|  |  |  |
| --- | --- | --- |
|  | PRAXE |  |
| 1 | SPŠ Chomutov | Fous Hrdý |
| 7.12.2020 | DCF příjem časové informace | V4 |

# Zadání

Vytvořte program, který zpracuje a vyhodnotí časovou informaci z DCF přijímače.

**Program musí:**

Rozpoznat začátek vysílání.

Zpracovat signály a rozdělit přijaté bity na správné časové údaje.

Vyhodnotit, podle paritních bitů, zda jsou údaje správné.

Vypsat získanou informaci.

# Teorie

**Základní informace**

DCF-77 je vysílač časové informace, který vysílá signály na dlouhé vlně 77,5kHz. Signál je dosažitelný v okruhu ±2 000 km.

**Kódování informace**

Vysílač vždy vysílá informace o následující minutě. Konkrétně vysílá středoevropský čas a datum. Na začátku každé sekundy je vysílána značka, které odpovídá log. 0 (50÷120 ms) nebo log.1 (140÷240 ms). Datum a čas je vždy doplněn o paritní bit, který udává, zda byl sudý nebo lichý počet log. 1.

**Význam bitů**

Bit 0 – označuje začátek časové informace, tj. první sestupná hrana v sekundě

Bit 1 – 14 – tyto bity zatím nemají žádný význam. Všechny tyto bity jsou v log. 0.

Bit 15 – označuje, zda je signál vysílán rezervní anténou. Pokud ano tak je v log. 1.

Bit 16 – upozorňuje, jestli dojde/došlo ke změně letního/zimního času. Log. 1 se vysílá hodinu před a hodinu po změně času

Bit 17 – v log. 1 označuje letní čas

Bit 18 – označuje zimní čas v log. 1.

Bit 21 – 27 – udává minuty podle vah bitů (21 = 1, 22 = 2, 23 = 4, ...)

Bit 28 – první parita (pro minuty)

Bit 29 – 34 – udává hodiny podle vah bitů

Bit 35 – druhá parita (pro hodiny)

Bit 36 -41 – odpovídá dnu v měsíci

Bit 42 – 44 – odpovídá dnu v týdnu

Bit 45- 49 – udává měsíc

Bit 50-57 – udává rok

Bit 58 – třetí parita (pro měsíc, rok, den a den v týdnu)

Bit 59 – poslední bit, který se nevysílá. Podle tohoto bitu se poznává konec/začátek časové informace

**Hardware**

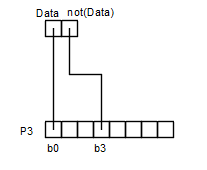
Hardware byl virtualizován na TCE Linuxu. Virtuální hardware je navržen jako dokonalý hardware bez vnějšího rušení.

**Přístup k portům**

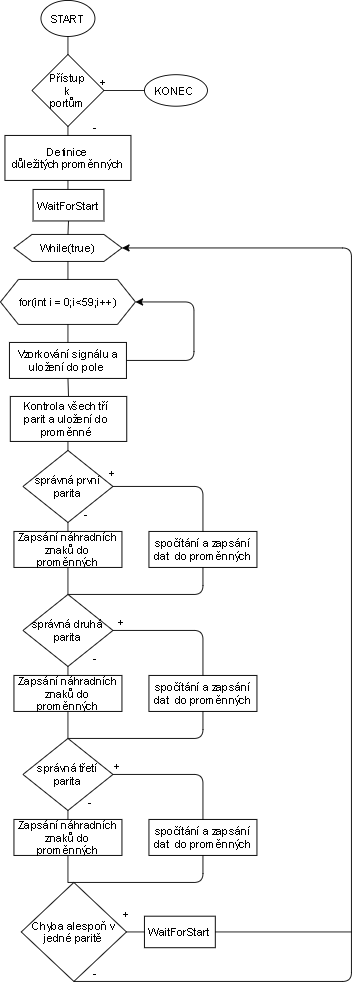
K portům se používá virtuální knihovna <sim\_io.h>, která funguje stejně jako knihovna <sys/io.h>.

Využívaly se porty P1 a P2 pro výstup a porty P3 a P4 pro vstup. Nacházejí se na adrese 0x300 a 0x301.

# Schéma zapojení



# Vývojový diagram



# Popis programu

Program začíná funkcí WaitForStart, která čeká a nalezne poslední bit DCF packetu.

Poté už program funguje v jedné nekonečné smyčce. Na začátku smyčky se v cyklu na vzorkuje packet a zapíše se do pole DCFdata. Následně proběhne kontrola všech tří parit.a výsledky se zapíší do příslušných proměnných (viz rozbor funkcí a proměnných).

Pokud je vždy příslušná parita správně, program dopočítá časový údaj a uloží jej do proměnné.

Na konci se zobrazí výsledek. Pokud alespoň jedna parita nebyla úspěšná, program znovu vyvolá funkci WaitForStart.

# Rozbor funkcí a proměnných

**Důležité proměnné:**

* int DCFdata[59] – pole s uloženými přijatými bity
* int P1, P2, P3 – určují zda byla chyba v paritě nebo ne
* char sdenvtyd[] ,sden[] ,smes[] ,srok[] ,shod[] ,sminuty[] – nabývají hodnot podle toho, jaký je právě čas ve stringu(pole charů).
* int denvtyd, den, mes, rok, hod, minuty – nabývají hodnot podle toho, jaký je právě čas v intu
* int start, end – začátek a konec časové aktivity procesoru. Rozdíl udává, jak dlouho program mezi nimi trval.
* int začátek, konec – určují o jaký interval bitů se jedná

**Důležité funkce:**

* VzorkujDCF – Tato funkce nepřijímá žádné parametry. Měří délku logické nuly a podle toho vrací jedničku nebo nulu.
* WaitForStart – Voidová funkce, která počká na start (najde poslední bit, který je po celou sekundu v logické jedničce).
* KontrolaParity– Funkce přijímá parametry a podle toho vrací hodnotu symbolizující chybu nebo správnost parity (0 nebo 1)
* SectiBity – Podle parametrů počet bitů a vrátí jejich součet.

# Odpovědi na otázky

1. Důležitým parametrem každého datového přenosu je tzv. latence. Pokuste se definovat tuto hodnotu pro příjem DCF normálu pro ČR. Je tato hodnota konstantní?

Vzdálenost Frankfurtu od Chomutova je vzdušnou čarou 315,2 km. Rychlost šíření rádiových vln se blíží rychlosti světla ve vakuu, tj. 300 000 km/s.

315,2/300 000 = 1,06 ms

Latence je přibližně 1,06 ms

Tato hodnota není konstantní. Vlnění je ovlivňováno materiálem, kterým prochází, a proto latenci může ovlivnit jen změna hustoty vzduchu natož pak třeba bouřka.

1. Existují další standardy, které je možné využít pro získání časové reference bez nutnosti síťového připojení? Pokud ano, popište principy jednoho z takových standardů.

Ano existují.

Používá se GPS synchronizace, kde přijímač přijímá časovou informaci z 4 družic a díky tomu získá polohu v prostoru. Pokud zná přijímač svou polohu v 3D prostoru, tak dokáže vypočítat zpoždění času z jedné z družic a získá tak čas. Tento signál je dostupný po celém světě. Každá z družic obsahuje atomové hodiny.

Přijímač přijímá na frekvenci 1575,42 MHz s šířkou pásma ±1,023 MHz[[1]](#footnote-1)

1. Protokol DCF využívá pulzní kódování pro přenos stavu jednotlivých bitů. Vyberte si libovolný jiný princip kódování (encoding) informace vhodný pro bezdrátový přenos a popište jej.

Manchester kódování.

Toto kódování využívá dat a hodinového signálu. Pokud je součet přechodů rozdílný a je roven přechodu z log.1 na log.0 signál je v log. 0, ale pokud je přechod z log. 0 na log.1 signál odpovídá log.1. Jestliže se součet přechodů vyruší, signál zůstává ve své pozici.

Diferenciální Manchester kódování (přesně opačné) se používá u technologie Ethernet.

# Závěr

Program byl schopen na virtualizovaném prostředí splňovat požadované funkce DCF přijímače. K dosažení výsledku využíval program nedokonalost virtuální periferie.

1. Diplomová práce BC. Petr Švábeník VUT Brno Synchronizace času pomocí GPS

   soubor.eu Ing. Martin Šimek, Přenos dat [↑](#footnote-ref-1)