

Mikrokontrollerek Alkalmazástechnikája Házi Feladat

Szikra István részére

Neptun kód: URLJRN

Feladat címe: **Infra Távirányító vevő, dekódoló**

Feladat konzulense: **dr. Benesóczky Zoltán**

Feladat leírása:

A feladat célja egy olyan mikrokontrolleres rendszer megtervezése és megvalósítása, amely képes SPACE ENCODING kódolással küldött infra jel demodulálására, kódminták felismerésre, az N bites kód visszafejtésére és átküldésére számítógépre.

A számítógéppel a kommunikáció RS232 porton soros kommunikációval valósul meg. A dekódoláshoz szükséges paraméterek (bitszám, fejléc, kifutórész, ismétlés, '1', '0' kódmintái) a mikrokontroller EEPROM-jában tárolódnak, és számítógépről átkonfigurálhatók. A számítógépről egyéb műveletek is végezhetők (diagnosztika).

Specifikáció: Az egység 36kHz-el modulált SPACE ENC kódolású infra jel vételére alkalmas. A kódolás default beállításait egy AIWA RC-AVR15 típusú távirányítóra határoztam meg (winlirc program analízátor funkciója, és hangkártyával történő digitalizálás alapján). Ezek a következők:

- bits	32	bitszám
- header	8981 4446	fejléc
- one	623 514	egy
- zero	623 1626	nulla
- ptrail	623	kifutó rész
- repeat	8980 2199	ismétlés

Az időzítési értékek μ s-ban értendők, és átlag értékek.

A belső számláló 128 μ s-onként lép egyet (1024/8MHz). A paraméterekben tartományokat kell megadni (a jel időzítési- és a "mintavételből" adódó pontatlanság miatt) amik a számláló értékére vonatkoznak. Pl. ebben az esetben 8981 μ s/128 μ s-ra 70.1 adódik, ezért a paramétert [70-72]-re választottam, 4446 μ s/128 μ s-ra 34,73 -> [34-36]... 514 μ s/128 μ s -> 4.01 -> [3-6]...

Ezzel a beállítással többféle távirányító is működik, de új konfiguráció is beállítható az alábbi módon:

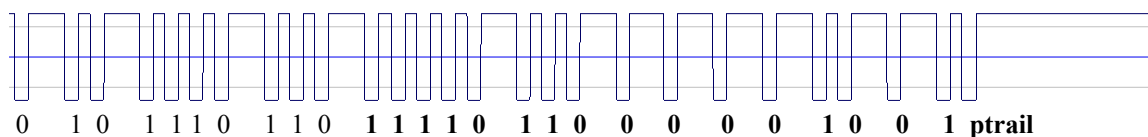
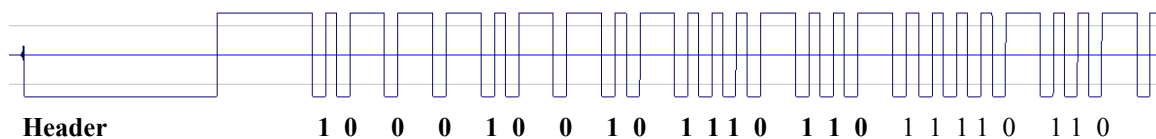
Soros porton (38400 kbps-mal, paritás nélkül, 8 adatbittel) kiküldjük a konfigurációs mód kiválasztását : 'C' (byte); majd a bitszámot 1..64 (1 byte); fejlécet (alacsony min, alacsony max, magas min, magas max) (egyenként 1..255) (4 byte); ismétlést (4 byte); 'egy' -t (4 byte); 'nulla' -t (4 byte) és végül a kifutó rész időzítéseit (min, max) (2 byte). Ezután az EEPROM-ba való mentéshez ki kell küldeni egy 'S' -et (byte).

Az EEPROM-ból a paraméterek betöltését 'L' -el végezhetjük. Amennyiben az EEPROM tartalmaz elmentett konfigurációt, az reset után automatikusan betöltődik, ellenkező esetben a default beállítások lesznek érvényben.

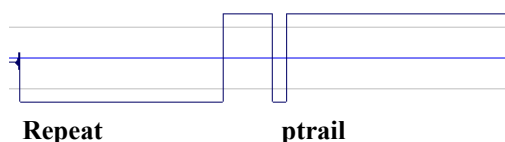
Az EEPROM tartalmának olvasását 'E', a RAM tartalmának olvasását 'R' küldésével kezdeményezhetjük, és ezután a mikrokontroller a soros porton továbbítja a kért adatokat. Így lehetőség van a kódolási paraméterek meghatározására a memória dump-ból is.

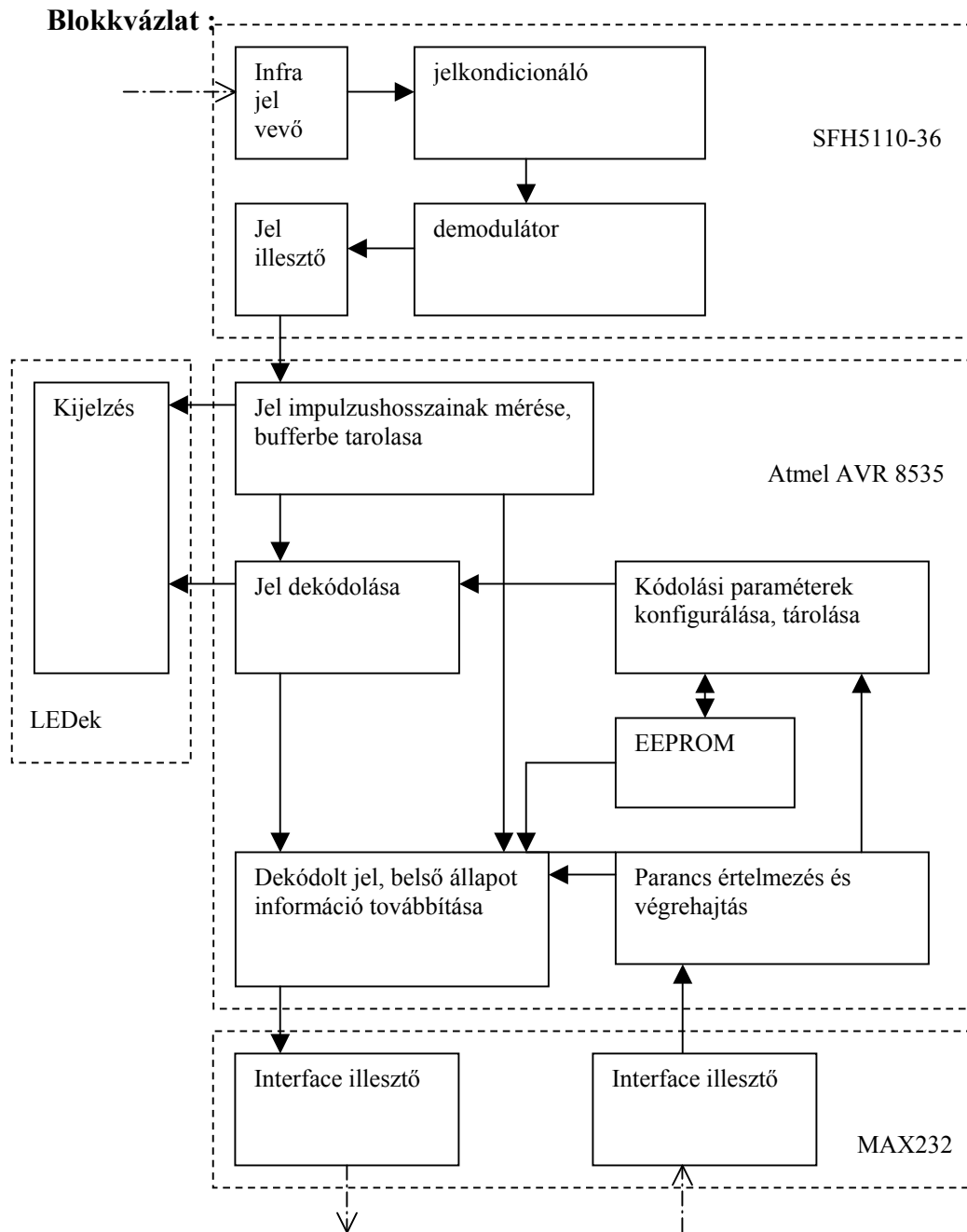
SPACE ENCODING kódolás (az alábbi jeleket hangkártyával digitalizáltam 48kHz-el):

Kód: *16 bites távirányító kód (0x8976), 16 bites gomb kód (0x3EC1)*



Ismétlés:





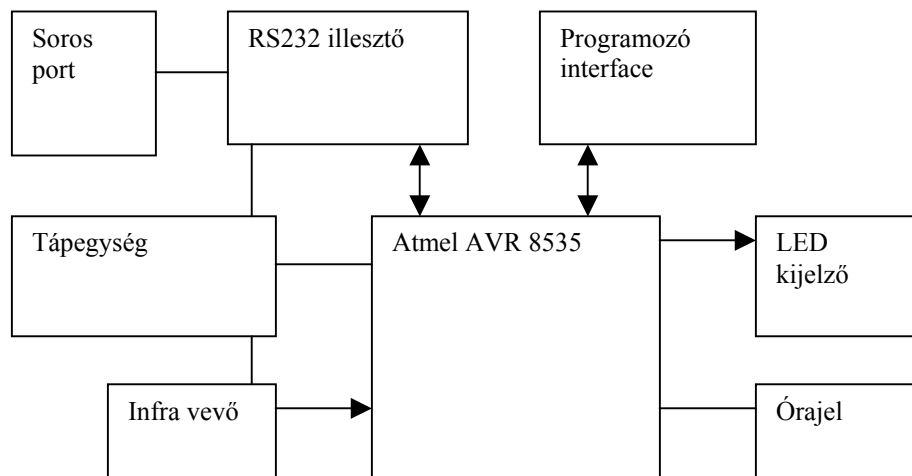
A jelkondicionáló felerősíti az infra vevő által vett jelet, és a demoduláló kiválasztja a 36kHz-es vivőfrekvenciájú komponenst, ezután az illesztő TTL jelszinté alakítja. A következő blokk feladata a jelben megtalálható impulzusok hosszának a megmérése, az adatok tárolása. Ha impulzus érkezik, azt a kijelző felé is továbbítja, az időzítési adatokhoz pedig hozzáférést biztosít a dekódoló, és a kommunikációs egység számára. A dekódoló előállítja a kódot az időzítési adatokból a kódolási paraméterek alapján, és a dekódolás folyamatát kijelzi. Ezután a kimeneti kommunikációs egység továbbítja a számítógép felé az illesztőn keresztül. A bemeneti interfészillesztőn keresztül

az adat a parancsértelmezőbe kerül, ami az adat alapján elvégzi a szükséges művelet. Ez lehet:

- az időzítési adatok kiolvasása, amikor a kommunikációs egység továbbítja a buffer tartalmát a PC felé
- a kódolási paraméterek változtatása, amikor a PC-ről érkező adatokat betölti a paramétertárolóba
- paraméterek mentése, amikor a paramétertároló a paramétereket az EEPROM-ba írja
- paraméterek töltése, amikor a paramétertároló beolvassa a paramétereket az EEPROM-ból
- EEPROM olvasás, amikor a kommunikációs egység továbbítja az EEPROM tartalmát a PC felé

A hardware elemek minimalizálása érdekében egy integrált infra érzékelőt, néhány LEDet, és egy RS232-es illesztőt választottam a mikrokontroller mellé. Ezeket a szaggatott vonallal rajzolt dobozok jelölik.

Hardware blokkvázlat:



A soros port D-SUB 9-es csatlakozót jelöl, az RS232 illesztő, pedig jelszint illesztő (MAX232, és kondenzátorok, ahogy az a kapcsolási rajzon látszik). A tápegység +5V-ot szolgáltat. A LED kijelző zöld, sárga, piros LEDekből és ellenállásokból áll. Az órajelet 8MHz-es kvarc állítja elő. Az infra vevő SFH5110-36-os IC, ami a mikrokontroller interrupt lábára van kötve. Részletesebb információk a mellékelt kapcsolási rajzon.

Szoftver:

A program egy főprogramból, hozzá tartozó eljárásokból és megszakításokból épül fel. Ezek globális adatstruktúrákon keresztül kommunikálnak egymással, így nincsenek paramétereik. (Az eljárás által módosított adatstruktúrákat zárójelben jelzem)

Megszakítások:

- *UART_REC*: (bemeneti FIFO)

Soros portról érkező adat bemeneti FIFO-ba mentése.

- *UART_TRANS*: (kimeneti buffer)

Kimeneti bufferből adat küldése soros portra.

- *OWEFLOW0* : (Infra kód, LED, Timeout)

Timer interrupt. Működést jelző ledet villogtatja. Egy bizonyos idő eltelt után ($\text{timeout} = \text{timeout} + 1$, ha $\text{timeout} > X$) törli az infra kódot, hogy ha úgy érkezik ismétlés, hogy előtte nem érkezett érvényes kód, akkor ne küldje ki az előző érvényes kódot.

- *INTERRUPT0*: (Infra idő buffer, LED, Timeout)

Külső megszakítás (felfutó és lefutó élre működik). Törli a Timeoutot. Az impulzus szélességét belerakja az Infra idő buffer-be. LED-et villogtat.

Eljárások:

- *uart_send(buffer)*: (kimeneti buffer)

(Egyedül ennek az eljárásnak van paramétere.) Megvárja, amíg a kimeneti buffer kiürül. Ezután a kimeneti bufferbe tölti a paraméterben megadott buffer-t, és elindítja az adatátvitelt.

- *uartEventHandler*: (bemeneti FIFO, kódolási paraméterek)

Ha a bemeneti FIFO nem üres, akkor kivesz belőle egy byte-ot, és az értékének megfelelő műveletet hajtja végre:

‘L’ : kódolási paraméterek betöltése EEPROMból

‘S’ : kódolási paraméterek elmentése EEPROMba

‘C’ : 19 byte-ot kiolvas a bemeneti FIFO-ból (addig vár, amíg megjön az összes adat), és beírja a kódolási paraméterekbe

‘R’ : A RAM tartalmát kiküldi soros porton (*uart_send* eljárással)

‘E’ : Az EEPROM tartalmát kiküldi soros porton

- *Infra_SpaceDec*: (Infra kód, Infra idő buffer)

Ha van elegendő adat az Infra idő buffer-ben, megvizsgálja, hogy az ismétlés időzítésein belül van-e. Ha igen, akkor kiveszi a bufferből, és soros porton kiküldi az Infra kódot, és egy LED-et is kigyújt. Ha nem, akkor megvizsgálja, hogy az a fejléc időzítésein belül van-e. Ha igen, és megvan az összes adat a teljes kód dekódolásához, akkor kiveszi a fejléct az Infra idő buffer-ből. Ezután egy Infra bitszám lépésű ciklust indít: megvizsgálja, hogy egyesnek vagy nullásnak megfelelő adat van-e a bufferben. Ennek megfelelően beállítja, vagy törli a megfelelő bitet, LED-et, és kiveszi az adatokat a bufferből. Ha egyikkel sem egyezik, akkor hibajelzéssel kilép a ciklusból. A ciklus lejártá után, ha nem volt hiba, elküldi a kódot az *uart_send*-del. Ha se fejléc, se ismétlés nem ismerhető fel, akkor kivesz egy adatot a bufferből, ezzel lehetővé téve, hogy a hibásan vett adatoktól megszabaduljon.

Főprogram:

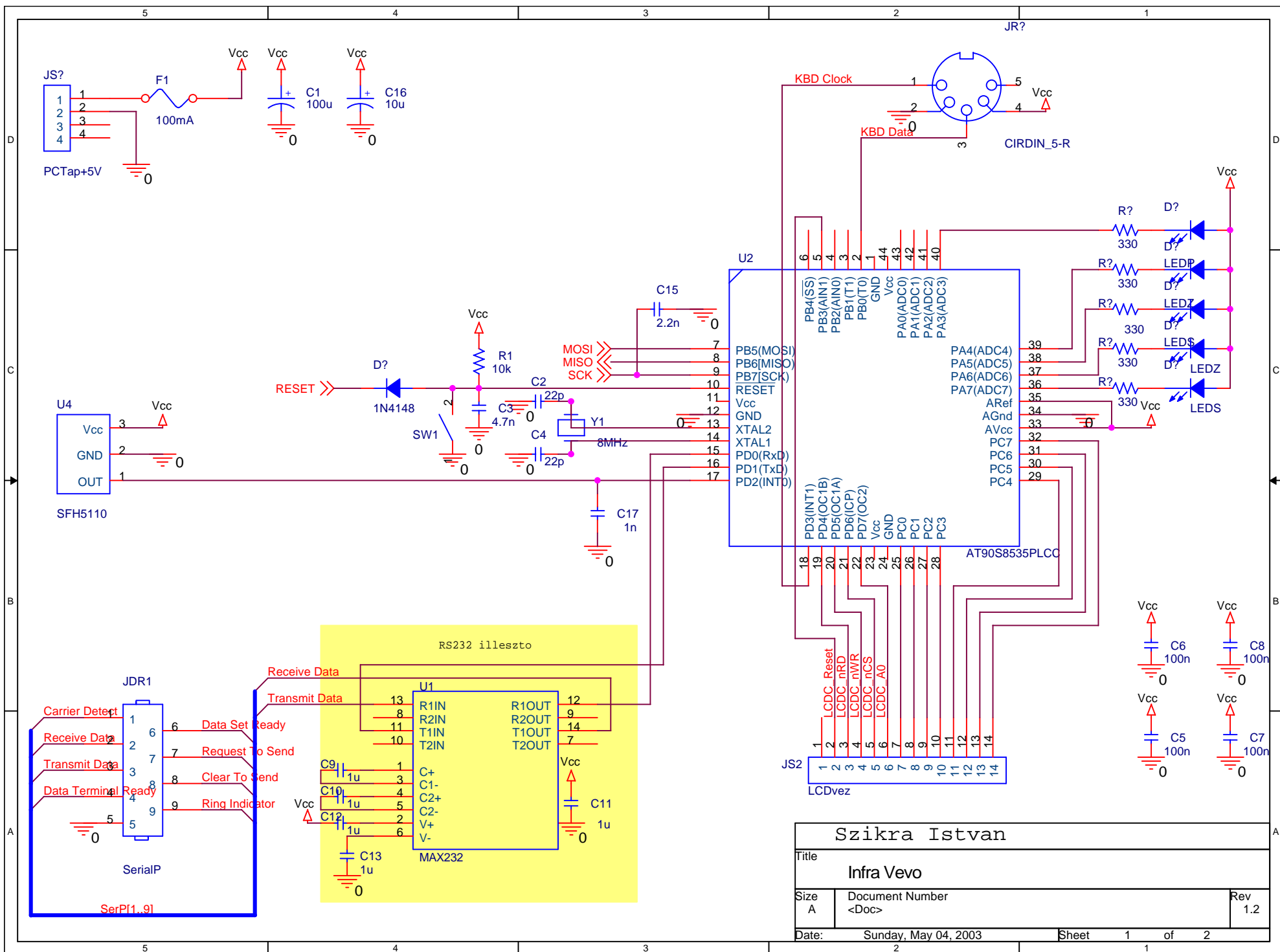
Beállítja a portok irányát. Kezdőértéket ad a változóknak (pl. stack pointer). Ha van EEPROMba mentett beállítás, akkor azt betölti. Inicializálja a megszakításokat, majd engedélyezi azokat. Egy végtelen ciklusban meghívja a decode, és az uart_handler rutinokat.

A FIFO, buffer mint adatstruktúrák magukba foglalják a tároláshoz használt memóriaterületet, a címzéshez használt pointreket/indexeket, és esetenként flageket (pl buffer üres). (A buffer- ill. FIFO - ba/ból való írás/olvasás természetesen a buffer pointereinek, FIFO indexeinek változtatását (növelését) is magába foglalják.)

A program részletesebb leírása megtalálható a mellékelt forráskódban.

Felhasználói leírás:

- Üzembe helyezés: A készüléket a COM1 portra kell csatlakoztatni. Az 5V-os tápot be kell kapcsolni.
 - Konfigurálás: A mellékelt programban ki kell választani a távirányító típusát (szükség esetén egyedi beállítások definiálására is van lehetőség).
 - Installáció: Ezután telepíteni kell a kódokat fogadó szoftvert, amellyel irányítani kívánjuk a számítógépet. A soros port sebességét 38 400 kbps-re kell állítani, majd az egyes kódokhoz funkciókat kell rendelni (ezek a szoftver használati útmutatójában (user manual) leírt módon végezhetők el). Több ilyen szoftver is létezik, az egyik az ingyenes IRAssistant, melyhez külön kell telepíteni a general COM drivert. Ezek letölthetők az alábbi címről: http://www.irassistant.com/dotcom/v_current/index.php?url=download.inc
 - Használat: Az infra távirányító gombjainak nyomogatásával távirányíthatjuk a PC-t. Kiegészítő funkció: a készülék PCs billentyűzet csatlakoztatására is fel van készítve, így ezzel is vezérelhetjük a PC-t.
- Természetesen nem csak PC-hez, hanem bármilyen soros porttal rendelkező egységhez csatlakoztatható a készülék.



Mennyiség	Érték	Leírás
1	100mA	biztosíték
1	100u	elektrolit kondenzátor
1	10u	elektrolit kondenzátor
5	1u	kerámia kondenzátor
4	100n	SMD kondenzátor
1	4.7n	SMD kondenzátor
1	2.2n	SMD kondenzátor
1	1n	SMD kondenzátor
2	22p	SMD kondenzátor
5	330	ellenállás
1	10k	ellenállás
2	LEDZ	zöld LED
2	LEDS	sárga LED
1	LEDP	piros LED
1	1N4148	dióda
1	8MHz	Kvarc
1	AT90S8535-PLCC	mikrokontroller
1	MAX232-DIP	RS232 illesztő
1	SFH5110-36	Infra vevő
1	DSUB-9	soros port csatlakozó (anya)
1		kisfloppy táp csatlakozó
1		nyomógomb