

Programová knihovna

ZigBee modulu

AP13192USLK

Programová knihovna

- Obsahuje funkce, které řídí komunikační modul. Prototypy funkcí jsou uvedeny v souboru **ZigBee.h**, samotný kód pak v souboru **ZigBee.c**.
- Pro začlenění knihovny do vašeho projektu vložte řádek:
`#include "ZigBee.h"`
- A definujte přerušení v souboru „Project.prm“ takto:
`VECTOR 0 _Startup`
`VECTOR 2 IRQIsr // toto je přerušení vyvolané modulem`

Funkce knihovny

- Nastavení modulu (vysílací frekvence, výkon)
- Adresace odesilatele a příjemce
- Odesílání dat
- Příjem dat

- Kontrola přijatých dat pomocí 8bit CRC součtu
- Kontrola počtu přijatých vadných paketů
- Kontrola počtu žádostí o přeposlání paketu
- Nastavení interního zdroje impulzů na 20MHz

Funkce Init_ZigBee

- Proveďte inicializaci modulu.
- **Funkci se předávají 3 parametry:**
 - Číslo kanálu (1 až 15)
 - Číslo modulu (1 až 255)
 - Vysílací výkon (0 až 15)

Pozn.:
Funkce knihovny vrací
datový typ *unsigned char*.

- Při nedodržení hodnot vrací funkce hodnotu **0**.
- Při dodržení hodnot vrací funkce hodnotu **1**.

Init_ZigBee – příklad použití

- Prototyp funkce:

```
unsigned char Init_ZigBee(unsigned char c_kanal, unsigned char  
Nr_teto_stanice, unsigned char vykon);
```

- Příklad volání funkce:

```
unsigned char stav;
```

```
stav = Init_ZigBee(10,1,15);
```

Funkce Send_Data

- Odešle vaše data určenému příjemci.

- **Funkci se předávají 3 parametry:**

- Data (ukazatel na datový typ unsigned char)
- Počet bajtů, které chcete z bufferu odeslat (1 až 118)
- Číslo modulu, kterému jsou data určena (1 až 255)

Pozn.:
Pokud po odeslání paketu
není potvrzeno jeho přijetí,
je paket automaticky přeposlán.
Maximálně však 4x.

- Při nedodržení hodnot vrací funkce hodnotu **0**.
- Pokud příjemce data přijal, vrací hodnotu **1**.
- Pokud nepřijal, vrací hodnotu **2**.

Send_Data – příklad použití

- Prototyp funkce:

```
unsigned char Send_Data(unsigned char *data, unsigned char Qty_Tx_B,  
unsigned char příjemce);
```

- Příklad volání funkce:

```
unsigned char stav1, stav2;  
unsigned char *text;  
unsigned int cislo = 1500;  
text = "Hello world";  
  
stav1 = Send_Data(text,5,2);  
stav2 = Send_Data((char*)&cislo,2,2);
```

Funkce Read_Data

- Čeká na přijetí dat od určitého odesilatele po stanovenou dobu.
- **Funkci se předávají 4 parametry:**
 - Ukazatel na buffer, do kterého se mají zapsat přijatá data
 - Počet bajtů, které se mají do bufferu zapsat (1 až 118)
 - Číslo modulu, od kterého mají data přijít (1 až 255)
 - Doba čekání (1 až 0xFFFFF -> 0 až 67,109 sekund)
- Při nedodržení hodnot vrací funkce hodnotu **0**.
- Jsou-li data přijata, vrací **počet zapsaných bajtů**
- Nebyla-li v dané době data přijata, vrací hodnotu **255**.

Read_Data – příklad použití

- Prototyp funkce:

```
unsigned char Read_Data(unsigned char *st_buffer, unsigned char l_buffer,  
unsigned char odesilatel, unsigned long doba_cekani);
```

- Příklad volání funkce:

```
unsigned char stav;  
unsigned char my_buffer[119];  
  
stav = Read_Data(my_buffer,25,2,0xFFFF);
```

Doba čekání je 0,262 sekund, dle vzorce:

$$T_{\text{wait}} = \frac{0xFFFF * \frac{1}{0,25}}{1 * 10^6}$$

, kde hodnota 0,25 je frekvence časovače modulu v MHz.

Indikace CRC chyb

- **Funkce Get_CRC_Errors**
- Vrací **počet přijatých vadných paketů** (CRC ERROR) po ukončení funkce Read_Data.
- Funkci se nepředává žádný parametr.
- Pozn: Není nutné znovu volat funkci Read_Data – při přijetí vadných dat se čeká automaticky dál (interval čekání je prodloužen).
- **Funkce Get_CRC_Warnings**
- Vrací **počet žádostí o přeposlání** paketu (příjemce přijal vadná data) po ukončení funkce Send_Data.
- Funkci se nepředává žádný parametr.
- Pozn: Není nutné volat funkci Send_Data – přeposlání je automatické.

Indikace CRC chyb – použití

- Prototypy funkcí:

```
unsigned char Get_CRC_Errors(void);  
unsigned char Get_CRC_Warnings(void);
```

- Příklad volání funkcí:

```
unsigned char stav1, stav2;  
unsigned int CRC_Errors, CRC_Warnings;  
  
stav1 = Read_Data(my_buffer, 25, 2, 0xFFFF);  
CRC_Errors = Get_CRC_Errors();  
  
stav2 = Send_Data("Hello world", 11, 2);  
CRC_Warnings = Get_CRC_Warnings();
```

Funkce Init_Clock_Generator

- Funkce nastaví frekvenci interního zdroje impulzů na 20MHz.
- Funkci se nepředává žádný parametr.
- Externí zdroj impulzů (COM port) pak není potřeba a je možné kabel RS232 po flashnutí programu odpojit.
- Funkci je potřeba volat hned na začátku vašeho programu.

Init_Clock_Generator - použití

- Prototyp funkce:

```
void Init_Clock_Generator(void);
```

- Příklad volání funkce:

```
Init_Clock_Generator();
```

Příklad

- **Zadání – stanice č.1:**
- Modul vysílá stav tlačítek SW₁ až SW₄.
- Modul vysílá 10 bitovou hodnotu z A/D převodníku.
 - 1 = vysílá , 0 = nevysílá
- Na LCD displeji je zobrazeno:
 - Počet odeslaných bajtů
 - Počet přijatých bajtů
 - Počet ztracených bajtů (odeslány, ale nepotvrzeny)
 - Počet přijatých žádostí o přeposlání paketu
 - Počet odeslaných žádostí o přeposlání paketu

Příklad

- **Princip – stanice č.1:**

Inicializovat modul, displej, povolit přerušení od tlačítek a AD převodníku.

V cyklu čekat na data a kontrolovat zda přišla informace, že se má vysílat.

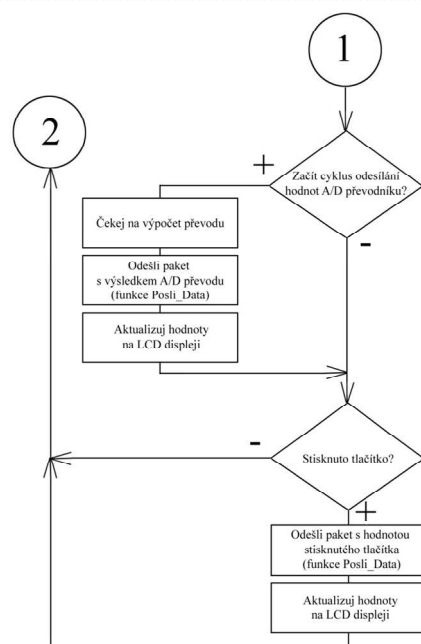
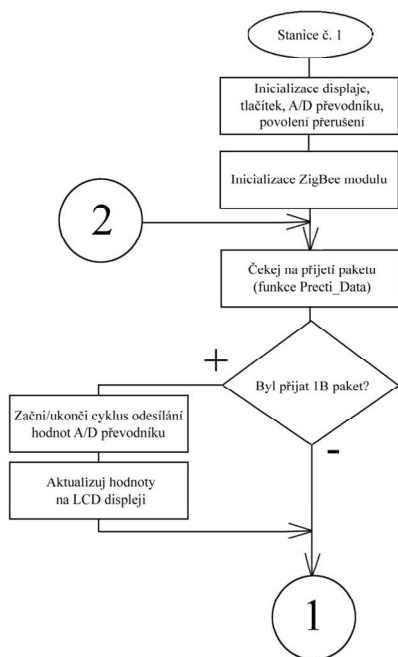
Pokud se má vysílat, začne se neustále odesílat hodnota A/D převodu (potenciometr).

V přerušení od tlačítka se pouze kontroluje, které tlačítko bylo stisknuto a hodnota tlačítka se uloží. V cyklu se tedy čeká na data, poté se odesílá hodnota A/D převodu nebo tlačítka (bylo-li stisknuto).

Pro získání hodnot, které se mají vypisovat na LCD, je potřeba kontrolovat hodnoty, které jsou vráceny funkcemi knihovny.

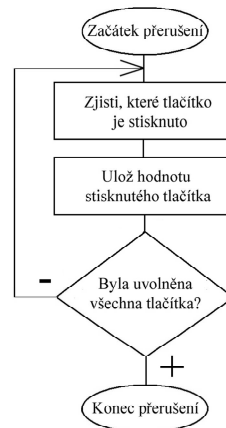
Pozn.:
A/D převod trvá 14us.
Je potřeba tuto dobu vždy čekat a
Kontrolovat stav registru ATD1SC.

Stanice č.1 – Vývojový diagram



Stanice č.1 – Vývojový diagram

Přerušení od SW



Příklad

- **Zadání – stanice č.2:**
- Modul přijímá stav tlačítek a rozsvěcuje nebo zhasíná LED diody 1 až 4.
- Modul zobrazí přijatý údaj z A/D převodníku na displeji, převede jej na teplotu v rozsahu 0-100 °C a také zobrazí.
- Na LCD displeji je zobrazeno:
 - Počet odeslaných bajtů
 - Počet přijatých bajtů
 - Počet ztracených bajtů (odeslány, ale nepotvrzeny)
 - Počet přijatých žádostí o přeposlání paketu
- Modul posílá informaci (tl. SW₁) zda má stanice č.1 odesílat výsledek A/D převodu.

Příklad

• Princip:

Inicializovat modul, displej, povolit přerušení od tlačítka.

V cyklu čekat na data a zároveň kontrolovat, zda nebylo stisknuto tlačítko SW1.

Pokud přišla data, musí se zkontrolovat zda byl přijat 1B (LED diody) nebo 2B (výsledek AD).

Podle toho rozsvítit LED diodu nebo zobrazit hodnotu na displeji.

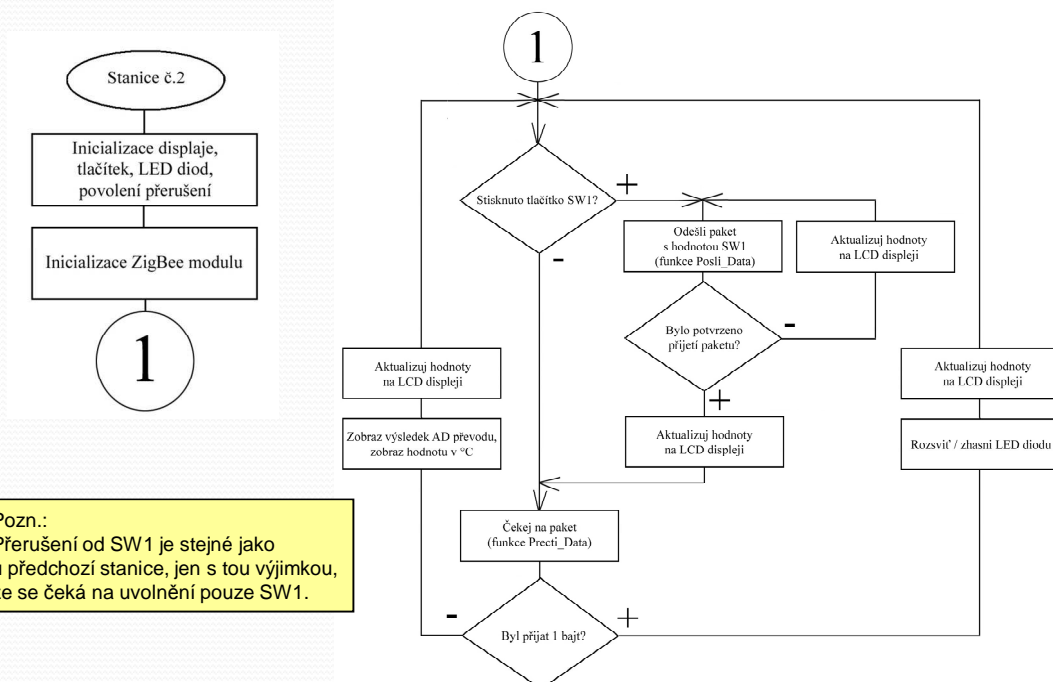
• Pozn.:

Jestliže stanice č.1 neustále odesílá výsledky A/D převodu, je možné, že nebude ve stavu, ve kterém by mohla přijmout informaci o tom, že má přestat s vysíláním.

• Po stisknutí SW1 je tedy potřeba kontrolovat, zda byla informace doručena, případně paket přeposlat.

• Pro získání hodnot, které se mají vypisovat na LCD, je potřeba kontrolovat hodnoty, které jsou vráceny funkcemi knihovny.

Stanice č.2 – Vývojový diagram





Konec prezentace

Nyní jste schopni zvládnout obsluhu ZigBee modulu.