

# Časování a synchronizace v embedded systémech s $\mu$ C + FPGA

Ing. Tomáš Navrátil | Ryston Electronics s. r. o.

## ■ Úvod

V Rystonu se dlouhodobě zabýváme vývojem komunikačních zařízení pro náročné aplikace, které kromě základní funkce přenosu dat splňují požadavky odolnosti proti poruchám, redundance, zabezpečení dat a vzdáleného dohledu. Jedním z projektů vlastního vývoje byla jednotka NAG obsahující vestavný mikro počítač s OS RT-Linux spolupracující s obvodem FPGA Xilinx-Spartan. K tomuto systému je připojeno jedno nebo více rozhraní E1 (G.703) s framerem FALC56.

Vedení E1 je plesiochronní, tedy s přibližně (ale ne zcela) synchronními hodinami bitové sekvence. Podřízené rozhraní se musí zasynchronizovat na přicházející signál od nadřazeného uzlu sítě, jehož frekvence může v určitých mezích (zhruba  $\pm 50$  ppm) kolísat podle charakteru přenosové trasy, např. vlivem dopplerovského posunu vzniklého pohybem telekomunikačních družic.

Je-li připojeno více rozhraní E1, popř. E2 a vyšších, jedno z nich řídí synchronizaci a ta ostatní jsou „zavěšena“ na svých zdrojích signálu, a tak vzniká potřeba sladit několik nesynchronních systémů. To může mít za následek měnící se fázevá zpoždění mezi signály a následné hazardní stavy.

## ■ CPUi

Obvod FPGA, mimo další funkce, funguje jako „lepídlo“ mezi doménami systému podléhajících různým zdrojům časování: hodinám procesoru, které řídí cykly sběrnice a přijímaným hodinám rozhraní E1.

Obvod FPGA musí být po zapnutí nebo resetu nainicializován z procesoru, protože jeho vnitřní obvody jsou volatilní (závislé na napájení). Vznikla otázka, jak nejlépe propojit procesor a FPGA, aby jejich komunikace v prostředí víceúlohového operačního systému probíhala optimálně.

Proto ze strany procesoru byl obvod FPGA umístěn do jeho 32bitového paměťového prostoru a k přístupu do něj slouží dva z procesorem generovaných selektivních signálů. Obvod FPGA se chová vůči procesoru jako oblast rychlé dvouportové paměti RAM, takže běžící procesy mohou velmi rychle nezávisle komunikovat s blokem, který je jim přidělen. Velikost, struktura a významy jednotlivých bitů jsou určeny firmwarem FPGA.

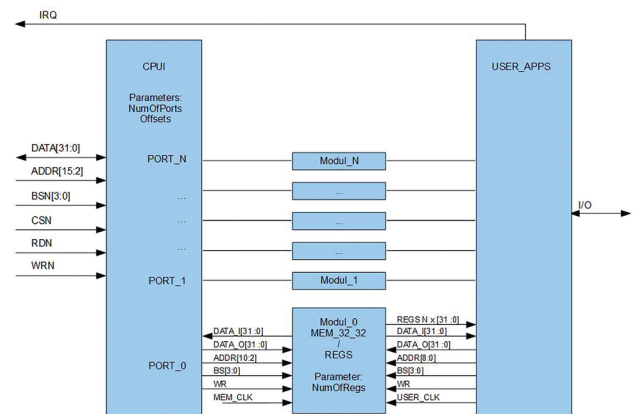
Modul CPUi je částí programu v FPGA a tvoří datové (sběrnice) rozhraní mezi nadřazeným procesorem a uživatelskou aplikací v FPGA. Skládá se z určitého počtu „bran“ – bloků paměti s asynchronním přístupem.

Brána může fungovat například jako dvouportová, semafor/ukazatel řízená paměť zpráv, do níž se z venku zapisuje a zprávy jsou vybírány nějakým procesem uvnitř FPGA, nebo naopak.

Počet bran je limitován velikostí FPGA a volným adresovým prostorem procesoru. Konkrétní počet je dán parametrem NumOfPorts.

Každá z bran zaujímá prostor 2048 bytů v adresovém prostoru procesoru, a jeví se tedy jako kus jeho paměti.

Offset bran je definován parametrem Offsets(n). Offsets je pole typu integer, počet prvků je roven NumOfPorts. Prvky pole jsou násobkem 2048 a navzájem různé. Brána má 32bitový vstup/výstup dat na datovou sběrnici, 9bitovou adresovou sběrnici, 4bitový signál BSn pro práci s jednotlivými byty v 32bitovém slově a signál zápisu WRn a čtení RDn a souhrnnou žádost o přerušení pro procesor. Viz obr. 1.



Obr. 1 Rozhraní CPUi

Přítomnost spojení obou subsystémů přes sběrnici je velice volná a umožňuje multiprocesovou komunikaci.

## ■ Modul MEM/REGS

Pokud aplikace v FPGA využívá pro komunikaci s nadřazeným procesorem dvouportové paměti, je možno použít modul mem/regs. Tento modul zaujímá adresový prostor 2048 bytů. V tomto adresovém prostoru může být pole 32bitových registrů a vlastní paměť. Počet registrů je určen parametrem NumOfRegs. Registry jsou mapovány od konce adresového prostoru a zmenšují velikost využitelné dvouportové paměti. Počet registrů je limitován velikostí adresového prostoru, tj. max. 512 registrů na jednu bránu, a také velikostí FPGA. V praxi není velké množství registrů potřeba.

Komunikace přes dvouportovou paměť výrazně šetří zdroje FPGA. Modul obsahuje dvě brány, jednu směrem k CPUi a druhou směrem k uživatelské aplikaci. Velikost portu REGS je závislá na počtu registrů daných parametrem NumOfRegs, tj. obsahuje  $\text{NumOfRegs} \times 32$  bitů. Komunikace s dvoupor-

tovou paměť je v obou směrech 32bitová. Modul umožňuje oddělení taktové domény procesoru (taktový signál mem\_clk) a aplikace, resp. její části (taktový signál user\_clk).

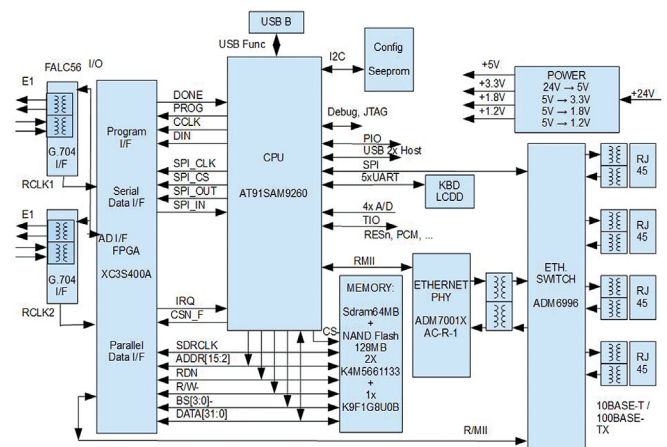
### ■ Synchronizace na vnější signály

Na plesiochronní signály přijímané z E1 je nutno nejprve „zavěsit“ zdroj regenerovaných hodin, který řídí vnitřní procesy a vysílač obvodu Framer, FALC56. Tento obvod je výsledkem dlouhého vývoje firem Siemens (pak Infineon, Lantiq a nyní Intel).

Ladění lokálního oscilátoru u různých systémů se provádí buď digitálním fázovým závěsem (DPLL), nebo programovatelnou děličkou provedenou v PLD, závěsem přes zpoždění (delay-locked loop) pomocí „pullable“ (jemně laditelného) krystalu paralelně připojeným varikapem, jehož proměnná kapacita zatěžuje výbrus, anebo pomocí analogového PLL pomocí napětím řízeného krystalového oscilátoru (VCXO). V průběhu 25 let vývoje telekomunikačních aplikací jsme je vyzkoušeli všechny.

Obvod FALC56 potřebuje jen přesný a stabilní zdroj kmitočtu, např. 20 MHz, a pomocí programovatelné děličky racionálním číslem si vytváří signál RRCLK (received route clock). Obvod Xilinx si může multiplexerem vybírat tyto signály od různých rozhraní a tím určit Mastera hodin.

Nesynchronně měnící se signály je na vstupu FPGA ještě nutno převzorkovat řadou několika klopných obvodů D, kte-



Obr. 2 Blokové schéma jednotky NAG se 2x E1 vstupy a ethernet switchem na výstupu

ré klopi na hranu taktovacího signálu vnitřní domény FPGA. Aby byl dodržen setup- a hold-time (předstih a přesah stabilní úrovně signálu oproti taktu), je nutno použít alespoň dva vzorovací obvody D za sebou.

Rozhraní datové sběrnice obvodu FALC56 vede do FPGA – to kvůli rychlosti, protože sběrnice procesoru je rychlá a nemá vstup READY na prodloužení cyklu sběrnice. FPGA simuluje řídicí signály datové multiplexované sběrnice AD (Intel).

**VÝVOJ, VÝROBA ELEKTRONIKY**  
Sběr, měření, přenos a zpracování dat, průmyslové řízení, automobilní vestavné počítače, zákaznické systémy na míru.

**VÝROBA MECHANIKY**  
Automobilní konstrukce a nástavby, zdvihací mechanismy, hydraulika.

**DODÁVKY**  
Jsme předním dodavatelem komunálních nástavb a systémů pro nákladní automobily, s elektronickým řízením. Prodej podporujeme programováním.

**TECHNICKÁ PODPORA**  
Servis a dodávky spotřebního materiálu jsou nedílnou součástí podpory zákazníka.

**KVALITA**  
Máme SMJ dle ISO 9001:2015 pro vývoj, výrobu a prodej. Součástky od nás mají garantovaný původ a jsou k nim dostupné všechny technické podklady.

Další informace:  
[www.ryston.cz](http://www.ryston.cz)  
[www.pivnicka.eu](http://www.pivnicka.eu)

## RFID systémy

**pro Vaše automobilní a komunální aplikace + vždy něco navíc:**

**HW:** Stavebnice moderních vestavných počítačů s 32-bitovými procesory a FPGA, s porty RS-485, CAN, 1W, opto I/O pro sensory polohy, výkonové moduly pro silové spínače, řízení motorů. Tlačítkové ovladače, indikátory a displeje, jednotky GPS, GSM, WiFi, čtečky RFID, bar-kódů, tenzometrické váhy, PoE/LAN IoT integrace, radiové moduly, zálohované zdroje... + **rychlé propojení sběrnice s Vaší logikou v FPGA.**

**SW:** Aplikace pod OS Linux s reálným časem umožňují rychlé konkurenční řídicí procesy, a současně prací se soubory, připojení k internetu a databázím na cloudu, dálkový dohled a aktualizaci SW, zabezpečení + **VHDL / grafická tvorba log. systému.**

**Aplikace:** Řízení a monitoring, sběr, přenos, zpracování a zabezpečení dat v automobilních systémech např. pro komunální služby, drážní komunikaci, ekologii, řízení technologie, potravinářství, lékařství + **Vaše příští aplikace může začít právě zde.**

Ryston Electronics s.r.o., Pod vinicí 2045/18, 143 00, Praha 4 - Modřany

## ■ Projektové soubory FPGA

Výše popsaný systém bran jako rozhraní procesoru byl vyvinut za použití jazyka VHDL. Pro použití ve vyšší sestavě jsou nutné tři soubory. Soubor CPU1.VHD implementuje modul CPU1, soubor MEM\_BLOK\_32\_32.VHD implementuje modul mem/regs a v souboru CPU\_INT\_LIB.VHD jsou definovány potřebné datové typy a konstanty. V souboru APP\_TOP.VHD je uveden příklad použití modulů CPU1 a MEM/REGS. Projekt je uložen ve složce CPU\_INTERFACE.

Na tyto bloky pak zpravidla navazují samotná logická struktura realizovaná mocnými prostředky použitého obvodu FPGA (desetitisíce hradel a klopných obvodů). V naší aplikaci je to člen rodiny Spartan. Obvodem je možno realizovat velmi složité stavové automaty, radiče, komunikační obvody, specializované procesory nebo vstupní či výstupní obvody, které navazují procesor na okolní prostředí. V jedné naší aplikaci je to například řada PWM výstupních portů (převodníků číslo–střída) pro výkonové řízení spínanými regulátory.

V této aplikaci byly napsány automaty vybírající z přijatých dat E1 rámce dle ITU-T V.110 a konstruující rámce V.110 do kanálů vysílaného směru E1. Vlastně jde o dva specializované procesory pro jednu E1 připravující data V.24 do/z portů CPU1. Tak se podařilo realizovat multikanálový obousměrný terminálový adaptér V.24 / V.110, který nejen nahradil nedostupné obvody Infineon ITAC, ISDN Terminal Adapter Circuit, ale jejich výkon řádově předčil.

Spolupráce CPU a FPGA tohoto typu se ukázala tak úspěšnou, dobfé programovatelnou a odladitelnou, že jsme tento koncept převzali do dalších konstrukcí a dále jsme ho „recyklovali“ do malého modulu vestavného mikropočítače s názvem Combi. Výrobce procesoru Atmel dále přišel s modernizovaným obvodem AT91SAMG20 s taktem 400 MHz a vylepšeným instrukčním souborem, a navíc je k dispozici nové vydání operačního souboru RT-Linux podporující periferie jako USB, ethernet, čítače a PCM na čipu. Modul Combi-G20 velikosti vizitky se stal základem nejen inovací jednotky NAG, ale i nových konstrukcí.

Na jedné malé desce vícevrstvého plošného spoje je kompletní mikropočítač s jádrem ARM9 na 400 MHz, paměťmi a podpůrnými obvody, propojený 32bitovou sběrnicí s obvodem FPGA. Na vnější konektor modulu jsou vyvedeny periferní signály a sériová rozhraní (UARTy, SPI, I2C) s příslušnými selekt-signály, analogové vstupy a další užitečné signály, zatímco na opačnou stranu desky jsou vyvedeny periferní I/O signály z FPGA.

To přináší překvapivé možnosti kombinace velmi rychlých a složitých logických struktur v FPGA, doplňovaných o softwarovou podporu z procesoru.

Blokové schéma embedded modulu Combi-G20 je na obr. 3.

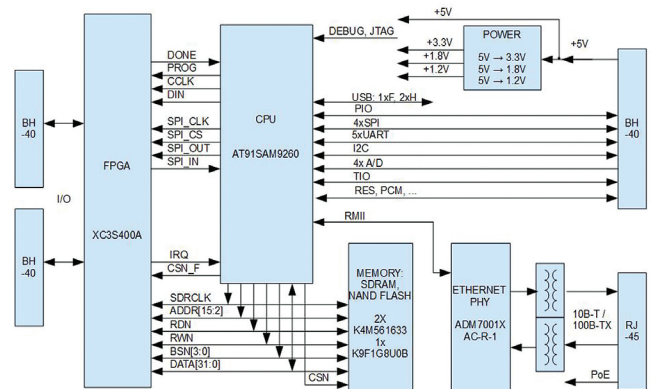
Podoba obr. 3 a obr. 2 je zřejmá. Liší se okrajovým propojením. V modulu Combi jsou signály procesoru a FPGA vyvedeny na dutinky konektoru BH-40 obráceného „dospodu“, takže modul Combi je možno nasunout na 40špičkové konektory – „hřebínky“ zapájené v základní nosné desce. Tyto konektory mají vývody v rastru 100 mils, což usnadňuje vývoj základní desky.

Rozhraní ethernet je vyvedeno na konektor RJ-45 a je též připraveno pro možné dálkové napájení PoE přes DC/DC převodník. Rozhraní USB Function je vyvedeno na konektor USB B (+5 V z kabelu je možno využít pro napájení modulu) a rozhraní 2× Host je připraveno na hřebínkovém konektoru pro plochý kabel, stejně jak se to dělá v PC. Konektory USB B a RJ-45 jsou umístěny na kraji modulu, aby je bylo možno prostrčit výřezem v krytu vestavného systému.

## ■ Praktické použití

Uvedené programové i hardwarové moduly byly prakticky vyzkoušeny již v několika aplikacích. Jednou z nich je zpracování telekomunikačního rozhraní E1, kde ve 30B kanálech probíhá duplexní komunikace podle V.110. Zjednodušeně je činnost tato: Jestliže přijde platný blok dat, je uložen v příslušném bloku dvouportové paměti a FPGA vygeneruje do fronty zprávu procesoru o této události a aktivuje žádost o přerušení. Procesor v obslužné rutině přečte zprávu, tím shodí žádost o přerušení, vyčte data a uloží je do struktury v paměti pro zpracování vyššími vrstvami protokolového zásobníku. Tato rutina je velice rychlá, protože je to jen jednoduchý přenos bloku paměti plus ošetření přerušení, a proto nehrozí přetečení zásobníku.

Pokud se komunikuje s více rozhraními E1, je možno vybrat řídicí rozhraní z hlediska časování. Ostatní E1 musejí použít



Obr. 3 Blokové schéma modulu Combi-G20

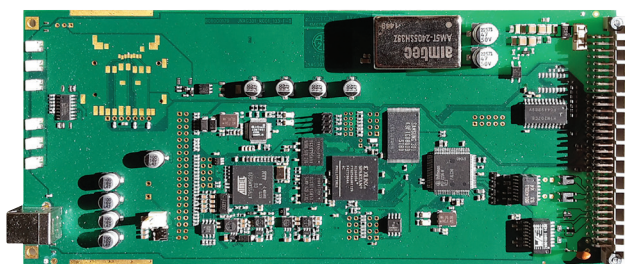
pružný rámcový buffer v obvodu FALC. Pokud řídicí rozhraní postihne výpadek, systém běží volně až do zvolení náhradního zdroje časování. Převzorkování a synchronizace umožní hladké přepnutí na jiný zdroj.

Vyšší vrstvy protokolu už pracují ve víceúlohovém režimu podle toho, jak jim systém přiděluje prostředky.

V opačném směru komunikace je činnost podobná: Když proces získá data pro odeslání, uloží je do příslušných míst sdílené paměti a nastaví v registru bit „vyšli“. Po odeslání je vygenerována zpráva a žádost o přerušení do procesoru, který doplní data do příslušné dvouportové paměti, aby se mohlo vysílat bez přestávek.

Další aplikací je například řízení polohy systému (robota) poháněného servomotorem. Poloha je odměřována dvojicí

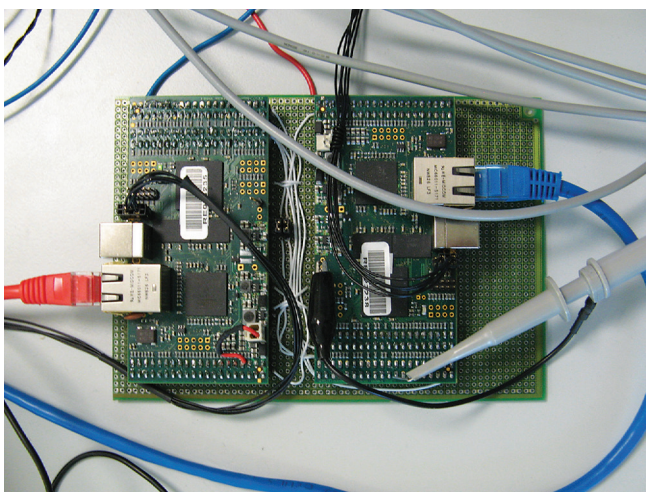




Obr. 4 Recyklace modulu Combi-G20 v jednotce NAG

kvadrurních (postupně zatmívaných) signálů, z nichž jsou dekodovány impulsy UP/DN pro 32bitový obousměrný čítač polohy, který je v FPGA. Procesor tento čítač může kdykoli přečíst přes jeho dvouportovou paměť (zde opět jde o dva nesynchronní procesy, které synchronizuje odpamatovací-vzorkovací obvod registru polohy). Procesor tak jediným atomickým čtením 32bitového registru získá platnou polohu. Další optický vstup funguje jako koncové čidlo, které definuje nulový stav (polohu).

Periodicky spouštěný řídicí program (pod RT Linuxem) si může časovou diferenciací polohy zjistit údaj o rychlosti a pomocí adaptivního modelu vypočítá pro žádaný pohyb hodnotu



Obr. 5 Vývoj aplikace se dvěma moduly Combi-G20

řídicí veličiny. Tou je výstup PWM nebo D/A převodníku do výkonového zesilovače, jímž je motor napájen. Vyšší vrstvy programového vybavení komunikují s nadřazeným systémem a rozkládají trajektorii složeného pohybu do elementárních pohybů, zatímco další proces porovnává realitu s chováním modelu a průběžně ho adaptuje. Předpokládá se přitom, že řízený systém je objektem s mnoha nelinearitami a vazbami (např. různá setrvačnost při různých ramenech).

Prakticky bylo ověřeno, že použití jediného přerušovacího signálu s navazujícím mechanismem bran ve dvouportové paměti je časově úsporné a spolehlivé (neztrácí se žádosti o obsluhu ani „nevisí“ rozpracovaná obsluha události, a to ani ve velmi složitých situacích). Tento mechanismus je také velmi vhodný pro prostředí víceúlohového operačního systému s mnoha paralelními procesy, kterým Linux je.

## ■ Závěr

Popsaná rozhraní mezi vstupy z reálného světa do FPGA i uvnitř něj, mezi FPGA a procesorem a synchronizace procesů v RT-Linuxovském prostředí jsou výsledkem snahy usnadnit praktické aplikace vestavného počítače se vším nabízeným komfortem pro práci v reálném čase.

Bylo vyřešeno zavěšení časové základny telekomunikačních obvodů na vstupní signály, implementováno synchronizované rozhraní vstupů/výstupů FPGA pro hladké přepnutí synchronizace a optimalizováno jádro operačního systému pro kompromisní soft-RT řízení procesů.

Základ systému byl „recyklován“ do embedded-modulu Combi-G20 obsahujícího vylepšený procesor, FPGA a podpůrné obvody. Dále byly vyvinuty hardwarové prostředky pro připojení modulu k ovládacím a snímacím obvodům pro běžné průmyslové řízení jako opticky oddělené nebo reléové či triakové vstupy/výstupy, PWM můstky, analogové I/O, rozhraní CAN nebo RS-485, přijímače a vysílače proudové smyčky 4–20 mA apod. Toto jádro opakovaně používáme v různých zakázkových systémech a snažíme se příznivě „vedlejší efekty“ tohoto spojení zúročit v neustále vylepšovaných projektech. Naštěstí jsme se, zdá se, „trefili“ do dlouhodobě podporovaného typu procesoru i logického obvodu, takže jsou pro ně dostupné aktualizace jak operačního systému, tak knihovnic prvků i projektů třetích stran.

180 x 55 mm