ulaznih i izlaznih pinova mikrokontrolera

Port	Smer	Opis
RA2	izlaz	PinOperation - Led dioda. Simulira da je rampa u funkciji (svetli zeleno ili crveno svetlo)
RA3	izlaz	PinRampOpen - Led dioda. Simulira da je rampa otvorena
RB0	ulaz	PinTaster - taster - Zahtev vozača za preuzimanje kartice
RB1	ulaz	PinSensor – taster - Senzor prolaska vozila kroz rampu
RB4	ulaz	PinError – prekidač - Ako je RB4='1', simulira neispravan rad automata ili da kartice nedostaju
RB5	ulaz	PinEXTRampOpen – prekidač - Ako je RB5='1', rampa se otvara.
RD3, RD2	ulazi	Kategorija vozila
RC0	izlaz	RS - <i>Data/instruction select</i> kontrolni signal LCD displeja
RC1	izlaz	RW - <i>Read/Write</i> kontrolni signal LCD displeja
RC2	izlaz	E - <i>Enable</i> kontrolni signal LCD displeja
RC5	izlaz	DR - <i>Write/Read</i> kontrola RS485 drajvera SN65176; ako je DR='1' kolo je u <i>write</i> modu rada, u suprotnom - <i>read</i> mod
RC6	izlaz	TX - <i>Transmit</i> signal UART-a mikrokontrolera
RC7	ulaz	RX - <i>Receive</i> signal UART-a
RD4 - RD7	izlaz	Magistrala podataka za prenos informacija između mikrokontrolera i LCD displeja.

Raspored korišćenih promenljivih

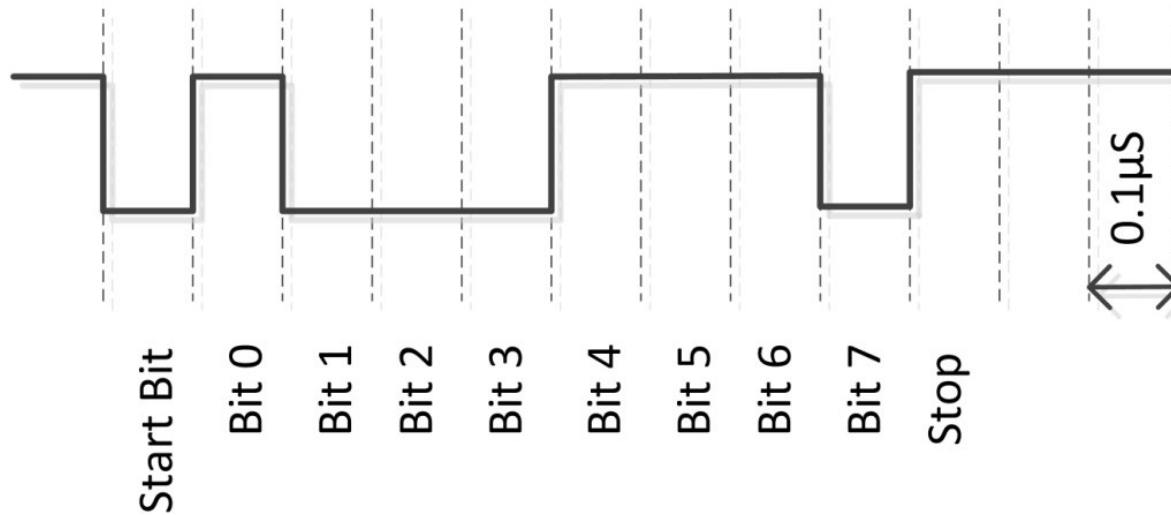
Promenljive	Tip podataka	Objašnjenje promenljivih
Operation, Operation2	<i>bit</i>	Rampa u funkciji ako je Operation=1, Operation se programira kada Slejv dobije komadu za radi; Operation2 uzima u obzir stanje na Error pinu.
RampOpen, RampOpen2	<i>bit</i>	Rampa podignuta ako je RampOpen=1, RampOpen se postavlja kada vozač dobije karticu; RampOpen2 uzima u obzir stanje na PinEXTRampOpen pinu.
Event	<i>bit</i>	Setuje se kada novo vozilo dođe na rampu, resetuje kada se informacije (sati, minuti, sekunde, kategorija) pošalju Masteru. Pritom se rampa otvara.
BytesToReceive	<i>unsigned char</i>	Broj bajtova koji treba da se primi do dekodiranja, time se ostvaruje kontrola primljenih podataka
Command	<i>unsigned char</i>	bajt komande koji se prima
CommandModified	<i>unsigned char</i>	bajt koji se vraća Masteru
CallFlag, RTCSetupFlag, UpdateLCDFlag	<i>bit</i>	Oznaka da je stigao zahtev za prozivkom i Slejv treba da vrati odgovor; Oznaka da je stigao zahtev za podešavanjem vremena i Slejv treba da vrati odgovor; Oznaka da Slejv treba da inkrementira vreme za 1 sekundu.

Raspored korišćenih funkcija

Naziv	Tip	Objašnjenje funkcija
init_variables()	<i>void</i>	inicijalizacija promenljivih
init()	<i>void</i>	podešava rad pinova mikrokontrolera, konfigurira periferije kontrolera, AD konvertor, prekide, kola tajmera; inicijalizuje UART
transmit()	<i>void</i>	obavlja slanje bajtova preko kola UART-a. Kao ulazni parameter funkcija uzima 8-bitni podatak.
UpdateLCD()	<i>void</i>	ispisuje stanje senzora, aktuatora i tajmera na LCD displej
processInputs()	<i>void</i>	prikuplja informacije sa senzora i menja stanja promenljivih koje čine status ulazne rampe. Funkcija se poziva periodično - nakon svake sekunde
interrupt()	<i>void</i>	koristi se za obradu prekida mikrokontrolera
main()	<i>void</i>	glavna funkcija programa


```
void transmit (unsigned char DATA8b)
{
    TXREG = DATA8b;
    while (!TXSTA.TRMT);
}
```

0x71, 8N1 (8 Data bits, No Parity, 1 Stop)



```
void DecodeTime () {  
    Seconds= (Sec_X10<<4) +Sec_X1;  
    Minutes= (Min_X10<<4) +Min_X1;  
    Hours= (Hour_X10<<4) +Hour_X1;  
}
```

```

void IncrementTime () {
    if (Sec_X1==9) {
        Sec_X1=0;
        if (Sec_X10==5) {
            Sec_X10=0;
            if (Min_X1==9) {
                Min_X1=0;
                if (Min_X10==5) {
                    Min_X10=0;
                    if
((Hour_X1==9) || ((Hour_X10==2) && (Hour_X1==3))) {
                        Hour_X1=0;
                        if (Hour_X10==2) {
                            Hour_X10=0;
                        } else Hour_X10++;
                    } else Hour_X1++;
                } else Min_X10++;
            } else Min_X1++;
        } else Sec_X10++;
    } else Sec_X1++;
}

```

```
void ConvertTime(unsigned char ch) {  
    X1=ch;  
    X10=0x00;  
    while (X1>9) {  
        X1=X1-10;  
        X10++;  
    }  
}
```

```

void UpdateLCD() {

    Lcd_Out(1, 1, "Time:");
    Lcd_Chr(1, 6, (Hour_X10+0x30));
    Lcd_Chr(1, 7, (Hour_X1+0x30));
    Lcd_Chr(1, 8, ':');
    Lcd_Chr(1, 9, (Min_X10+0x30));
    Lcd_Chr(1, 10, (Min_X1+0x30));
    Lcd_Chr(1, 11, ':');
    Lcd_Chr(1, 12, (Sec_X10+0x30));
    Lcd_Chr(1, 13, (Sec_X1+0x30));
    if (Operation2==0x01) Lcd_Out(2, 1, "Operating");
    else if (Error==0x01) Lcd_Out(2, 1, "Error    ");
    else Lcd_Out(2, 1, "          ");
    if (RampOpen2==0x01) Lcd_Out(2, 11, "Opened");
    else Lcd_Out(2, 11, "Closed");

}

```

```

void interrupt() {

    if ((PIE1.TMR1IE) && (PIR1.TMR1IF)) {
// prekid tajmera na svakih 100ms

        PIR1.TMR1IF = 0; // brise se flag
        if (Counter==9) {
            Counter=0;
            IncrementTime();
            DecodeTime();
            UpdateLCDFlag=1;
        }
        else Counter++;

        if (Counter2>0) Counter2--;
        else Counter2=0;

        ProcessInputs();

        TMR1H = 0x0B; // startne vrednosti Tajmera 1
        TMR1L = 0xDC;
    }
}

```


Master

Prozivka

KOMANDA		ID	RAMPE	Y $\in\{0,1\}$			
0	0	1	X	Y	Y	Y	Y

Podešavanje RTC. Posle ovog bajta, Slejv automat šalje još tri dodatna bajta: *sekunde, minuti, sati*

KOMANDA		ID	RAMPE	Y $\in\{0,1\}$			
0	1	1	X	Y	Y	Y	Y

Slejv

Automat nema kartica.

KOMANDA		ID	RAMPE	Y $\in\{0,1\}$			
0	0	0	X	Y	Y	Y	Y

Automat radi ali nije pritisnut taster

KOMANDA		ID	RAMPE	Y $\in\{0,1\}$			
0	0	1	X	Y	Y	Y	Y

Pritisnut je taster, tj. prošlo je vozilo kroz rampu. Posle ovog bajta, Slejv automat šalje još četiri dodatna bajta: *sekunde, minuti, sati i kategorija vozila.*

KOMANDA		ID	RAMPE	Y $\in\{0,1\}$			
0	1	0	X	Y	Y	Y	Y

Odgovor da je RTC podešen

KOMANDA		ID	RAMPE	Y $\in\{0,1\}$			
0	1	1	X	Y	Y	Y	Y

```

if ((PIE1.RCIE) && (PIR1.RCIF)) {

    ch=RCREG; // citanje primljenog bajta
    if (BytesToReceive==0x00) {
        if ((ch&0x0F)== RAMP_ID) { // adresa slejva se poklapa
            Command=ch;
            if ((ch&0xE0)==0x20) { // bajt prozivke 001
                BytesToReceive=0x00;
                CallFlag=1;
            }
            else if ((ch&0xE0)==0x60) { // 011
// primljeni bajt za podesavanje sata realnog vremena

                BytesToReceive=0x03;
                Counter2=3; // 300 ms je vreme tokom kojeg
                    // treba da stignu preostali bajtovi
            }
        } // od if ((ch&0x0F)== RAMP_ID
    } // od if (BytesToReceive==0x00)

```

```

else if (BytesToReceive==0x03) {
    BytesToReceive=0x02;
    ch=ch-0x30;
    if (ch>59) ch=59;
    ConvertTime(ch);
    Tmp_Sec_X1=X1; Tmp_Sec_X10=X10;
}
else if (BytesToReceive==0x02) {
    BytesToReceive=0x01;
    ch=ch-0x30;
    if (ch>59) ch=59;
    ConvertTime(ch);
    Tmp_Min_X1=X1; Tmp_Min_X10=X10;
}
else if (BytesToReceive==0x01) {
    BytesToReceive=0x00;
    ch=ch-0x30;
    if (ch>23) ch=23;
    ConvertTime(ch);
    Tmp_Hour_X1=X1; Tmp_Hour_X10=X10;
    RTCSetupFlag=1;
}
} // od if ((PIE1.RCIE) && (PIR1.RCIF))
} // od interrupt()

```

mikroC PRO for PIC v.6.6.3 - D:\Users\Borko\Predmeti\Projektovanje elektronskih sistema\Racunske vezbe\Naplatne rampe\Sljv\Program\Slave.mcppi

File Edit View Project Build Run Tools Help

Code Explorer Start Page Slave.c

```

• // definicija ulaznih pinova
• #define PinTaster PORTB.F0
• #define PinSensor PORTB.F1
•
• #define PinError PORTB.F4
• #define PinEXTRampOpen PORTB.F5
•
• #define PinOperation PORTA.F2
10 #define PinRampOpen PORTA.F3
• #define PinEvent PORTA.F4
•
• #define DR PORTC.F5
•
• unsigned char RAMP_ID = 0x00;
• bit Operation;
• bit Operation2;
•
• bit RampOpen;
• bit RampOpen2;
20
• bit Event;
• bit EXTRampOpen;
• bit Error;
• bit Sensor;
•
• unsigned char Category = 0x00;
•
• // ID broj rampe
• unsigned char BytesToReceive = 0x00;
30 // broj bajtova koji treba da se primi do dekodiranja
• unsigned char ch = 0x00;
• // primljeni bajt
• unsigned char Command = 0x00;

```

Project Settings

Device

Name: P16F877A

MCU Clock

Frequency: 20.000000 MHz

Build/Debugger Type

Build Type

Release ICD Debug

Debugger

Software mikroLCD

Project Manager [1/1] - Slave.mcppi

Slave.mcppi

- Sources
- Header Files
- Binaries
- Project Level Defines
- Image Files
- EEPROM Files
- Active Comments Files
- Output Files
- Other Files

Library Manager Project Explorer

LibStok

- FLASH
- Glcd
- Glcd_Fonts
- I2C
- Keypad4x4
- Lcd
- Lcd_Constants
- Manchester
- MemManager
- One_Wire
- Port_Expander
- PS2
- PWM
- RS485
- Software_I2C
- Software_SPI
- Software_UART
- Sound
- SPI
- SPI_Ethernet
- SPI_Ethernet_24j600
- SPI_Glcd
- SPI_Lcd
- SPI_Lcd8
- SPI_T6963C
- Sprinti
- Sprinti
- T6963C
- Time
- TouchPanel
- Trigonometry
- UART

Messages Quick Converter

Errors Warnings Hints

Line	Message No.	Message Text

430: 13 Insert D:\...Borko\Predmeti\Projektovanje elektronskih sistema\Racunske vezbe\Naplatne

Edit Project



Oscillator Selection

HS oscillator

Watchdog Timer

Disabled

Power-up Timer

Disabled

Brown-out Reset

Disabled

Low-Voltage (Single-Supply) In-Circuit Serial Program...

Disabled

Data EEPROM Memory Code Protection

Disabled

Flash Program Memory Write

Disabled

In-Circuit Debugger Mode

Disabled

Flash Program Memory Code Protection

Disabled

MCU and Oscillator

MCU Name P16F877A

MCU Clock Frequency [MHz] 20.000000

Build Type

Release ICD Debug

Heap

Size 2000

Configuration Registers

```
CONFIG : $2007 : 0x2F0A
```

Load Scheme

Save Scheme

Default

OK

Cancel

General Output Settings ...



Optio... Operate Power Settings x

Operate

Family: Mid-Range 8-bit MCUs (PIC10/12/1...
Device: PIC16F877A
Tool: PICKit3 S.No : BUR184352307

Apply Disconnect

Results

CP=OFF Checksum: 0FCF
Checksum: 0FCF
Pass Count: 153
Fail Count: 4
Total Count: 157

Program Erase Read Verify Blank Check

Hex File: Click on browse to select a hex file
SQTP File: Click on browse to select a SQTP file

Browse Clear selection

Browse Clear selection

Output - IPE x

```
Firmware type.....PIC18F
Now Downloading new Firmware for target device: PIC16F877A
Downloading AP...
AP download complete
Programming download...

Currently loaded firmware on PICKit 3
Firmware Suite Version.....01.55.01
Firmware type.....Midrange
Target voltage detected
Target device PIC16F877A found.
Device Revision ID = 8
```

Optio... Operate Power Settings x

Operate
Power
Memory
Environment
SQTP
Production
Settings
Logout

Device and Tool Selection

Family: Mid-Range 8-bit MCUs (PIC10/12/1...
Device: PIC16F877A
Tool: PICKit3 S.No : BUR184352307

Apply Disconnect

Results

CP=OFF Checksum: 05BB
Checksum: 05BB
Pass Count: 154
Fail Count: 4
Total Count: 158

Program Erase Read Verify Blank Check

Hex File: D:\Users\Borko\Predmeti\Projektovanje elektronskih sistema\Racunske vezbe\Naplatne rampe\Slej... Browse Clear selection
SQTP File: Click on browse to select a SQTP file Browse Clear selection

Output - IPE x

```
2020-09-23 18:45:56 +0200 - Programming...  
  
Device Erased...  
  
Programming...  
  
The following memory area(s) will be programmed:  
program memory: start address = 0x0, end address = 0x4ef  
configuration memory  
Programming/Verify complete  
2020-09-23 18:46:02 +0200 - Programming complete
```