

# **Mikrovezérlők alkalmazása**

## **Projekt feladat**

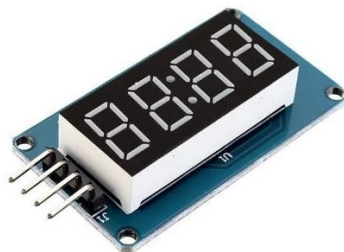
### **L06-06: Stopperóra**

**Kárpáti Péter Milán**

**VW19BD**

**Kaszás Péter**

**HGKW8H**



**2023-12-07**

## Tartalomjegyzék

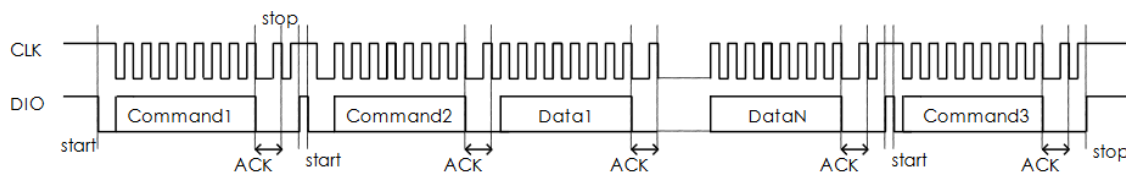
1	A Feladat ismeretése .....	3
2	A TM-1637 modul működése .....	3
3	A programkód ismertetése .....	3
4	A nyák felépítése .....	6

## 1 A Feladat ismeretese

A feladatunk az L06-06-os feladat volt, melyben egy stopper órát kellett készítenünk start, stop és körítő mérés funkcióval. Az időt a TM-1637-es modulon kellett megjelenítenünk. A start, stop és körítő mérés funkciókat a mikrovezérlőn található gombokkal kapcsolhatjuk. Emellett egy számítógépes programot is kellett készítenünk, amely távolról irányítja a stoppert, logolja és megjeleníti a mérési adatokat. A számítógépes programról is lehetőségünk van vezérelni a modult.

## 2 A TM-1637 modul működése

A TM1637 modul az I<sup>2</sup>C-hez hasonló soros kommunikációs protokollal működik. A legfőbb különbség, hogy ebben az esetben nem szükséges Slave address címet kiküldeni. A modullal való kommunikációs séma az alábbi ábrán látható.



1. ábra TM-1637 kommunikációja

1. Parancs: az adatírás és az adatolvasás beállítására szolgál.
2. Parancs: a kijelző regisztercímének beállítására szolgál.
3. Adat
4. Parancs: kijelző fényerejének beállítása.

## 3 A programkód ismertetése

Az adatkiküldést a CLK, illetve a DIO lábainak fel és lehúzásával lehet vezérelni. Magát a kommunikációt pedig az alábbi függvény teszi lehetővé:

```
void comm(unsigned char c){  
    for(int i=0;i<=7;i++){  
        setCLKlow();  
        if((c>>i)&0x01){  
            setDIOhigh();  
        }  
        else{  
            setDIOlow();  
        }  
        setCLKhigh();  
    }  
}
```

Ahol a függvény bitenként vizsgálja az adatot és ennek megfelelően a DIO pin értékét magasra vagy alacsonyra állítja. A teljes adatküldés pedig a következőképpen hajtodik végre:

```
startComm();
startComm();
comm(adress);
checkACK();

//ADAT KÜLDÉS
comm(digits[0]);
checkACK();
comm(digits[1]+128);
checkACK();
comm(digits[2]);
checkACK();
comm(digits[3]);
checkACK();

//DISPLAY SETTING
startComm();
comm(brightness);
checkACK();
stopComm();
```

Látható, hogy az adatküldés az 1. ábra alapján hajtodik végre. A programrészleten még az is látszik, hogy minden egyes adatkiküldést egy acknowledgment követ, amely, úgy történik, hogy a TM-1637 modul a nyolcadik órajel után lehúzza az adatvezetékét. Ez annak ellenőrzésére szolgál, hogy a modul megkapta-e az adott adatcsomagot. Az acknowledgment ellenőrzését az alábbi függvény végzi:

```
void checkACK(){
    setCLKlow();
    DELAY_US(defdelay);
    DIO_SetDigitalInput();

    while(DIO_GetValue()==1);
    setCLKhigh();
    DELAY_US(defdelay);
    setCLKlow();
    DIO_SetDigitalOutput();
    setDIOlow();
}
```

A gombok lekezelését állapotgéppel valósítottuk meg, melynek az állapotai a következők:

```
enum States {
    INIT,
    STOP,
    STOPPER,
    KOR
};
enum States STATE=INIT;
```

- **case INIT:**

Az INIT állapotban reseteljük és megállítjuk a timert és az egyes változók értékét. Az SW1 gomb megnyomásával pedig átlépünk a STOPPER állapotba.

- **case STOPPER:**

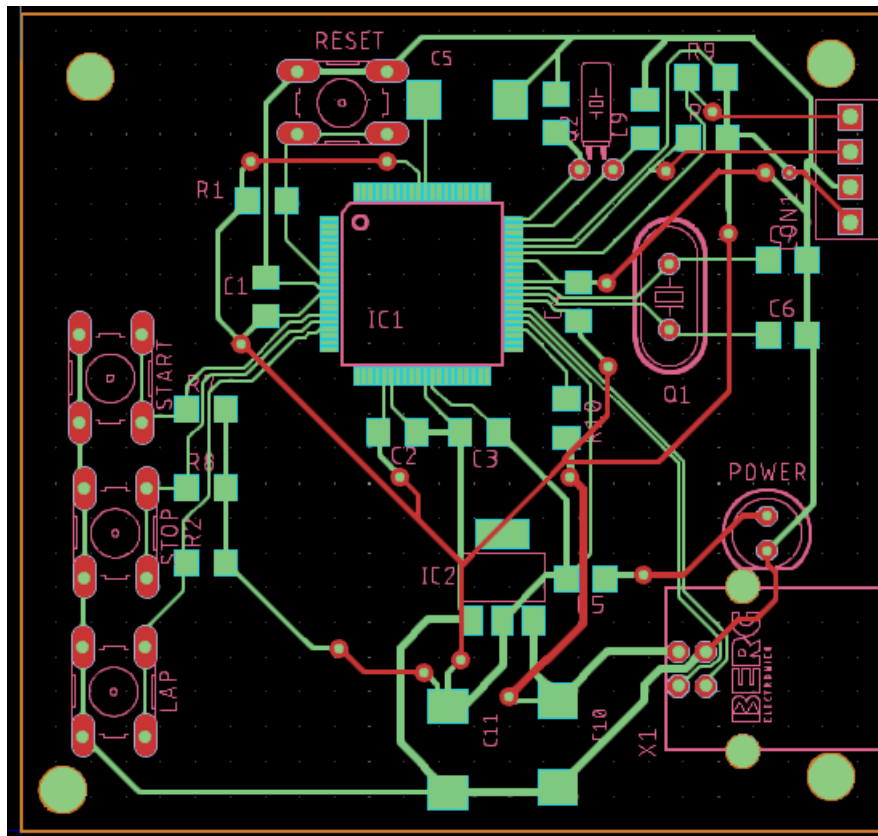
Ez egy köztes állapot, innen az SW3 gomb megnyomásával mentjük a köridő értékeket, az SW2 gomb nyomásával pedig átmegyünk a STOP állapotba.

- **case STOP:**

A timert megállítjuk. SW2 újbóli megnyomásával átugrunk az INIT állapotba és nullázzuk az órát. SW1 megnyomásával pedig újra elindíthatjuk azt, értelemszerűen ilyenkor visszalépünk a STOPPER állapotba.

Az egyes adatokat átkonvertálás után USB-n keresztül küldjük át a PC-s programnak.

## 4 A nyák felépítése



2. ábra: A stopperóra mikrovezérlője

A stopper vezérlését a PIC24FJ256GB108 mikrokontroller végzi. Az órajelet a 12 MHz-es oszcillátor szolgáltatja. Emellett a stopper szempontjából kiemelt fontosságú a másodlagos oszcillátor, amely a stopper időzítéséért felel. Továbbá lényeges kiemelni, hogy a TM-1637 modul az I<sup>2</sup>C pinekre csatlakozik, amely felhúzó ellenállásokkal rendelkezik. A nyákon emellett megtalálhatóak a stopper vezérlésére szolgáló gombok, illetve a Master Clear Reset gomb, amely a mikrovezérlő újraindítására szolgál. A számítógéppel való kommunikációra USB-t használunk illetve ez szolgáltatja a stopper tápellátását is. A mikrovezérlő 3,3V-on működik ehhez szükség van egy LM1117-33 Lineáris feszültség regulátorra, amely az USB-ből érkező 5V-ból 3,3V-ot állít elő.