

dátum:

a mérést végezte:

USART egység használata

- m é r é s i j e g y z ő k ö n y v -

1. Soros kommunikáció a PIC-ben futó felhasználói program és a PC között, az USART egység használata

A PIC-be épített USART (USART=Universal Synchronous Asynchronous Receiver Transmitter) egységet regisztereken keresztül az alábbi módokban lehet felprogramozni és használni:

- aszinkron kommunikáció,
- szinkron kommunikáció - Master üzemmód,
- szinkron kommunikáció - Slave üzemmód.

Az elsőt többnyire PIC és PC közötti soros adattovábbítás céljaira használják, míg a következő kettő PIC és perifériák (*AD konverter, EEPROM*) közötti kommunikációt segíti.

A *mikroPascal* szerencsére tartalmaz egy *USART Library* beépített könyvtárat az *USART* egység kezelésére, így annak beállításait a PIC regisztereinek írása helyett egyszerű parancsokkal végezhetjük.

Egy nagyon egyszerű feladatot tűzzünk ki magunk elé célul, a PIC-ben levő program küldjön adatokat (számokat) a PC-n futó terminálablak számára.

```
program Usart_teszt;
var i:byte; t1, t2:string[20];
{-----}
begin
  i:=0; t1:='Számokat küldözget'; t2:='ek a következőkben:';
  Usart_Init(9600);
  Usart_Write_Text(t1);
  Usart_Write_Text(t2);
  while true do begin
    Usart_Write(i);
    Delay_ms(500);
    inc(i);
  end;
end.
```

A program elején feltöltünk kezdőértékkel két karaktertömböt (t_1 és t_2), majd az *USART* egység 9600 *baud* átviteli sebességre való felprogramozása után (*USART_Init*), ezt a két tömböt kiküldjük a PC *soros portja* felé. A továbbiakban belépünk a végtelen ciklusba, ahol félmásodpercenként egy *byte*-os változó lépésenként növelt értékét továbbítjuk sorosan.

Hogy mindezt kellőképpen élvezhessük a PC oldalon szükségünk van egy terminálablakra, amit a *mikroPascal Tools/USART Terminal* menüjével nyithatunk. A terminálablakban egy *Connect*-el kapcsolódjunk PIC-ünkhöz és figyeljük a beérkező adatokat különböző formátumokban *ASCII, HEX, DEC*!

Figyeljük meg azt is, mi történik miután reseteljük a PIC-et (vajon mi az a közvetlen újraindítást követő adatfolyam?)!

Megj.: A *bootloader* használata előtt, mindenképp zárjuk le a terminálablak soros kommunikációját, különben nem tudjuk új programmal frissíteni a PIC tartalmát!

2. Valósítsunk meg kétirányú kommunikációt , egy menüvezérlés vázával demonstráljuk is.