



PROJEKTOVÁNÍ MIKROPOČÍTAČOVÝCH SYSTÉMŮ

doc. Ing. Jan Janeček, CSc.

PROJEKTOVÁNÍ MIKROPOČÍTAČOVÝCH SYSTÉMŮ

doc. Ing. Jan Janeček, CSc.

2022

České vysoké učení technické v Praze

(C) Jan Janeček, 1994, 2022

ISBN 978-80-01-06973-8 (electronic)

978-80-01-01289-5 (print)

Tato publikace podléhá licenci Creative Commons



Předmluva

Tento text je učební pomůckou pro studenty denního studia Elektrotechnické fakulty ČVUT, kteří si zapsali volitelný předmět Projektování mikropočítačových systémů. Jeho studium předpokládá základní znalosti z architektury počítačů a jejich programování ve strojovém kódu.

Zaměření textu vychází ze současné situace, kdy se konstruktér mikropočítačového aplikačního systému nejčastěji setká s použitím jednočipových mikropočítačů a jednočipových řadičů. Úvodní část je proto věnována obvodům rozhraní, která jsou pro jednočipové mikropočítače a řadiče typická - paralelnímu rozhraní, obvodům pro měření času a čítání událostí, obvodům pro různé formy sériové komunikace a obvodům pro zpracování analogových signálů. Podrobnější informace o procesorech jsou uvedeny v následujícím přehledu mikropočítačových prvků, kde jsou předpokládány základní znalosti struktury procesoru.

Přehled mikropočítačových prvků, které má v současné době konstruktér mikropočítačové aplikace k dispozici, tvoří největší část textu. Všímá si typických osmibitových řadičů (čtyřbitové řadiče, které mají mnohem chudší možnosti už necháváme stranou), osmibitových a šestnáctibitových jednočipových mikropočítačů. Doplněny jsou základní údaje o paměťových obvodech, bez nichž se většina aplikací neobejde.

Závěrečná část je věnována technickým a programovým prostředkům, které podporují a podstatně zrychlují ladění mikropočítačových aplikací.

Na tomto místě bych chtěl poděkovat všem, kteří mi s přípravou textu pomohli, zvláště Ing. Martinu Bílému za řešení problémů spojených se sazbou v systému L^AT_EX, Ing. Mirkovi Skrbkovi za korekci textu a lektorovi Doc. Janu Bílkovi, jehož cenné připomínky přispěly k přehlednosti a užitečnosti předkládaného materiálu.

Text vychází v této formě poprvé a autor uvítá poznámky pečlivého čtenáře k jeho formě a obsahu.

Tábor, listopad 1994

Jan Janeček

(C) Jan Janeček, 1995

ISBN 80-01-01289-1

Obsah

1 Úvod	5
2 Paralelní rozhraní	6
3 Měření času a kmitočtu	8
3.1 Časovače a čítače	8
Univerzální čítač/časovač	8
Záchytný registr	10
Komparační registr	11
Čítače událostí	12
Generátor periody	12
Časovací procesory	13
3.2 Obvody reálného času	15
3.3 Hlídací obvody Watchdog a Power-Fail	17
3.4 Obvody PWM	20
4 Sériová komunikace	21
4.1 Elektrické standardy	21
RS-232C	21
RS-423	22
RS-422	23
RS-485	24
Proudová smyčka	24
4.2 Komunikační protokoly	25
Modbus	25
Devítibitová multipočítačová sběrnice Intel	26
Bitbus	27
Fieldbus	28
CAN	29
4.3 Sériové sběrnice	31
Motorola SPI	31
MicroWire	32
I ² C	34
5 Analogové vstupy a výstupy	38

<i>Obsah</i>	3
5.1 Aproximační A/D převodníky	38
5.2 Integrovaní A/D převodníky	40
5.3 Alternativní řešení A/D převodu	41
5.4 D/A převodníky	42
6 Zobrazovače	44
6.1 Zobrazovače LED	44
6.2 Zobrazovače LCD	46
7 Osmibitové jednočipové řadiče	47
7.1 Motorola MC68HC05	48
7.2 SGS-Thompson ST62	50
7.3 Microchip PIC16C5x	54
8 Osmibitové jednočipové mikropočítače	58
8.1 Intel 8051	59
8.2 Philips 80C552	72
8.3 Siemens SAB 80C515/80C535	81
8.4 Siemens SAB 80C517/80C537	83
8.5 Dallas DS80C320	85
8.6 Motorola MC68HC11	86
8.7 Hitachi H8/330	105
9 Šestnáctibitové jednočipové mikropočítače	107
9.1 Intel 80C196	108
9.2 Siemens 80C166	113
9.3 Motorola MC68HC16	116
9.4 Motorola MC68332	118
10 Dvaatřicetibitové mikropočítače	120
11 Paměťové obvody	121
11.1 Paměti RAM	121
11.2 Paměti EPROM	122
11.3 Paměti EEPROM	123
11.4 Paměti Flash EEPROM	125
12 Vývojové prostředky	127

12.1 Ladění bez podpůrných prostředků	127
12.2 Emulace paměti programu	128
12.3 Sledování výpočtu, logická analýza	129
12.4 Obvodová emulace	130

1. Úvod

Návrh a vývoj jednoúčelových aplikací opírajících se o mikropočítačové řízení je v současnosti rozhodně perspektivní oblastí. Mikropočítačové řízení v integrované formě, jako jednočipových mikropočítačů, dnes najdeme v řadě přístrojů a zařízení, se kterými se běžně denně setkáváme. Mikropočítače řídí kancelářská zařízení (faxy, digitální telefony, telefonní ústředny, kopírky), domácí a spotřební elektroniku a často si jejich existenci v těchto zařízeních ani neuvědomujeme. Tuto skutečnost obvykle charakterizujeme označením, které pro taková použití mikropočítačů používáme, *embedded* (vestavěné) mikropočítače. Z našeho pohledu konstruktéra mikropočítačových aplikačních systémů ale budou zřejmě zajímavější jednoúčelové aplikace v měřicí, automatizační a regulační technice.

Dnešním standardem v oblasti jednoúčelových mikropočítačových aplikací jsou jednočipové mikropočítače. S výjimkou konzervativních oblastí již prakticky vytlačily starší mikroprocesory, které jsme považovali za standard před deseti lety (Intel 8085, Zilog Z80, Motorola 6800). Hlavním důvodem byl zřejmě fakt, že dokázaly dát v jediném pouzdře k dispozici soubor prvků, které u starší technologie vyžadovaly jednu nebo více desek plošného spoje. Důležité jsou pochopitelně i podstatně nižší nároky na příkon a nižší cena novějších jednočipových řešení.

Dnešním standardem v mikropočítačovém řízení jsou jednočipové osmibitové mikropočítače s poměrně jednoduchými procesory — řady Motorola 68HC05, Intel 8051 (a mnoho variant tohoto mikropočítače od řady výrobců), Motorola 68HC11 a Zilog Z8. S rostoucími nároky na výpočetní výkon, zvláště ze strany automobilek, lze očekávat přechod na technologie šestnáctibitové, opírající se o moderní struktury procesoru, vhodným příkladem může být např. Siemens 80C166. V blízké budoucnosti lze předpokládat rostoucí význam dvaatřicetibitových mikropočítačů, které jsou dnes používány v laserových tiskárnách, a signálových procesorů. Tento posun bude zřejmě důsledkem rozvoje multimediálních technik.

Současná cena jednodušších programovatelných prvků, opírajících se o princip počítačového zpracování informace, poklesla při jejich dostatečném výkonu na takovou úroveň, že mikropočítačová řešení v řadě oblastí vytlačují nebo již vytlačila pevnou a programovatelnou logiku. Tento trend je zvláště patrný v poslední době, kdy se objevují rychlé jednočipové radiče, které mají minimální nároky na napájení (např. Microchip 16C54). Najdeme je v elektronických klíčkách (kontaktních nebo infračervených), v čipových kartách pro bankovní automaty, ap..

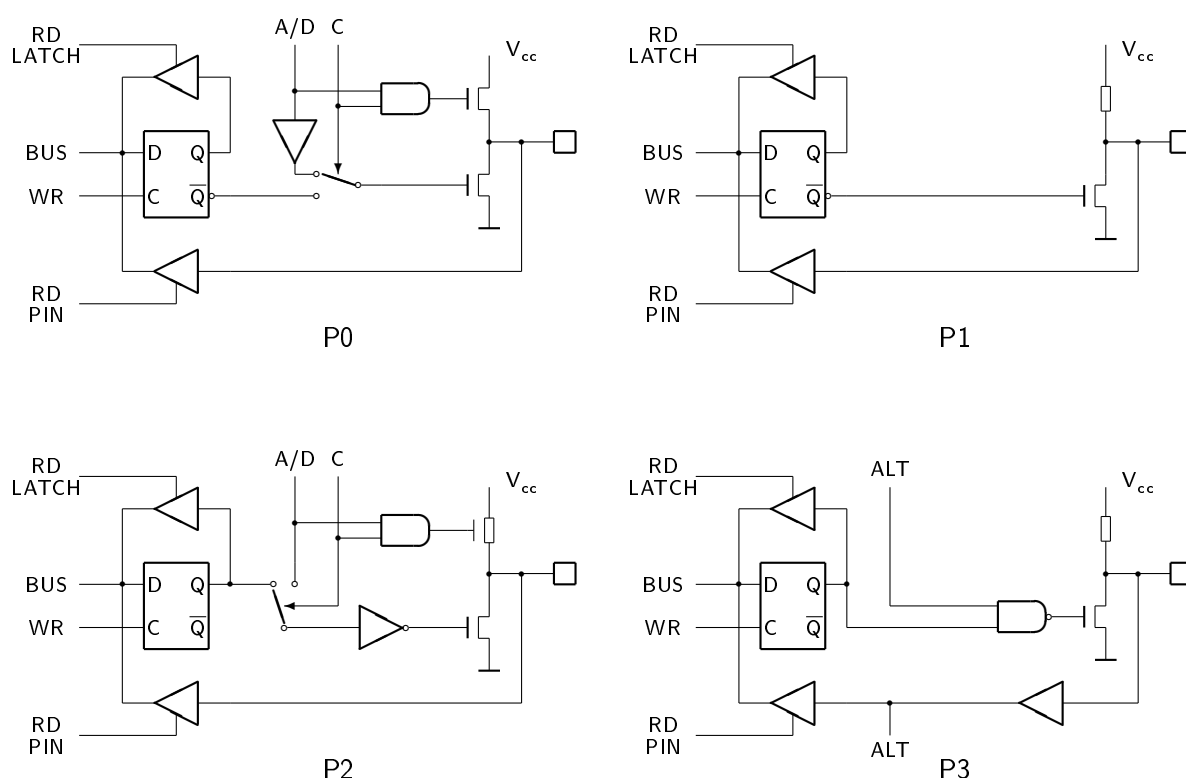
Perspektivním způsobem realizace aplikačních systémů, zejména těch, u kterých není požadována kriticky rychlá odezva a podpora speciálních obvodů rozhraní (zvláště časovačů), je využití standardní architektury *osobních počítačů*. Nejdůležitější motivací tohoto trendu je široké rozšíření této architektury, její dobrá znalost, dobrá dostupnost a nízká cena vývojových prostředků (překladačů), stále rostoucí výkonnost a klesající cena a nároky na napájení. V aplikacích se lze opřít o řadu existujících řešení (uveďme si zde např. využití grafických rozhraní pro komunikaci s obsluhou, přenos dat kartami PCMCIA).

Uvedená cesta má však i své problémy. Mají své zdroje v tom, že architektura dnešních osobních počítačů, bohužel, v době svého vzniku řadu zřejmých požadavků ignorovala. Proto se musíme u vážnějších aplikací opřít o lepší operační systém (resp. jádro reálného času) než je DOS nebo jeho modifikace, a o dražší a spolehlivější vývojové nástroje (zvláště překladače). Současná výkonová omezení (např. průchodnost sběrnice ISA) již zřejmě na překážku nebudou.

Takto pojaté aplikace však nemusí v budoucnosti nutně trpět slabými technickými a programovými prostředky současných osobních mikropočítačů. Jako zajímavost si uveďme, že např. Motorola již oznámila vstup do oblasti aplikačních systémů s rodinou dvaatřicetibitových mikropočítačů RMCU500, které jsou modifikací mikropočítačů řady Power PC.

2. Paralelní rozhraní

Nejjednodušším rozhráním, které najdeme prakticky na každém jednočipovém mikropočítači je *rozhraní paralelní*. Dovoluje programem ovládat jednotlivé signálové vývody mikropočítače (I/O vývody) buď jednotlivě, nebo jako skupiny (u osmibitových mikropočítačů většinou osmic vývodů). Takové skupiny obvykle označujeme jako *brány* nebo častěji *porty*. Typické vlastnosti jednodušších bran si můžeme ilustrovat na příkladě vnitřní struktury I/O vývodů jednočipového mikropočítače 8051 (obr.2.1).



Obrázek 2.1: Zapojení I/O vývodů mikropočítače 8051

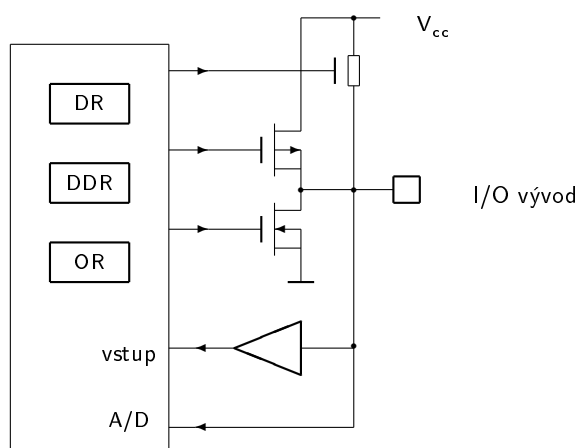
Nejjednodušší strukturu najdeme u brány označené jako P1. Každý I/O vývod je zde zapojen na budič s otevřeným kolektorem ovládaný hodnotou v datovém registru. Současně může být úroveň na I/O vývodu přenesena na vnitřní sběrnici mikropočítače. To je však možné pouze za předpokladu, že výstupní tranzistor nebyl zápisem nulové hodnoty do datového registru vývodu otevřen, a za předpokladu, že můžeme zanedbat vliv zvyšovacího odporu na zdroj vnějšího signálu. Za zmínku stojí možnost volby mezi čtením hodnoty signálu na I/O vývodu nebo na výstupu datového registru. Tato možnost umožňuje u mikropočítače 8051 odstínit vliv vnějších obvodů na funkci instrukcí, které modifikují hodnoty na paralelních branách (instrukce typu Read-Modify-Write).

Poněkud komplikovanější strukturu mají I/O vývody brány P3, které jsou alternativně využívány jako vstupy a výstupy pro řídicí signály vnější sběrnice (\overline{RD} , \overline{WR}) a pro signály dalších periferních rozhraní (čítače/časovače, sériová rozhraní, přerušovací systém). Jejich schéma upozorňuje na nutnost ponechání jednotkové hodnoty na I/O vývodu, nulová úroveň alternativní funkci blokuje.

Konečně, brány P0 a P2 lze alternativně využít pro vytvoření vnější sběrnice. Multiplexo-

vaná část sběrnice (signály AD_0 - AD_7) vyžaduje vyšší rychlost a je realizována jako třístavová. Nemultiplexovaná část sběrnice (A_8 - A_{15}) je realizována otevřenými kolektory, zvyšovací odpor je ve skutečnosti realizován tranzistorem MOSFET, jejichž impedance je při záznamu jednotkové hodnoty na výstup snižována (pro zvýšení rychlosti).

Pružnější řízení paralelního rozhraní dovoluje řešení, které definuje funkci každého I/O vývodu hodnotou v dalším - řídicím registru. Takové řešení zvolili i konstruktéři mikrořadiče ST62, strukturu odpovídající jednomu I/O vývodu uvádí obr.2.2.



Obrázek 2.2: Zapojení jednoho I/O vývodu mikropočítače ST62

Každý I/O vývod je samostatně programovatelný. Může pracovat jako třístavový výstup nebo výstup s otevřeným kolektorem. Ve funkci vstupu může mít zapojen zvyšovací odpor, může být zdrojem přerušení nebo může být připojen na vstup A/D převodníku. (Jako zajímavost uveďme, že při připojení dvou a více vývodů na vstup A/D převodníku dojde k jejich vzájemnému propojení, jedná se o netypické řešení analogového multiplexu.)

Pro řízení každé brány (osm I/O vývodů) máme k dispozici trojici registrů - datový registr (DR - I/O Port Data Register), registr směru přenosu (DDR - I/O Port Data Direction Register) a registr funkce I/O vývodu (OR - I/O Port Option Register). Použití jednotlivých bitů registru pro programování I/O vývodu uvádí tabulka 2.1.

DDR	OR	DR	
0	0	0	vstup se zvyš. odporem, bez přerušování
0	0	1	vstup bez zvyš. odporu, bez přerušování
0	1	0	vstup se zvyš. odporem, přerušování
0	1	1	vstup bez zvyš. odporu, analogový
1	0	x	výstup s otevřeným kolektorem
1	1	x	třístavový výstup

Tabulka 2.1: Programování I/O vývodů ST62

3. Měření času a kmitočtu

Důležitou funkcí jednočipových mikropočítačů a řadičů je čítání vnějších událostí, časových intervalů mezi nimi a/nebo jejich frekvence. Vnější události jsou na vstupech systému běžně prezentovány úrovněmi nebo změnami logických signálů. Od jednočipového mikropočítače a řadiče obvykle vyžadujeme, aby byl schopen v určitých časových okamžicích generovat podněty pro vnější prostředí — řídicí signály pro nejrůznější spínače. Konečně, u řady systémů je nutná vazba programu na reálný — „astronomický“ čas.

Čítání vnějších událostí lze u pomalejších dějů zajistit programem, a to buď průběžným sledováním hodnoty binárního signálu na příslušném vstupu, nebo lépe reakcí na přerušení vyvolané změnou hodnoty tohoto signálu. Některé jednočipové mikropočítače a řadiče druhé řešení dobře podporují, příkladem jsou řadiče ST62 (str.50)).

Přímé měření nebo generování časových intervalů programem je možné jen ve velmi omezené míře (vycházíme ze známého kmitočtu hodin procesoru a počtu hodinových cyklů, které vyžaduje provedení každé instrukce). Je možné ho chápat jen jako nouzové řešení pro jednoduché aplikace, kdy není k dispozici podpora časovačem. V praxi se s takovýmto řešením setkáváme v případě, kdy potřebujeme časově omezit (s nepřilíš velkou přesností) čekání v čekací smyčce, kdy potřebujeme definovat časový limit — *time-out*. V některých aplikacích přichází v úvahu i nepřímé měření a generování delších časových intervalů čítáním událostí o známé frekvenci (např. průchodů síťového napětí nulou) programem.

Ve většině aplikací je měření a generování časových intervalů opírající se čistě o běh programu nepřijatelné. Důležitou funkcí ve struktuře jednočipových mikropočítačů a řadičů proto mají podpůrné obvody schopné čítat vnější události a měřit a generovat časové intervaly nezávisle na procesoru.

3.1 Časovače a čítače

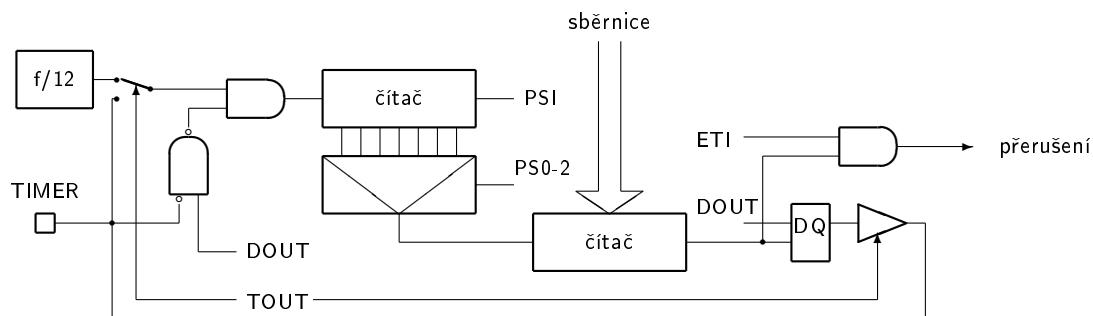
Obvody umožňující čítání vnějších událostí prodělaly od časů prvních mikropočítačů určitý vývoj. Starší jednoduché univerzální čítače/časovače mají řadu nevýhod a jsou pro měření času stále častěji nahrazovány volně běžícím nebo synchronizovaným čítačem se záchytnými a komparačními registry. Běžně se vyskytují specializované čítače a časovače (obvody reálného času, obvody Watchdog a obvody PWM).

Univerzální čítač/časovač

Nejjednodušším obvodem používaným pro čítání událostí je jednoduchý binární čítač. Ten bývá pro měření časových intervalů doplněn o předřazený dělič kmitočtu připojený na hodiny procesoru. Jako příklad si uvedeme strukturu jednoduchého *univerzálního čítače/časovače*, který nalezneme u jednočipových řadičů řady ST62 (obr.3.1).

Na hlavní (osmibitový) čítač lze přímo přivést signál ze vstupu TIMER — obvod pak plní funkci *čítače* událostí. Druhou možností je přivést na vstup hlavního čítače signál ze zvoleného výstupu předřazeného (zde osmibitového) čítače připojeného na hodinový generátor procesoru. Obvod pak plní funkci *časovače*. Vstup TIMER může být využit k hradlování vstupu hodinového signálu, hradlování dovolí zvýšit přesnost měření délky intervalu určené úrovní vnějšího signálu (např. jednotkovou) a dovolí měřit střídu i u vstupních signálů s vyšším kmitočtem.

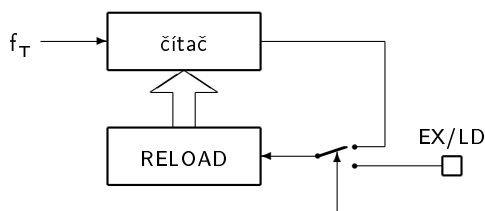
Vlastní čítač lze programem přednastavit na libovolnou hodnotu, okamžitou hodnotu čítače



Obrázek 3.1: Jednoduchý čítač/časovač mikropočítače ST62

lze programem při jeho běhu číst. (Problémy vznikají u delších čítačů, u kterých nelze hodnotu získat jedinou operací čtení). Průchod čítače nulou může být procesoru indikován jako přerušeni. Omezená délka čítače pak nemusí být na závadu, přerušovací rutina reagující na průchod nulou realizuje potřebné rozšíření rozsahu čítače v operační paměti. Vývod TIMER lze použít i jako výstup, jehož hodnotu lze měnit jak programem, tak při průchodu hodnoty hlavního čítače nulou. (Tato funkce chybí u jednoduchých čítačů/časovačů mikropočítačů řady 8051 a řady 6805).

Univerzální čítač/časovač splňuje požadavky kladené na čítání vnějších událostí a je schopen efektivně podpořit většinu funkcí spojených s časem. Jeho slabým místem je generování periodických signálů, které se u výše uvedené univerzální struktury neobejde bez opakovaného přednastavování hlavního čítače a případné ztráty přesnosti. Tam, kde nelze ztrátu výpočetního výkonu procesoru a/nebo riziko pozdního a/nebo nepřesného přednastavení zanedbat, musíme přednastavení zajistit obvodově. Příklad řešení používaného u čítačů generujících hodinový kmitočet pro komunikační obvody uvádí obr.3.2. Opírá se o registr (označovaný jako *Reload Register*), do kterého lze přednastavit hodnotu, která je potom přepsána do vlastního čítače při průchodu nulou (nebo hranou vnějšího signálu).

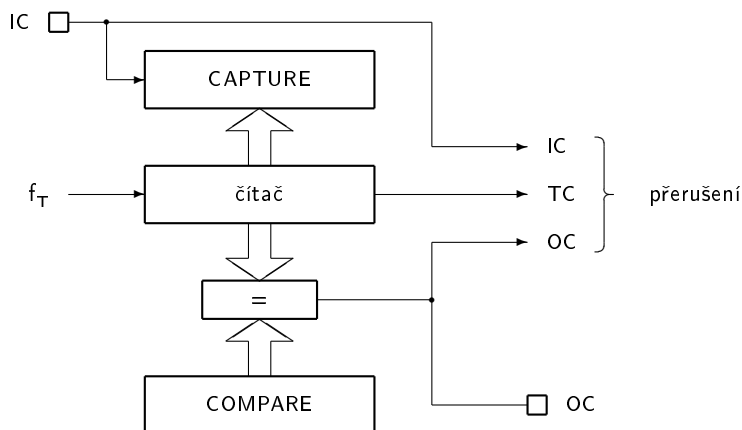


Obrázek 3.2: Automatické přednastavení univerzálního čítače/časovače

Potřebujeme-li souběžně realizovat více časových funkcí, pak s jediným univerzálním čítačem/časovačem nevystačíme. Jednočipové mikropočítače určené pro složitější aplikace mají obvykle takových čítačů/časovačů několik (např. 8051 má dva, 8052 tři čítače). V praxi se setkáme s možností konfigurovat univerzální čítače podle potřeb aplikace. Například univerzální časovače jednočipového mikropočítače 8051 lze konfigurovat jako (ne oba a ne nezávisle): 8-bitový čítač/časovač s 8-bitovým předřazeným děličem, 16-bitový čítač/časovač, 8-bitový časovač s 8-bitovým registrem pro přednastavení (Reload), kombinace 8-bitového čítače/časovače a 8-bitového časovače.

Záchytný registr

Pokud potřebujeme určit čas, ve kterém došlo k určité vnější události a máme přitom k dispozici univerzální čítač/časovač, který nám vytváří časovou základnu, postupujeme obvykle tak, že v obslužné rutině aktivované vnější událostí přečteme okamžitou hodnotu čítače/časovače a vypočteme odpovídající časový údaj. Doba, která uplyne mezi vnější událostí a čtením čítače/časovače je chybou metody. Chybu lze kompenzovat pouze částečně, protože k žádosti o přerušeni může dojít v době, kdy je obsluhováno jiné přerušeni (s vyšší prioritou), nebo kdy je přerušovací systém zamaskován.



Obrázek 3.3: Typická struktura moderního časovače

Moderní časovače řeší tento problém zapojením zjednodušeně znázorněným na obr.3.3. Volně běžící časovač, jehož přetečení je indikováno procesoru přerušením, je doplněn o *záchytný registr* (*Capture Register*). Signál informující o vnější události vygeneruje žádost o přerušeni a současně přepíše hodnotu čítače do záchytného registru. Aktivovaná přerušovací rutina čte časový údaj ze záchytného registru. Zpoždění způsobené vlastním kódem rutiny a skutečností, že rutina nemohla být spuštěna okamžitě, je tak eliminováno.

Záchytný registr dovolí přesně změřit délku časového intervalu, metoda toleruje i značně zpožděný start obslužné rutiny. Pro jednoduché řadiče vystačíme s jediným záchytným registrem (najdeme ho např. u většiny řadičů řady MC68HC05). Pro složitější aplikace, kde potřebujeme měřit více nezávislých časových intervalů je potřeba použít více záchytných registrů. Jako příklad si uvedeme jednočipové mikropočítače:

Intel 80C52

1 záchytný registr na doplněném šestnáctibitovém čítači (procesor má dále k dispozici dva univerzální čítače/časovače odpovídající jednočipovému mikropočítači 8051),

Motorola MC68HC11

4-5 záchytných registrů na šestnáctibitovém čítači a

Philips 83C552

4 záchytné registry na doplněném šestnáctibitovém čítači (procesor má dále k dispozici dva univerzální čítače/časovače odpovídající jednočipovému mikropočítači 8051).

Komparační registr

Záchytné registry zpřesňují měření časových intervalů a zjednodušují programování. Podobného zpřesnění a zjednodušení lze dosáhnout i při generování časových intervalů (jako výstupních signálů) použitím *komparačních (srovnávacích) registrů (Compare Register)*.

Obsah komparačního registru (obr.3.3) je komparátorem srovnáván s okamžitou hodnotou volně běžícího čítače. Při shodě je jednak realizována předprogramovaná změna výstupního signálu (přechod do nuly, přechod do jedničky, změna hodnoty), jednak je vygenerován požadavek na přerušení. Aktivovaná přerušovací rutina připraví další hodnotu v komparačním registru a (případně) předprogramuje další změnu výstupního signálu.

Komparační registr (a jeho podpůrné obvody) dovoluje eliminovat zpoždění způsobené přerušovací rutinou při generování signálů (např. univerzální čítače/časovače mikropočítače 8051 nemohou indikovat přímo výstupním signálem vypršení předprogramovaného intervalu). Samozřejmostí je také eliminace vlivu prodlevy mezi vypršením jednoho časového intervalu a startem dalšího, přesně lze realizovat i značně složité periodické signály.

Podobně jako u záchytných registrů, jediný komparační registr najdeme pouze u jednočipových řadičů (např. většina řadičů řady Motorola MC68HC05), moderní jednočipové mikropočítače mají počty komparačních registrů srovnatelné s počty registrů záchytných, například:

Intel 80C52

má 1 komparační registr na doplněném šestnáctibitovém čítači (procesor má dále k dispozici dva univerzální čítače/časovače odpovídající jednočipovému mikropočítači 8051),

Motorola MC68HC11

má 4-5 komparačních registrů,

Philips 83C552

má 3 komparační registry na doplněném šestnáctibitovém čítači (procesor má dále k dispozici dva univerzální čítače/časovače odpovídající jednočipovému mikropočítači 8051).

V řadě případů lze použít podpůrné registry čítače/časovače v obou funkcích. Jako příklad si uvedeme jednočipové mikropočítače

Siemens SAB80C515(535),

má čtyři univerzální registry připojené k jednomu přidanému šestnáctibitovému čítači (vedle dvou univerzálních čítačů/časovačů odpovídajících procesoru 8051), které lze jednotlivě použít jako registry záchytné nebo komparační, jeden z nich lze navíc použít k přednastavení čítače při přetečení nebo při hraně vnějšího signálu (funkce Reload) a

Siemens SAB80C517(317),

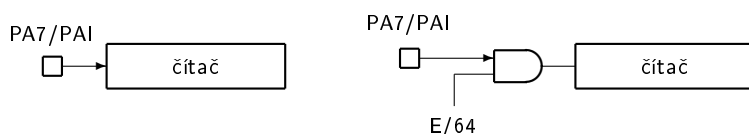
má dva přídavné čítače s možností nezávislého nastavení předděliče a přednastavení, čtyři univerzální registry použitelné jako záchytné nebo komparační pro kterýkoliv z čítačů a dalších osm komparačních registrů pevně připojených k jednomu z čítačů.

Poslední výstup, který u většiny jednotek časovačů s volně běžícím čítačem najdeme, je signál přetečení čítače. Příznak nastavovaný tímto signálem lze využít v obslužných rutinách záchytných a komparačních registrů. Signálem může být aktivována i samostatná přerušovací rutina, která pak může vytvářet časovou osu a řídit přidělování procesoru v jádrech reálného času.

Čítače událostí

Časovače vybavené záchytnými a komparačními registry jsou výhodné pro realizaci časových funkcí. Při konfiguraci, jakou najdeme u mikropočítače Motorola MC68HC11, však můžeme snadno postrádat jednoduchý čítač vnějších událostí. Jeho realizace je sice možná programově, přerušovací rutinou, při větším požadavku na rychlost nebo při větším zatížení systému s ní však nemusíme vystačit.

Pomocí zde může velmi jednoduchý osmibitový čítač, označovaný v MC68HC11 jako *Pulse Accumulator*. Jeho vnitřní struktura není nijak zajímavá, uvedeme si ji spíše pro ilustraci (obr.3.4).

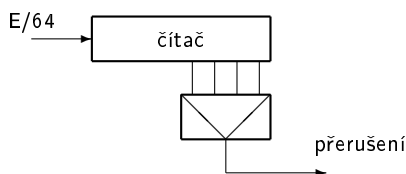


Obrázek 3.4: Čítač událostí mikropočítače MC68HC11

Čítač lze použít vedle prostého čítání vnějších událostí (Event Counting Mode) také pro měření časových intervalů (Time Accumulation Mode). Pak čítáme periody konstantního kmitočtu (např. předdělený kmitočet sběrnice - $E/64$) po dobu aktivní hodnoty vstupního signálu. V režimu čítání událostí lze volit polaritu aktivní hrany vstupního signálu a případné přerušování odvodit od zvolené hrany nebo přetečení čítače. V režimu měření intervalů může být přerušování odvozeno od přetečení čítače a/nebo od uzavření hradla.

Generátor periody

Primitivní čítač událostí nás přivedl ke specialitám mikropočítače MC68HC11. Zdržme se zde ještě chvíli a představme si další velice jednoduchý, ale o to více užitečný časovač, který je napájen z hodinového generátoru, přesněji (stejně jako čítač událostí) ze zdroje kmitočtu sběrnice (signál E). Je označován jako *generátor periody* nebo *periodický časovač* (Periodic Real Time Interrupt) a jeho úkolem je generovat přerušování každých 4.096, 8.192, 16.384 nebo 32.768 ms (pro krystal 8 MHz) pro měření času a řízení jádra reálného času. Podobně jako u čítače událostí, struktura generátoru periody je velice jednoduchá (obr.3.5).



Obrázek 3.5: Generátor periody mikropočítače MC68HC11

Generované přerušování je pro své použití jádrem reálného času označováno jako RTI (Real Time Interrupt).

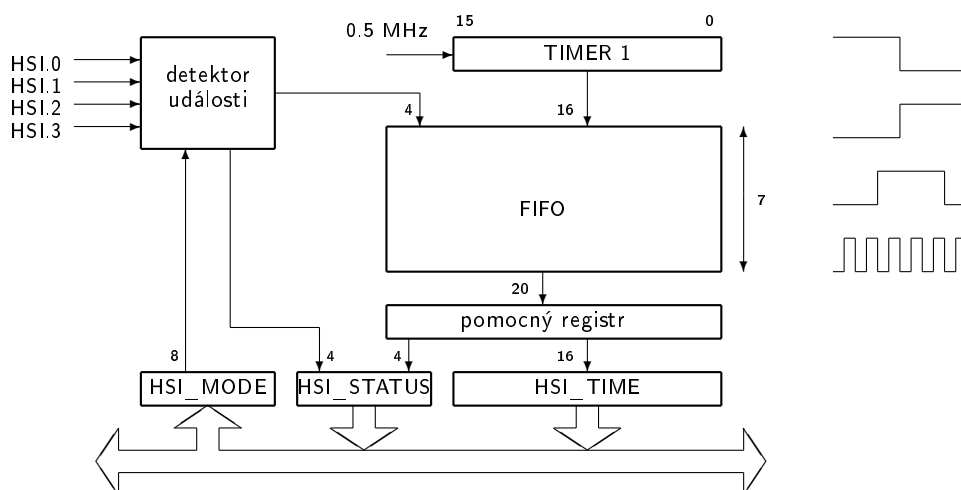
Časovací procesory

Šestnáctibitové mikropočítače, jako jsou Intel 80196, Siemens 80C166 nebo Motorola MC68HC16, jsou vybaveny ještě podstatně výkonnějším časovacím systémem než moderní mikropočítače osmibitové.

Časovačový a čítačový systém mikropočítače Intel 80196 tvoří dvě oddělené jednotky, HSI (High Speed Input) a HSO (High Speed Output), které se opírají o dva šestnáctibitové čítače.

První z čítačů, TIMER 1, je pevně připojen na dělič hodinového kmitočtu (0.5 MHz pro krystal 12 MHz). Lze ho využít pro práci jednotek HSI i HSO a může generovat přerušení při přetečení (zhruba každých 130 ms). Druhý čítač, TIMER 2, využívá vnějšího signálu a může tedy pracovat jako čítač/časovač. Může být využit pouze pro práci jednotky HSO.

Jednotka HSI je schopna zaznamenávat události (událostí je náběžná hrana, sestupná hrana, libovolná hrana nebo posloupnost osmi hran) na čtyřech vstupních kanálech (vstupy HSI.0 - HSI.3) s rozlišením 2 μ s. Informace o událostech jsou spolu s aktuálním stavem časovače TIMER 1 ukládány do paměti FIFO s hloubkou 7 záznamů. Procesor může být o uložení záznamu informován přerušením, a to buď bezprostředně nebo až po zaplnění šesti buněk paměti FIFO. Strukturu jednotky HSI uvádí obr.3.6.

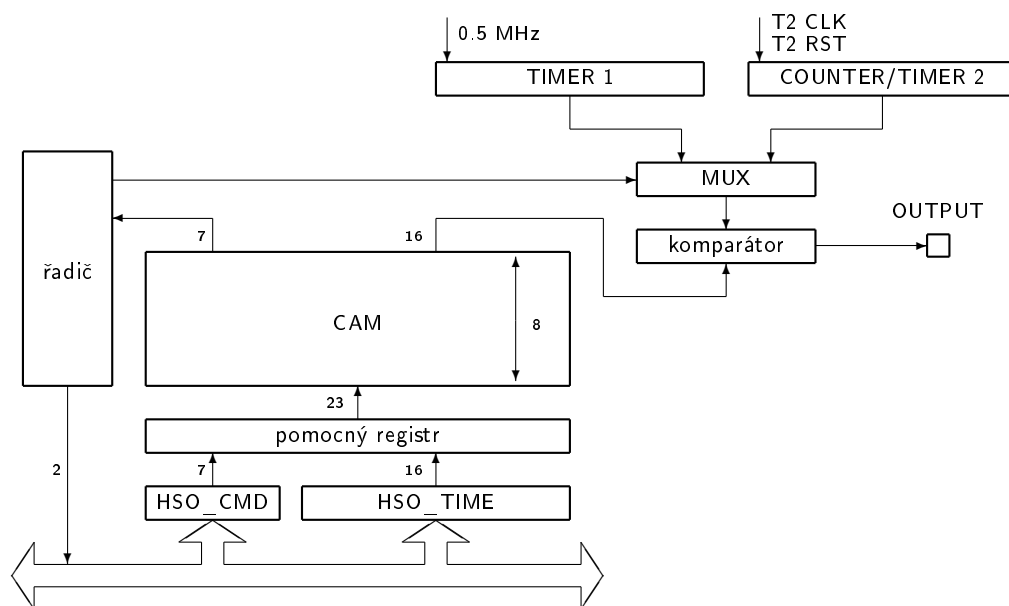


Obrázek 3.6: Jednotka HSI mikropočítače 80196

Jednotka HSO generuje naprogramované události, nastavení nebo vynulování bitu, vztažené k hodnotě jednoho z čítačů. Naprogramovanou událostí je buď změna některého z šesti výstupů jednotky HSO (výstupy HSO.1 - HSO.6), nastavení jednoho ze čtyř systémových příznaků v registru IOS1, vynulování čítače TIMER 2 nebo spuštění A/D převodníku. Pro záznam událostí je k dispozici paměť CAM (Content Addressable - výběr je řízen shodou s hodnotou ve zvoleném čítači) o kapacitě 8 záznamů. Strukturu jednotky HSO najdeme na obr.3.7

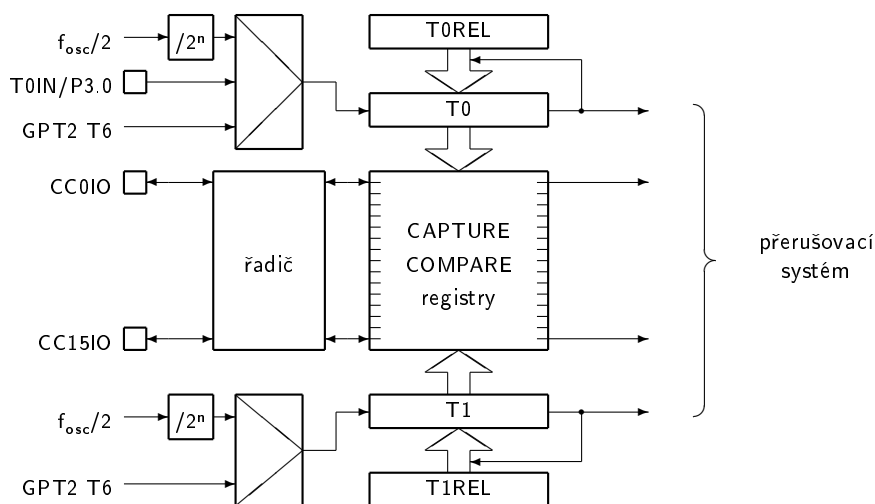
Kromě této základní časovací/čítačí jednotky u mikropočítače 80196 najdeme generátor PWM a generátor přenosové rychlosti pro sériový kanál.

Mikropočítač Siemens SAB80C166 je vybaven jednotkou CAPCOM, což je pole šestnácti univerzálních registrů, které lze ve vazbě na dva šestnáctibitové čítače použít jako záchytné a komparační. První z čítačů, T0, lze napájet z předřazeného děliče hodinového signálu, z výstupu univerzálního čítače GPT2, nebo ze signálu na vývodu T0IN (P3.0). Čítač T1 lze napájet pouze vyděleným hodinovým signálem nebo z výstupu čítače GPT2. Oba čítače jsou vybaveny přednastavením z pomocného registru (funkce Reload) a lze tak libovolně definovat jejich



Obrázek 3.7: Jednotka HSO mikro počítače 80196

periodu čítání. Strukturu jednotky CAPCOM určené hlavně pro méně náročná měření času a generování signálů PWM uvádí obr.3.8.



Obrázek 3.8: Jednotka záchytných a komparačních registrů mikro počítače 80C166

Čítačová jednotka CAPCOM tvoří pouze část vybavení mikro počítače 80166 pro potřeby čítání/časování. K dispozici máme ještě pět šestnáctibitových čítačů rozdělených do dvou modulů GPT1 a GPT2. Tyto čítače vedle běžných funkcí, jakými jsou měření časových intervalů s využitím vnějšího hradlovacího signálu a prosté čítání, dovolují realizovat i složitější funkce jako jsou obousměrné čítání a dělení kmitočtu.

Konečně, přechod k nové generaci časovačů/čítačů zřejmě představuje specializovaný procesor, kterým je vybaven mikro počítač Motorola MC68HC16. Ten je, vedle časovače s pomocnými registry, osmibitového čítače, dvou PWM kanálů a obvodu Watchdog převzatých od MC68HC11, vybaven programovatelnou jednotkou TPU (Time Procesor Unit). Ta obsahuje šestnáct pro-

gramovatelných kanálů časovače s vlastním jednoduchým procesorem, kterému jsou parametry předávány v paměti RAM.

3.2 Obvody reálného času

Řídící, měřicí a dohlížecí systémy se často neobejdou bez informace o reálném (skutečném, „astronomickém“) čase. V mnoha případech nepostačí relativní údaje, vztažené například k okamžiku spuštění programu, nebo údaje o délkách časových intervalů. Potřebujeme absolutní údaj o vteřině, minutě, hodině, dni, měsíci a roce, v němž ta která událost nastala.

V méně náročných aplikacích lze měření reálného času zvládnout s prostředky samotného mikropočítače. Stačí zajistit periodické generování přerušování vnitřním čítačem a v přerušovací rutině zajistit střídání odpovídajících časových přírůstků a převod nastřádané hodnoty do běžně používané soustavy (rok, měsíc, den, hodina, minuta, vteřina). Frekvence přerušování bude zřejmě kompromisem mezi požadavkem na časové rozlišení, požadavkem na minimální ztrátu výpočetní kapacity (obětované na pravidelně spouštěnou obslužnou rutinu) a požadavkem vyloučit riziko, že nestačíme obsloužit všechny žádosti o přerušování.

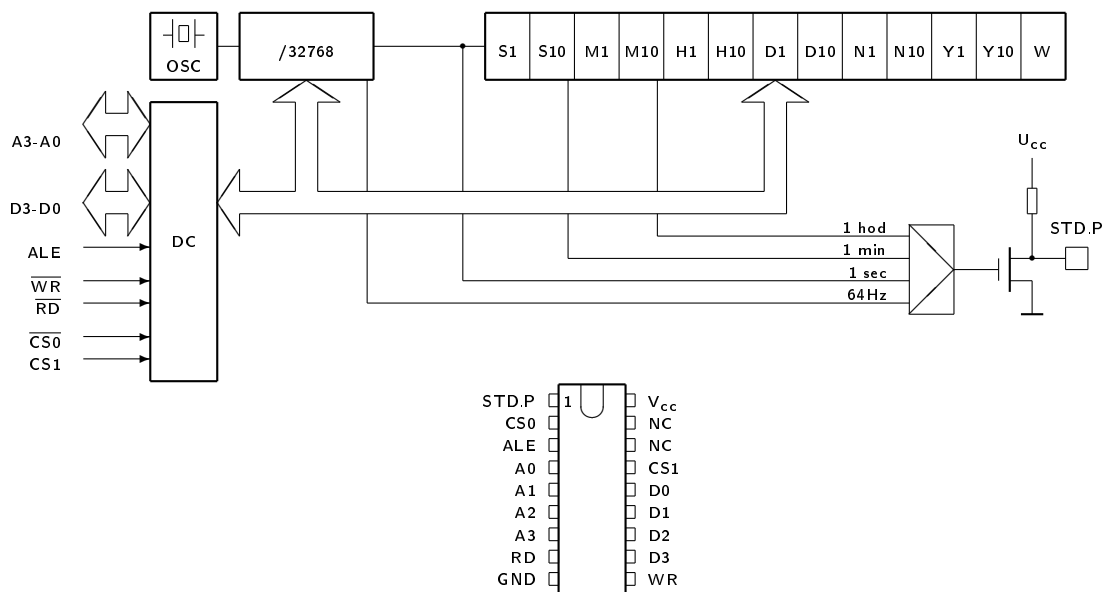
Měření reálného času přímo mikropočítačem je často použitelné, má však, bohužel, řadu nepříjemných vlastností. Přesnost měření přímo závisí na přesnosti a stabilitě hodinového generátoru, které jsou už z hlediska principu běžně používaného zapojení (krystal ve zpětné vazbě invertoru) nedostatečné. Daleko podstatnější nevýhodou je však nutnost trvalé práce hodinového generátoru na vysokém kmitočtu (a tedy s poměrně vysokou spotřebou). To téměř vylučuje možnost měřit reálný čas přímo mikropočítačem v případech, kdy jsme závislí na bateriovém zdroji, nebo kdy vyžadujeme měření reálného času při přepnutí na záložní bateriový zdroj.

Pokud nám jednoduché měření reálného času pro některý z uvedených důvodů nevyhoví, nebo pokud si nechceme komplikovat program mikropočítače obsluhou periodického přerušování, můžeme využít obvody označované jako *obvody reálného času*. Struktura těchto obvodů se podobá obvodům digitálních hodinek, trvale běžící oscilátor pracující s krystalem 32.768 kHz je navržen s ohledem na stabilitu a má velmi malou spotřebu. Krystal je často vestaven přímo do pouzdra obvodu, kmitočet oscilátoru je běžně dodržen s přesností ± 50 ppm (part per milion), což odpovídá maximální chybě zhruba 5 sec za den, výběrové obvody zaručují přesnost ± 10 ppm. Údaje nasčítané v dekadických čítačích (jednotky vteřin, vteřiny, jednotky minut, minuty, jednotky hodin, hodiny, den, den v týdnu, dekáda, měsíc, rok) lze přecházet na nejčastěji čtyřbitové datové sběrnici. Jednotlivé řády čítačů lze pochopitelně přednastavit, čítač lze pozastavit a znovu spustit.

Jednoduché obvody reálného času produkují periodický signál využitelný např. jako vstup pro periodické přerušování (resp. vzbuzení mikropočítače) každou vteřinu, minutu, apod. Složitější obvody jsou vybavovány programovatelným budíkem (funkce Alarm), možností vygenerovat výstupní signál při dosažení předvoleného času. Obvody reálného času musí být trvale napájeny (vlastní baterií nebo z bateriového záložního zdroje) a jsou proto často doplňovány o malou paměť RAM, do které lze uložit data, která mají být zachována při vypnutí/výpadku hlavního napájecího zdroje.

Z řady obvodů různých výrobců jsou nejdostupnější a nejčastěji používané jednoduché obvody Sony RTC-62421/423 a RTC-72421/423. Jejich vnitřní struktura odpovídá obr.3.9.

Sběrnice obvodu dovoluje přímé připojení na dostatečně širokou (alespoň čtyřbitovou) ne-multiplexovanou nebo multiplexovanou (adresa/data) sběrnici, ve druhém případě rozlišení adres a dat na propojených vývodech zajišťuje signál ALE (odpovídající signálu ALE mikropo-



Obrázek 3.9: Struktura obvodu reálného času Sony RTC-72421

čítače 8051). Jednotlivé stupně řadiče jsou přímo adresovatelné, stejně jako tři řídicí registry. Využití adresového prostoru uvádí tab.3.1.

Adr	A3	A2	A1	A0	Jm,no	D3	D2	D1	D0	
0	0	0	0	0	S1	s8	s4	s2	s1	jednotky vteřin
1	0	0	0	1	S10		s40	s20	s10	desítky vteřin
2	0	0	1	0	M1	m8	m4	m2	m1	jednotky minut
3	0	0	1	1	M10		m40	m20	m10	desítky minut
4	0	1	0	0	H1	h8	h4	h2	h1	jednotky hodin
5	0	1	0	1	H10		P/AM	h20	h10	desítky hodin
6	0	1	1	0	D1	d8	d4	d2	d1	jednotky dní
7	0	1	1	1	D10			d20	d10	desítky dní
8	1	0	0	0	N1	n8	n4	n2	n1	jednotky měsíců
9	1	0	0	1	N10				n10	desítky měsíců
A	1	0	1	0	Y1	y8	y4	y2	y1	jednotky roků
B	1	0	1	1	Y10	y80	y40	y20	y10	desítky roků
C	1	1	0	0	W		w4	w2	w1	den v týdnu
D	1	1	0	1	CD	ADJ30	IRQ	BUSY	HOLD	řídicí registr D
E	1	1	1	0	CE	t1	t0	I/S	MASK	řídicí registr E
F	1	1	1	1	CF	TEST	24/12	STOP	RST	řídicí registr F

Tabulka 3.1: Adresy registrů obvodu reálného času Sony RTC-72421

Obvod je určen pro napájení v rozmezí 4.5 až 5.5 V a má odběr pouhých 10 μA . Je vyráběn v pouzdrech DIL-18 a SOIP-24.

Použití obvodů reálného času se neomezuje pouze na mikropočítače s vnější paralelní sběrnicí. Najdeme mezi nimi i moderní obvody Sony RTC-4503 a RTC-4553 určené pro sériovou sběrnicí Microwire, (odběr pod 1 μA , pracují od 1.1 V (RTC-4503) a 2.7 V (RTC-4553)), pro sběrnicí I²C je určen např. obvod Philips PCF8583.

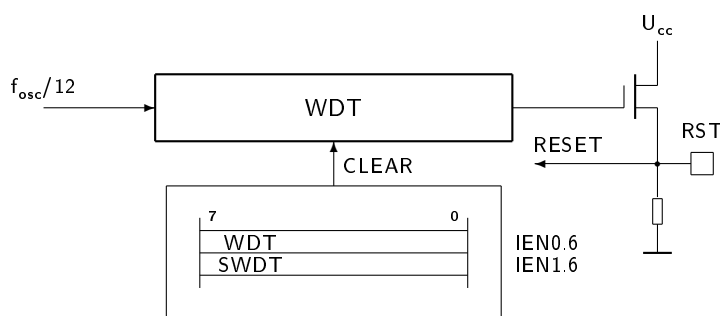
3.3 Hlídací obvody Watchdog a Power-Fail

Termínem hlídací (nebo dohlížecí) obvody obvykle označujeme obvody, které dohlíží na správný běh programu (jsou známé pod anglickým termínem Watchdog), a obvody, které dohlíží na dodržení základní podmínky, za které je mikroprocesor schopen bezporuchově pracovat, tedy odpovídající hodnoty napájecího napětí (jsou známé pod anglickým termínem Power-Fail).

Hlídací obvod *Watchdog* je analogový nebo digitální časovač, buď přímo vestavěný do struktury mikroprocesoru, nebo připojený zvnějšku, který je odstartován buď bezprostředně po resetu mikroprocesoru nebo někdy později z programu. Jeho časová konstanta může být pevně svázána s hodinami mikroprocesoru, nebo může být nějakým způsobem nastavena (zapojením a hodnotami vnějších prvků nebo programem). Po vypršení časového limitu časovač vyvolá automatický reset, pokud ho ovšem dříve nějakým podnětem (signálem, hodnotou zapsanou do řídicího registru, speciální instrukcí) nevynulujeme, tj. nevrátíme do počátečního stavu.

Využití hlídacího obvodu Watchdog je založeno na jednoduché myšlence: do programu jsme schopni vložit příkazy nulující časovač hlídacího obvodu Watchdog tak, že při jeho „správné“ funkci nedojde k vypršení časového limitu. Porucha, způsobená například náhodnou změnou dat v paměti, nebo chyba v programu, například neošetřená určitá kombinace vstupních dat, často vyvolá změnu chování programu. V řídicích systémech lze některé chyby chování tolerovat, protože nevedou k problémům v řízeném systému, jiné jsou netolerovatelné. Netolerovatelnou chybou chování je jistě zacyklení programu, chybné zamaskování přerušovacího signálu nebo překonfigurování periferních obvodů. A právě v těchto případech je hlídací obvod Watchdog užitečný, program v nekonečném cyklu, nebo program, která není pravidelně aktivován vnějšími podněty nevynuluje do časového limitu časovač obvodu Watchdog, a ten program z nežádaného stavu vyvede resetem. Obvod Watchdog není tedy nic jiného než technická implementace našeho chování před obrazovkou osobního počítače, který se v důsledku náhodné chyby technických prostředků, „operačního systému“ nebo aplikačního programu, „zakousne“.

Nejjednodušším hlídacím obvodem Watchdog je vestavěný, pevně nastavený čítač, připojený na vnitřní hodiny mikroprocesoru. Jako příklad si můžeme uvést hlídací obvod Watchdog mikroprocesoru Siemens SAB80C515/535 (obr.3.10).

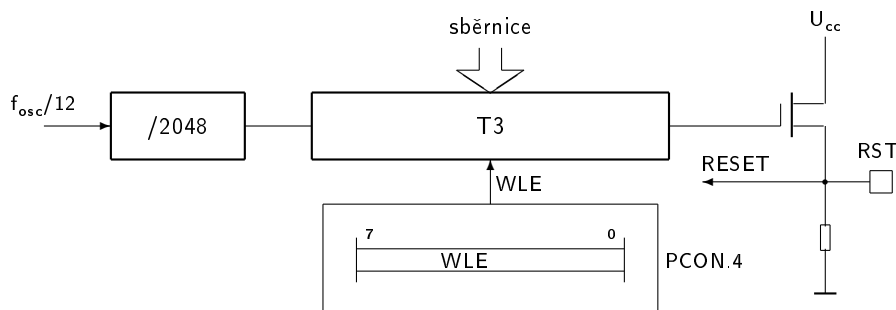


Obrázek 3.10: Struktura obvodu Watchdog mikroprocesoru SAB80C515/535

Hlídací obvod lze programem aktivovat (vypnout ho však již pochopitelně nelze), čítač WDT je nulován postupným nastavením bitů WDT a SWDT ve dvou řídicích registrech.

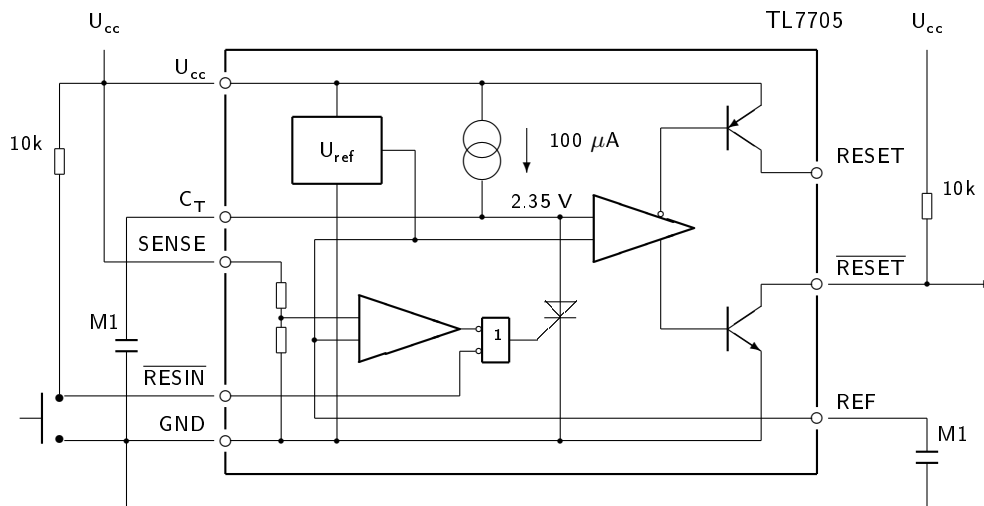
Nevýhodou takto jednoduchého řešení je pevný časový limit (v tomto případě odpovídá přetečení šestnáctibitového čítače napájeného dvanáctinou hodinového signálu 12 MHz zhruba 65 ms), který může podstatně zkomplikovat programování složitějších aplikací. Složitější strukturu obvodu Watchdog najdeme u téměř všech současných mikroprocesorů. Většinou jde o předřazení

programovatelného děliče kmitočtu hodinového signálu (např. Motorola MC68HC11 nebo PIC16C5x) nebo o možnost nastavit počáteční hodnotu čítače obvodu Watchdog. Příklad posledního řešení obvodu Watchdog v mikroočítači Philips 80C552 uvádí obr.3.11.



Obrázek 3.11: Struktura obvodu Watchdog mikroočítače 80C552

Obvod Watchdog musí být chráněn proti náhodnému nulování nebo dokonce vypnutí chybně běžícím programem. Náhodnému nulování se obvykle bráníme tím, že nulování není reakcí na změnu hodnoty jediného bitu v řídicím registru, ale je vyžadována určitá posloupnost změn ve více bitech nebo řídicích registrech. Ochranou proti náhodnému vypnutí může být možnost konfigurovat obvod Watchdog pouze při programování (např. u mikrořadiče PIC16C5x, kde je bit aktivující obvod Watchdog realizován jako buňka EPROM), povolit konfiguraci pouze v omezeném časovém intervalu po resetu (např. u mikroočítačů Motorola MC68HC11), nebo dovolit pouze aktivaci, ale už ne vypnutí (např. u mikroočítačů Siemens).

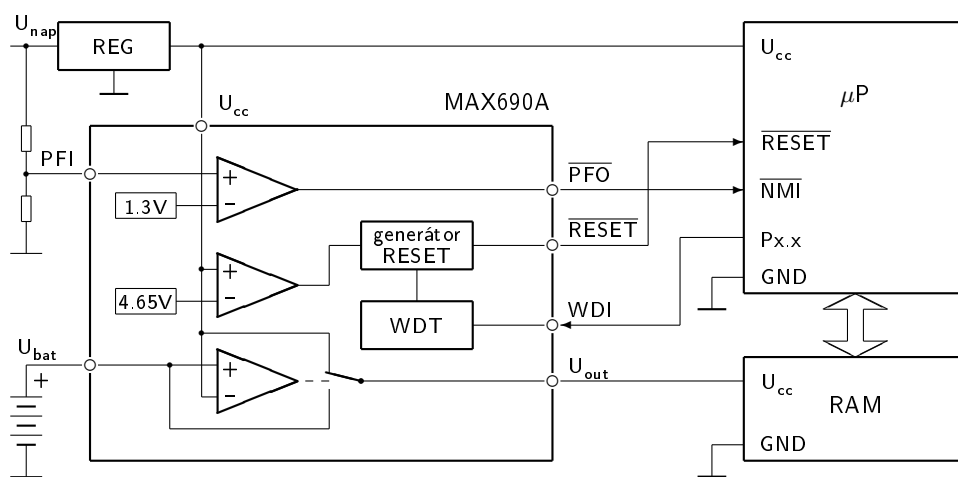


Obrázek 3.12: Obvody Texas Instruments TL77xxA a jejich použití

Funkci hlídání správného rozmezí napájecích napětí zajišťují obvody označované jako obvody *Power-Fail*. Tyto obvody dovolují informovat mikroočítač o snížení napájecího napětí pod stanovenou mez a dát mu tak šanci bezpečně ukončit svou funkci před následujícím výpadkem (např. u systémů bez zálohování napětí pro procesor nebo paměť RAM uložit rychle důležité údaje do paměti EEPROM a nedovolit další zápisy do této paměti). Jsou běžně kombinovány se startovacími obvody, které zajistí bezpečný reset při náběhu napájecího napětí (obvody *Power-On Reset*). Nejjednodušší obvody dovolující realizovat uvedené funkce obsahují zdroj referenčního napětí, komparátor, obvod dovolující generovat časovou prodlevu a

jednoduchou řídicí logiku. Strukturu často používaných obvodů Texas Instruments TL77xxA a jejich použití pro funkce Power-Fail a Power-On Reset uvádí obr.3.12. (Na místě znaků xx najdeme dvoumístné číslo udávající napětí, pro které je obvod určen, pro mikropočítačové aplikace přichází v úvahu varianta TL7705A.)

Řada návrhářů dává před vestavěnými obvody Watchdog přednost obvodům vnějším, které současně hlídají napájecí napětí. Typickým představitelem těchto obvodů jsou obvody řady MAX690(A) firmy Maxim (a její ekvivalenty produkované dalšími výrobci). Jsou kombinací jednoduchého obvodu Watchdog s pevnou časovou konstantou (zhruba 1.6 s) s obvodem pro hlídání napájecího napětí, s obvodem pro přepnutí na záložní zdroj a se startovacím obvodem. Strukturu obvodu MAX 690A a příklad jeho použití najde čtenář na obr.3.13.



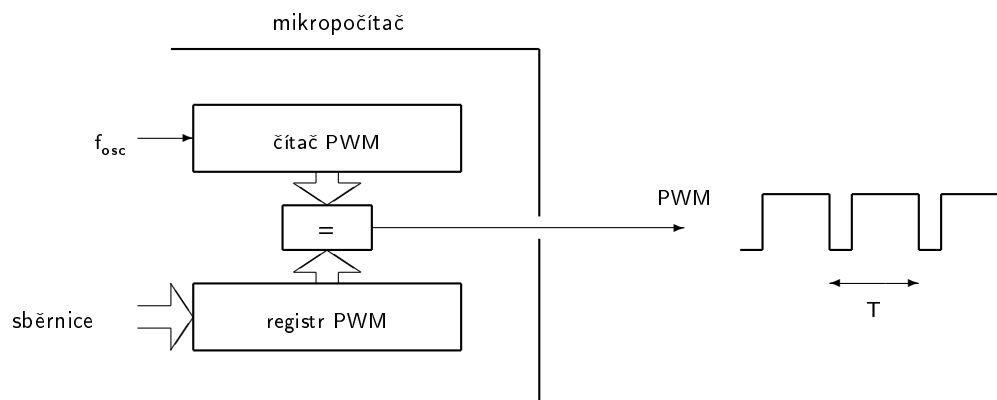
Obrázek 3.13: Hlídacího obvodu Maxim MAX 690A a jeho použití

Při náběhu napájecího napětí udrží obvod MAX690A na výstupu $\overline{\text{RESET}}$ nulovou hodnotu ještě 140 ms po vzrůstu napětí na vývodu U_{cc} nad 4.65 V a zajistí tak bezpečný reset mikropočítače. Pokud napětí na vývodu U_{cc} poklesne pod 4.65 V, obvod MAX690A na vývodu $\overline{\text{RESET}}$ nulovou úroveň bezpečně mikropočítač zastaví. Komparátor s výstupem $\overline{\text{PFO}}$ dovolí v předstihu informovat mikropočítač o očekávaném poklesu napájecího napětí (jeho vstup PFI připojujeme obvykle před stabilizátor napájecího napětí). Další komparátor dovolí připnout k výstupu U_{out} pomocnou baterii a to při poklesu napětí U_{cc} pod napětí baterie U_{bat} , v našem případě baterie zálohuje napájení pro vnější paměť RAM. Konečně, vstup WDI dovoluje nulovat čítač obvodu Watchdog, který, není-li po dobu 1.6 s vynulován, způsobí vygenerování signálu $\overline{\text{RESET}}$.

3.4 Obvody PWM

Obvody *pulsně-šířkové modulace* (*PWM* — Pulse Width Modulation) řadíme do této kapitoly z toho důvodu, že jejich vnitřní struktura je obdobná čítačům (starší univerzální čítače a moderní čítače s komparačním registrem lze konečně ke generování signálu PWM snadno využít). Z hlediska použití bychom měli obvody PWM spíše zařadit do kapitoly, zabývající se generováním analogových signálů.

Struktura jednobanárového obvodu PWM je uvedena na obr.3.14:



Obrázek 3.14: Struktura obvodu PWM mikropočítače Intel 80196

Hodnota volně běžícího čítače hodinového signálu (výjimečně hodinového signálu po jeho předběžném vydělení konstantou) PWM je srovnávána s hodnotou nastavenou programem v registru PWM. Výsledek srovnání je prezentován jako logická úroveň na vývodu PWM. Perioda výstupního signálu je pevná a je definována čítaným signálem a modulem čítače CLK. Střída výstupního signálu je úměrná hodnotě zapsané do registru PWM.

Obvod PWM je dnes běžným vybavením jednočipových mikropočítačů. Najdeme ho u řady variant mikrořadiče MC68HC05, u jednočipových mikropočítačů MC68HC11, PCB83C752 a dalších. Často se setkáme s obvody vícekanálovými, příkladem mohou být dva kanály PWM mikropočítače PCB80C552. Takové obvody PWM se obvykle opírají o jediný volně běžící čítač, pouze registry PWM a komparátory jsou samostatné.

4. Sériová komunikace

Důležitým prvkem, který najdeme téměř v každém mikropočítačovém systému nebo jednočipovém mikropočítači, je rozhraní pro sériovou komunikaci. Sériová rozhraní pro *dvoubodové spoje* vycházejí z telekomunikačních standardů a umožňují připojit k mikropočítači jednotlivá zařízení samostatnými vedeními na vzdálenost jednotek až stovek metrů.

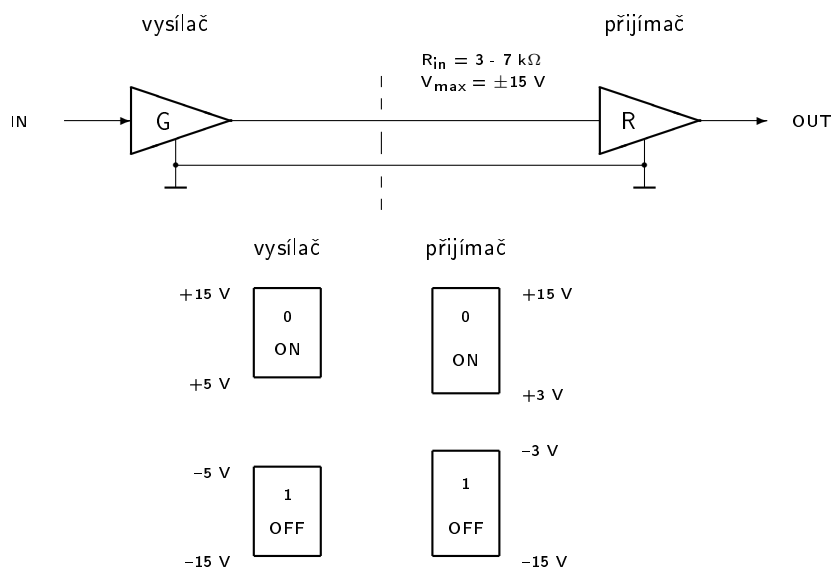
Sériová rozhraní dovolující vytvářet *vícebodové spoje* na jediném vedení vznikla buď modifikací standardů pro dvoubodovou komunikaci nebo využívají běžné logické signály TTL. Z hlediska elektrických parametrů tak můžeme rozhraní pro vícebodovou komunikaci rozdělit do dvou skupin. Prvou skupinu tvoří rozhraní podporující vytváření soustav měřících a řídicích prvků propojených dvoudrátovými spoji na vzdálenost jednotek až desítek metrů. Jde se o ekonomickou variantu lokálních sítí pro technologické řízení, takové lokální sítě jsou označovány termínem *FieldBus*. Druhou skupinu tvoří rozhraní podporující komunikaci mezi mikropočítačem a pomocnými obvody v rámci jednoho konstrukčního modulu, jejich účelem je zjednodušit konstrukci. Spoje takto vytvářené budeme označovat termínem *sériová sběrnice*.

4.1 Elektrické standardy

Elektrické parametry obvodů rozhraní určují, jakou rychlostí a na jakou vzdálenost mohou propojená zařízení komunikovat. Signály TTL lze přenášet jen na malé vzdálenosti a jsou navíc citlivé na rušení. Pro spolehlivější přenos na větší vzdálenosti je nutné použít signálů s odlišnými parametry. V následujících odstavcích si uvedeme ty, které jsou nejčastěji používány.

RS-232C

Díky svému rozšíření jako sériového rozhraní osobních počítačů se telekomunikační rozhraní RS-232C EIA, resp. V.24/V.28 CCITT (ve značně zjednodušené konfiguraci) stalo široce používaným standardem pro sériový přenos i v mikropočítačové řídicí technice.

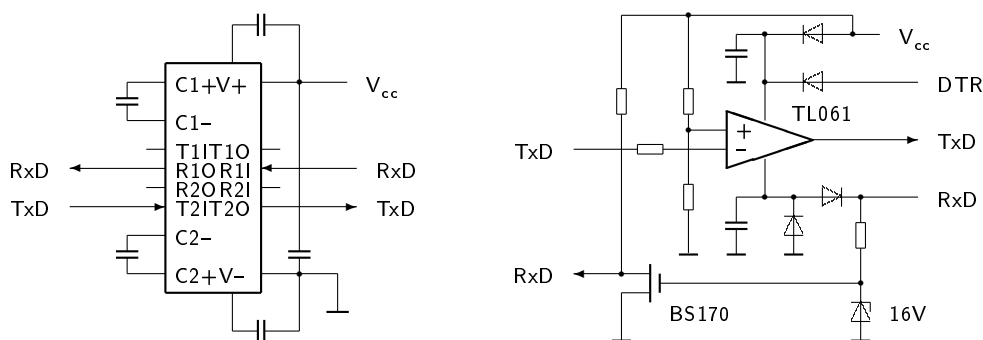


Obrázek 4.1: Sériové rozhraní RS-232C

Elektrické parametry rozhraní, které bylo původně určené pro přenos dat rychlostí do 19.2 kb/s na vzdálenost do 20 m (propojení modemu s terminálem nebo počítačem) shrnuje obr.4.1.

Nesymetrický vysílač rozhraní G má poměrně vysoký vnitřní odpor (ochrana proti zkratům na vedení) a generuje napěťový signál vztažený k zemnímu potenciálu vysílače. Nesymetrický přijímač rozhraní R tento signál vyhodnocuje vzhledem k zemnímu potenciálu přijímače. Pro vysokou vnitřní impedanci vysílače a nezbytnou kapacitu vedení nelze signál přenášet na větší vzdálenost. Nesymetrický přijímač R je značně citlivý na napěťové rozdíly v zemních potenciálech vysílače a přijímače.

Převod napěťových úrovní pro toto rozhraní je podporován řadou obvodů mnoha výrobců. Dnes už historické obvody SN75150 (dvojitý vysílač), MC1488 (čtyřnásobný vysílač), SN75154 a MC1489 (čtyřnásobné přijímače) vyžadují pro svou činnost symetrické napájecí napětí. V současné mikropočítačové technice jsou nahrazovány aplikačně jednoduššími obvody s vestavěným měničem napětí. Typickým obvodem tohoto typu je Maxim MAX232, zapojení převodníku pro sériové rozhraní RS-232C s tímto obvodem uvádí obr.4.2.

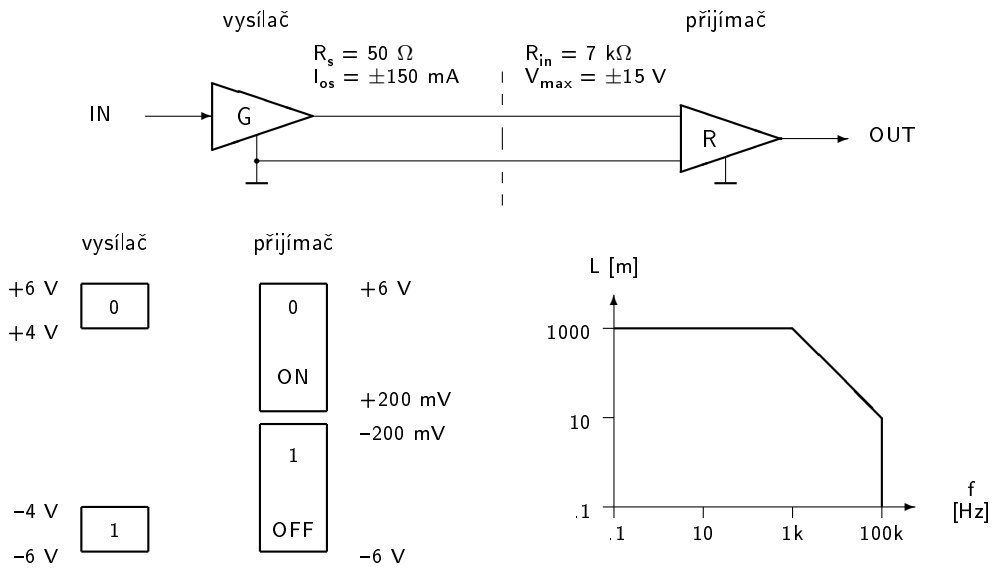


Obrázek 4.2: Obvod sériového rozhraní s převodníkem úrovní MAX232

Rozhraní RS-232C je pravidelně využíváno i pro připojování mikropočítačových zařízení k osobním počítačům při ladění programů, k nastavování parametrů řídicích programů, k předávání nasbíraných dat nebo diagnostických údajů, tedy pro činnosti, které nejsou časté. V takovýchto případech může být použití měniče pro napájení převodníku (o zvláštním napájecím napětí ani nemluvě) neekonomické. Často se proto můžeme setkat s převodníkem, který své napájecí napětí získává ze signálů sériového rozhraní osobního počítače. Příklad řešení takového převodníku uvádí obr.4.2.

RS-423

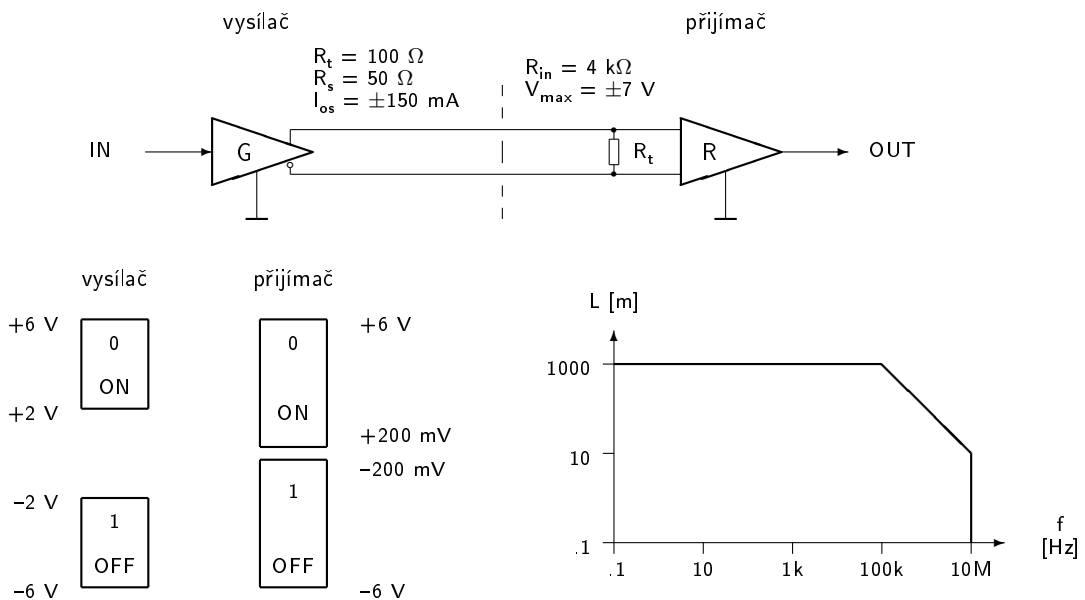
Náhradou nesymetrického přijímače přijímačem symetrickým, úpravou napěťových úrovní vysílače, ale hlavně snížením jeho vnitřní impedance dostáváme modernější komunikační rozhraní RS-423 EIA, resp. V.10/X.26 CCITT. Jeho využití pro komunikaci mikropočítačových zařízení není zajímavé, uvádíme ho zde proto, že se jedná o krok k rozhraním výhodnějším. Jeho elektrické parametry a graf závislosti mezi přenosovou rychlostí a překlenutou vzdáleností shrnuje obr.4.3.



Obrázek 4.3: Sériové rozhraní RS-423

RS-422

Symetrický vysílač a zatížení vedení mezi vysílačem a přijímačem charakteristickou impedancí je dalším krokem na cestě k vyšší rychlosti a překlenuté vzdálenosti. Parametry rozhraní RS-422 EIA, resp. V.11/X.27 CCITT uvádí obr.4.4.



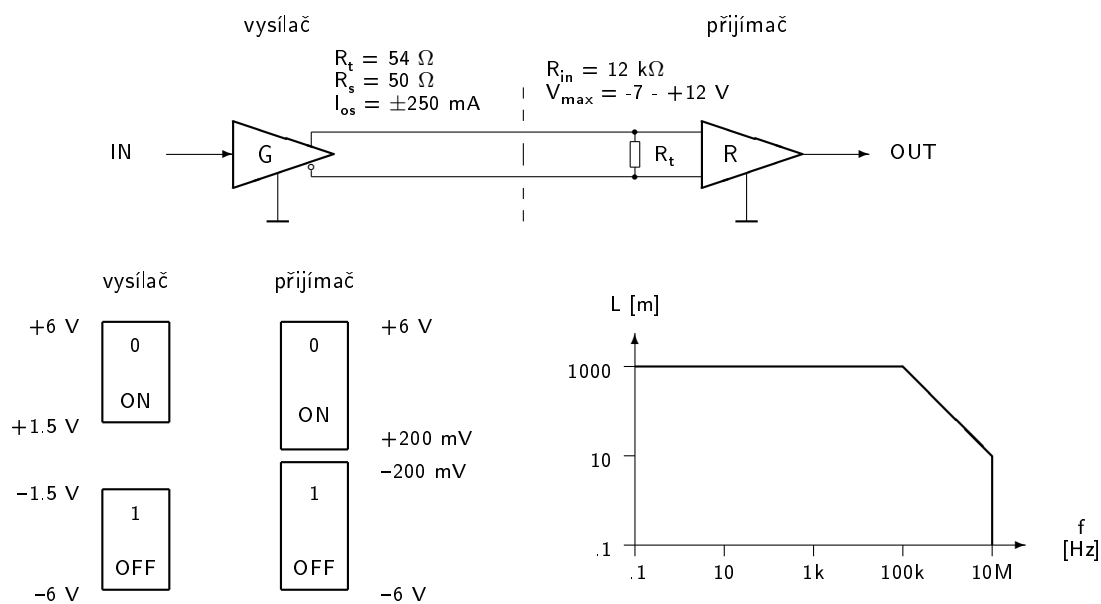
Obrázek 4.4: Sériové rozhraní RS422

Příjemným důsledkem použití symetrického vysílače je, že vystačíme s jediným napájecím napětím 5 V. Poměrně vysoká vstupní impedance přijímače dovoluje propojení jednoho vysílače s více přijímači, sběrniceový provoz není běžně předpokládán. Typickými převodníky úrovní pro rozhraní RS-422 jsou obvody 26LS31 (čtyřnásobný vysílač) a 26LS32 (čtyřnásobný přijímač).

RS-485

Dosud uváděná sériová rozhraní jsou určena pro dvoubodové propojení jednoho vysílače s jedním případně více (u rozhraní RS-422) přijímači. Při komunikaci ve skupině mikropočítačových zařízení však obvykle vyžadujeme obousměrný přenos dat, schopnost koordinované práce více vysílačů na jediném vedení. Vytváříme sériovou sběrnici.

Sériová sběrnice RS-485 EIA, která se v praxi používá téměř výlučně, vznikla úpravou posledního uváděného rozhraní RS-422. Úprava se týká jednak oboustranného zakončení dvoudrátového vedení charakteristickou impedancí, jednak změny mezních napětí na vstupu přijímače (obr.4.5).



Obrázek 4.5: Sériové rozhraní RS485

Typickými převodníky napětí pro sběrnici RS-485 jsou obvody SN75174 (čtyřnásobný vysílač) a SN75175 (čtyřnásobný přijímač). V praxi se však nejčastěji setkáme s kombinovaným obvodem SN75176 (transceiver, vysílač a přijímač v osmivývodovém pouzdře).

V prostředí, kde nejsou propojená zařízení napájena z jediného zdroje, nebo v prostředí, kde se mohou objevit rušivé signály s úrovněmi, které přesahují mezní napětí vysílače a přijímače, je nutné vážně uvažovat o optické izolaci vlastního převodníku rozhraní od zbytku zařízení optočlenem. To se týká libovolného z uvedených standardů (ale s ohledem na četnost aplikací hlavně rozhraní RS-232C a RS-485).

Proudová smyčka

Posledním důležitým rozhraním, kterého si musíme přes jeho jednoduchost všimnout je proudová smyčka. Pro jednosměrný přenos se využívá dvoudrát, signálem je proud (jednička odpovídá 20 mA, nule proud nulový). Vysílačem je (polovodičový, opticky oddělený) spínač, přijímačem optočlen. Pro obousměrnou komunikaci je nutné vytvořit dva kanály. Výhodou je odolnost proti rušení, přechodovým odporům na vedení, elektrické oddělení zařízení od vedení. Nepříjemností je potřeba izolovaného zdroje (vyžadujeme-li izolaci vysílače) a nízká úroveň standardizace, která komplikuje propojování různých zařízení.

4.2 Komunikační protokoly

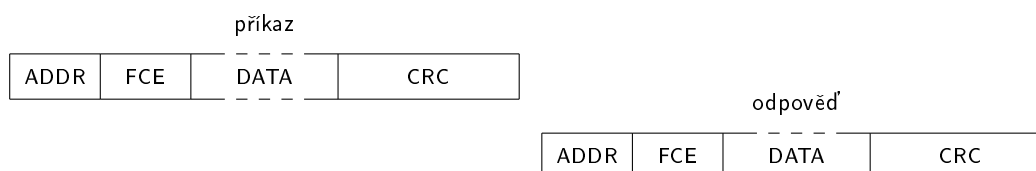
Dosud jsme se zabývali elektrickými vlastnostmi sériových rozhraní a odpovídajícími převodníky napěťových úrovní. Pro komunikaci jsou však stejně důležitá pravidla, podle kterých jsou předávány informace mezi komunikujícími zařízeními, *komunikační protokoly*.

Komunikační protokol (nejnižší, linkové úrovni) má zajistit bezchybný přenos dat. Musí zajistit takový formát zobrazení přenášených dat (znaků, bloků znaků), který lze na straně přijímače jednoznačně dekodovat, a poskytnout možnost vyloučit chyby způsobené přenosem signálu. V praxi je používána celá řada řešení. Jako příklady si uvedeme jednoduchý asynchronní protokol Modbus, devítibitovou komunikaci, pro kterou jsou vybaveny mikropočítače 8051 a 68HC11, protokol Bitbus. Současně si všimneme principů na nichž jsou budovány nejnižší úrovně sítí pro technologické řízení označované jako sítě typu Fieldbus. Závěrem se budeme věnovat protokolu CAN, který byl navržen pro automobilovou techniku a který vychází z moderních přístupů k lokálním počítačovým komunikacím.

Modbus

Protokol Modbus je starý protokol, navržený firmou Modicon pro ovládání programovatelných automatů (PLC - Programmable Logic Controller) řídicím počítačem. (Programovatelné automaty můžeme považovat za náhradu réleových schémat, reagují na vstupní binární signály a podle zadané sekvenční funkce generují výstupní binární signály. Navíc jsou běžně vybavovány schopností generovat časové prodlevy.)

Síť Modbus se opírá o sběrnici RS-485, která propojuje skupinu programovatelných automatů a řídicí počítač. Povel a odpovědi mezi řídicím počítačem a programovatelnými automaty mohou být vyměňovány buď binárně jako posloupnosti osmibitových slabik (RTU mód) nebo jako texty složené ze sedmibitových znaků (ASCII mód). Ve druhém případě je každá slabika při přenosu reprezentována dvojicí hexadecimálních číslic, zpráva začíná středníkem a končí přechodem na začátek nové řádky. Výhodou je možnost sledování činnosti sítě běžným terminálem nebo emulátorem terminálu (s potřebným převodníkem elektrických úrovní). Vlastní přenos je asynchronní, typická přenosová rychlost je 9600 b/s. Ochrana proti chybám zajišťuje v ASCII módu podélná parita (LRC), v binárním módu je využíván šestnáctibitový cyklický kód CRC-16. Povel jsou obvykle potvrzovány odpovědí adresovaného řadiče.



Obrázek 4.6: Formát příkazu a odpovědi v síti Modbus

Většina příkazů a odpovědí protokolu Modbus má malou délku a jednoduchou strukturu (obr.4.6). Typicky obsahují adresu programovatelného automatu (i u odpovědi), číslo požadované funkce, její parametry a zabezpečovací kód. Zadávají programovatelnému automatu požadavek na sepnutí adresovaného kontaktu nebo nastavení hodnoty proměnné, přečtení hodnoty adresovaného vstupního binárního signálu nebo proměnné, ap.. Odpovědí je informace o provedení požadované akce, hodnota vstupního signálu nebo proměnné. Určitou výjimkou jsou delší zprávy, ty předávají programovatelnému automatu program, který potom automat autonomně vykonává (tedy např. sekvenční logickou funkci definující vztahy mezi vstupními

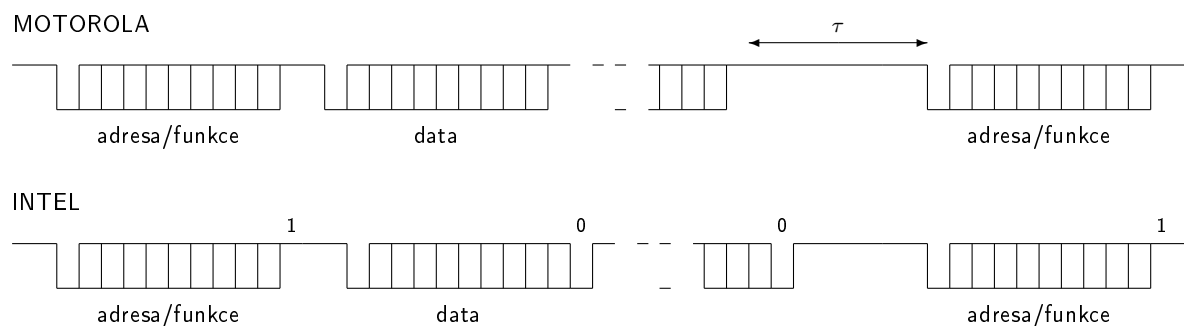
binárními signály a generovanými výstupy, případně doplněnou o generování časových prodlev a reakcí na ně).

Devítibitová multipočítačová sběrnice Intel

Výhodou asynchronní komunikace použité u sítě Modbus je dostupnost asynchronního sériového rozhraní na většině jednočipových mikropočítačů nebo poměrně přijatelná možnost realizace takového rozhraní programově. Jednoduchá asynchronní komunikace na sběrnici propojující řadu mikropočítačů má však i svá úskalí. Nejvýznamnějším je přídavná zátěž, kterou generuje komunikace na sběrnici ve všech připojených mikropočítačích, a to i těch, které se dané komunikace nezúčastní.

Omezení této přídavné zátěže vyžaduje doplnění komunikačního řadiče o schopnost ignorovat zprávy, které nejsou danému mikropočítači určeny. Musíme však dovolit příjem zpráv, které mu určeny jsou. Technická podpora takové funkce vyžaduje odlišit adresační informaci (obvykle krátkou) od vlastního textu zprávy (obvykle delšího), generovat přerušení (čekání na příjem znaku nebudeme asi uvažovat) pro příjem adresační informace a dát procesoru šanci vypnout přerušení pro text zprávy. V praxi se setkáme se dvěma řešeními tohoto problému.

Řešení, které používají mikropočítače Motorola, se opírá o možnost vypnout generování přerušení asynchronním rozhraním na žádost, kterou vydá procesor, jakmile rozpozná, že mu daná zpráva není určena (obr.4.7). Přerušení musí být opět povoleno po ukončení zprávy. To zajistí obvod rozhraní autonomně, když zjistí klid na sběrnici delší, než kolik odpovídá jednomu přenesenému znaku. Správná funkce tohoto schématu ovšem vyžaduje, aby vysílající stanice nevkládaly do vysílaných zpráv prodlevy (příjímač by zbytek zprávy nesprávně analyzoval jako začátek zprávy následující), a aby stanice před zahájením vysílání ponechávaly na médiu po dostatečně dlouhou dobu klidový stav.



Obrázek 4.7: Multiprocesorová komunikace Motorola a Intel

Alternativním řešením problému je doplnění dalšího, devátého, bitu ke každému osmibitovému znaku přenášenému po sběrnici (Intel 8051). Tento *devátý bit* odlišuje důležitou adresační informaci od vlastního textu zprávy. Asynchronní rozhraní lze uvést do stavu, kdy generuje přerušení pouze při příjmu znaku s jednotkovým devátým bitem. Pokud se mikropočítač na základě přijaté adresační informace rozhodne přijmout i zbytek zprávy, stačí, když převede asynchronní rozhraní do stavu, kdy generuje přerušení po každém přijatém znaku.

Devítibitová komunikace nemá nevýhodu citlivosti na časové prodlevy v přenášených zprávách (vysílající mikropočítač může mít na starosti mnohem důležitější činnost, než je bez prodlev vyslaná odpověď nadřízenému řídicímu systému), zaplatíme za to však jedním bitem navíc na každý přenášený znak. Typickým formátem znaku je start-bit, osm datových bitů, devátý řídicí bit a jeden stop-bit.

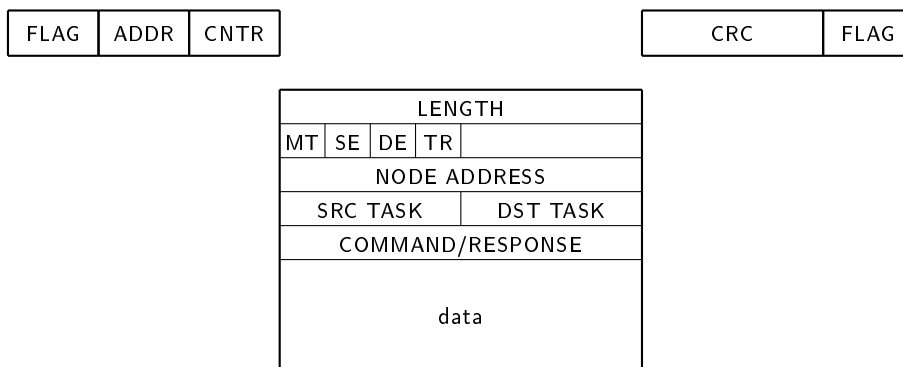
Bitbus

Asynchronní přenos lze snadno realizovat, jeho nevýhodou je *nízká efektivita* (na jeden znak musíme přenést start-bit, osm datových bitů, případný devátý bit a jeden až dva stop-bity). Dalším problémem je *transparence* přenášených dat, zajištění, že přijímač začínající přijímat v libovolném okamžiku správně rozpozná začátek a konec přenášené zprávy (zajištění transparence označením začátku každé zprávy (adresy) nastaveným devátým bitem je výhodou devítibitové komunikace).

Potřebu alespoň dvou synchronizačních bitů na jeden přenesený znak u asynchronního přenosu eliminuje přenos synchronní, u kterého jsou přenášeny pouze datové bity. Hlavním problémem synchronního přenosu je rozpoznání začátku a konce zprávy (a současně i zajištění transparence) a udržení bitové synchronizace mezi vysílačem a přijímačem bez ohledu na přenášená data. Bitová synchronizace vyžaduje dostatečnou frekvenci napěťových změn v přenášeném signálu (problém vzniká při přenosu dlouhých posloupností nul nebo jedniček). Znaková synchronizace (správné členění bitů na znaky a rozpoznání začátku zprávy) je obvykle zajištěna vyčleněním zvláštního znaku (znak SYN kódu ASCII), který označuje začátek zprávy. Transparenci lze zajistit vyčleněním znaku pro konec zprávy (např. znak EOT kódu ASCII) a prefixací tohoto znaku v přenášených datech jiným vyčleněným znakem (znak DLE kódu ASCII). Protokoly tohoto typu označujeme jako *znakově-synchronní*, vyžadují vybavení synchronního rozhraní obvodu pro bitovou a znakovou synchronizaci, a dnes se s nimi již prakticky nesetkáme.

Výhodnější vlastnosti než znakově synchronní protokoly mají protokoly *bitově-synchronní*, které technickými prostředky odliší začátek a konec zprávy od textu, a dovolí přijímači správně se synchronizovat i při přenosu dlouhých posloupností nul (dlouhé posloupnosti jedniček lze eliminovat vhodným kódováním, např. kódem NRZI). Typickým představitelem takových protokolů (vycházejí ze společného základu, kterým je protokol HDLC - High Level Data Link Communication) v oblasti mikropočítačové komunikace je protokol *Bitbus* firmy Intel.

Přenosovým médiem sítě Bitbus je dvoudrátové vedení odpovídající RS-485. Data jsou přenášena rychlostí 375 kb/s (segment sběrnice pak může mít délku 330 m), nebo 62.5 kb/s (segment sběrnice pak může být dlouhý až 1300 m) v kódu NRZI. Přenášené rámce zahrnují adresu podřízené stanice (v příkaze i v odpovědi), řídicí informaci (číslo přenášeného rámce a číslo potvrzení), vlastní předávaná data a zabezpečení cyklickým kódem. Struktura přenášené zprávy zahrnuje identifikaci odesílajícího a adresovaného procesu, protokol předpokládá komunikaci mezi mikropočítači s paralelním zpracováním procesů nad jádrem reálného času.



Obrázek 4.8: Formát rámce a zprávy v síti Bitbus

Výhodou komunikačního protokolu Bitbus je dokonalé zabezpečení přenosu a dobrá efektivita (okénkové potvrzovací schéma). Nepříjemnou nevýhodou je fakt, že není podporován

komunikačními řadiči jednočipových mikropočítačů. Výjimkou byl mikropočítač Intel 8044, což byla kombinace jednočipového mikropočítače řady 8051 a obvodu bitově-synchronního sériové rozhraní pro Bitbus nebo počítač National Semiconductor HP16083, který je vybaven obvodem rozhraní HDLC (určeným pro podporu protokolu LAPD zařízení ISDN). Pokud chceme tento protokol použít u jiného mikropočítače, nezbyvá, než připojit vnější komunikační řadič, například běžný dvoukanálový Zilog 85C30.

Protokol Bitbus není (přes jeho nedostatečnou podporu u běžných jednočipových mikropočítačů) pouhým firemním standardem. Na použití protokolu Bitbus je založena linková vrstva standardu sběrnice pro propojení lékařských přístrojů lůžka pacienta MIB (Medical Information Bus) - P1073. Data jsou kódována do signálu NRZI a přenášena rychlostí 375 kb/s.

Fieldbus

Vedle "de facto"standardů v oblasti komunikačních protokolů pro řídicí systémy na sběrnici RS-485 vzniklo několik doporučení pro řízení přenosu a řízení přístupu. Tyto standardy jsou označovány jako Fieldbus. Patří sem mezinárodní standard IEC Fieldbus, francouzský standard FIP (Factory Instrumentation Protocol), německý standard PROFIBUS (PROcess FIEld BUS) a mezinárodní standard ISA SP50.

Tato doporučení definují fyzické vlastnosti přenosu, protokol linkové vrstvy a řízení přístupu k médiu, a využití sběrnice v aplikaci. Na fyzické úrovni jsou definovány přenosové rychlosti, počet zařízení na sběrnici, oddělení zařízení od sběrnice, kódování dat a redundance. Na aplikační úrovni jsou definovány typy a formáty zpráv a časové (doba odezvy) a výkonové limity (počet zpráv za sekundu).

Přenosové rychlosti se pohybují u těchto norem v rozsahu od 31.25 kb/s do 2.5 Mb/s. Na sběrnici může být připojeno nejvýše 32 zařízení, pouze FIP dovoluje až 256 zařízení. Zařízení jsou připojována ke (stíněné) sběrnici RS-485 izolovaně, alternativním médiem je mimo PROFIBUSu optická hvězda, u IEC Fieldbus a u ISA SP50 i radiový kanál. Pro přenos jsou data obvykle kódována diferenciatlním kódem Manchester (nule odpovídá orientace hrany shodná s hranou předchozí, jednička orientace opačná). Normy IEC Fieldbus, PROFIBUS a ISA SP50 předpokládají redundantní zdvojení sběrnice.

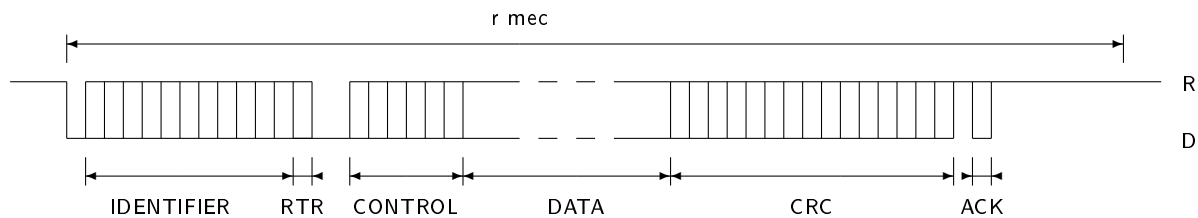
Přístup k médiu bývá volitelně centrální (Polling) nebo distribuovaný (Token Passing), zabezpečení přenosu je potvrzovacím schématem, PROFIBUS se opírá o podélnou paritu, ISA SP50 o cyklický kód CRC-16 nebo CRC-32. Ve formátech aplikačních zpráv se doporučení liší, doby odezvy jsou v rozmezí 25 až 100 ms.

CAN

Na závěr přehledu komunikačních protokolů (a předcházejícího přehledu elektrických standardů) si uvedeme komunikační sběrnici navrženou pro komunikaci mezi zařízeními v automobilech navrženou firmou Bosch. Komunikační protokol využívá poněkud modifikované sběrnice RS-485 (řešení kolize více vysílačů ve prospěch dominantního).

Řízení sběrnice CAN se opírá o přenos dvou úrovní signálu, jedna je označována jako dominantní (D), druhá jako recesivní (R). Současné přivedení více signálů na sběrnici má jako výsledek recesivní (R) úroveň pouze tehdy, mají-li všechny přivedené signály recesivní úroveň. Jediný signál s dominantní (D) úrovní dává dominantní (D) úroveň na sběrnici. Jedná se o princip, který známe u logických obvodů jako funkci *Wired-OR* (dominantní úrovní odpovídá nula, recesivní úrovní jednička).

Data jsou předávána v krátkých rámcích, krátké zprávy jsou konečně typické pro komunikaci u většiny řídicích systémů, délka předávané zprávy může být nejvýše osm znaků. Struktura rámce je značně odlišná od běžných komunikačních protokolů, uvádí ji obr.4.9.



Obrázek 4.9: Formát rámce síti CAN

Dominantní bit Start-of-Frame označuje začátek předávané zprávy. Je následován jedenáctibitovou identifikací zprávy a jednobitovým polem RTR (Remote Transmission Request), které odlišuje zprávu s daty (bit RTR má dominantní úroveň) od požadavku na vyslání dat jinou stanicí (bit RTR má recesivní úroveň). Celá posloupnost dvanácti bitů slouží pro arbitráž na sběrnici, zprávy (a požadavky na jejich přenos) s vyšší hodnotou identifikace (jednička kódována jako D, nula jako R) mají přednost, při shodné identifikaci má zpráva přednost před požadavkem. Bit RTR má funkci odpovídající bitu R/\overline{W} u sběrnice I²C, recesivní úroveň bitu RTR "předává sběrnici" adresované protistanici.

Šestibitové pole Control informuje o délce následujícího bloku dat (v bitech, pole dat může mít délku 1 až 64 bitů), dva bity jsou vyhrazeny pro budoucí rozšiřování funkcí. Data jsou přenášena počínaje nejvýznamnějším bitem. Přenos je zabezpečen patnáctibitovým BCH kódem. Ten dovoluje zajistit blok dat do délky 112 bitů (využito je pouze 84 bitů), má Hammingovu vzdálenost šest bitů a dovoluje detekovat až pět samostatných chyb v rámci a všechny chyby v lichém počtu bitů. Zbytková chybovost přenosu je $3 \cdot 10^{-5}$, vzhledem ke kontrole struktury rámce je však ještě nižší. Vysílání rámce končí recesivním bitem CRC Delimiter.

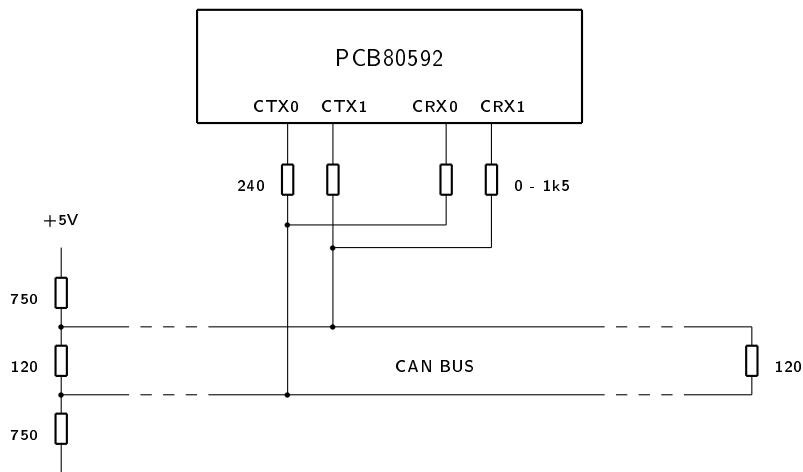
Příjem rámce musí potvrdit alespoň jeden z příjemců dominantní úrovní bitu ACK. Následujícím recesivním bitem ACK Delimiter a sedmibitovým recesivním polem End-of-Frame přenos rámce končí.

Pro zajištění bitové synchronizace přijímače (přijímačů) s vysílačem je za každou posloupnost pěti bitů se shodnou úrovní vkládán bit s úrovní opačnou (modifikace bit-stuffingu). Posloupnost šesti dominantních a šesti recesivních bitů dovoluje přijímači indikovat chybový stav a testovat akceptování tohoto stavu vysílačem.

Pro rozhraní CAN existují specializované obvody a najdeme je i u řady mikropočítačů integrované přímo na čipu. Jako příklady řadičů si můžeme uvést obvody Intel 82526 a

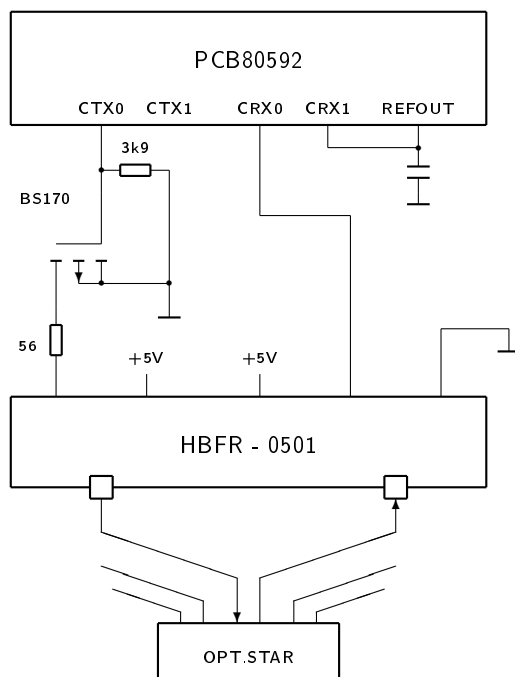
Philips PCA82C200. Jako příklady mikropočítačů vybavených rozhraním CAN si můžeme uvést Philips PCB80C592 (modifikace PCB80C552 s rozhraním CAN namísto I²C) a Motorola MC68HC05X16.

Nejjednodušším způsobem lze připojit řadič CAN nebo mikropočítač vybavený tímto rozhraním na sběrnici RS-485 podle obr.4.10, v automobilových aplikacích je požadováno oddělení výstupů vysílače vnějšími tranzistory a přídavná ochrana vstupů.



Obrázek 4.10: Sběrnice CAN

Dalším možným přenosovým prostředím je optická hvězda, jako převodníky elektrického signálu na světlo lze použít např. transceivery typu HBFR-05011.



Obrázek 4.11: Optická hvězda CAN

4.3 Sériové sběrnice

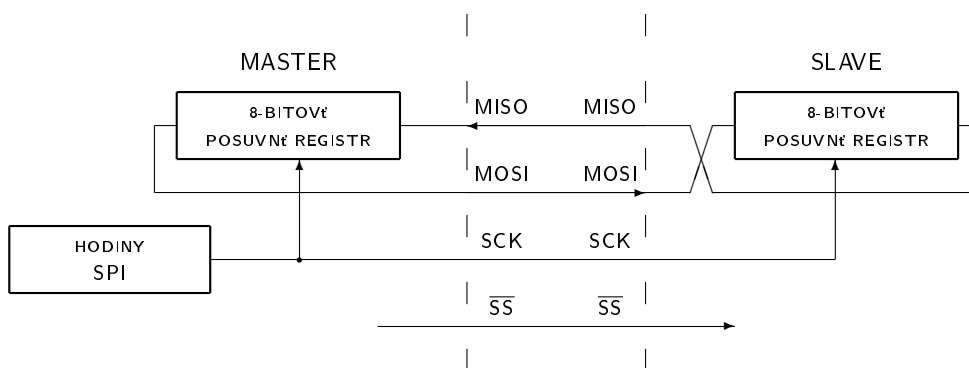
Jako sériové sběrnice obvykle označujeme propojovací systémy, které spojují mikropočítač s pomocnými obvody v rámci jednoho zařízení. Sériová sběrnice šetří počet vývodů procesoru a periferních obvodů, zjednodušuje konstrukci. Je typicky tvořena dvojicí signálových vodičů. Jeden přenáší hodinový signál, hrany hodinového signálu definují časové okamžiky, ve kterých jsou na druhém vodiči presentovány jednotlivé bity přenášených dat.

Se sériovými sběrnici se setkáváme u procesorů, které (třeba s ohledem na počet vývodů, např. Philips 83C752) nemají vyvedenu sběrnici paralelní, nebo u aplikací, ve kterých by použití paralelní sběrnice vedlo na nepřijatelné zvýšení ceny obvodů a složitosti, a tím i ceny spoje. Pro některé obvody, jako jsou malé paměti EEPROM (např. NM93C46) je použití sériové sběrnice typické. Sériová sběrnice je pochopitelně pomalejší než sběrnice paralelní.

Synchronní přenos po sériové sběrnici je z obvodového hlediska jednodušší než přenos asynchronní. Navíc lze vhodným ovládním hodinového signálu zajistit vzájemnou synchronizaci vysílače a přijímače (pozdržení vysílače na dobu potřebnou pro převzetí nebo zpracování přijatého znaku) nebo arbitráž v konfiguracích s více řadiči.

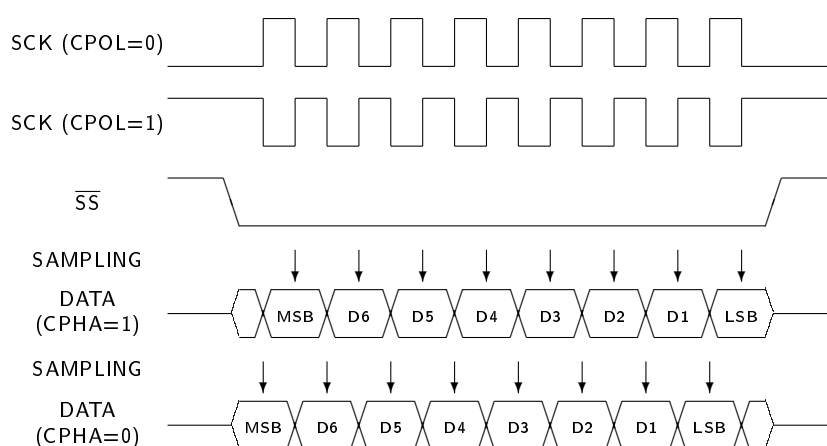
Motorola SPI

Jednoduchou sériovou sběrnici, označovanou jako SPI (Serial Peripheral Interface) najdeme u mikropočítačů firmy Motorola (řady MC68HC05, MC68HC11, MC68HC16). Sběrnice je typicky používána pro připojení periferních obvodů (nebo podřízených mikropočítačů) k jednočipovému mikropočítači na desce spoje. Je tvořena trojicí signálů. Hodinový signál SCK (Serial Clock) je generován řadičem (mikropočítačem), signály MOSI (Master Out/Slave In) a MISO (Master In/Slave Out) propojují posuvné registry řadiče a podřízeného obvodu do kruhu (obr.4.12). Podřízený obvod musí být aktivován signálem \overline{SS} (Slave Select); vývod \overline{SS} dovoluje připojit k jednomu řadiči více podřízených obvodů (za cenu více adresacích signálů a případně i vnější adresacní logiky).



Obrázek 4.12: Sériová sběrnice SPI

Vlastní předání dat je velmi jednoduché, operace nad SPI zamění údaje v datových registrech řadiče a podřízeného obvodu. Přesun je řízen hodinami mikropočítače a je velmi rychlý, např. mikropočítač MC68HC11 jako řadič dovoluje výběr z frekvencí 62.5, 125 a 500 kHz, nebo 1 MHz. Aby bylo možné připojit i obvody, které neodpovídají jednoznačné volbě polaritě hodinového a datového signálu, lze polaritu signálu SCK a polaritu datových signálů MOSI a MISO na straně řadiče nastavit (obr.4.13).



Obrázek 4.13: Signály sériové sběrnice SPI

Sběrnice SPI zůstala omezena na produkty vyvinuté firmou Motorola. Pro informaci uvedeme několik periferních obvodů ovládaných po sběrnici SPI:

- MC14489 - budič segmentového zobrazovače LED,
- MC14499 - dekódér/budič sedmisegmentového zobrazovače LED,
- MC145453 - budič nemultiplexovaného zobrazovače LCD,
- MC145000/1 - budič multiplexovaného zobrazovače LCD,
- MC68HC68T1 - hodiny reálného času a paměť RAM,
- MCM2814 - paměť EEPROM 256*8,
- MC145040/1 - jedenáctivstupový osmibitový A/D převodník,
- MC145050/1 - jedenáctivstupový desetibitový A/D převodník,
- MC144110/1 - šesti/čtyřnásobný šestibitový D/A převodník,
- MC14515x - obvody pro kmitočtovou syntézu PLL,

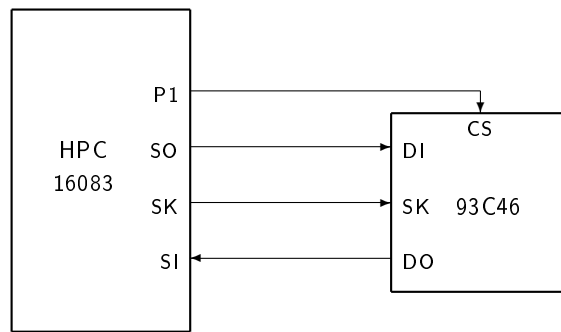
Kromě uvedených obvodů lze na sběrnici SPI připojit i univerzální obvody jako:

- 74HC595 - osmibitový posuvný registr (Serial-In/Parallel-Out),
- 74HC589 - osmibitový posuvný registr (Parallel-In/Serial-Out)

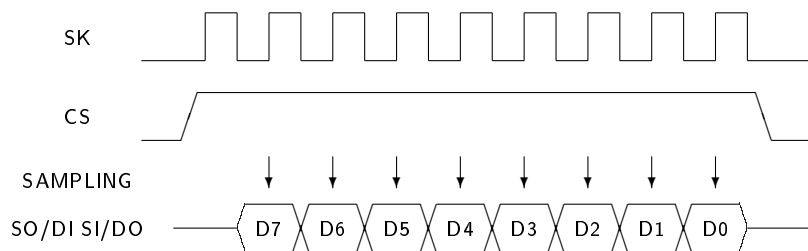
MicroWire

Poněkud univerzálnější použití než sběrnice SPI má sběrnice MicroWire firmy National Semiconductor. Sběrnice dovoluje připojit skupinu periferních obvodů k nadřízenému řadiči (mikropočítači). Sběrnice je tvořena trojicí vodičů SK, SO/DI a SI/DO. Hodinový signál SK řídí přenos po dvou datových vodičích. První propojuje výstup řadiče SO (Serial Out) se vstupem DI (Data In) periferních obvodů, druhý připojuje výstupy DO (Data Out) periferních obvodů na vstup SI (Serial In) řadiče. Typické připojení paměti EEPROM typu NM93C46 k šestnáctibitovému mikropočítači National Semiconductor HPC 16083 uvádí obr.4.14.

K jednomu řadiči lze připojit skupinu periferních obvodů. Výběr periferního obvodu, se kterým řadič komunikuje, vyžaduje použití dalších - výběrových vodičů CS (a případně i dekódéru adresy). Polarita hodinového signálu a časování datových signálů je na rozdíl od SPI pevně definováno, úrovně na datových vodičích SO/DI a SI/DO se mění se sestupnou hranou hodin SK, signály jsou čteny s náběžnou hranou hodin SK (obr.4.15).



Obrázek 4.14: Připojení paměti EEPROM sériovou sběrnicí Microware



Obrázek 4.15: Signály sériové sběrnice Microware

Sběrnice MicroWire se podobá sběrnici SPI, není však pevně definována délka předávaného slova (obvykle se jedná o celočíselný násobek délky posuvného registru, ten je např. u procesorů National Semiconductor COP800 dlouhý čtyři bity). Data jsou předávána jako pouhé posloupnosti bitů pouze u nejjednodušších obvodů (např. řadič zobrazovače COP470), většina periferních obvodů předpokládá zprávu s určitou strukturou, je definován primitivní protokol. Po počáteční nule (nulách) následuje synchronizační jednička, identifikátor požadované funkce a/nebo adresa a konečně předávaná data. Každé pole má délku specifickou pro daný periferní obvod.

Jako příklad „protokolu“ si uvedeme ovládání paměti EEPROM NM94C56. Tento protokol používá příkazy (bitové posloupnosti):

		Start	Kód	Adresa	Data
READ	čtení slova	1	10	A5-A0	D15-D0
WRITE	zápis slova	1	01	A5-A0	D15-D0
WRAL	zápis do všech slov	1	00	01xxxx	D15-D0
WEN	povolení zápisu	1	00	11xxxx	
WDS	zákaz zápisu	1	00	00xxxx	
ERASE	smazání slova	1	11	A5-A0	
ERAL	smazání celé paměti	1	00	10xxxx	

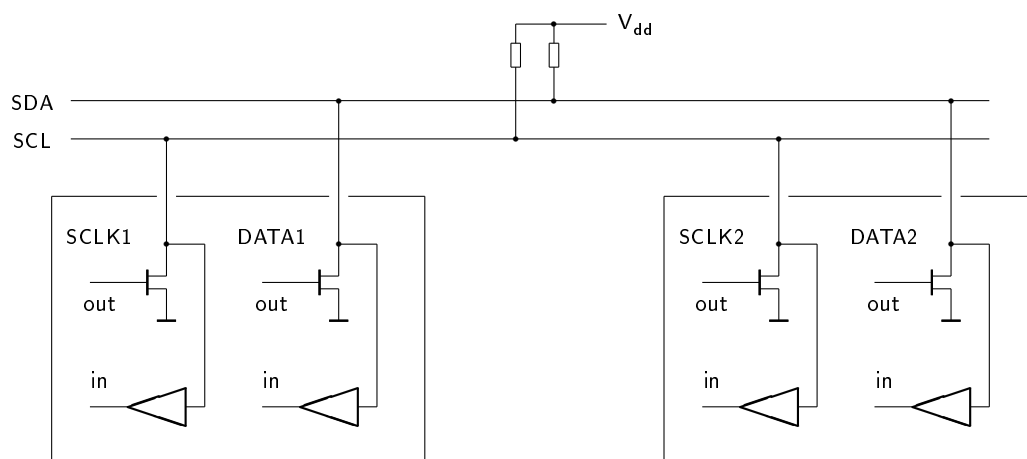
Pozn.: U operace čtení po zadání adresy řadičem vysílá data periferní obvod.

Sběrnici MicroWire najdeme u mikropočítačů National Semiconductor (COP800/820/880 a u řady HPC). Řada periferních obvodů, jejichž ovládání odpovídá sběrnici MicroWire, zahrnuje obvody, které jsou často využívány i v aplikacích mikropočítačů jiných firem, kde „komunikační protokol“ realizujeme programově:

ADC0831 - jednobitový osmibitový A/D převodník
 ADC0834 - čtyřkanálový osmibitový A/D převodník
 ADC0852 - D/A převodník s komparátorem a multiplexerem
 ADC0854 - D/A převodník s komparátorem a multiplexerem
 MM5483 - řadič LCD, 31 segmentů
 NM93C46 - paměť EEPROM 64*16 / 128*8
 NM93C56 - paměť EEPROM 128*16 / 256*8
 NM93C66 - paměť EEPROM 256*16 / 512*8
 74HC595 - osmibitový posuvný registr (Serial-In/Parallel-Out),
 74HC589 - osmibitový posuvný registr (Parallel-In/Serial-Out)

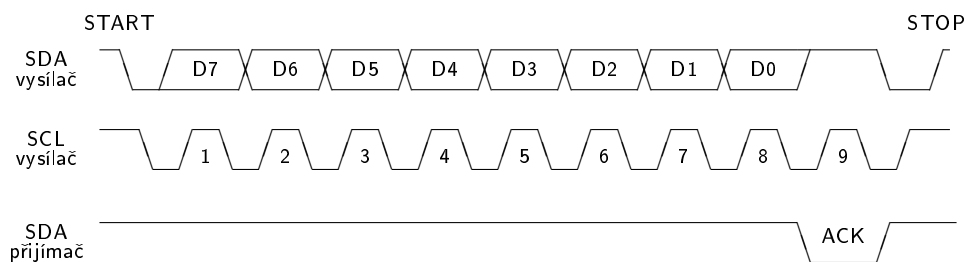
I²C

Z hlediska univerzálnosti použití i z hlediska standardizace má největší praktický význam sběrnice I²C (Inter-Integrated-Circuit Bus) vytvořená firmou Philips. Podporu sběrnice I²C nalezneme u procesorů této firmy (80C528, 80C552, 83C752) ale i u procesorů jiných výrobců (Motorola MC68HC05E0, MC68HC05T7).



Obrázek 4.16: Struktura sběrnice I²C

Sběrnice je tvořena dvojicí vodičů, na kterých je v klidovém stavu zvyšovací odpory udržováno kladné napětí (obr.4.16). Jeden z vodičů, značený SCL, přenáší hodinový signál, druhý, značený SDA, slouží synchronnímu přenosu dat (obr.4.17). Budiče s otevřeným kolektorem umožňují, aby sběrnice byla využívána skupinou rovnoprávných řadičů (multi-master konfigurace); odpovídající arbitrážní protokol, který řeší přidělení sběrnice při současném požadavku více řadičů si popíšeme později.



Obrázek 4.17: Signály sběrnice I²C

Komunikaci zahajuje řadič prvkem START (sestupná hrana signálu SDA při úrovni HIGH signálu SCL). Osmibitový znak je vyslán po vodiči SDA počínajíc nejvýznamnějším bitem (na rozdíl od běžné sériové komunikace), úroveň na vodiči SDA musí zůstat stabilní po dobu trvání úrovně HIGH signálu SCL. Vyslaný znak je potvrzován přijímačem stažením signálu SDA na úroveň LOW v době, která odpovídá devátému bitu (prvek ACK).

Přijímač může pozastavit vysílač podržením signálu SCL na úrovni LOW (prvek HOLD). To se týká každé periody signálu SCL a pozastavení může vyvolat libovolný prvek sběrnice. Sestupná hrana signálu SCL navíc synchronizuje generátory signálu SCL všech připojených prvků.

Norma I²C definuje, na rozdíl od MicroWire, i formát přenášených dat, jejich potvrzování a předávání řízení mezi účastníky komunikace. Po vyslání prvního znaku, jehož významnějších sedm bitů adresuje podřízený obvod, může pokračovat ve vysílání dalších znaků jak řadič tak podřízený obvod. Směr přenosu dalších znaků je určen osmým bitem, úroveň LOW indikuje vysílání řadiče (funkce WRITE), úroveň HIGH indikuje vysílání podřízeného obvodu (funkce READ). Komunikaci ukončuje řadič prvkem STOP (náběžná hrana signálu SDA při úrovni HIGH signálu SCL).

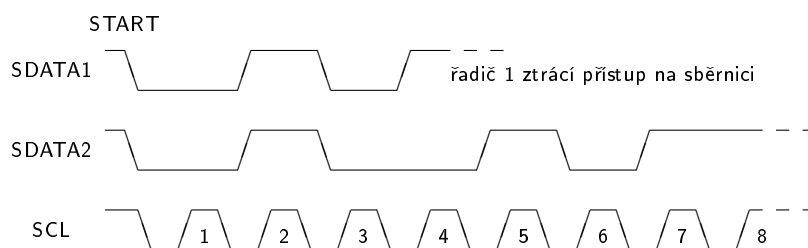
V předchozím odstavci jsme si uvedli, že první znak adresuje podřízený periferní obvod. Přiřazení adres je přitom určeno jednotlivým typům periferních obvodů výrobcem, každý typ obvodu má část adresy pevně definovanou, zbylé bity lze nastavit zapojením adresacích vývodů obvodu. Podobně jako u sběrnic SPI a MicroWire si uvedeme několik obvodů užitečných pro univerzální aplikace spolu s jejich adresami (písmeno a v tabulce adres odpovídá hodnotě nastavené zapojením příslušného adresacního vývodu obvodu, bit A0 zadává směr přenosu):

		A7	A6	A5	A4	A3	A2	A1
PCF8574	- osmibitový vstup/výstup	0	1	0	0	a	a	a
PCF8577	- 64-segmentový řadič LCD	0	1	1	1	0	1	0
PCF8578	- řadič bodového LCD	0	1	1	1	1	0	a
PCF8582A	- paměť EEPROM 256*8	1	0	1	0	a	a	a
PCF8583	- hodiny/kalendář a paměť RAM	1	0	1	0	0	0	a
PCF8591	- osmibitový A/D, D/A převodník	1	0	0	1	a	a	a
SAA3028	- přijímač kódu RC5 (infra)	0	1	0	0	1	1	0
TDA8444	- osminásobný D/A převodník	0	1	0	0	a	a	a

Pro aplikace z oblasti spotřební elektroniky (audio, video) a telekomunikací mají firmy Philips a Siemens v řadách obvodů PCF, SAA, TDA, TEA a TSA mnoho prvků řízených sběrnici I²C.

Samotná jednoduchá adresace prvním znakem neposkytuje dostatečnou pružnost. Adresacní možnosti sběrnice I²C jsou dále rozšířeny možností použít dvouznakové adresy. Adresa začínající znakem 00_H adresuje všechny obvody na sběrnici (General Call Address — broadcast adresa), její druhý znak definuje funkci, která je vyžadována na straně přijímačů (reset, programování), nebo identifikuje vysílač. Prefix 01_H slouží jako prefix běžného provozu a dovoluje zefektivnit práci prvků, které realizují obsluhu I²C programově. Prefix 02_H dovoluje připojit přijímače odpovídající staršímu standardu CBus, prefixy 03_H - 0F_H dovolují další rozšiřování funkcí. Konečně, adresy s prvním znakem F8_H - FF_H podporují desetibitovou adresaci (poslední tři bity znaku jsou nejvyššími řády adresy, zbývajících sedm bitů adresy je ve druhém znaku).

Důležitou součástí protokolu I²C je arbitráž v konfiguracích s více řadiči (obr.4.18). Přístup libovolného řadiče na sběrnici, která je v klidovém stavu (jednotková úroveň signálů SCL i SDA po dobu alespoň t_{HIGH}) je zahajován sestupnou hranou signálu SDA. Pokud je prvek START generován současně více řadiči, je nutné vybrat jeden, který smí sběrnici použít. Nutnou

Obrázek 4.18: Arbitráž na sběrnici I²C

podmínkou arbitrážní procedury je synchronizace hodinových signálů generovaných řadiči, které soupeří o sběrnici. Je zajištěna stejným způsobem jako synchronizace řadiče s podřízeným obvodem: řadič, který ukončil fázi LOW hodinového signálu SCL musí počkat na přechod signálu SCL do úrovně HIGH. Vlastní arbitráž se opírá (podobně jako synchronizace hodinového signálu) o funkci sběrnice s otevřenými kolektory. Řadič, který v době fáze HIGH hodinového signálu čte na datovém vodiči úroveň, kterou sám generuje, smí pokračovat v komunikaci. Řadič, který generuje úroveň HIGH ale čte úroveň LOW, musí přenos ukončit a pokusit se o přenos později. Jedná se tedy systém s prioritním přístupem, zprávy s nižší adresou (nebo identifikací funkce) jsou předávány přednostně.

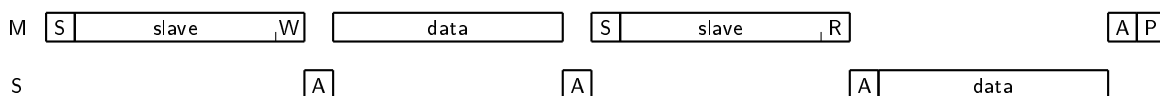
VYSKLŽN



PžJEM



KOMBINACE (VYSKLŽN + PžJEM)

Obrázek 4.19: Předávání dat po sběrnici I²C

Vlastní přenos dat mezi řadičem, který získal přístup ke sběrnici, a podřízenou stanicí je definován jednoduchým protokolem (obr.4.19). Ten, vedle potvrzování devátým bitem, dovoluje předat sběrnici podřízené stanici pro přenos dat. Řadič, který jednou sběrnici získá, si ji může, po zopakování prvku START a pozitivním výsledku arbitráže, podržet i pro další přenos.

Přenosová rychlost je určena hodinovým signálem, může být u standardních obvodů až 100 kb/s (Standard Mode), novější obvody dovolují jít až na 400 kb/s (Fast Mode). Spodní hranice není omezena, obvody I²C lze ovládat i programem na paralelním I/O rozhraní (podobně jako obvody SPI nebo MicroWire).

Elektrické parametry rozhraní I²C dovolují oddělit vývody obvodů od vodičů sběrnice odpory 300 Ω a chránit je tak před napěťovými špičkami.

<i>Parametr</i>		<i>Standard</i>		<i>Fast</i>		
		min	max	min	max	
Úroveň vstupu LOW	V_{IL}	-0.5	1.5	-0.5	1.5	V
Úroveň vstupu HIGH	V_{IH}	3.0		3.0		V
Úroveň výstupu LOW	V_{OL}		0.4		0.4	V
Proud vstupu	I_i	-10	10	-10	10	μA
Kapacita vstupu	C_i		10		10	pF
Kapacita sběrnice	C_b		400		100	pF
Kmitočet SCL	f_{SCL}	0	100	0	400	kHz
Úroveň LOW SCL	t_{LOW}	4.7		1.3		μs
Úroveň HIGH SCL	t_{HIGH}	4.0		0.6		μs
Čas náběhu	t_R		1000		300	ns
Čas doběhu	t_F		300		300	ns

Tabulka 4.1: Elektrické parametry rozhraní I²C

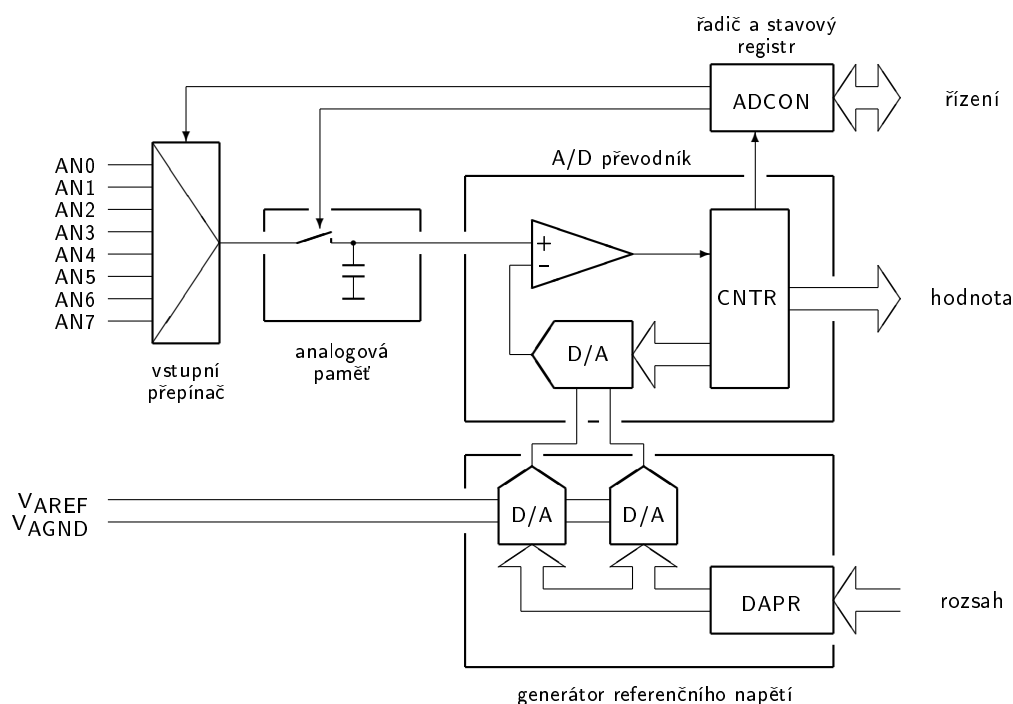
5. Analogové vstupy a výstupy

Jednočipové mikroočítače a řadiče jsou určeny pro aplikace, které zpracovávají a produkují analogové signály. Je tedy přirozené, že jsou buď přímo vybavovány analogově-číslicovými a číslicově-analogovými převodníky, nebo že tyto převodníky najdeme mezi jejich podpůrnými obvody.

V této kapitole si budeme všimnout vybavení a prvků, které se používá v oblasti měřící a řídicí techniky, stranou necháme prvky, které souvisí s telekomunikacemi (DTFM generátory a přijímače, PCM převodníky), domácí elektronikou (číslicově řízené potenciometry, filtry) a hlasovou komunikací (řečové procesory).

5.1 Aproximační A/D převodníky

Moderní jednočipové mikroočítače a řadiče jsou běžně vybavovány vícevstupovými aproximačními A/D převodníky. Jejich typickou vnitřní strukturu si představíme na příkladě převodníku mikroočítače Siemens SAB80C515/535 na obr.5.1.



Obrázek 5.1: Struktura A/D převodníku mikroočítače SAB80C515/535

Běžný vícevstupový A/D převodník se skládá z analogového přepínače vstupů (obvykle osm až dvanáct vstupů), analogové paměti (obvod Sample-and-Hold, označovaný zkratkou S/H) a vlastního A/D převodníku pracujícího zpravidla na principu *postupné aproximace*. Metoda postupné aproximace se opírá o rekurzivní půlení intervalu referenčního napětí a jeho porovnávání s napětím vstupním, algoritmus postupné aproximace je zpravidla realizován řadičem vlastního A/D převodníku. Aproximační převodníky mívají rozlišení osm, deset až dvanáct bitů a jsou obvykle schopné zpracovat vstupní signál v intervalu referenčního napětí s přesností odpovídající polovině až jednotce nejnižšího převáděného řádu. Převodníky A/D moderních mikroočítačů jsou schopné převést vstupní napětí na osmibitový až dvanáctibitový

údaj do $20 \mu\text{s}$, u starších A/D převodníků (např. často používané vnější převodníky National Semiconductor ADC0804, ADC0808 a ADC0809) musíme počítat s dobou převodu kolem $100 \mu\text{s}$.

Převodník mikropočítače SAB80C515/535 (ale i SAB80C517/537), který jsme si uvedli jako příklad (obr.5.1), je navíc ještě vybaven programovatelným děličem referenčního napětí (dvojice čtyřbitových D/A převodníků). Tento dělič dovoluje realizovat převod ve zvoleném podintervalu referenčního napětí a zvýšit tak v tomto intervalu rozlišení.

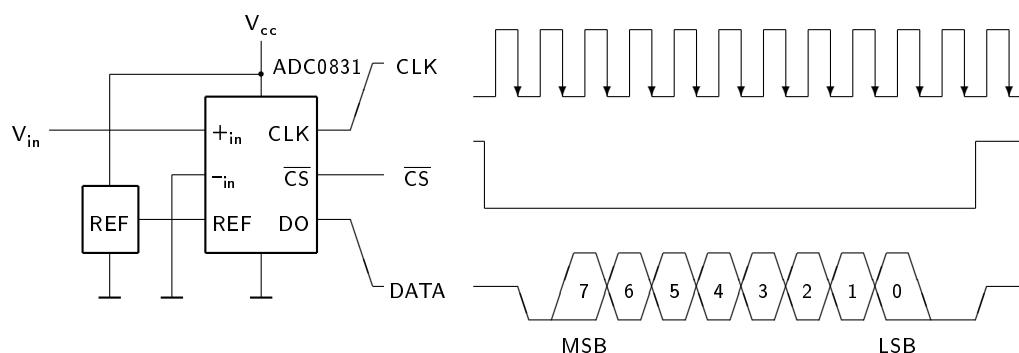
Pro výběr vstupního kanálu a start převodu slouží bity v řídicím registru, informaci o ukončení převodu obvykle získáme ve stavovém registru a/nebo jako žádost o přerušení. Převedenou hodnotu si můžeme přečíst v datovém registru.

U některých vnitřních A/D převodníků analogovou paměť nenajdeme (příkladem mohou být mikropočítače Philips řady 83C552 s osmivstupovým desetibitovým převodníkem). V takovém případě pak musíme počítat s možnými chybami při převodu měnících se signálů a případně je v dalším zpracování naměřených dat vyloučit.

Jinde je vnitřní řadič rozšířen o možnost spustit posloupnost převodů na jednom nebo více vstupních kanálech. Příkladem může být vnitřní A/D převodník mikropočítače Motorola MC68HC11, který je vybaven čtyřmi datovými registry. Lze do nich nahrát hodnoty čtyř bezprostředně následujících měření na jediném kanále, nebo bezprostředně následující měření čtyř kanálů. Je dokonce možné spustit kontinuální měření, v tomto režimu najdeme v datových registrech hodnoty čtyř posledních měření (na jednom nebo čtyřech kanálech).

Zcela výjimečně se můžeme setkat s programovou realizací algoritmu postupné aproximace. Takové řešení se opírá o D/A převodník produkující napětí, které komparátorem porovnáváme s napětím měřeným. Generování srovnávacího napětí D/A převodníkem řídí program. Vedle postupné aproximace lze snadno realizovat i jinou aproximační metodu, například sledování (změna srovnávacího napětí o jednotkové kroky). S programovou obsluhou se můžeme setkat nejčastěji ve starších systémech, důvodem byly rozdíly v dostupnosti a v ceně D/A a A/D převodníků. Obvody kombinující D/A převodník a komparátor s prepínačem více vstupních kanálů však najdeme i mezi prvky pro sériovou sběrnici MicroWire (National Semiconductor ADC0852 a ADC0854), zde je hlavním důvodem snaha poskytnout obě funkce, A/D i D/A převodník, v jediném pouzdře.

O něco výše jsme se zmínili o vnějších A/D převodnících komunikujících s mikropočítačem po sběrnici MicroWire. Protože se jedná o elegantní řešení, uvedeme si zde základní zapojení s jednokanálovým převodníkem National Semiconductor ADC0831 nebo jeho obdobou Texas Instruments TLC549 (obr.5.2).



Obrázek 5.2: Připojení A/D převodníku ADC0831 k mikropočítači

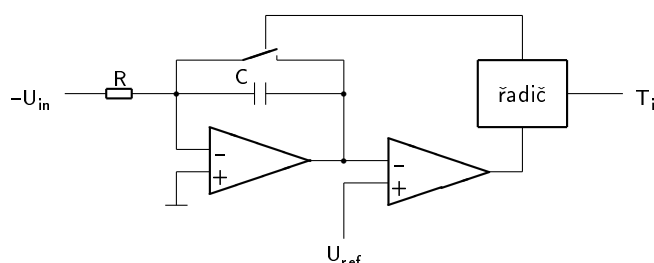
Data jsou na datovém vodiči D0 zveřejňována od nejvyššího bitu, tak jak je poskytuje algoritmus postupné aproximace. U vícekanálových převodníků ADC0832, ADC0834 a ADC0838

je nutné vstupním kanálem DI nejdříve zadat převodníku parametry měření: měření proti zemi (asymetrický vstup) nebo rozdílové měření (symetrický vstup), výsledek se znaménkem nebo bez znaménka a číslo kanálu.

5.2 Integrační A/D převodníky

Pokud nám nezáleží na rychlosti převodu a pokud potřebujeme dosáhnout vyšší přesnosti měření a vyloučit vliv rušivých signálů (číslicového „šumu“ na spoji, všudypřítomného rušivého signálu sítě 50 Hz) obracíme se obvykle k integračním převodníkům.

Princip integračního převodníku je velmi jednoduchý: proud tekoucí do vstupu operačního zesilovače kompensujeme proudem, který nabíjí kondensátor zapojený ve zpětné vazbě (obr.5.3).



Obrázek 5.3: Integrační převodník

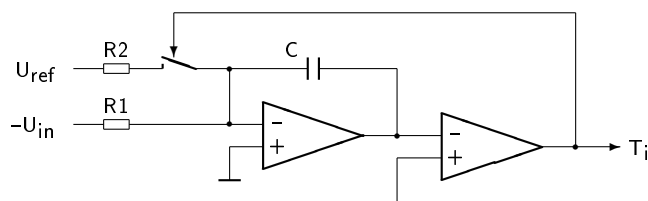
Primitivní metoda, kdy opakovaně nabíjíme vybitý kondenzátor (nulové napětí na výstupu integrátoru) na napětí referenční (srovnání zajistí komparátor na výstupu integrátoru) a z potřebného času určujeme měřené napětí, závisí na znalosti přesné hodnoty odporu R a kondensátoru C . Navíc, ve hře je i kvalita kondenzátoru, lépe řečeno jeho dielektrika, které musí mít zanedbatelnou hysterezi (běžné dielektrikum si při krátkodobém vybití kondenzátoru zkrátka „pamatuje“ předchozí polarizaci a na kondenzátoru se po rozpojení zkratu objeví zbytkové napětí), a vlastnosti operačního zesilovače.

Chceme-li vyloučit vliv tolerance a změn hodnot součástek, můžeme srovnávat dobu potřebnou k nabití kondenzátoru měřeným napětím s dobou potřebnou k nabití kondenzátoru napětím referenčním. Při použití kvalitních prvků lze i tímto jednoduchým způsobem dosáhnout přesnosti potřebné pro dvanáctibitový převod.

Zjednodušení převodníku lze dosáhnout úpravou zapojení, při které komparátor srovnává napětí na výstupu komparátoru s nulou. Kondenzátor, který jsme po pevnou dobu nabíjeli měřeným napětím, vybijeme na nulu referenčním napětím (s opačnou polaritou) a cyklus opakujeme. Poměr mezi dobou nabíjení a dobou vybití odpovídá poměru mezi měřeným a referenčním napětím. Volíme-li dobu nabíjení (té pochopitelně odpovídají hodnoty součástek) s ohledem na převažující rušivý signál (v Evropě 50 Hz, v Americe 60 Hz) jako násobek jeho periody (20 ms, resp. 16.66 ms) dosáhneme vyloučení vlivu rušivého signálu na výsledek měření. Metoda je obvykle označována jako *metoda dvojí integrace*. Poměrové měření vylučuje vliv tolerancí a změn hodnot součástek. Kondenzátor není vybit zkrátka, podstatně se proto snižuje vliv hystereze dielektrika.

Malou úpravou řízení, takovou, že měřené napětí přivádíme na vstup integrátoru trvale a přiváděný náboj kompenzujeme konstantním proudem s opačnou polaritou přiváděným v pulzech ze zdroje referenčního napětí, se dostáváme k *metodě vyrovnávání náboje*. Základní metoda vyžaduje použít komparátor s hysterezí: pokud napětí na výstupu integrátoru vybočí

z pásma necitlivosti na jednu stranu, bude kompenzační proud zapnut, pokud vybočí na druhou stranu, bude vypnut. Spínání kompenzačního proudu zajistí přímo výstupní signál komparátoru, podíl času, po který byl kompenzační proud přiváděn můžeme zjistit hradlováním signálu čítače mikropočítače. Tento podíl přímo odpovídá poměru absolutní hodnoty vstupního a kompenzačního napětí, vliv tolerancí součástek a jejich změn je kompenzován (s výjimkou poměru R_1 / R_2). Vliv rušení sítě je vyloučen, pokud je celková doba měření celočíselným násobkem periody síťového napětí.

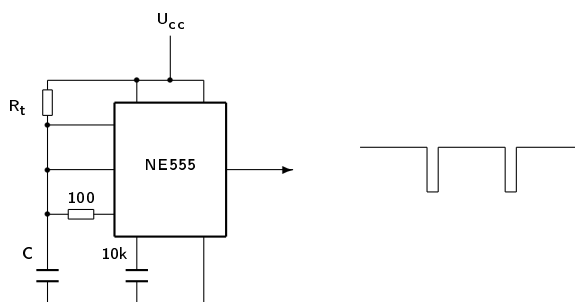


Obrázek 5.4: Vyrovnávání náboje

Napětovou hysterezi klasické metody vyrovnávání náboje lze nahradit „časovou hysterezí“ vyvolanou periodickým vzorkováním výstupu komparátoru (v konstantních časových intervalech). Zapojení se pak podstatně zjednoduší (obr.5.4), postačí nám porovnání výstupu integrátoru s nulou. Změřené napětí (jeho střední hodnota za dobu měření) je zde úměrné počtu intervalů, po které byl kompenzační zdroj připojen k celkovému počtu měřících intervalů.

5.3 Alternativní řešení A/D převodu

V řadě případů používáme A/D převodník k měření neelektrických veličin, které jsou vlastním senzorem převáděny na jinou elektrickou veličinu a teprve potom na napětí. Příkladem může být měření teploty levnými termistorovými čidly nebo měření tlaku kapacitními sondami. V takových případech může být výhodnější převést hodnotu odporu nebo kapacity na kmitočet nebo délku intervalu, pro jejichž měření jsou i ty nejjednodušší mikropočítače a řadiče dobře vybaveny. Příklad velice jednoduchého převodníku odpor/kmitočet s obvodem NE555 určeného pro měření teploty termistorovým čidlem uvádí obr.5.5.

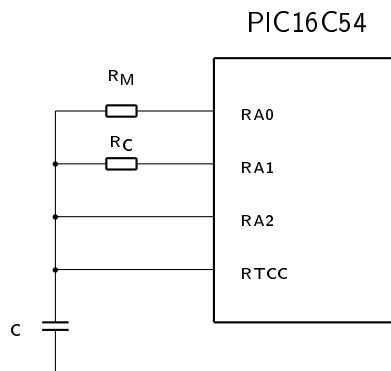


Obrázek 5.5: Převodník teplota/kmitočet

Převodník není pochopitelně zcela lineární, stejně jako není lineární závislost odporu termistoru na teplotě. Kompozici obou závislostí lze buď linearizovat (pro práci v malém intervalu teplot, například pro regulaci ústředního topení) nebo kompenzovat, například tabulkou (pro práci ve větším rozsahu teplot).

Funkci odpovídající předchozímu zapojení lze realizovat i samotným jednočipovým mikro-

počítačem, pokud je vybaven napětovým komparátorem. Takový komparátor najdeme např. u mikrořadiče Microchip 16C54, vybavení komparátorem je typické i pro mikrořadiče Zilog Z86Cxx. Jako příklad si uvedeme měření odporu (např. termistoru) řadičem PIC16C54 na obr.5.6.



Obrázek 5.6: Měření teploty mikrořadičem 16C54

Komparátor na vstupu RTCC porovnává napětí na kondenzátoru s pevným vnitřním referenčním napětím. Kondenzátor je nabíjen jednotkovou úrovní na I/O vývodu RA₀ přes měřený odpor R_M a je vybíjen nulovou úrovní na vývodu RA₂. Vliv nepřesné vnitřní reference a tolerance kondenzátoru dovoluje kompenzovat srovnávací měření pevného odporu R_C připojeného k I/O vývodu RA₁. Připomeňme si, že I/O vývody řadiče 16C54 lze (na rozdíl od vývodů mikropočítače 8051) převést do stavu vysoké impedance.

Použití frekvence periodického signálu jako prostředníka pro předání údaje o jiné elektrické i neelektrické veličině je výhodné, máme-li tento údaj přenést na větší vzdálenost. Zde najdou uplatnění i lineární převodníky napětí/kmitočet (například obvod National Semiconductor LM231), slušný lineární převodník napětí/frekvence lze sestavit i z běžných součástek.

Zde jsme se dostali do místa, kde bychom se asi měli zastavit, protože dalším krokem by byly inteligentní senzory, převodníky elektrických a neelektrických veličin schopné předat naměřenou hodnotu na žádost (ale i z vlastní iniciativy) po standardizovaném rozhraní (např. RS-485). Takový inteligentní sensor už ale není svou strukturou pouhým převodníkem, jedná se úplně mikropočítač v integrované formě schopný komunikace s nadřazeným systémem.

5.4 D/A převodníky

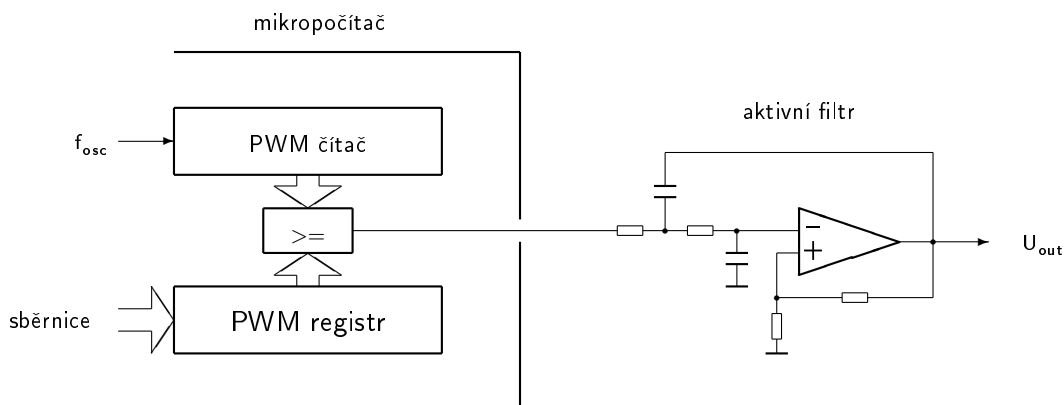
Číslicově-analogový převodník je prvek, který uvnitř struktury jednočipových mikropočítačů a řadičů prakticky nenajdeme. Pokud tedy chceme pro generování analogového výstupního signálu použít klasický D/A převodník s odporovým děličem musíme ho připojit zvnějšku. Nejjednodušší řešení, přímé připojení samostatného D/A převodníku na I/O vývody mikropočítače je použitelné, pokud nám nevadí velký počet obsazených I/O vývodů.

Pokud z jakéhokoliv důvodu vytváříme na mikropočítači vnější paralelní sběrnici, můžeme tuto sběrnici využít pro připojení D/A převodníku s vestavěným registrem, například typu Analog Devices 7524.

Konečně, existují D/A převodníky, které lze ovládat po sériové sběrnici SPI (šestinásobný šestibitový D/A převodník Motorola MC144110 nebo čtyřnásobný šestibitový D/A převodník Motorola MC144111), MicroWire (kombinace osmibitového D/A převodníku s komparátorem a přepínačem kanálů National Semiconductor ADC0852 a ADC0854) nebo I²C (osmikanálový

šestibitový D/A převodník Philips TDA8444 nebo čtyřkanálový A/D převodník s osmibitovým D/A převodníkem Philips PCF8591).

Pro generování stabilních, nebo pomalu se měnících analogových signálů jsou D/A převodníky nahrazovány generátorem signálu PWM. Generátor PWM produkuje logický signál s pevnou periodou (např. 255 instrukčních cyklů) a proměnnou střídou, na jeho výstup je zařazený filtr, jehož úkolem je odfiltrovat základní kmitočet PWM generátoru (odpovídá jeho periodě) a jeho vyšší harmonické. Příklad D/A převodníku opírajícího se o PWM převodník doplněný o aktivní filtr najdeme na obr.5.7.



Obrázek 5.7: PWM generátor

Závěrem si připomeňme, že mezní frekvence, do které lze generátor PWM použít je omezena *vzorkovacím teorémem*. Ten nám říká, že pro přesnou aproximaci periodického signálu potřebujeme tolik vzorků za vteřinu, kolik je dvojnásobek nejvyšší harmonické složky v signálu. Pro PWM převodník mikropočítače 80C552, který se opírá o osmibitový čítač se vstupním signálem o kmitočtu 1 MHz to například znamená, že jsme schopni generovat signály nejvýše o kmitočtu 1953 Hz (za předpokladu, že filtr, který na vývod PWM zařadíme bude tak kvalitní, že vše do tohoto kmitočtu bez útlumu přeneseme a základní kmitočet PWM generátoru (3906 Hz) a jeho vyšší harmonické beze zbytku potlačí).

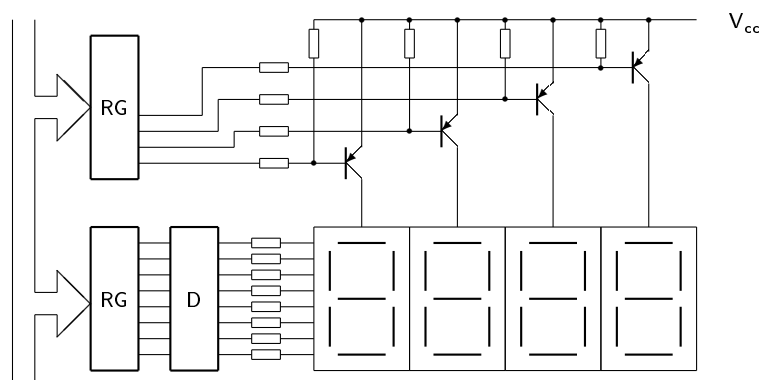
Když už jsme se dostali ke vzorkovacímu teorému, měli bychom zde alespoň upozornit na nutnost jeho respektování i při A/D převodu. Pokud se chceme vyhnout chybné interpretaci analogových signálů vzorkovaných A/D převodníkem, musíme zajistit, že frekvence vzorkování je alespoň dvojnásobkem nejvyšší významné harmonické obsažené v měřeném signálu (včetně rušení). Splnění této podmínky obvykle vyžaduje zařazení dolní propusti (aktivní nebo pasivní) na vstup A/D převodníku.

6. Zobrazovače

Důležitým prvkem řady mikropočítačových systémů je zobrazovač. Poskytuje obsluhu zařízení informace o výstupních veličinách programu, dovoluje realizovat dialog se zařízením při nastavování parametrů. Přes rozvoj technologií plochých zobrazovačů s malou spotřebou (LCD, plazmové zobrazovače), je stále nejběžnější technologií v aplikacích jednočipových mikropočítačů zobrazovač LED, a to buď ve formě vhodně uspořádané skupiny indikátorů - diod LED nebo ve formě několikamístného numerického (výjimečně alfanumerického) segmentového zobrazovače.

6.1 Zobrazovače LED

Jednoduché a často používané připojení vícemístného sedmisegmentového zobrazovače LED se společnými anodami k mikropočítačovému systému uvádí obr.6.1. Zobrazovač pracuje v multiplexním režimu, v každém okamžiku svítí pouze segmenty jediné číslovky, té, která je aktivována sepnutím anodového spínače. U této číslovky svítí pouze ty segmenty, pro které jsou aktivovány odpovídající budiče katod. Náš obrázek uvádí připojení na sběrnici, zjednodušení lze dosáhnout při využití registrů I/O bran na místě registrů RG.

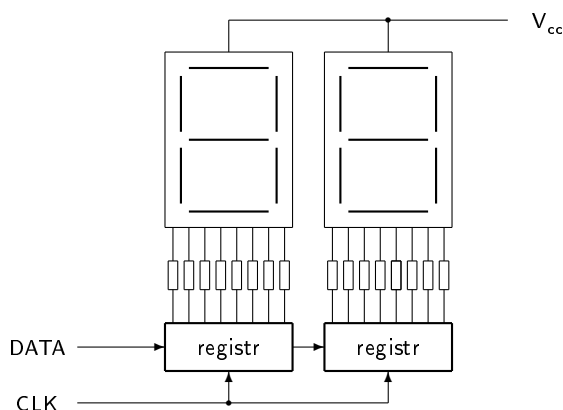


Obrázek 6.1: Zapojení jednoduchého dynamického zobrazovače

Výhodou klasického zapojení je jednoduchost i při vícemístném zobrazovači, nevýhodou, kterou lze poněkud kompenzovat zařazením generátoru znaků před budiče katod a dekóderu před spínače anod, je vysoký počet obsazených vývodů procesoru a nutnost relativně rychle a přitom pravidelně přepisovat data v registrech chceme-li zabránit nepříjemnému blikání (pro osmimístný zobrazovač musíme přepnout na další místo každé 2.5 ms). V řadě aplikací (např. indikace kmitočtu v mikropočítačem řízeném přijímači) je dynamická indikace s ohledem na způsobované rušení nepřijatelná.

V současné době se stále častěji setkáváme s alternativním statickým řízením segmentových zobrazovačů. Princip řešení si můžeme ilustrovat na zapojení dvoumístného sedmisegmentového zobrazovače (obr.6.2).

Zobrazovací prvky jsou připojeny na výstupy posuvného registru (např. typu 74HC165), data jsou do registru vkládána sériově po datovém vodiči DATA, přenos je řízen hodinami CLK. Řešení je vhodné pro menší počet číslovek, při větším počtu ovládaných segmentů je výhodné sáhnout po integrované verzi „řadiče“.



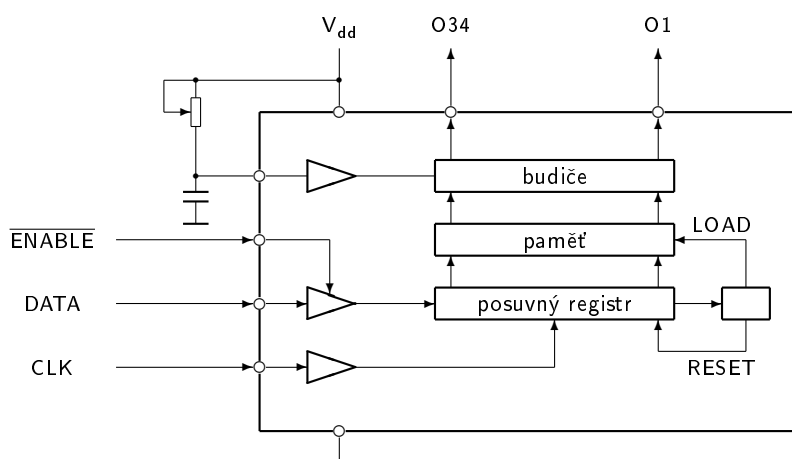
Obrázek 6.2: Využití posuvného registru pro zobrazovač LED

Nejznámějšími integrovanými řadiči segmentových zobrazovačů LED jsou obvody firmy SGS-Thompson řady M545x a M548x.

Řadiče LED M545x a M548x

Typ	segmentů	pouzdro
M5450	34	DIL40
M5451	35	DIL40
M5480	23	DIL28
M5481	14	DIL20

Vnitřní strukturu a zapojení obvodu M5450, který dovoluje řídit 34 segmentů LED, uvádí obr.6.3. Obvod M5451 umožňuje řídit až 35 segmentů, na úkor vývodu posledního segmentu chybí vstup výběrového signálu a není proto možné řídit více obvodů společnými vodiči DATA a CLK.



Obrázek 6.3: Řadič segmentových zobrazovačů LED M5451

Řízení obvodů M545x a M548x odpovídá sériové sběrnici Microwire. Data na datovém vodiči DATA jsou čtena s náběžnou hranou hodinového signálu CLK, obvod čeká na startovací jednotkový bit, následující bity po jejich načtení prezentuje na výstupech zobrazovače.

Obvod je vybaven řízením jasu segmentů, vedle úspory velkého počtu odporů omezujících proud segmenty získáváme možnost regulovat intenzitu svitu zobrazovače s ohledem na intenzitu vnějšího osvětlení.

S maticovými zobrazovači LED, které se používají jako informační panely a reklamní tabla, se v typických mikropočítačových aplikacích prakticky nesetkáme. Jejich řešení se s ohledem na vysoký počet diod LED opírá o multiplex a může být, až na značné požadavky na výkonové parametry spínačů a efektivitu diod LED, „obdobou“ jednoduchého několikamístného segmentového zobrazovače.

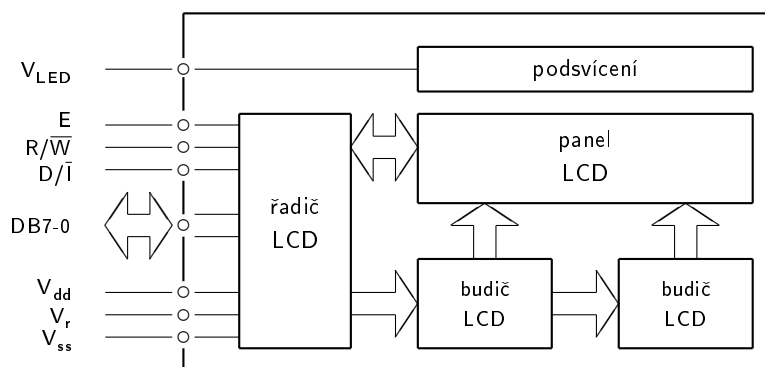
6.2 Zobrazovače LCD

Zobrazovače LED jsou sice dobře čitelné i při malém nebo naopak velkém vnějším osvětlení, mají však jednu podstatnou nevýhodu, kterou je vysoká spotřeba. Navíc se až na výjimky (informační tabla) jedná o malý počet sedmisegmentových zobrazovačů určených pro presentaci číselných údajů.

Nevýhodu velké spotřeby lze obejít náhradou zobrazovačů LED zobrazovači LCD. S jejich ovládním bez použití speciálních řadičů (jako je např. 31-segmentový řadič M5483) se v praxi nesetkáme.

Podstatně efektnější i pro malé aplikace jsou však dnes řádkové alfanumerické zobrazovače LCD. Obvykle dovolují zobrazit dva řádky o délce 16, 20, 24 nebo 40 znaků, jsou běžně vyráběny v reflexním provedení, pro provoz za horších světelných podmínek mohou být podsvícené elektroluminescenčním panelem nebo panelem LED.

Ovládání samotného panelu LCD přímo mikropočítačem by bylo značně komplikované, proto jsou LCD panely dodávány jako moduly s vlastním řadičem. Vnitřní strukturu takového modulu uvádí obr.6.4.



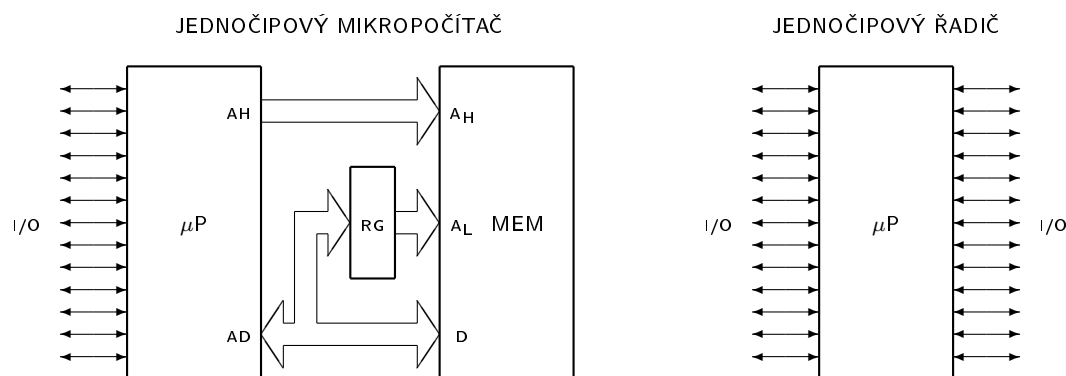
Obrázek 6.4: Modul zobrazovače LCD

Řadičem LCD panelu v modulu je běžně obvod Hitachi HD44780, který může přebírat data z osmibitové datové sběrnice mikropočítače (vodiče DB7-DB0) nebo jen z její čtyřbitové části (vodiče DB7-DB4, předávány jsou nejdříve čtyři vyšší řády potom čtyři nižší řády). Předávání dat je řízeno signálem E (Enable), údaje jsou vzorkovány závěrnou hranou. Signál R/\overline{W} řídí směr přenosu mezi mikropočítačem a řadičem zobrazovače, signál D/\overline{I} odlišuje předávaná data od příkazů (inicializace zobrazovače, ovládání kurzoru). Vodiče V_{ss} a V_{dd} propojují zem a přivádějí napájení, napětí na vodiči V_r ovládá kontrast.

7. Osmibitové jednočipové řadiče

Jako *jednočipové řadiče* budeme označovat prvky, které konstrukčně soustřeďují základní funkce mikropočítačové struktury, tedy mikroprocesor, paměť programu, paměť dat a obvody rozhraní, na jediném čipu. Reagují na digitální nebo analogové signály přiváděné na jejich vstupy a produkují digitální nebo analogové signály na svých výstupech. Pro signály na vstupech je přitom předpokládána jen minimální předřazená úprava (zesílení, převod na napětí, úprava úrovně a/nebo hran), výstupy jsou běžně schopné přímo ovládat indikační prvky (diody LED, panely LCD) a/nebo produkovat řídicí signály pro výkonové prvky (relé, tyristory).

Na rozdíl od klasických *jednočipových mikropočítačů* nezpřístupňují jednočipové řadiče na svých vývodech signály sběrnice mezi mikroprocesorem, paměťmi a obvody rozhraní. Není možné rozšiřovat rozsah programu nad mez danou kapacitou vnitřní paměti, připojování přidavných vnějších obvodů rozhraní je limitováno počtem I/O vývodů. Nemožnost připojit vnější obvody rozhraní paralelně lze eliminovat vytvořením sériové sběrnice, která je pro pomalejší obvody rozhraní dostačující. Některé jednočipové řadiče rozhraní sériové sběrnice zahrnují (některé prvky řady Motorola 6805/68HC05 mají řadič SPI, řadič Philips 83C752 má rozhraní I²C), na ostatních lze obsluhu sériového rozhraní realizovat programově.



Obrázek 7.1: Struktura jednočipového mikropočítače a jednočipového řadiče

Doménou jednočipových řadičů jsou hromadně vyráběné řídicí obvody pro domácí spotřebiče, domácí audio-vizuální techniku, zabezpečovací zařízení, telefonní přístroje. Typickými aplikacemi jsou i obsluha vstupních zařízení počítačů (klávesnice, myši), řízení zobrazovacích panelů v automobilech, zpracování signálů v inteligentních senzorech a jednoúčelové řízení motorů pro průmyslovou automatizaci. Vzhledem k nízké ceně jsou i vhodnou alternativou k logickým obvodům tam, kde není vyžadována vysoká rychlost. Zatímco ještě před pár lety byly jednočipové řadiče typicky čtyřbitové a jejich aplikační možnosti se omezovaly na logické řízení, dnes se setkáváme prakticky výlučně s řadiči osmibitovými. Pro náročnější aplikace (řízení automobilových motorů, ABS systémy) lze počítat s příchodem jednočipových řadičů šestnáctibitových.

Hranice mezi jednočipovými řadiči a jednočipovými mikropočítači, jak jsme si ji definovali, není ostrá a jednotlivé produkty ji překrývají. Jednočipové mikropočítače, schopné vytvářet vnější sběrnici, se mohou často omezit na vnitřní paměť programu i dat a mohou tak být použity (při vyšší ceně prvku a spoje) ve funkci jednočipového řadiče. Někteří výrobci dokonce dodávají levné varianty jednočipových mikropočítačů v pouzdrech s malým počtem vývodů, u kterých je vnější rozšíření vyloučeno. Jako příklad si můžeme uvést jednočipový „řadič“ Philips 83C752 (87C572) v pouzdře DIL28 (PLCC28), který je modifikací jednočipového

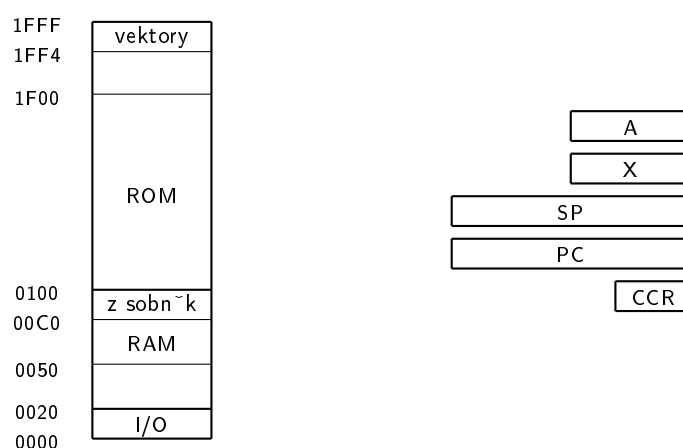
mikropočítače Philips 83C552. Na druhou stranu, řady jednočipových řadičů jsou rozšiřovány o prvky dovolující vytvořit vnější sběrnici. Takové modifikace podstatně zjednodušují vývoj programového vybavení, který se jinak musí opírat pouze o programové simulátory a opakované programování vnitřní paměti programu EPROM nebo EEPROM, případně o drahé emulační čipy. Jako příklad rozšíření řady jednočipových řadičů směrem k jednočipovým mikropočítačům si můžeme uvést prvek Microchip 16C71, který rozšiřuje řadu řadičů Microchip 16C5x. Podobně v řadě řadičů Motorola 6805 lze nalézt prvek 6805E0, který používá vnější paměť programu.

V následujícím textu si stručně popíšeme vnitřní strukturu pro nás nejzajímavějších jednočipových řadičů. Budou to Motorola 68HC05, Signetics ST62 a Microchip 16C5x.

7.1 Motorola MC68HC05

Jednočipové řadiče Motorola řady 68HC05 [1] jsou typickými představiteli své kategorie. Dnešní typy 68HC05 jsou vyráběny technologií HCMOS, starší typy 6805 byly vyráběny technologií HMOS. Řadiče 68HC05 jsou plně statické a mohou pracovat od nulového kmitočtu hodin až po 2-4 MHz na vnitřní sběrnici

Struktura jádra vychází z historického mikroprocesoru 6800 a staršího jednočipového mikropočítače 6801. Jádro řadiče 68HC05 bylo proti mikropočítači 6801 poněkud zjednodušeno (chybí druhý střadač a šestnáctibitové instrukce), na druhé straně bylo doplněno o instrukce pro práci s bity v paměti RAM a v registrech periférií. Procesor je doplňován perifériemi tak, aby co nejlépe vyhovoval požadavkům konkrétní aplikace. Současná nabídka firmy Motorola dnes zahrnuje více než 100 variant v pouzdrech s šestnácti až stošedesáti vývody, řada modifikací je vytvářena na základě požadavků zákazníků jako obvody CSIC (Consumer Specified Integrated Circuit) z knihovnic modulů.



Obrázek 7.2: Rozdělení paměti 68HC05, registry procesoru

Procesor pracuje s osmibitovým střadačem A, podporou pro adresaci je osmibitový indexregistru X. Vnitřní zásobník je adresován šesti- až jedenáctibitovým registrem SP, programový čítač PC je jedenácti- až šestnáctibitový (obr.7.2). Do jediného paměťového prostoru je mapována paměť ROM, RAM i periférie. Paměť RAM má podle typu kapacitu od 32 B do 2 KB. Vnitřní paměť programu ROM má kapacitu od 2 KB do 32 KB, existují modifikace s pamětí programu EPROM (značené 68HC705) v pouzdře s mazacím okénkem pro vývoj a modifikace

OTP v pouzdře bez mazacího okénka pro malé a ověřovací série. U některých typů jsou k dispozici i verze s pamětí programu EEPROM (značené 68HC805). Zajímavostí je standardní vybavení pamětí ROM s kódem testu řadiče a s kódem dovolujícím naprogramování vnitřní paměti EPROM z vnějšku připojené standardní paměti EPROM, tedy bez specializovaného programátoru.

Instrukční repertoár procesoru 68HC05 tvoří 59 instrukcí. Instrukce využívají řadu adresačních módů, instrukční repertoár je značně ortogonální. Operandem může být implicitní registr (Inherent Mode), přímý operand (Immediate Mode), operand adresovaný osmibitovou (Direct Mode) nebo šestnáctibitovou (Extended Mode) adresou. Operand může být adresován registrem X (Indexed Mode) s případným osmibitovým nebo šestnáctibitovým doplňkem. Instrukce podmíněného skoku používají relativní (Relative Mode) adresaci v rozsahu ± 128 B. U bitových operací lze adresovat libovolný bit v prvních 128 B paměti (Bit Mode).

Provedení aritmetické instrukce potřebuje 2 cykly, přídavné cykly jsou nutné pro čtení delších operandů, nepřímou adresaci, apod. Vzhledem k podobnosti instrukčního repertoáru řadiče 68HC05 a jednočipového mikropočítače 68HC11 odkazujeme zájemce o podrobnosti na tabulku 8.2, (str.91).

Řadič 68HC05 je vybavován širokou škálou periferií. Téměř standardním vybavením je šestnáctibitový čítač s jedním komparačním a jedním záchytným registrem. U jednodušších modelů najdeme jednoduchý osmibitový čítač/časovač se sedmibitovým předřazeným programovatelným děličem. Běžným vybavením řadičů 68HC05 je i hlídací časovač Watchdog, asynchronní sériové rozhraní SCI a/nebo sériová sběrnice SPI (případně univerzální sériové rozhraní SIOP) a čtyř- nebo osmivstupový osmibitový A/D převodník s postupnou aproximací bez analogové paměti (Sample/Hold). Řadiče určené pro specifické aplikace jsou doplňovány pamětmi EEPROM, sériovými sběrnicemi I²C, obvody PWM, obvody dvoutónové volby DTFM, řadiči panelů LCD, generátory televizního signálu OSD (On Screen Display), fázovými závěsy PLL a podobně.

Prioritní přerušovací systém řadičů 68HC05 vychází z koncepce procesoru 6800, zdrojem přerušování může být vnější nemaskovatelný signál (NMI), vnější maskovatelný signál (INT), přetečení vnitřního čítače nebo programové přerušování (instrukce SWI).

Pro zajímavost si uvedeme několik příkladů konkrétní konfigurace řadičů 68HC05.

Typ	ROM	RAM	E ²	SIO	A/D	PWM	I/O	Pouzdro	
68HC05B4	4K	176		SCI	8*8	2*8	32	56B,52FN	
68HC05B8	8K	176	256	SCI	8*8	2*8	32	56B,52FN	
68HC05C4	4K	176		SCI,SPI			31	40P,44FN	
68HC05D9	16K	352		SCI		5*8	32	40P,44FN	
68HC05F6	4K	320					26	42B,44FB	DTFM
68HC05G8	8K	304			8*8		64	160FT	PC chip
68HC05H2	2K	96		SIOP		2*8	16	40P,44DW	
68HC05J1	1K	64					14	20P,20DW	
68HC05K1	512	32					10	16P,16DW	
68HC05P4	4K	176		SIOP			21	28P,28DW	
68HC05T7	5K	320		I ² C	1*8	9*8	28	58B	
68HC05X4	4K	176					16	20DW	CAN
68HC05X16	16K	352	256	SCI	8*8	2*8	32	68FN,64FB	CAN

Pro označení pouzder jsou použity zkratky:

P = DIP, B = SDIP, DW = SOIC, FB = QFP (10*10 mm), FN = PLCC, FT = QFP (28*28 mm)

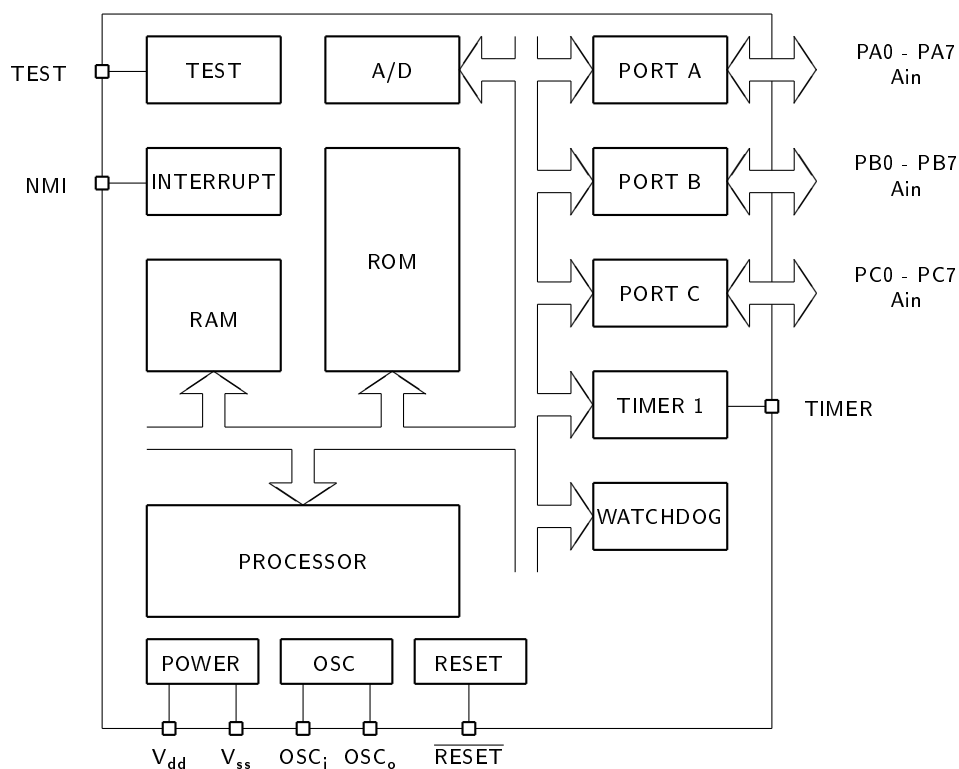
Typickými aplikacemi speciálně konfigurovaných řadičů Motorola jsou telefonní přístroje (řada 68HC05F), řízení krokových motorů (řada 68HC05H), zabezpečovací systémy, regulátory teploty, bezdrátové telefony, řadiče domácích spotřebičů (řady 68HC05J, 68HC05K, 68HC05P), čipové bankovní karty (Smartcards - 68HC05SC), televizory (68HC05T) a automobilové aplikace (68HC05X).

7.2 SGS-Thompson ST62

Osmibitové řadiče ST6210, ST6215, ST6220 a ST6225 firmy SGS-Thompson [2] jsou přechodem mezi klasickými jednočipovými mikropočítači a jednočipovými řadiči.

Jádro je tvořeno osmibitovým procesorem harvardského typu s malým počtem instrukcí, ale s dobrými adresačními možnostmi. Procesor je schopen pracovat na kmitočtu až 8 MHz (typická instrukce trvá 4 cykly). Procesor lze programově zastavit (STOP Mode). Součástí jádra je malá vnitřní datová paměť RAM o kapacitě 64 B a vnitřní paměť programu ROM o kapacitě 2-4 kB. Existují modifikace vybavené pamětí EPROM v klasickém provedení určené pro vývoj a modifikace vybavené pamětí EPROM v provedení OTP pro malé série.

Mikropočítačové jádro je doplněno o jednoduchý osmibitový časovač/čítač s předřazeným programovatelným děličem, obvod Watchdog a osmibitový A/D převodník s osmivstupovým (ST6210,15) nebo šestnáctivstupovým (ST6220,25) multiplexorem. Celkový počet I/O špiček, které mohou být použity jako digitální vstupy/výstupy je dvanáct (ST6210,15) nebo dvacet (ST6220,25). Čtyři z nich jsou bez posílení schopné ovládat diody LED nebo spínače TRIAC (dokáží sepnout až 20 mA). Strukturu obvodu popisuje obr.7.3



Obrázek 7.3: Struktura řadičů ST62

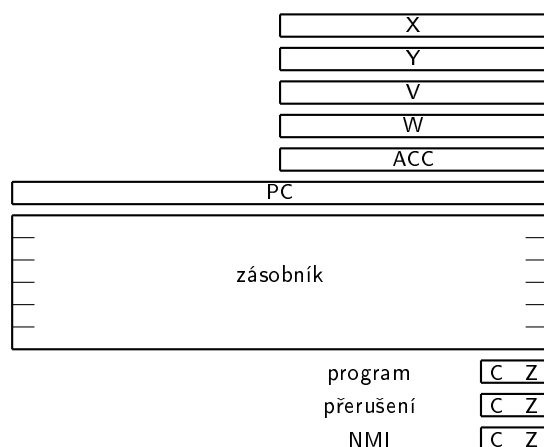
Základní řada je doplňována o typy ST6240,40 a 60 s řadiči zobrazovačů LCD, se sériovým rozhraním, s většími paměťovými prostory, s pamětí EEPROM.

Řadiče mají hezky řešené programování funkce vstupně/výstupních vývodů. Pro každý vývod lze samostatně definovat směr přenosu (vstup/výstup). U vývodů naprogramovaných jako vstupy lze programově připojit zvyšovací odpor, napojit je na přerušovací systém nebo využít jako vstupy A/D převodníku. Zde stojí za zmínku, že programovací mechanismus nahrazuje běžný analogový multiplexor, na což musíme dávat při programování pozor. U výstupů lze volit mezi otevřeným kolektorem a výstupem Push-Pull.

Časovač/čítač je tvořen osmibitovým čítačem a sedmibitovým programovatelným děličem. Vstupem je buď signál oscilátoru (vydělený dvanácti) nebo signál přivedený na špičku TIMER. Tento vývod může být alternativně využit pro hradlování vnitřního hodinového signálu, nebo jako výstup, který bude nastaven na naprogramovanou hodnotu při nulové hodnotě čítače. Obvodu Watchdog lze naprogramovat pracovní interval mezi 384 μ s a 24.576 ms (pro krystal 8 MHz).

Osmibitový A/D převodník pracuje s postupnou aproximací, potřebuje pro převod 70 μ s (pro krystal 8 MHz), má poměrně malou přesnost a není vybaven analogovou pamětí (Sample/Hold).

Registry procesoru zahrnují osmibitový akumulátor, dva pomocné registry (V, W), dva registry pro nepřímou adresaci (X,Y), dvanáctibitový čítač instrukcí a šestiúrovňový zásobník.

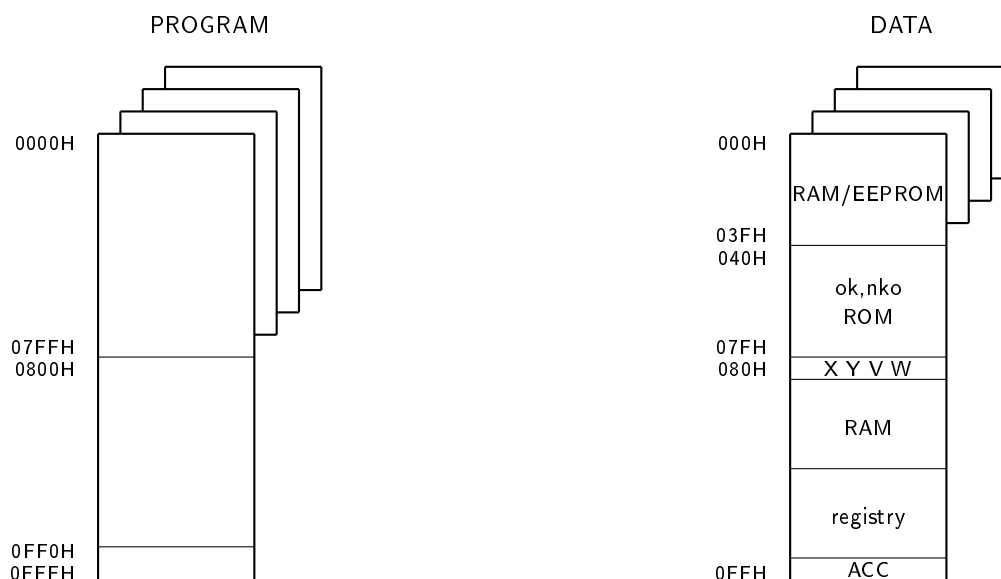


Obrázek 7.4: Registry procesoru řadičů ST62

Pro každý ze tří režimů práce procesoru (normální výpočet, přerušení, nemaskovatelné přerušení) máme samostatnou dvojici příznaků C a Z. Příznak C ukládá přenosový bit a příznak Z informaci o nulovém výsledku poslední aritmetické nebo logické operace. Přepnutí na odpovídající dvojici příznaků je spojeno se vstupem do přerušovací rutiny a návratem z ní.

Registry W, X a Y jsou součástí paměti RAM (celkem 64 B). Ta je spolu s akumulátorem a registry periférií mapována do datového paměťového prostoru o rozsahu 256 B. Zajímavostí je okénko o délce 64 B, do kterého lze mapovat zvolenou oblast paměťového prostoru programu (jde o jedinou možnost, jak zpřístupnit konstanty v paměti ROM). Paměťový prostor programu má rozsah 4 kB, s možností rozšíření o stránky o délce 2 kB, stránkování dovozuje i paměť dat. Strukturu paměťových prostorů popisuje obr.7.5

Instrukční repertoár procesoru je poměrně malý, malé délky kódu procesor dosahuje použitím devíti adresačních módů (nejde zde o ortogonalitu). Operandem instrukcí jsou osmibitové konstanty (Immediate), osmi bity přímo adresovaná datová paměť (Direct), dvěma bity určený registr X, Y, V nebo W (Short Direct), dvanáctibitové adresy skoku nebo podpro-

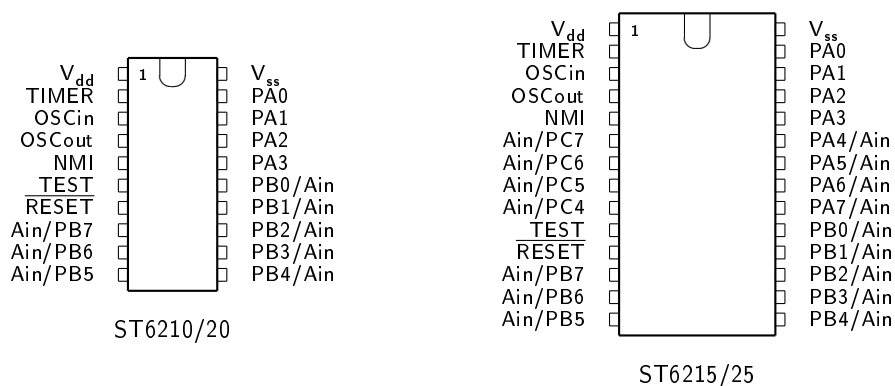


Obrázek 7.5: Struktura paměťových prostorů řadičů ST62

gramu (Extended), pětibitové relativní adresy pro podmíněné skoky (Relative), jedenácti bity přímo adresovaný bit v datové paměti (Bit Direct), datová paměť adresovaná hodnotou v registru X nebo Y (Indirect), implicitní operand operace (Inherent). Jako samostatný adresační mód je označována kombinace jedenáctibitové adresy bitu a osmibitové relativní adresy u podmíněných skoků (Bit Test & Branch). Procesor je poměrně pomalý, provedení jedné aritmetické instrukce (ADD) vyžaduje $6.5 \mu s$. Přehled všech instrukcí uvádí tab.7.1.

Přerušovací systém je pětiúrovňový, zdroji přerušení jsou: nemaskovatelné přerušení (špička NMI), vývody brány A, vývody bran B a C, časovač a A/D převodník. Snížení spotřeby lze dosáhnout převedením procesoru do stavu WAIT (zastavený výpočet, běží oscilátor) nebo STOP (zastavený oscilátor).

Obvody řady ST62 jsou schopné pracovat v rozsahu napájecího napětí 3 až 6 V, od -40 do $+85^{\circ}C$. Obvody ST6210 a ST6215 jsou vyráběny v pouzdech DIP20 a SOP20, obvody ST6220 a ST6225 jsou pouzřeny do pouzder DIP28 a SOP28, rozložení špiček uvádí obr.7.6.



Obrázek 7.6: Rozložení vývodů řadičů ST62

Aritmetické a logické operace			
Instrukce	Adresace	Délka	Příznak
ADD A,(X)	Indirect	1	Z,C
ADD A,(Y)	Indirect	1	Z,C
ADD A,rr	Direct	2	Z,C
ADDI A,#n	Immediate	2	Z,C
AND A,(X)	Indirect	1	Z
AND A,(Y)	Indirect	1	Z
AND A,rr	Direct	2	Z
ANDI A,#n	Immediate	2	Z
CLR A	Short	2	Z,C
CLR r	Direct	3	
COM A	Inherent	1	Z,C
CP A,(X)	Indirect	1	Z,C
CP A,(Y)	Indirect	1	Z,C
CP A,rr	Direct	2	Z,C
CPI A,#n	Immediate	2	Z,C
DEC X	Short	1	Z
DEC Y	Short	1	Z
DEC V	Short	1	Z
DEC W	Short	1	Z
DEC A	Direct	2	Z
DEC rr	Direct	2	Z
DEC (X)	Indirect	1	Z
DEC (Y)	Indirect	1	Z
INC X	Short	1	Z
INC Y	Short	1	Z
INC V	Short	1	Z
INC W	Short	1	Z
INC A	Direct	2	Z
INC rr	Direct	2	Z
INC (X)	Indirect	1	Z
INC (Y)	Indirect	1	Z
RLC A	Inherent	1	Z,C
SLA A	Inherent	2	Z,C
SUB A,(X)	Indirect	1	Z,C
SUB A,(Y)	Indirect	1	Z,C
SUB A,rr	Direct	2	Z,C
SUBI A,#n	Immediate	2	Z,C

Přesuny			
Instrukce	Adresace	Délka	Příznak
LD A,X	Short	1	Z
LD A,Y	Short	1	Z
LD A,V	Short	1	Z
LD A,W	Short	1	Z
LD X,A	Short	1	Z
LD Y,A	Short	1	Z
LD V,A	Short	1	Z
LD W,A	Short	1	Z
LD A,rr	Direct	2	Z
LD rr,A	Direct	2	Z
LD A,(X)	Indirect	1	Z
LD A,(Y)	Indirect	1	Z
LD (X),A	Indirect	1	Z
LD (Y),A	Indirect	1	Z
LD A,#n	Immediate	2	Z
LD rr,#n	Immediate	3	Z

Podmíněné skoky			
Instrukce	Podmínka	Délka	Příznak
JRC e	$C = 1$	1	
JNRC e	$C = 0$	1	
JRZ e	$Z = 1$	1	
JNRZ e	$Z = 0$	1	
JRR b,rr,e	$bit = 1$	3	C
JRS b,rr,e	$bit = 0$	3	C

Bitové operace			
Instrukce	Adresace	Délka	Příznak
SET b,rr	Bit Direct	2	
RES b,rr	Bit Direct	2	

Řízení			
Instrukce	Adresace	Délka	Příznak
NOP	Inherent	1	
RET	Inherent	1	
RETI	Inherent	1	Z,C
STOP	Inherent	1	
WAIT	Inherent	1	
CALL abc	Extended	2	
JP abc	Extended	2	

Tabulka 7.1: Instrukční soubor mikrořadičů ST62

7.3 Microchip PIC16C5x

Osmibitové řadiče PIC16C5x firmy Microchip Technology [3] jsou zajímavé svou vnitřní strukturou, vysokým výkonem, nízkou spotřebou a nízkou cenou.

Jádrem je plně statický, osmibitový procesor schopný pracovat s hodinovým kmitočtem 20 MHz. Při tomto kmitočtu je instrukce provedena za 200 ns, (s výjimkou instrukcí skoku, které potřebují dva instrukční cykly). Procesor využívá překrývání cyklů FETCH a EXECUTE (a výrobce ho označuje jako procesor typu RISC), během zpracování jedné instrukce je další instrukce čtena z paměti programu. Instrukční soubor tvoří pouhých 33 instrukcí. Procesor má typickou harvardskou architekturu s oddělenou pamětí programu a dat a odlišnou délkou slova dat a instrukce. Zatímco délka datového slova je 8 bitů, délka instrukce je 12 bitů. Programová paměť je typu EPROM s kapacitou 512 B až 2 KB, řadiče jsou dodávány v keramickém pouzdře s okénkem pro vývoj a v provedení OTP (One Time Programmable) v levném plastovém pouzdře pro výrobní sérii.

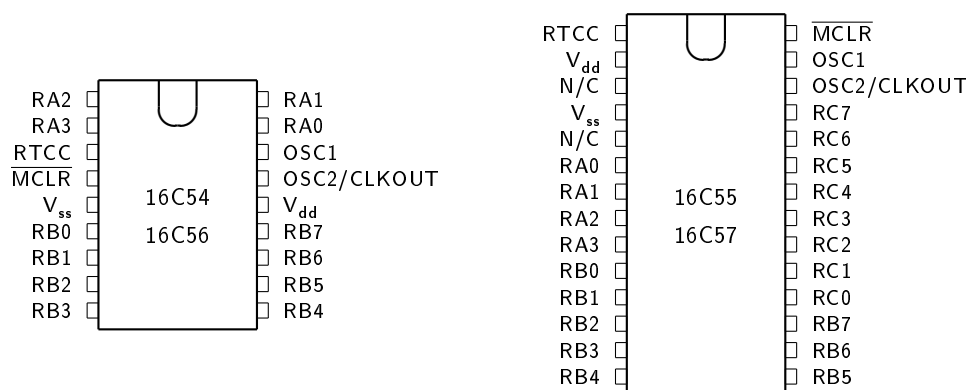
Řada PIC16C5X je tvořena čtyřmi řadiči, které se liší kapacitou paměti programu EPROM, kapacitou paměti RAM a počtem periferních vývodů.

Přehled řadičů řady PIC16C5X

Typ	EPROM	RAM	I/O	Pouzdro
PIC16C54	512*12	32*8	13	DIP18, SOIC18, SSOP20
PIC16C55	512*12	32*8	21	DIP28, SOIC28, SSOP28
PIC16C56	1K*12	32*8	13	DIP18, SOIC18, SSOP20
PIC16C57	2K*12	80*8	21	DIP28, SOIC28, SSOP28

Kapacita paměti RAM zahrnuje i registry
Mezi I/O vývody je zahrnut i vývod RTCC

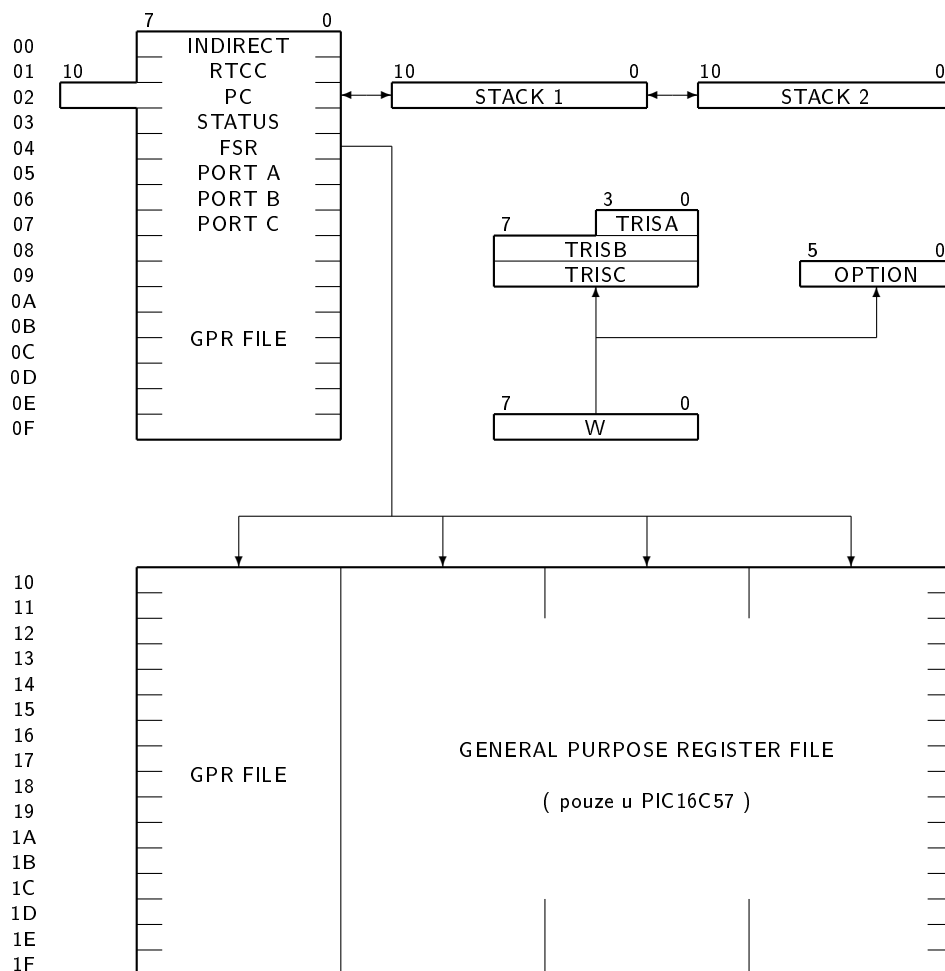
Řadiče PIC16C5X jsou schopné pracovat v rozsahu napájecích napětí 2.5 až 6 V, od -55 do +125°C. Hodinový generátor může být řízen krystalem nebo obvodem RC, lze použít i vnější hodinový generátor (volbu provádíme při programování EPROM). Zapsaný program lze pojistkou v paměti EPROM chránit proti čtení a přepsání. Při kmitočtu krystalu 4 MHz má řadič spotřebu pod 2 mA, při kmitočtu 32 kHz 15 μ A a ve stavu STANDBY pouze 3 μ A. Řadiče jsou vyráběny v pouzdech DIP-18 (DIP-28), SOIC-18 (SOIC-28) a SSOP-20 (SSOP-28). Rozložení vývodů pro pouzdra DIP a SOIC uvádí obr.7.7.



Obrázek 7.7: Rozložení vývodů řadičů PIC16C5X

Strukturu registrů (paměti RAM) řadiče popisuje obr.7.8. Adresní prostor paměti RAM má

rozsah 32 až 80 B a zahrnuje i registry. Je rozdělen na 16 přímo adresovatelných registrů, dalších 16 buněk je adresovatelných pouze nepřímo, stejně jako doplňkových 48 buněk u řadiče 16C57. Do počátku paměťového prostoru RAM jsou mapovány registry procesoru (W, PC, STATUS, FSR, STACK1, STACK2) a registry periférií (RTCC, OPTION, PORTA, PORTB, PORTC, TRISA, TRISB a TRISC).



Obrázek 7.8: Struktura registrů PIC16C5X

Buňky nepřímo adresované zóny jsou dostupné zapsáním jejich adresy do registru $F04_H$ (FSR) a použitím adresy $F00_H$ (Indirect Address) v příslušné instrukci. (Pozn. $F00_H$ není registr, adresa 00_H pouze označuje nepřímou adresaci.)

Adresa $F01_H$ (RTCC - Real Time Clock/Counter) zpřístupňuje osmibitový čítač/časovač, kterému lze předřadit osmibitový předřazený dělič (nastavením příslušných bitů v konfiguračním registru OPTION). Zdroj vstupního signálu (instrukční cyklus nebo vývod RTCC) a reakci na náběžnou nebo sestupnou hranu určuje opět registr OPTION.

Čítač programu (PC - Program Counter, přesněji jeho méně významných osm bitů) je dostupný na adrese $F02_H$. Zápisem na adresu $F02_H$ lze také realizovat skok (přesněji, nastavit osm méně významných bitů adresy, devátý bit bude nulový a desátý a jedenáctý bit bude určen hodnotami bitů PA0 a PA1 registru $F03_H$ (STATUS). Návrátové adresy při volání podprogramu jsou ukládány do dvouúrovňového zásobníku STACK1 a STACK2.

Registr $F03_H$ (STATUS) zpřístupňuje příznaky Z (Zero), C (Carry/Borrow) a DC (Digit

Carry/Borrow — přenos mezi čtvrtým a pátým bitem při operacích sčítání a odčítání). Bit TO (Time-Out) je nastavován při startu procesoru, instrukcí CLRWDT a instrukcí SLEEP, nulován je při vynulování čítače Watchdog. Registr PD (Power-Down) je nastavován při startu procesoru a instrukcí CLRWDT, nulován je instrukcí SLEEP.

F03	PA2	PA1	PA0	TO	PD	Z	DC	C
------------	-----	-----	-----	----	----	---	----	---

Paměťový prostor přímo adresovatelný instrukcí skoku má rozsah 512 B (při volání podprogramu dokonce pouze 256 B). Bity PA0 a PA1 registru STATUS dovolují pracovat s pamětí do 2 KB, přednastavují adresu stránky programu (512 B) pro instrukce skoku a volání podprogramu, bit PA2 lze volně používat (je rezervou pro další budoucí rozšíření paměti).

Registr F04_H (FSR — File Select Registr) adresuje jeden z 32 registrů určených bity 0-4. Bity 5 a 6 dovolují u procesoru 16C57 vybrat jednu ze čtyř stránek registrů.

Adresa F05_H zpřístupňuje čtyřbitový port A. Čtením z adresy F05_H získáme (v méně významných řádech) informaci o logických úrovních na čtyřech vývodech portu A. Zápis na adresu F05_H odpovídá výstupu na vývody portu A, které jsou nakonfigurované jako výstupní zápisem hodnoty 0 v příslušných bitech konfiguračního registru TRIS A. Na adrese F06_H lze podobně číst údaje z portu B a zapisovat na ni, konfiguračním registrem portu B je registr TRIS B. Konečně, na adrese F07_H je zpřístupněn port C (pouze u řadičů 16C55 a 16C57), jeho konfiguračním registrem je TRIS C. Výstupní budiče vývodů jsou třístavové, při vhodném ovládní je lze používat i jako výstupy s otevřeným kolektorem.

Registry F08_H - F1F_H (a u řadiče 16C57 dále F20_H - F7F_H) jsou volně použitelné paměťové buňky.

Kromě adresovatelných registrů F00_H - F1F_H (F00_H - F7F_H) procesor pracuje s registry W, TRISA, TRISB a TRISC. Pro zápis do registrů TRISA, TRISB a TRISC máme zvláštní instrukci TRIS.

Registr OPTION nastavovaný instrukcí OPTION konfiguruje čítač/časovač. Bit RTS volí vstupní signál (0 - vnitřní hodiny, 1 - vnější signál), bit RTE volí aktivní hranu (0 - náběžná hrana, 1 - sestupná hrana), bit PSA zařazuje programovatelný dělič před čítač/časovač (PSA=0) nebo za generátor signálu pro Watchdog (PSA=1).

OPTION	-	-	RTS	RTE	PSA	PS2	PS1	PS0
---------------	---	---	-----	-----	-----	-----	-----	-----

Bity PS2, PS1 a PS0 určují dělicí poměr předřazeného děliče (2, 4, 8, 16, 32, 64, 128, 256 pro časovač, 1, 2, 4, 8, 16, 32, 64, 128 pro obvod Watchdog).

Hlídací obvod Watchdog má svůj vlastní generátor hodin (vnitřní obvod RC), má základní periodu 18 ms. Pokud není v této době restartován instrukcí CLRWDT, zresetuje procesor. Základní periodu hlídacího obvodu lze prodloužit až 128-krát předřazením programovatelného děliče (nastavením bitu PSA a bitů PS2, PS1 a PS0).

Instrukční soubor řadiče PIC16C5x uvádí tabulka 7.2. Zajímavostí souboru je možnost výběru registru výsledku aritmetických a logických operací bitem d v instrukci. Nulová hodnota tohoto bitu udává jako registr výsledku pracovní registr/akumulátor W, jednotková hodnota dovoluje zapsat výsledek do přímo nebo nepřímo adresovatelného registru f.

Znakové operace		(11-6)	(5)	(4-0)	
		OC	d	#F	
Op. kód	Operandy	Operace			Příznaky
NOP					
MOVWF	f	$W \rightarrow f$			
CLRW		$0 \rightarrow W$			Z
CLRF	f	$0 \rightarrow f$			Z
SUBWF	f,d	$f - W \rightarrow d[f + \overline{W} + 1 \rightarrow d]$			C,DC,Z
DECF	f,d	$f - 1 \rightarrow d$			Z
IORWF	f,d	$W \vee f \rightarrow d$			Z
ANDWF	f,d	$W \& f \rightarrow d$			Z
XORWF	f,d	$W \oplus f \rightarrow d$			Z
ADDWF	f,d	$W + f \rightarrow d$			C,DC,Z
MOVF	f,d	$f \rightarrow d$			Z
COMF	f,d	$\overline{f} \rightarrow d$			Z
INCF	f,d	$f + 1 \rightarrow d$			Z
DECFSZ	f,d	$f - 1 \rightarrow d$, skip if zero			
RRF	f,d	$f(n) \rightarrow d(n-1), f(0) \rightarrow C, C \rightarrow d(7)$			C
RLF	f,d	$f(n) \rightarrow d(n+1), f(7) \rightarrow C, C \rightarrow d(0)$			C
SWAPF	f,d	$f(0-3) \leftrightarrow f(4-7) \rightarrow d$			
INCFSZ	f,d	$f + 1 \rightarrow d$, skip if zero			

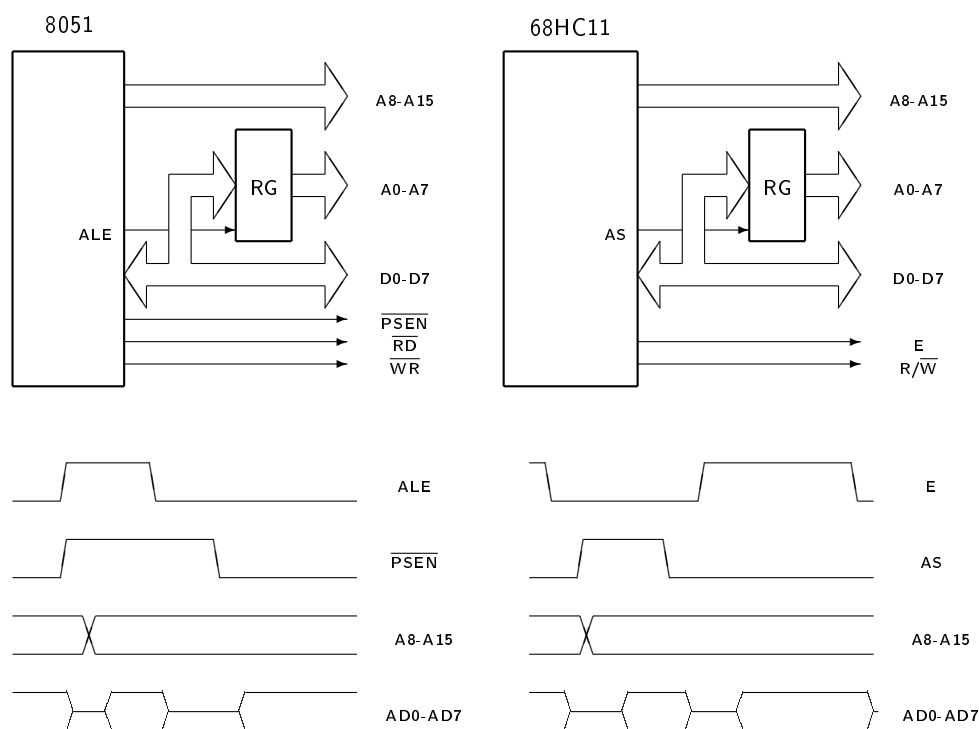
Bitové operace		(11-8)	(7-5)	(4-0)	
		OC	#b	#F	
Op. kód	Operandy	Operace			Příznaky
BCF	f,b	$0 \rightarrow f(b)$			
BSF	f,b	$1 \rightarrow f(b)$			
BTFSC	f,b	test bit(b) in file(f), skip if clear			
BTFSS	f,b	test bit(b) in file(f), skip if set			

Řízení a operace s konstantou		(11-8)	(7-0)	
		OC	LIT	
Op. kód	Operandy	Operace		Příznaky
OPTION		$W \rightarrow OPTION$		
SLEEP		$0 \rightarrow WDT$, stop oscillator		TO,PD
CLRWDT		$0 \rightarrow WDT$		TO,PD
TRIS	f	$W \rightarrow I/O$ control register		
RETLW	k	$k \rightarrow W, Stack \rightarrow PC$		
CALL	k	$PC + 1 \rightarrow Stack, k \rightarrow PC$		
GOTO	k	$k \rightarrow PC$ (9 bits)		
MOVLW	k	$k \rightarrow W$		
IORLW	k	$k \vee W \rightarrow W$		Z
ANDLW	k	$k \& W \rightarrow W$		Z
XORLW	k	$k \oplus W \rightarrow W$		Z

Tabulka 7.2: Instrukční soubor mikrořadičů PIC 16C5X

8. Osmibitové jednočipové mikroočítače

Jednočipové mikroočítače jsme si v předcházející kapitole charakterizovali jako prvky, které na jediném čipu sdružují procesor, paměť programu typu ROM (případně OTPROM, EPROM nebo EEPROM), paměť dat RAM a obvody rozhraní pro běžné aplikace. Přitom u nich existuje možnost vytvořit na jejich vývodech (na úkor vstupů a výstupů aplikace) multiplexovanou vnější sběrnici, na kterou lze připojit rozšiřující paměť programu a/nebo dat a doplňkové obvody rozhraní. Způsob vytvoření osmibitové multiplexované sběrnice pro mikroočítače nejběžnějších řad osmibitových mikroočítačů Intel 8051 a Motorola MC68HC11 uvádí obr.8.1.



Obrázek 8.1: Vytvoření vnější sběrnice na mikroočítačích Intel 8051 a Motorola MC68HC11

V některých případech (Motorola MC68HC11) lze ztrátu vývodů použitých pro vytvoření vnější sběrnice kompenzovat připojením speciálního vnějšího obvodu (Motorola MC6824), který původní vývody plně nahrazuje (bez modifikace programu). V řadě případů (např. Philips 83C451 nebo 83C552) je však počet I/O vývodů mikroočítače dostatečně vysoký, a ztráta dvou osmibitových bran pro vytvoření sběrnice nám tak ani nemusí v řadě aplikací vadit.

Většina současných jednočipových mikroočítačů vznikla postupným doplňováním vnitřních pamětí a obvodů rozhraní k starším, široce používaným mikroprocesorům. Typickým příkladem jsou řady Motorola 6801/02, 6805/68HC05, 68HC11 a 68HC16 vycházející z architektury procesoru 6800, řada šestnáctibitových mikroočítačů Motorola 6833x vychází z architektury procesoru 68000, řady Zilog Z8 a Z86Cxx, Hitachi HD64180 a Toshiba Z84C0xx vycházejí z architektury procesoru Zilog Z80. Výhodou tohoto přístupu je programová kompatibilita se staršími mikroprocesory a možnost využít jejich vývojové prostředky a akumulované zkušenosti.

Alternativním přístupem je na starších mikroprocesorech nezávislý návrh jednočipových řadičů a mikroočítačů, který je charakteristický pro firmu Intel. Dnes již nepoužívané jednočipové mikroočítače Intel 8035/41/48 se opírají o vlastní procesor nezávislý na mikropro-

cesoru Intel 8080, z řady Intel 8035/41/48 vychází velice úspěšný typ 8051 vyráběný a modifikovaný dalšími firmami (Siemens, Philips). Podobně, nezávislou architekturou se vyznačuje i procesor řady šestnáctibitových jednočipových mikropočítačů Intel 8096/80C196.

Pro současné osmibitové jednočipové mikropočítače je typické přebírání instrukčního souboru starších mikroprocesorů, které lze charakterizovat jako procesory typu CISC. Pouze okrajově se zde setkáváme s některými rysy architektury RISC, příkladem může být řada procesorů Hitachi H8/330.

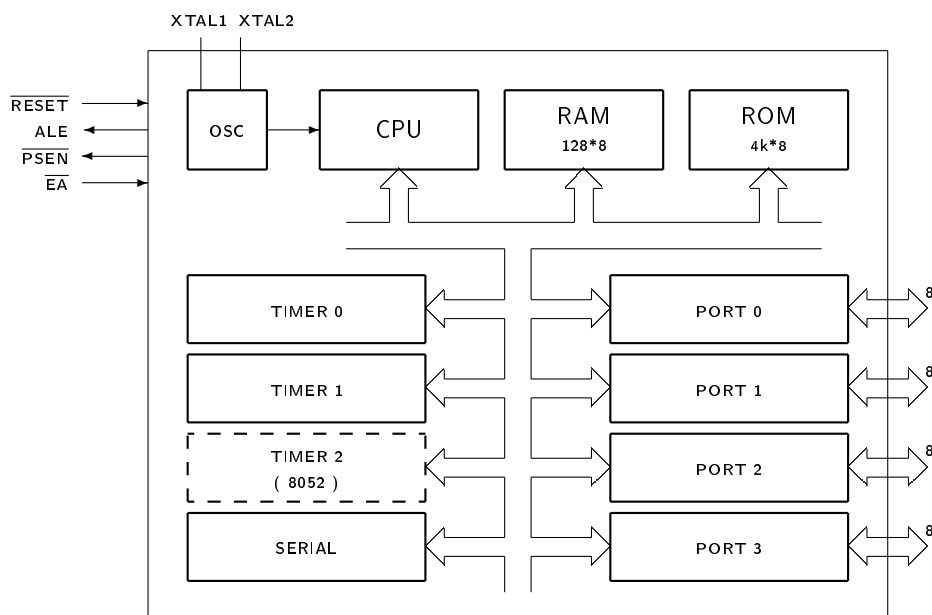
V této kapitole si budeme všimnout typických představitelů kategorie osmibitových jednočipových mikropočítačů - mikropočítače Intel 8051, jeho modifikací Philips 80C552, Siemens 80C515, Siemens 80C717 a mikropočítače Motorola MC68HC11. Jako příklad jednočipového mikropočítače s „architekturou RISC“, který však již patří spíše mezi šestnáctibitové si uvedeme mikropočítač Hitachi H8/330.

8.1 Intel 8051

Osmibitový jednočipový mikropočítač 8051 vyvinutý firmou Intel v roce 1983 [4] se stal základem rozsáhlé třídy mikropočítačů vyráběných dnes v mnoha modifikacích řadou výrobců (AMD, Intel, Matra-MHS, OKI, Philips, Siemens/Signetics). My si zde uvedeme základní informace potřebné k práci s typem 8051 (nebo jeho praktickým rozšířením 8052), hrubé informace o procesorech, které z 8051 vycházejí najdete v samostatných sekcích tohoto textu.

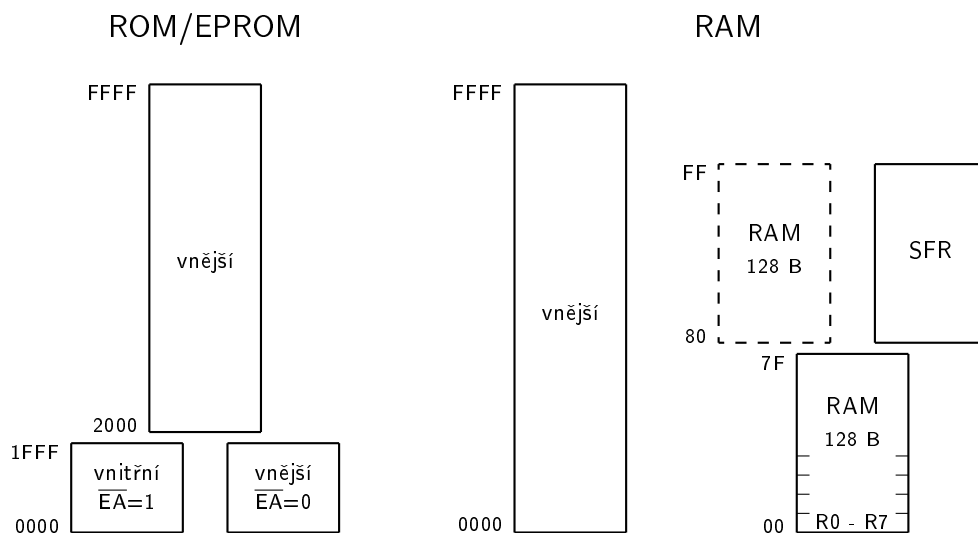
Základní typ mikropočítače 8051 zahrnuje vedle procesoru a vnitřních pamětí čtveřici I/O portů P0, P1, P2 a P3, dvojici univerzálních šestnáctibitových čítačů/časovačů T0 a T1 (další čítač T2 u typu 8052 je vybaven záchytným/přednastavovacím registrem) a obvody asynchronního sériového rozhraní.

Přerušovací systém mikropočítače 8051 reaguje na tři vnitřní (dva čítače/časovače a sériové rozhraní) a dva vnější zdroje přerušování, generuje pět vektorů, z nichž každý lze maskovat a definovat pro něj jednu ze dvou úrovní priority.



Obrázek 8.2: Struktura jednočipového mikropočítače 8051/8052

Procesor mikro počítače 8051 je harvardského typu, s oddělenou pamětí programu a dat (obr.8.3). Program může být umístěn ve vnitřní paměti o kapacitě 4 KB (8 KB u 8052), nebo ve vnější paměti. Jsou vyráběny modifikace s programovatelnou vnitřní pamětí programu typu EPROM (8751, 8752) a modifikace bez vnitřní paměti programu (8031, 8032). Vnitřní paměť programu lze nulovou úrovní na vývodu \overline{EA} odpojit, vnější paměť programu má rozsah 64 KB.



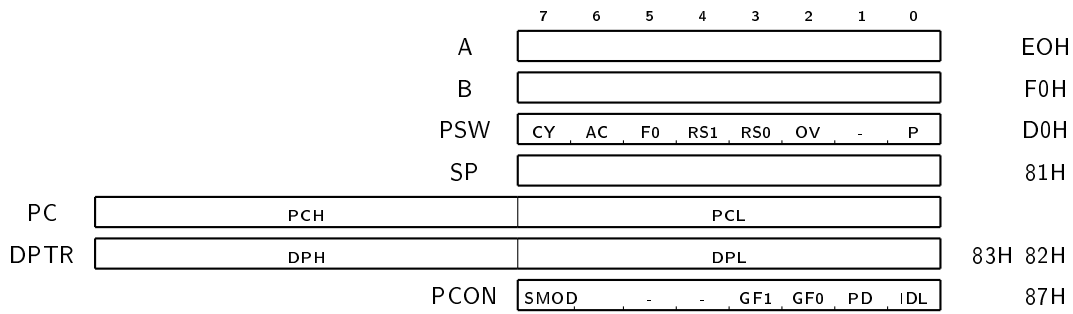
Obrázek 8.3: Paměťové prostory jednočipového mikro počítače 8051/8052

Pro data má procesor k dispozici dva paměťové prostory - vnitřní a vnější. Vnitřní datový prostor má rozsah 256 B, je využit pro přístup k vnitřní paměti RAM o kapacitě 128 B (256 B u 8052), a pro přístup k registrům procesoru a periférií. Ty jsou označovány jako funkční registry (SFR - Special Function Register, nadále budeme mluvit o SFR registrech) a jsou mapovány na adresy 80_H až FF_H . Přístup k vnitřní paměti RAM nad 128 B (u 8052 a řady odvozených typů) je od přístupu k registrům SFR odlišen adresací; přímou adresací se dostaneme k registrům SFR, nepřímou adresací získáme přístup k paměti RAM v rozsahu 80_H až FF_H .

Část vnitřní paměti RAM má specializované využití. Prvých 32 B paměti (adresy 00_H - $1F_H$) lze využít jako čtyři skupiny pracovních registrů R0 - R7. Výběr právě aktivní skupiny je řízen bity RS0 a RS1 SFR registru PSW, přepínání je užitečné při programování přerušovacích rutin a nezávislých procesů. Paměť RAM v rozsahu 20_H - $2F_H$ je adresovatelná po jednotlivých bitech (instrukcemi pro práci s bity), celkově zde máme k dispozici 128 jednobitových buněk (bitové adresy 00_H - $7F_H$). Zbývajících 128 bitových adres (80_H - FF_H) je využito pro přímý přístup k bitům SFR registrů na adresách 80_H , 90_H , ... $F0_H$. Na tyto adresy jsou mapovány některé vnitřní registry procesoru (A, B, PSW) a registry vnějších portů.

Vnější paměť dat má kapacitu 64 KB a je adresovatelná pouze nepřímo prostřednictvím šestnáctibitového SFR registru DPTR nebo prostřednictvím osmibitových registrů R0 a R1 (dovolují adresovat v rozsahu stránky 256 B). Procesory, které rozšiřují svou vnitřní paměť nad 256 B vnitřního paměťového prostoru (např. pro automobilní elektroniku určený Philips 80C592), překrývají touto rozšířenou vnitřní pamětí začátek vnějšího paměťového prostoru. Rozšířenou vnitřní paměť RAM lze adresovat pouze nepřímo, stejně jako vnější paměť dat.

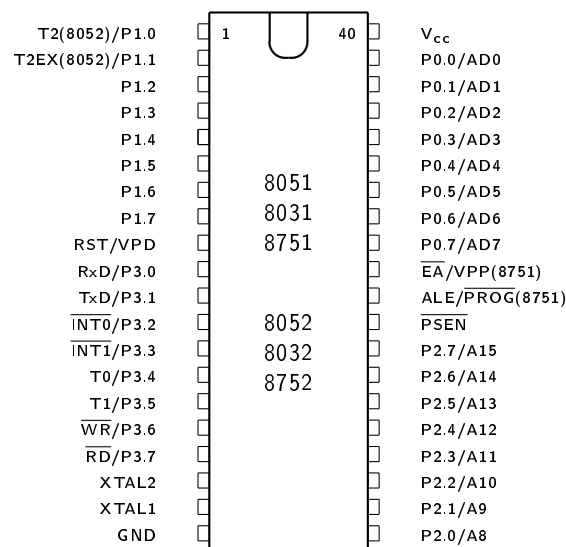
Přístup k vnější paměti programu a dat vyžaduje vytvoření vnější sběrnice. Tomuto účelu je nutné obětovat I/O porty P0, P2 a případně i dva bity portu P3. Nižší řady adresy a data sdílí port P0, multiplex je řízen signálem ALE (obr.8.6), vyšší řady adresy jsou předávány na portu P2. Přístup k vnější paměti programu je řízen signálem \overline{PSEN} , řízení přístupu k vnější paměti dat používá bity P3.7 (\overline{RD}) a P3.6 (\overline{WR}) portu P3.



Obrázek 8.4: Procesor jednočipového mikropočítače 8051/8052

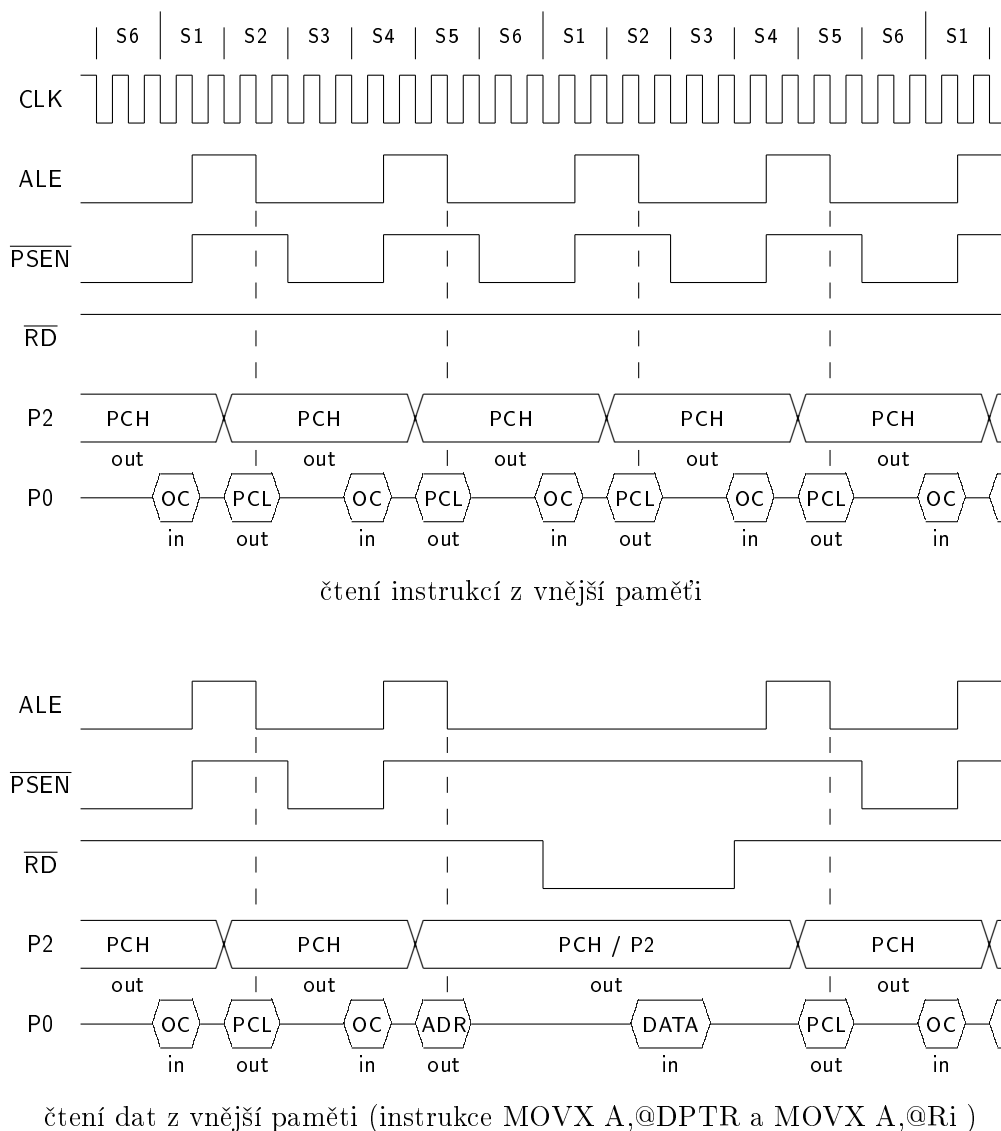
Struktura procesoru 8051 je poměrně jednoduchá (obr.8.4). Aritmeticko-logická jednotka pracuje s osmibitovým střadačem A (nebo ACC). Registr B je využíván pro násobení a dělení, jinak je k dispozici jako pomocný registr. Příznakový registr PSW ukládá příznaky přenosu (C - Carry) a pomocného přenosu (AC - Auxiliary Carry, přenos mezi čtveřicemi bitů při osmibitové operaci), příznak přeplnění (OV - Overflow) a příznak parity (P - Parity). Bity RS0 a RS1 dovolují volit skupinu pracovních registrů (Register Bank), příznak F0 a nepoužitý bit PSW.1 jsou k dispozici uživateli. Osmibitový registr SP adresuje vnitřní zásobník, na který jsou ukládány návratové adresy, a který využívají instrukce PUSH a POP. Šestnáctibitový programový čítač PC adresuje paměť programu, další šestnáctibitový registr DPTR zpřístupňuje vnější paměť dat. Pro nepřímou adresaci vnitřní paměti RAM a vnější datové paměti v rozsahu stránky 256 B lze použít pracovní registry R0 a R1.

Jednočipové mikropočítače 8051 (8052) byly vyráběny technologií HMOS, v současné době se používá technologie CHMOS a procesory jsou značené jako 80C51 (87C51, 80C31, 80C52, 87C52, 80C32). Základní typy pracují s krystalem do 12 MHz, některé modifikace dovolují až 40 MHz. CHMOS varianty dokáží pracovat v rozsahu napájecích napětí 2 - 6 V, jejich typická spotřeba při napětí 5 V je 12 mA pro krystal 12 MHz. U CHMOS verzí lze programově pozastavit procesor (Idle Mode - nastavením bitu IDL v SFR registru PCON) nebo dokonce vypnout hodinový generátor (nastavením bitu PD v SFR registru PCON). Procesory jsou pouzřeny do pouzder DIP-40 (obr.8.5), alternativně jsou používána i pouzdra PLCC-44 a QFP-44.



Obrázek 8.5: Rozložení vývodů jednočipového mikropočítače 8051/8052

Strojový cyklus procesoru (obr.8.6) je tvořen posloupností šesti stavů (S1 - S6), v každém stavu procesor setrvává po dobu dvou period hodinového signálu. Pro krystal 12 MHz má tedy strojový cyklus délku 1 μ s.



Obrázek 8.6: Instrukční cyklus a signály vnější sběrnice jednočipového mikropočítače 8051/8052

Jednomu strojovému cyklu odpovídají dva přístupy do paměti programu. Pokud procesor druhý z bytů nepotřebuje, ignoruje jej a jeho čtení opakuje v následujícím strojovém cyklu. Instrukce, které pro své provedení potřebují více než jeden strojový cyklus, takto opakuji čtení až čtyřikrát. Doba pro přístup do paměti programu je poměrně krátká (500 ns pro krystal 12 MHz) a procesor neumožňuje její prodloužení (bez snížení hodinového kmitočtu). Tento fakt je nutné respektovat při volbě vnější paměti programu, multiplexu sběrnice a adresacích obvodů. Výjimkou z naprosto pravidelného režimu práce procesoru jsou instrukce pro práci s vnější pamětí, které věnují celý strojový cyklus na přístup k (pomalejší) vnější paměti nebo periférii.

Instrukční soubor procesoru 8051 zahrnuje 111 instrukcí, pro adresaci je využíváno pět adresacích módů. Přehled instrukcí uvádí tab.8.1. V tabulce jsou použity následující symboly:

Rn	registr R0 - R7 aktivní skupiny
direct	osmibitová adresa vnitřního datového prostoru
@Ri	nepřímý přístup do datového prostoru přes registr R0 nebo R1
@DPTR	nepřímý přístup do vnějšího datového prostoru přes registr DPTR
#data	osmibitová konstanta
#data16	šestnáctibitová konstanta
addr11	jedenáctibitová adresa programové paměti
addr16	šestnáctibitová adresa programové paměti
rel8	osmibitová hodnota (se znaménkem) pro relativní adresaci
bit	osmibitová adresa bitu
Stk	vrchol zásobníku

Instrukční soubor dovoluje pracovat s vnitřní pamětí dat a s registry SFR, k dispozici je řada operací pro práci s jednotlivými bity (lze pracovat s jednotlivými bity v zóně paměti na adresách 20_H až $2F_H$ a s jednotlivými bity registrů s adresami 80_H , 88_H , 90_H , ..., $F8_H$. Přístup k vnější paměti dat je poněkud komplikovaný, opírá se o jediný adresační registr DPTR. Některé modernější modifikace mikropočítače 8051 (Siemens 80C517, Dallas DS80C320) počet registrů DPTR zvětšují.

8051 - Aritmetické operace

Kód	Operandy	Operace	Délka	Cykly	Příznaky C OV AC
ADD	A,Rn	$A \leftarrow A + Rn$	1	1	x x x
	A,direct	$A \leftarrow A + direct$	2	1	x x x
	A,@Ri	$A \leftarrow A + @Ri$	1	1	x x x
	A,#data	$A \leftarrow A + #data$	2	1	x x x
ADDC	A,Rn	$A \leftarrow A + Rn + C$	1	1	x x x
	A,direct	$A \leftarrow A + direct + C$	2	1	x x x
	A,@Ri	$A \leftarrow A + @Ri + C$	1	1	x x x
	A,#data	$A \leftarrow A + #data + C$	2	1	x x x
SUBB	A,Rn	$A \leftarrow A - Rn - C$	1	1	x x x
	A,direct	$A \leftarrow A - direct - C$	2	1	x x x
	A,@Ri	$A \leftarrow A - @Ri - C$	1	1	x x x
	A,#data	$A \leftarrow A - #data - C$	2	1	x x x
INC	A	$A \leftarrow A + 1$	1	1	
	Rn	$Rn \leftarrow Rn + 1$	1	1	
	direct	$direct \leftarrow direct + 1$	2	1	
	@Ri	$@Ri \leftarrow @Ri + 1$	1	1	
DEC	A	$A \leftarrow A - 1$	1	1	
	Rn	$Rn \leftarrow Rn - 1$	1	1	
	direct	$direct \leftarrow direct - 1$	2	1	
	@Ri	$@Ri \leftarrow @Ri - 1$	1	1	
INC	DPTR	$DPTR \leftarrow DPTR + 1$	1	2	
MUL	AB	$AB \leftarrow A * B$	1	4	0 x
DIV	AB	$A, B \leftarrow A / B$	1	4	0 x
DA	AB	dekadická korekce	1	1	x

Registry SFR

Pro praktické programování jednočipového mikropočítače jsou nezbytností podrobné informace o funkci řídicích a datových registrů vlastního procesoru a jeho periférií. Abychom zde vyšli

8051 - Logické operace

Kód	Operandy	Operace	Délka	Cykly	Příznaky C OV AC
ANL	A,Rn	$A \leftarrow A \wedge Rn$	1	1	
	A,direct	$A \leftarrow A \wedge direct$	2	1	
	A,@Ri	$A \leftarrow A \wedge @Ri$	1	1	
	A,#data	$A \leftarrow A \wedge \#data$	2	1	
	direct,A	$direct \leftarrow direct \wedge A$	2	1	
	direct,#data	$direct \leftarrow direct \wedge \#data$	3	2	
ORL	A,Rn	$A \leftarrow A \vee Rn$	1	1	
	A,direct	$A \leftarrow A \vee direct$	2	1	
	A,@Ri	$A \leftarrow A \vee @Ri$	1	1	
	A,#data	$A \leftarrow A \vee \#data$	2	1	
	direct,A	$direct \leftarrow direct \vee A$	2	1	
	direct,#data	$direct \leftarrow direct \vee \#data$	3	2	
XRL	A,Rn	$A \leftarrow A \oplus Rn$	1	1	
	A,direct	$A \leftarrow A \oplus direct$	2	1	
	A,@Ri	$A \leftarrow A \oplus @Ri$	1	1	
	A,#data	$A \leftarrow A \oplus \#data$	2	1	
	direct,A	$direct \leftarrow direct \oplus A$	2	1	
	direct,#data	$direct \leftarrow direct \oplus \#data$	3	2	

8051 - Logické operace (pokračování)

Kód	Operandy	Operace	Délka	Cykly	Příznaky C OV AC
CLR	A	$A \leftarrow 0$	1	1	
CPL	A	$A \leftarrow \overline{A}$	1	1	
RL	A	rotace A vlevo	1	1	
RLC	A	rotace A a C vlevo	1	1	x
RR	A	rotace A vpravo	1	1	
RRC	A	rotace A a C vpravo	1	1	x
SWAP	A	prohození čtveřic bitů	1	1	

8051 - Přesuny

Kód	Operandy	Operace	Délka	Cykly	Příznaky C OV AC
MOV	A,Rn	$A \leftarrow Rn$	1	1	
	A,direct	$A \leftarrow direct$	2	1	
	A,@Ri	$A \leftarrow @Ri$	1	1	
	A,#data	$A \leftarrow \#data$	2	1	
	Rn,A	$Rn \leftarrow A$	1	1	
	Rn,direct	$Rn \leftarrow direct$	2	2	
	Rn,#data	$Rn \leftarrow \#data$	2	1	
	direct,A	$direct \leftarrow A$	2	1	
	direct,Rn	$direct \leftarrow Rn$	2	2	
	direct,direct	$direct \leftarrow direct$	3	2	
	direct,@Ri	$direct \leftarrow @Ri$	2	2	
	direct,#data	$direct \leftarrow \#data$	3	2	
	@Ri,A	$@Ri \leftarrow A$	1	1	
	@Ri,direct	$@Ri \leftarrow direct$	2	2	
	@Ri,#data	$@Ri \leftarrow \#data$	2	1	
MOV	DPTR,#data	$DPTR \leftarrow \#data16$	3	2	
MOVC	A,@A+DPTR	$A \leftarrow @A + DPTR$	1	2	
	A,@A+PC	$A \leftarrow @A + PC$	1	2	
MOVX	A,@Ri	$A \leftarrow @Ri$	1	2	
	A,@DPTR	$A \leftarrow @DPTR$	1	2	
	@Ri,A	$@Ri \leftarrow A$	1	2	
	@DPTR,A	$@DPTR \leftarrow A$	1	2	
PUSH	direct	$SP \leftarrow SP + 1,$ $Stk \leftarrow direct$	2	2	
POP	direct	$direct \leftarrow Stk,$ $SP \leftarrow SP - 1$	2	2	
XCH	A,Rn	$A \leftrightarrow Rn$	1	1	
	A,direct	$A \leftrightarrow direct$	2	1	
	A,@Ri	$A \leftrightarrow @Ri$	1	1	
XCHD	A,@Ri	$A_{3-0} \leftrightarrow @Ri_{3-0}$	1	1	

8051 - Bitové instrukce

Kód	Operandy	Operace	Délka	Cykly	Příznaky C OV AC
CLR	C	$C \leftarrow 0$	1	1	0
	bit	$bit \leftarrow 0$	2	1	
SETB	C	$C \leftarrow 1$	1	1	1
	bit	$bit \leftarrow 1$	2	1	
CPL	C	$C \leftarrow \overline{C}$	1	1	x
	bit	$bit \leftarrow \overline{bit}$	2	1	
ANL	C,bit	$C \leftarrow C \wedge bit$	2	2	x
	C,/bit	$C \leftarrow C \wedge \overline{bit}$	2	2	x
ORL	C,bit	$C \leftarrow C \vee bit$	2	2	x
	C,/bit	$C \leftarrow C \vee \overline{bit}$	2	2	x
MOV	C,bit	$C \leftarrow bit$	2	1	x
	bit,C	$bit \leftarrow C$	2	2	
JC	rel8	skok při $C = 1$	2	2	
JNC	rel8	skok při $C = 0$	2	2	
JB	bit,rel8	skok při $bit = 1$	3	2	
JNB	bit,rel8	skok při $bit = 0$	3	2	
JBC	bit,rel8	skok při $bit = 1, bit \leftarrow 0$	3	2	

8051 - Řízení

Kód	Operandy	Operace	Délka	Cykly	Příznaky C OV AC
ACALL	addr11	krátké volání podprogramu	2	2	
LCALL	addr16	volání podprogramu	3	2	
RET		návrat z podprogramu	1	2	
RETI		návrat z přerušovací rutiny	1	2	
AJMP	addr11	krátký nepodmíněný skok	2	2	
LJMP	addr16	nepodmíněný skok	3	2	
SJMP	rel8	relativní nepodmíněný skok	2	2	
JMP	@A+DPTR	nepřímý skok	1	2	
JZ	rel8	skok při $A = 0$	2	2	
JNZ	rel8	skok při $A \neq 0$	2	2	
CJNE	A,direct,rel8	skok při $A \neq direct$	3	2	x
	A,#data,rel8	skok při $A \neq \#data$	3	2	x
	Rn,#data,rel8	skok při $Rn \neq \#data$	3	2	x
	@Ri,#data,rel8	skok při $@Ri \neq \#data$	3	2	x
DJNZ	Rn,rel8	$Rn \leftarrow Rn - 1,$ skok při $Rn \neq 0$	2	2	
DJNZ	direct,rel8	$direct \leftarrow direct - 1,$ skok při $direct \neq 0$	3	2	
NOP		prázdná operace	1	1	

Tabulka 8.1: Instrukční soubor mikropočítače 8051

vstříc čtenáři tohoto textu, a neodkazovali ho v případě u nás jednoznačně nejpoužívanějšího mikropočítače na firemní literaturu, uvedeme si alespoň stručně přehled SFR registrů (kromě registrů procesoru ACC, B, SP a DPTR), význam jejich bitů a hodnot, na které jsou tyto bity nastaveny při resetu. Dále uvedené informace odpovídají procesoru 80C52 (80C32, 87C52), tedy vylepšené verzi procesoru. Příznaky, které na procesorech 80C51 (80C31, 87C51) nenajdeme, jsou označeny závorkami u hodnot po resetu.

PSW - Program Status Word

	0	0	0	0	0	0	0
D0 _H	CY	AC	F0	RS1	RS0	OV	P

Příznaky CY, AC a OV informují o výsledku aritmeticko-logických operací, bit P doplňuje počet jednotkových bitů akumulátoru na sudý. Bity RS1 a RS0 volíme jednu ze čtyř skupin pracovních registrů R0 - R7. Bity F0 a PSW.1 lze volně používat jako jednobitové příznaky.

PCON - Power Control Register

	0	x	x	x	0	0	0	0
87 _H	SMOD	-	-	-	GF1	GF0	PD	IDL

Bit SMOD při nastavení (SMOD=1) dovoluje zdvojnásobit přenosovou rychlost sériového přenosu v režimech 1, 2 a 3. Bity PD a IDL dovolují snížit spotřebu mikropočítačového systému. Bit IDL blokuje přívod hodin do procesoru (periférie pracují normálně), bit PD zastavuje hodinový generátor. Bity GF1 a GF0 jsou volně použitelné příznaky, bity PCON.4, PCON.5 a PCON.6 jsou rezervou pro další rozšiřování.

P0 - Parallel Port 0

	1	1	1	1	1	1	1	1
80 _H	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0

Osmibitový port P0 je alternativně využíván jako multiplexovaná sběrnice adresy (nižší řády) a dat. V této funkci má třístavový výstup. Ve funkci paralelního portu dovolí sepnout výstup na nulu, ale nemá na rozdíl od portů P1, P2 a P3 vestavěné zvyšovací odpory.

P1 - Parallel Port 1

	1	1	1	1	1	1	(1)	(1)
90 _H	-	-	-	-	-	-	T2EX	T2

Osmibitový port P1 je alternativně využíván pro vyvedení signálů čítače/časovače TIMER 2 (pouze 8052). Je vybaven vnitřními zvyšovacími odpory (ve skutečnosti kanály FET), přechod z 0 do 1 je posílen. Porty P0, P1, P2 a P3 nepotřebují zvláštní konfigurační registr, který by definoval směr přenosu, neboť jejich výstupy pracují jako otevřené kolektory (výjimkou je P0 v režimu vnější sběrnice). Vývod portu může být využit jako vstup pouze tehdy, je-li do příslušného bitu výstupního registru zapsána jednička. Některé instrukce nutně vyžadují odlišení hodnoty generované na vývod registrem portu od hodnoty skutečně přečtené na vývodu díky funkci otevřeného kolektoru. Takové instrukce (Read-Modify-Write) čtou přímo výstupy vnitřního registru portu, ostatní instrukce čtou hodnotu na vývodu.

P2 - Parallel Port 2

	1	1	1	1	1	1	1	1
A0 _H	A15	A14	A13	A12	A11	A10	A9	A8

Osmibitový paralelní port P2 je alternativně využíván pro vyšší řády adresy vnější sběrnice.

P3 - Parallel Port 3

	1	1	1	1	1	1	1
B0 _H	RD	WR	T1	T0	INT1	INT0	RxD

Osmibitový paralelní port P3 je alternativně využíván pro vyvedení signálů časovačů/čítačů, sériového rozhraní a jako vstupy pro vnější přerušení. Port podporuje i přístup k vnější paměti dat (signály RD a WR).

TCON - Timer/Counter Control Register

	0	0	0	0	0	0	0
88 _H	TF1	TR1	TF0	TR0	IE1	IT1	IE0

Jednotková hodnota příznaku TF1 indikuje přetečení časovače/čítače TIMER 1, příznak je nulován hardwarem při vstupu do obslužné rutiny. Bitem TR1 můžeme povolit (TR1=1) nebo zakázat (TR1=0) vstup čítaných pulsů a zastavit tak nebo spustit programově časovač/čítač TIMER 1. Bity TF0 a TR0 mají tutéž funkci pro časovač/čítač TIMER 0. Příznak IE1 indikuje žádost o přerušení vnějším signálem IE1 a je automaticky nulován hardwarem při vstupu do obslužné rutiny. Bitem IT1 můžeme volit mezi reakcí přerušovací logiky na sestupnou hranu (IT1=1) nebo nulovou úroveň (IT=0) vnějšího signálu INT1. Bity IE0 a IT0 mají tutéž funkci pro vnější signál INT0.

TMOD - Timer/Counter Mode Control Register

	0	0	0	0	0	0	0
89 _H	GATE	C/T	M1	M0	GATE	C/T	M1

Registr TMOD dovoluje nakonfigurovat časovače/čítače TIMER 1 a TIMER 0 do některého z několika pracovních režimů. Bity TMOD.0 až TMOD.3 přísluší časovači/čítači TIMER 0, bity TMOD.4 až TMOD.7 přísluší časovači/čítači TIMER 1. Bitem C/T volíme mezi funkcí časovače (C/T=0), kdy je zdrojem čítaných pulsů signál hodinového generátoru vydělený dvanácti, a funkcí čítače (C/T=1), kdy jsou čítány pulsy na příslušném vnějším vstupu (T0 nebo T1). Bit GATE při nastavení (GATE=1) povoluje hradlování vstupního signálu příslušného čítače/časovače vnějším signálem na vstupu INT0 nebo INT1. Bity MOD1 a MOD0 definují funkci čítače/časovače TIMER 1 a TIMER 0 takto:

MOD1	MOD0	režim	TIMER 0	TIMER 1
0	0	0	13-bitový	13-bitový
0	1	1	16-bitový	16-bitový
1	0	2	8-bitový (Reload)	8-bitový (Reload)
1	1	3	(2x) 8-bitový	-

V režimu 3 může pracovat pouze TIMER 0, a to jako dva osmibitové časovače, nebo jako jeden osmibitový čítač.

TL0, TH0 - Timer 0

	0	0	0	0	0	0	0
8C _H	c15	c14	c13	c12	c11	c10	c9

	0	0	0	0	0	0	0
8A _H	c7	c6	c5	c4	c3	c2	c0

TL0 a TH0 jsou vlastní čítače časovače/čítače TIMER 0, programem je lze číst i nastavovat.

TL1, TH1 - Timer 1

	0	0	0	0	0	0	0	0
8D _H	c15	c14	c13	c12	c11	c10	c9	c8
	0	0	0	0	0	0	0	0
8B _H	c7	c6	c5	c4	c3	c2	c1	c0

TL1 a TH1 jsou vlastní čítače časovače/čítače TIMER 1, programem je lze číst i nastavovat.

T2CON - Timer/Counter 2 Control Register

	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
C8 _H	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	C/RL2

Šestnáctibitový čítač TIMER 2 mikropočítače 8052 je doplněn pomocným RCAP registrem, který lze použít jako záchytný registr (funkce Capture) nebo registr pro automatické nastavení počáteční hodnoty (funkce Reload). Jeho konfigurace a řízení je zajišťována registrem T2CON. Jednotková hodnota příznaku TF2 indikuje přetečení časovače/čítače TIMER 2, příznak musí být vynulován programem. Příznak EXF2 indikuje zápis do záchytného registru nebo žádost o přerušeni vyvolanou vnějším signálem T2EX, musí být nulován programem. Bit RCLK (RCLK=1) dovoluje využít signálu přeplnění čítače TIMER 2 jako hodinového signálu pro přijímač sériového rozhraní (režimy 1 a 3). Bit TCLK (TCLK=1) dovoluje využít signálu přeplnění čítače TIMER 2 jako hodinového signálu pro vysílač sériového rozhraní. Bitem EXEN2 lze povolit zápis do záchytného registru nebo nastavení čítače/časovače TIMER 2 sestupnou hranou signálu T2EX (pokud není časovač použit jako generátor přenosové rychlosti). Bitem TR2 můžeme povolit (TR2=1) nebo zakázat (TR2=0) vstup čítaných pulsů a časovač/čítač TIMER 2 tak programově ovládat. Bit C/T2 volí mezi funkcí časovače (C/T2=0) a čítače (C/T2=1). Konečně, bit CP/RL2 volí mezi použitím záchytného registru (CP/RL2=1), pokud to dovoluje bit EXEN2. Druhou možností (CP/RL2=0) je automatické nastavení časovače/čítače při přeplnění a (pokud to dovoluje bit EXEN2 a čítač není použit pro generování hodin sériového rozhraní) při sestupné hraně signálu T2EX.

TL2, TH2 - Timer 2

	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
CD _H	c15	c14	c13	c12	c11	c10	c9	c8
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
CC _H	c7	c6	c5	c4	c3	c2	c1	c0

TL2 a TH2 jsou vlastní čítače časovače/čítače TIMER 2, programem je lze číst i nastavovat.

RCAP2L, RCAP2H - Reload/Capture Register 2

	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
CB _H	r15	r14	r13	r12	r11	r10	r9	r8
	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
CA _H	r7	r6	r5	r4	r3	r2	r1	r0

Registry RCAP2L a RCAP2H jsou osmibitové části šestnáctibitového registru. Slouží pro nastavení čítače při jeho přetečení (nebo pro nastavení vnějším signálem). Alternativní funkcí je použití registru jako registru záchytného (Capture).

SCON - Serial Port Control Register

	0	0	0	0	0	0	0	0
98 _H	SM0	SM1	SM2	REN	TB8	RB8	TI	RI

Bity SM0 a SM1 definují režim činnosti sériového rozhraní podle následující tabulky:

SM0	SM1	Režim	Funkce	Přenosová rychlost
0	0	0	posuvný registr	$f_{osc}/12$
0	1	1	8-bitový UART	programovatelná
1	0	2	9-bitový UART	$f_{osc}/64$ nebo $f_{osc}/32$
1	1	3	9-bitový UART	programovatelná

Bit SM2 dovoluje v režimech 2 a 3, které jsou určené pro multiprocesorovou komunikaci, jemněji ovládat přerušovací systém. Jeho nastavení (SM2=1) zakazuje přerušování při příjmu znaků, jejichž devátý bit je nulový. Pouze znaky s nastaveným devátým bitem vyvolají obsluhu (znaky s nastaveným devátým bitem se používají pro selektivní, skupinovou nebo univerzální adresaci, jako funkční kód, ap.). Na sériové sběrnici pak může spolupracovat více mikropočítačů bez toho, že by značná část jejich výpočetní kapacity byla vyplývána na příjem irelevantních dat. V režimu 1 nastavený bit (SM2=1) dovolí generovat přerušování přijímače až po přijatém stop bitu (což je vlastně devátý jednotkový bit). Bit TB8 určuje hodnotu vysílaného devátého bitu v režimech 2 a 3, bit RB8 ukládá hodnotu přijatého devátého bitu. Bitem REN lze programově blokovat (REN=0) vstup přijímače. Konečně, bit TI informuje o odeslání znaku (TI=1), bit RI informuje o příjmu znaku (RI=1). Oba příznaky je nutné nulovat programem.

SBUF - Serial Data Buffer

	x	x	x	x	x	x	x
99 _H	s7	s6	s5	s4	s3	s2	s0

Vyrovňovací paměť pro vysílač a přijímač sériového rozhraní.

Přenosová rychlost sériového rozhraní v režimu 0 je dvanáctinou hodinového kmitočtu procesoru, v režimu 2 lze volit mezi čtyřiašedesátinou (SMOD=0) a dvaatřicetinou (SMOD=1). V režimech 1 a 3 lze jako generátor přenosové rychlosti použít časovač TIMER 1 nebo TIMER 2 (pouze u 8052) v režimu s automatickým nastavením (Reload). Pro přenosovou rychlost platí:

TIMER 1:

$$BaudRate = \frac{(SMOD + 1) * OSC}{32 * 16 * [256 - TH1]}$$

TIMER 2:

$$BaudRate = \frac{OSC}{32 * [62536 - RCAP2L]}$$

IE - Interrupt Enable Register

	0	x	(0)	0	0	0	0	0
A8 _H	EA	-	ET2	ES	ET1	EX1	ET0	EX0

Nastavením bitu EA (EA=1) umožňujeme funkci přerušovacího systému, bity IE.0 až IE.5 při nastavení (IE.i=1) povolují přerušování při přepnutí nebo zápisu do záchytného registru časovače/čítače TIMER 2 (TF2) nebo přerušování od vnějšího signálu (EXF2, pouze u 8052), přerušování generované sériovým řadičem (RI & TI), přerušování při přepnutí časovače/čítače TIMER 1 (TF1), přerušování od vnějšího signálu INT1 (IE1), přerušování při přepnutí časovače/čítače TIMER 0 (TF0) a přerušování od vnějšího signálu INT0 (IE0).

Pro každý z uvedených typů přerušování je dána adresa, na které musí začínat příslušná přerušovací rutina. Tyto adresy pro jednotlivé třídy přerušování jsou:

Zdroj přerušení	Příznak	Adresa
IE0	EX0	0003 _H
TF0	ET0	000B _H
IE1	EX1	0013 _H
TF1	ET1	001B _H
RI & TI	ES	0023 _H
TF2 & EXF2	ET2	002B _H

IP - Interrupt Priority Register

	x	x	(0)	0	0	0	0	0
B8 _H	-	-	PT2	PS	PT1	PX1	PT0	PX0

Příznaky v registru IP definují pro jednotlivé typy přerušení (jsou uvedeny ve stejném pořadí jako v registru IE) vyšší (IP.i=1) nebo nižší (IP.i=0) prioritu obsluhy.

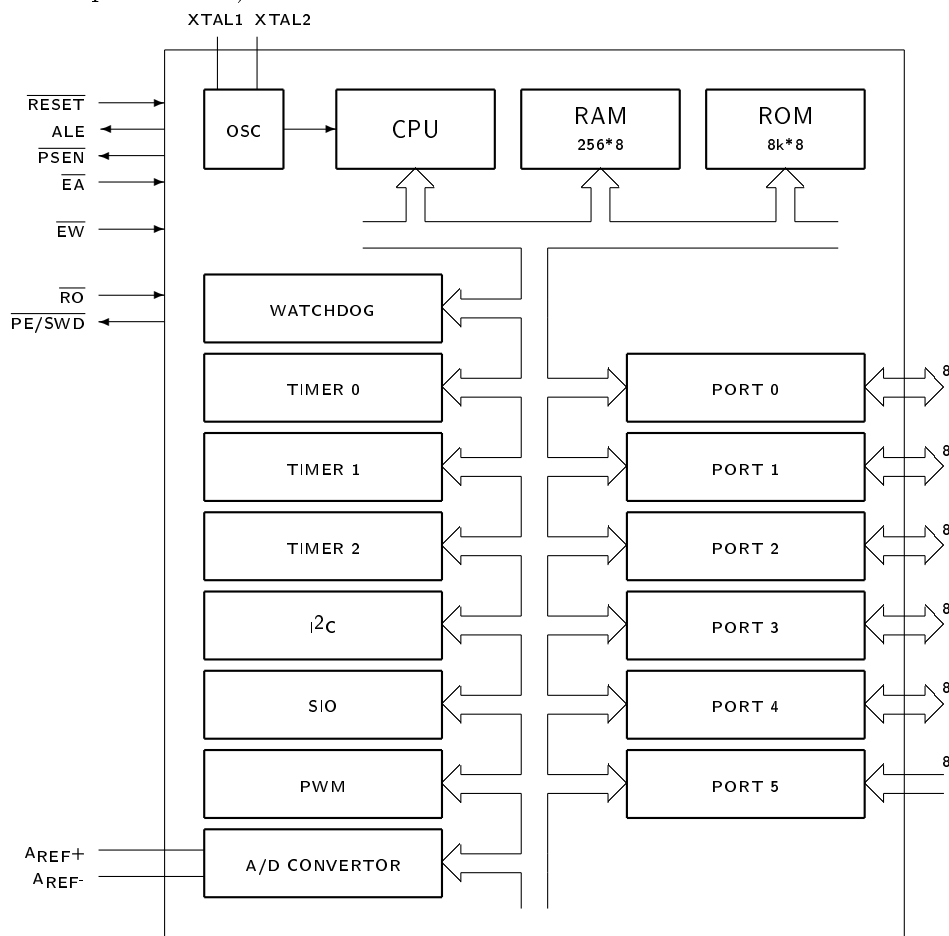
8.2 Philips 80C552

Mikroočítač 8051 [5] se stal základem pro celou řadu dalších modifikací, které obohacují jeho soubor periférií. Patří sem vedle dalších produktů firmy Intel hlavně mikroočítače Philips a Siemens. Přehled vybraných mikroočítačů Philips spolu s jejich základními charakteristikami uvádí následující tabulka. Mikroočítače Siemens si uvedeme později.

Jednočipové mikroočítače Philips

Typ	ROM	RAM	I/O	A/D	PWM		pouzdro
83CL51	4k	128	32			1.6-6V	DIL40
83C451	4k	128	52/56				DIL64,PLCC68
83C552	8k	256	48	8*10	2	I ² C	PLCC68,QFP80
83C592	16k	512	48	8*10	2	CAN	PLCC68
83C752	2k	64	21	1*8	1	I ² C	DIL28,PLCC28

Nejúspěšnějším prvkem z uvedeného výčtu je jednoznačně mikroočítač 80C552 (bez vnitřní paměti ROM), 83C552 (s vnitřní pamětí ROM o kapacitě 8 kB) a 87C552 (s vnitřní pamětí EPROM o kapacitě 8 kB).



Obrázek 8.7: Struktura jednočipového mikroočítače PCB 80C552

Mikroočítač PCB80C552 je proti základnímu typu 8051 doplněn o řadu obvodů rozhraní (obr.8.7). Výraznou změnou je doplnění dvou paralelních bran P4 a P5, brána P4 je alternativně

využívána pro vstupy záchytných registrů a pro výstupy komparátorů připojených k čítači T2, brána P5 je použitelná pouze jako vstupní a jsou na ni připojeny vstupy A/D převodníku.

Volně běžící šestnáctibitový čítač/časovač T2 je řízen signálem sběrnice ($f_{osc}/12$) nebo vnějším signálem na vývodu T2. Vstupní signál může být dále vydělen předřazeným programovatelným děličem dvěma, čtyřmi nebo osmi. Časovač je možné nulovat hranou vnějšího signálu RT2 (pokud to dovolí bit T2ER v registru TM2CON) a zajistit tak jeho synchronizaci s prostředím. Přetečení čítače nebo jeho poloviny lze indikovat a vyvolat jím přerušení. Čítač je vybaven čtyřmi záchytnými (CT0 - CT3) registry, zápis do nich je řízen signály CT3I - CT0I, v registru CTCON zadáváme zápis při náběžné a/nebo sestupné hraně jednotlivých signálů. Tři komparační registry CM0 - CM2 řídí osm výstupních obvodů, jejich výstupy jsou vyvedeny na bránu P4.

Osmibitový čítač označený jako T3 je součástí dohlížecího obvodu Watchdog. Jeho vstupem je signál sběrnice ($f_{osc}/12$) vydělený předřazeným jedenáctibitovým děličem. Čítač T3 čítá nahoru a při přetečení vyvolá reset. Čítač je možné volně číst, pro zápis nové hodnoty je nejdříve nutné nastavit bit WLE (Watchdog Load Enable - PCON.4). Obvod Watchdog je aktivován úrovní LOW na vývodu \overline{EW} (Enable Watchdog), programově ho nelze vypnout.

Dalším čítačem ve struktuře mikropočítače 80C552 je osmibitový čítač modulo 255 obvodu PWM. Čítá kmitočet oscilátoru dělený dvěma (6 MHz pro krystal 12 MHz) a předřazeným děličem PWMP, jehož koeficient je možné nastavit v rozsahu 1 - 256. Dvěma výstupním kanálům obvodu $\overline{PWM0}$ a $\overline{PWM1}$ přísluší dva komparační registry PWM0 a PWM1. Poměrná doba aktivní fáze (úroveň LOW) výstupního signálu odpovídá hodnotě v příslušném komparačním registru.

Desetibitový analogově-číslíkový převodník je poměrně pomalý, pro jeden převod potřebuje 50 cyklů sběrnice (50 μs pro krystal 12 MHz). Na vstup má připojen osmikanálový multiplexor, vstupy multiplexoru jsou k dispozici na bráně P5.

Přerušovací systém mikropočítače 80C552 reaguje na patnáct příčin přerušení. Pět z nich odpovídá přerušením mikropočítače 8051 ($\overline{INT0}$, $\overline{INT1}$, IT0, IT1 a SIO0). Přidaná přerušení odpovídají zápisům do záchytných registrů CT0 až CT3, porovnáním s komparačními registry CM0 až CM2, přetečení šestnáctibitového čítače T2 a/nebo jeho osmibitové části. Jedno přerušení je generováno při ukončení A/D převodu, další je svázáno s obvodem rozhraní sériové sběrnice I²C. Přerušovací systém se strukturou neliší od přerušovacího systému mikropočítače 8051, každé žádosti lze nezávisle přiřadit jednu ze dvou priorit.

Mikropočítač lze převést nastavením odpovídajícího bitu v registru PCON do režimu sníženého příkonu (Idle Mode), kdy je procesor zastaven a čeká na přerušení nebo signál RESET. Proti zhruba 30 mA spotřeby v běžném provozu (pro krystal 12 MHz) je spotřeba snižuje na 7 mA. Při vypnutí oscilátoru (Power-Down Mode) se příkon snižuje až na 50 μA , mikropočítač však pouze uchová data ve vnitřní paměti RAM a lze ho uvést do běhu pouze resetem.

Mikropočítač 80C552 je vyráběn v řadě variant pro komerční a průmyslový rozsah teplot (-40 až +85°C), pro krystal 12 MHz a 16 MHz. Má proti mikropočítači 8051 vyšší počet vývodů a je pouzdřen do pouzder PLCC-68 a QFP-80. Rozložení vývodů mikropočítače v pouzdře PLCC-68 uvádí obr.8.8.

registru CM0 (pokud to povoluje příslušný bit SP4i registru STE) a nulovány při shodě mezi hodnotou čítače T2 a komparačního registru CM1 (pokud to dovoluje příslušný bit RP4i registru RTE). Výstupy CMT1 a CMT0 přísluší klopným obvodům CMT1 a CMT0, které mění svůj stav při shodě mezi hodnotou čítače T2 a komparačního registru CM2 (pokud to dovoluje bit TG47 a TG46 registru STE).

P5 - Parallel Port 5

	x	x	x	x	x	x	x	x
C4 _H	adc7	adc6	adc5	adc4	adc3	adc2	adc1	adc0

Osmibitový paralelní port P5 je použitelný pouze jako osmibitový vstup a je alternativně využit pro vstupy A/D převodníku.

TM2CON - Timer 2 Control

	0	0	0	0	0	0	0	0
EA _H	T2IS1	T2IS0	T2ER	T2BO	T2P1	T2P0	T2MS1	T2MS0

TML2 - Timer Low 2

	0	0	0	0	0	0	0	0
EC _H	c7	c6	c5	c4	c3	c2	c1	c0

TMH2 - Timer High 2

	0	0	0	0	0	0	0	0
ED _H	c15	c14	c13	c12	c11	c10	c9	c8

Na adresách TMH2 a TML2 jsou dostupné vyšší a nižší řády čítače T2. Registr TM2CON čítač T2 ovládá. Nastavením bitu T2IS1 lze aktivovat přerušeni při přetečení šestnáctibitového čítače, nastavením bitu T2IS0 při přetečení jeho nižší slabiky. Nastavením bitu T2ER dovolujeme nulovat čítač T2 náběžnou hranou vnějšího signálu RT2. Bit T2BO při nastavení indikuje žádost o přerušeni vyvolané přetečením čítače. Bity T2P1 a T2P0 dovolují nastavit předřazený dělič čítače na dělicí poměr 1 (00), 2 (01), 4 (10) nebo 8 (11). Konečně, bity T2MS1 a T2MS0 dovolují odpojit vstup hodinových impulsů (00), použít signál oscilátoru vydělený dvanácti (01) nebo vnější signál přivedený na vývod T2 (11). Kombinace 10 signálů T2MS1, T2MS0 je určena pro testování.

CML0 - Compare Low 0

	0	0	0	0	0	0	0	0
A9 _H	c7	c6	c5	c4	c3	c2	c1	c0

CML1 - Compare Low 1

	0	0	0	0	0	0	0	0
AA _H	c7	c6	c5	c4	c3	c2	c1	c0

CML2 - Compare Low 2

	0	0	0	0	0	0	0	0
AB _H	c7	c6	c5	c4	c3	c2	c1	c0

CTL0 - Capture Low 0

	x	x	x	x	x	x	x	x
AC _H	c7	c6	c5	c4	c3	c2	c1	c0

CTL1 - Capture Low 1

	x	x	x	x	x	x	x	x
AD _H	c7	c6	c5	c4	c3	c2	c1	c0

CTL2 - Capture Low 2

	x	x	x	x	x	x	x	x
AE _H	c7	c6	c5	c4	c3	c2	c1	c0

CTL3 - Capture Low 3

	x	x	x	x	x	x	x	x
AF _H	c7	c6	c5	c4	c3	c2	c1	c0

CMH0 - Compare High 0

	0	0	0	0	0	0	0	0
C9 _H	c15	c14	c13	c12	c11	c10	c9	c8

CMH1 - Compare High 1

	0	0	0	0	0	0	0	0
CA _H	c15	c14	c13	c12	c11	c10	c9	c8

CMH2 - Compare High 2

	0	0	0	0	0	0	0	0
CB _H	c15	c14	c13	c12	c11	c10	c9	c8

CTH0 - Capture High 0

	x	x	x	x	x	x	x	x
CC _H	c15	c14	c13	c12	c11	c10	c9	c8

CTH1 - Capture High 1

	x	x	x	x	x	x	x	x
CD _H	c15	c14	c13	c12	c11	c10	c9	c8

CTH2 - Capture High 2

	x	x	x	x	x	x	x	x
CE _H	c15	c14	c13	c12	c11	c10	c9	c8

CTH3 - Capture High 3

	x	x	x	x	x	x	x	x
CF _H	c15	c14	c13	c12	c11	c10	c9	c8

Pod jmény CTH3 - CTH0 a CTL3 - CTL0 najdeme poloviny šestnáctibitových záchytných registrů, jména CMH2 - CMH0 a CML2 - CML0 označují komparační registry.

CTCON - Capture Control

	0	0	0	0	0	0	0	
EB _H	CTN3	CTP3	CTN2	CPT2	CTN1	CPT1	CTN0	CPT0

Register CTCON dovoluje pro každý záchytný registr nastavit citlivost na sestupnou (CTNi=1) a/nebo náběžnou (CTPi=1) hranu signálu na odpovídajícím vstupu.

STE - Set Enable Register

	1	1	0	0	0	0	0	
EE _H	TG47	TG46	SP45	SP44	SP43	SP42	SP41	SP40

Nastavením bitů SP45 - SP40 registru STE povolujeme nastavení klopných obvodů CMSR5 - CMSR0 (ovládají vývody CMSR5 - CMSR0 brány P4) při shodě čítače T2 a komparačního registru CM1. Bity TG47 a TG46 indikují příští hodnotu klopných obvodů CMT1 a CMT0 (ovládají vývody CMT1 a CMT0 brány P4) při shodě čítače T2 a komparačního registru CM2.

RTE - Reset/Togle Enable Register

	0	0	0	0	0	0	0	
EF _H	TP47	TP46	RP45	RP44	RP43	RP42	RP41	RP40

Nastavením bitů RP45 - RP40 registru RTE povolujeme nulování klopných obvodů CMSR5 - CMSR0 (ovládají vývody CMSR5 - CMSR0 brány P4) při shodě čítače T2 a komparačního registru CM1. Bity TP47 a TP46 povolují překlopení klopných obvodů CMT1 a CMT0 (ovládají vývody CMT1 a CMT0 brány P4) při shodě čítače T2 a komparačního registru CM2.

TM2IR - Timer 2 Interrupt Flag Register

	0	0	0	0	0	0	0	
C8 _H	T2OV	CMi2	CMi1	CMi0	CTi3	CTi2	CTi1	CTi0

Bit CTi3 - CTi0 registru TM2IR indikuje přerušení při přepisu hodnoty čítače T2 do záchytného registru CT3 - CT0 (přepis je vyžádán zvolenou hranou signálu CT3i - CT0i). Bit CMi2 - CMi0 indikuje přerušení při shodě čítače T2 s komparačním registrem CM2 - CM0. Konečně, bit T2OV indikuje přerušení při přetečení čítače T2 nebo jeho osmibitové části.

T3 - Timer 3

	0	0	0	0	0	0	0	
FF _H	c7	c6	c5	c4	c3	c2	c1	c0

Čítač T3 je součástí dohlížecího obvodu Watchdog, čítá nahoru. Zápis do něj je možný pouze po předchozím nastavení bitu WLE/PCON.4.

PWMP - PWM Prescaler

	0	0	0	0	0	0	0	
FE _H	p7	p6	p5	p4	p3	p2	p1	p0

PWM1 - PWM Register 1

	0	0	0	0	0	0	0	
FD _H	p7	p6	p5	p4	p3	p2	p1	p0

PWM0 - PWM Register 0

	0	0	0	0	0	0	0	0
FC _H	p7	p6	p5	p4	p3	p2	p1	p0

Registry PWM2, PWM1 a PWM0 jsou součástí generátoru signálu PWM. Hodnoty v registrech PWM0 a PWM1 určují poměr aktivní části periody (úroveň LOW) na vývodech $\overline{\text{PWM0}}$ a $\overline{\text{PWM1}}$. Registr PWMP dovoluje zadat koeficient předřazeného děliče.

S1CON - Serial 1 Control

	0	0	0	0	0	0	0	0
D8 _H	CR2	ENS1	STA	ST0	SI	AA	CR1	CR0

S1STA - Serial 1 Status

	1	1	1	1	1	0	0	0
D9 _H	SC4	SC3	SC2	SC1	SC0	0	0	0

S1DAT - Serial 1 Data

	0	0	0	0	0	0	0	0
DA _H	s7	s6	s5	s4	s3	s2	s1	s0

S1ADR - Serial 1 Address

	0	0	0	0	0	0	0	0
DB _H	a7	a6	a5	a4	a3	a2	a1	a0

Čtveřice registrů S1CON, S1STA, S1DAT a S1ADR slouží k ovládání obvodů sériové sběrnice I²C, rozhraní je označováno jako SIO1. Mikro počítač PCB80C552 může na sběrnici I²C pracovat jako řídicí (Master) nebo podřízený (Slave) a lze mu přidělit libovolnou adresu (s výjimkou adresy 00_H, která slouží pro funkci Broadcast). Připomeňme si, že adresy pro obvody ovládané po sběrnici I²C jsou určené výrobcem, obvody, u kterých se předpokládá, že jich bude více v jednom systému, mají některé bity adresy nastavitelné zapojením vývodů.

Registr S1DAT (SIO1 Data Register) je posuvný registr, do kterého vkládáme vysílaná data, a ze kterého čteme data přijímaná. Rozhraní nemá na rozdíl od asynchronního rozhraní SIO0 vyrovnávací paměti, ovládání obvodů SIO1 přerušovacími rutinami a možnost pozastavit vysílání hodinového signálu protistanicí (podržet signálu SCL na úrovni LOW) vyrovnávací paměti nevyžaduje. Do registru S1ADR (SIO1 Address Register) ukládáme adresu, na kterou bude rozhraní při příjmu reagovat (další takovou adresou je adresa 00_H, která slouží pro funkci Broadcast).

Registr S1CON (SIO1 Control Register) je řídicím registrem rozhraní I²C. Bit ENS1 (SIO1 Enable Bit) při nastavení povoluje činnost obvodů rozhraní I²C. Vodiče rozhraní jsou vyvedeny na vývody P1.6/SCL a P1.7/SDA, odpovídající budiče paralelní brány P1 musí být uvedeny do stavu HIGH. Bit STA (Start Flag) zadává požadavek na vyslání prvku START, bit STO (Stop Flag) požadavek na vyslání prvku STOP.

Bit SI (Serial Interrupt Flag) dovoluje reagovat na jeden z 25 stavů řadiče, indikovaných ve stavovém registru S1STA (SIO1 Status Register), podmínkou je povolení přerušování (EA=1 a ES1=1). Po dobu obsluhy je udržován vodič SCL na úrovni LOW. Bit AA (Assert Acknowledge Flag) zadává požadavek na vyslání potvrzení (prvek ACK) jako odpovědi na příjem adresy nebo znaku zprávy.

Konečně bity CR2, CR1 a CR0 zadávají přenosovou rychlost (základní kmitočet signálu SCL) v režimu Master. Konkrétní hodnoty přenosové rychlosti pro krystal 12 MHz uvádí tabulka, její poslední řádek (nastavení všech bitů CR2 - CR0) odpovídá řízení obvodu SIO1

čítačem T1 ve funkci generátoru přenosové rychlosti:

CR2	CR1	CR0	rychlost	děl.poměr
0	0	0	47	256
0	0	1	54	224
0	1	0	63	192
0	1	1	75	160
1	0	0	12.5	960
1	0	1	100	120
1	1	0	-	60
1	1	1	0.5-62.5	96*(256-TH1)

Stavový registr S1STA ve svých pěti významnějších bitech indikuje stav řadiče, na nějž je potřeba reagovat samostatnou obsluhou. Doporučené využití stavové informace přerušovací rutinou si uvedeme:

```
PUSH PSW
PUSH S1STA
PUSH HADD
RET
```

Přerušovací rutina uloží hodnotu registru S1STA na zásobník a po uložení adresy stránky (256 B) předá řízení konkrétní obsluze (která je většinou tak jednoduchá, že pro ni postačí osm slabik programu):

ADCON - A/D Convertor Control

	x	x	0	0	0	0	0	0
C5 _H	adc1	adc0	ADEX	ADCI	ADCS	AADR2	AADR1	AADR0

Registr ADCON řídí A/D převodník. Bitem ADEX (ADEX=1) dovoluujeme odstartování převodu náběžnou hranou na vývody STADC, nezávisle na nastavení tohoto bitu může být převod odstartován nastavením bitu ADCS. Bit ADCI je nastavován při ukončení převodu, programem může být pouze nulován. Bit ADCS je nastaven programem nebo při spuštění převodu vnějším signálem STADC, je nulován při ukončení převodu. Běžící převod nelze přerušit ani znovu odstartovat. Bity ADCON2, ADCON1 a ADCON0 ovládají vstupní multiplexer A/D převodníku.

ADCH - A/D Convertor High

	x	x	x	x	x	x	x	
C6 _H	adc9	adc8	adc7	adc6	adc5	adc4	adc3	adc2

Výstupní registr A/D převodníku obsahuje osm vyšších řádů výsledku, dva nižší řády najdeme v řídicím registru ADCON (bity ADCON.7 a ADCON.6).

IEN0 - Interrupt Enable 0

	0	0	0	0	0	0	0	0
A8 _H	EA	EAD	ES1	ES0	ET1	EX1	ET0	EX0

Většina bitů registru IEN0 má význam shodný s mikropočítačem 8051 (EA, ET1, EX1, ET0 a EX0). Bit ES0 odpovídá bitu ES mikropočítače 8051, bit ES1 dovoluje reagovat na

přerušeni generovaná obvodem rozhraní I²C, bit EAD povoluje přerušeni při ukončení práce A/D převodníku.

IP0 - Interrupt Priority 0

	0	0	0	0	0	0	0	
B8 _H	-	PAD	PS1	PS0	PT1	PX1	PT0	PX0

Několik bitů registru IP0 má (podobně jako u registru IEN0) význam shodný s mikropočítačem 8051 (PT1, PX1, PT0 a PX0). Bit PS0 odpovídá bitu PS mikropočítače 8051, bit PS1 zvyšuje prioritu přerušeni generovaných obvodem rozhraní I²C, bit PAD zvyšuje prioritu přerušeni generovaných A/D převodníkem.

IEN1 - Interrupt Enable 1

	0	0	0	0	0	0	0	
E8 _H	ET2	ECM2	ECM1	ECM0	ECT3	ECT2	ECT1	ECT0

Jednotlivé bity registru IEN1 při nastavení na jedničku povolují přerušeni při přetečení volně běžícího čítače T2 (ET2), při přepisu jeho hodnoty do záchytného registru (ECT3, ECT2, ECT1 a ECT0) a při shodě s komparačním registrem (ECM2, ECM1 a ECM0).

IP1 - Interrupt Priority 1

	0	0	0	0	0	0	0	
F8 _H	PT2	PCM2	PCM1	PCM0	PCT3	PCT2	PCT1	PCT0

Jednotlivé bity registru IP1 při nastavení na jedničku zvyšují prioritu přerušeni při přetečení volně běžícího čítače T2 (PT2), při zápisu jeho hodnoty do záchytného registru (PCT3, PCT2, PCT1 a PCT0) a při shodě s komparačním registrem (PCM2, PCM1 a PCM0).

PCON - Power Constrol Register

	0	0	0	0	0	0	0	
87 _H	SMOD	-	-	WLE	GF1	GF0	PD	IDL

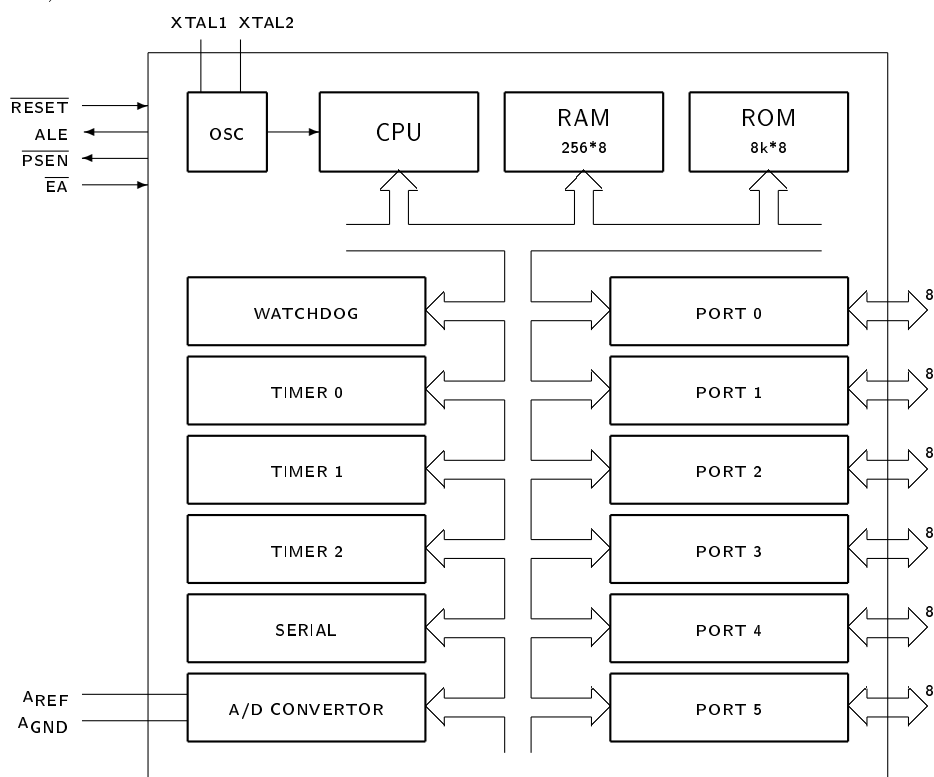
Některé z bitů registru PCON odpovídají těm, které najdeme u mikropočítače 8051 (SMOD - předdělič pro sériové rozhraní SIO0 a v aplikaci použitelné příznaky GF0 a GF1). Přidaný bit WLE musí být nastaven před každým zápisem nové hodnoty do čítače Watchdog (T3). Nastavení bitu IDL zastavuje procesor (režim Idle), nastavení bitu PD zastavuje oscilátor (režim Power-Down). Bit PD je funkční pouze, není-li aktivován obvod Watchdog (úroveň LOW na vývodu EW).

8.3 Siemens SAB 80C515/80C535

Jednočipový mikro počítač Siemens SAB 80C515/80C535 [6] je modernějším následníkem mikro počítače 8051. Jeho struktura vychází ze základního typu 8051 (programy vytvořené pro 8051 můžeme na SAB 80C515/80C535 použít), byla však podstatně rozšířena.

Paměť RAM mikro počítače SAB 80C515/80C535 je větší o 128 B (celková kapacita 256 B). Mikro počítač SAB 80C515 má paměť ROM programovanou maskou při výrobě (vnějším signálem \overline{EA} ji lze, podobně jako u mikro počítače 8051 odpojit), její kapacita je oproti 8051 větší o 4 KB (celkových 8 KB), verze SAB 80C535 vnitřní paměť neposkytuje (ROM-less verze).

Pro aplikace s bateriovým napájením jsou k dispozici dva úsporné módy: procesor lze programově zastavit (Idle Mode), přičemž periférie pracují, nebo lze vypnout oscilátor (Power Down Mode).

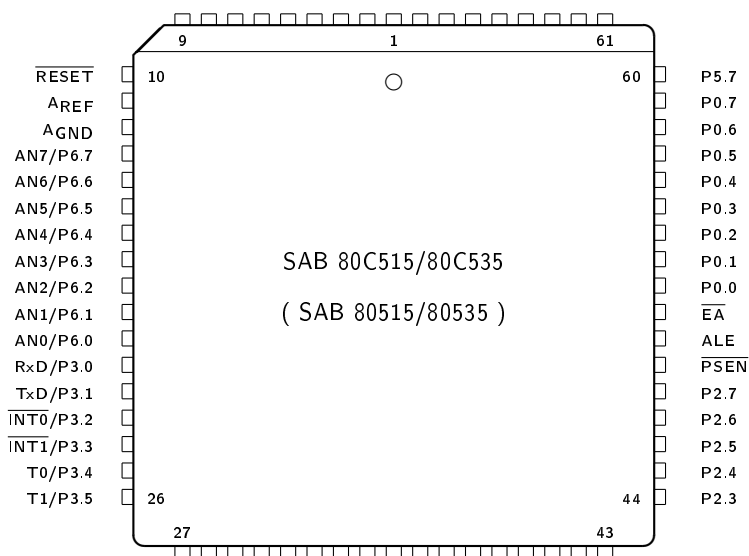


Obrázek 8.9: Struktura jednočipového mikro počítače SAB 80C515/80C535

Podstatně byl rozšířen soubor periferních obvodů (obr.8.9). Ke čtyřem osmibitovým portům mikro počítače 8051 byly přidány další dva osmibitové porty. K původním dvěma univerzálním čítačům/časovačům mikro počítače 8051 byl přidán moderní 16-bitový čítač/časovač s jedno-
stupňovým předřazeným děličem a čtyřmi registry, které mohou pracovat jako záchytné nebo komparační. Jeden z registrů lze navíc použít pro přednastavení (Reload), požadavek na přednastavení lze odvodit z přeplnění čítače a z vnějšího signálu T2EX (možnost synchronizace s vnějším periodickým dějem). Mikro počítač byl doplněn o jednoduchý 16-bitový hlídač obvodu Watchdog, který musí být spuštěn po startu procesoru (není spouštěn automaticky). Jeho čítač není programovatelný, pokud není programem nulován, vyvolá po 65532 strojových cyklech (zhruba 65 ms pro krystal 12 MHz) restart procesoru.

Osmibitový A/D převodník s osmivstupovým multiplexerem a obvodem analogové paměti (Sample/Hold) pracuje s postupnou aproximací a je schopen převést vstupní analogové napětí

na osmibitové slovo za 15 μ s. Programovatelný obvod referenčního napětí dovoluje zvýšit přesnost v užším intervalu vstupního napětí. Přidání řady periférií si vyžádalo podstatnou úpravu přerušovacího systému. Přerušovací systém mikropočítače SAB 80C515/80C535 reaguje na pět vnitřních a osm vnějších zdrojů přerušení, generuje 12 vektorů a pracuje se čtyřmi úrovněmi přerušení.



Obrázek 8.10: Rozložení vývodů jednočipového mikropočítače SAB 80C515/80C535

Rozšíření periferních obvodů si pochopitelně vyžádalo použití pouzdra s více vývody než má 8051, mikropočítač SAB 80C515/80C535 používá pouzdro PLLC-68 (obr.8.10).

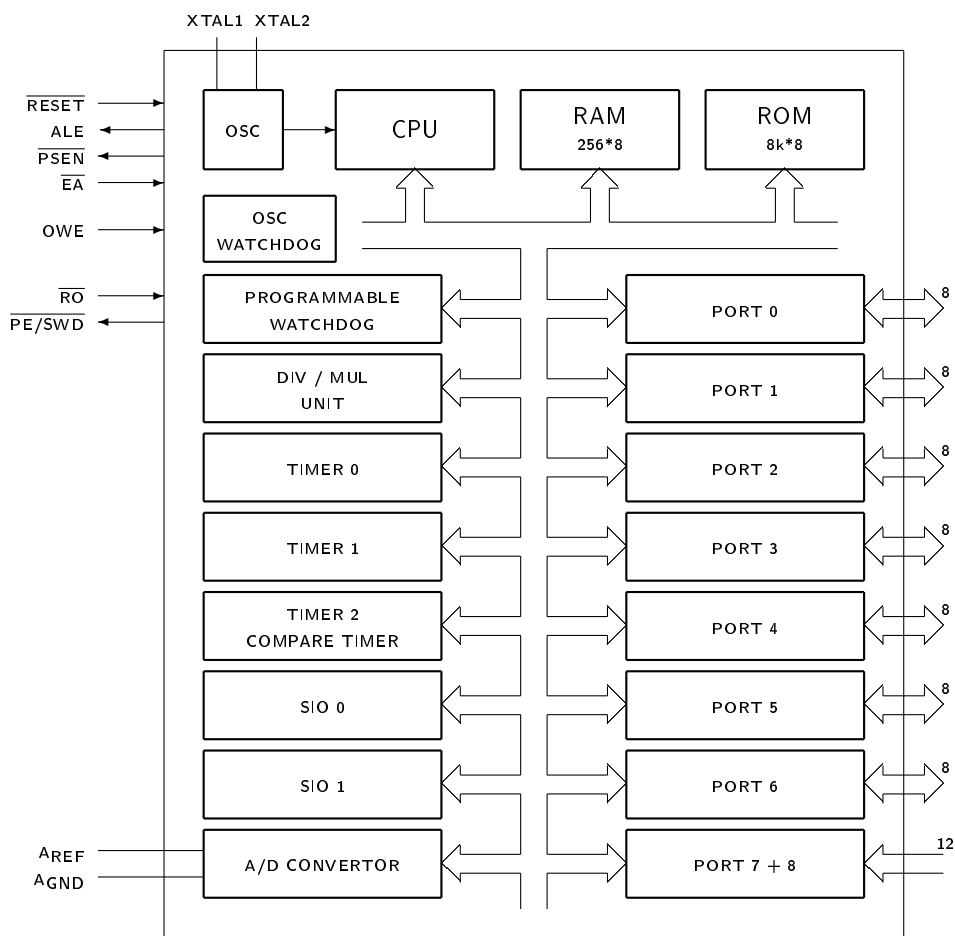
Základní typ SAB80515/80535 byl vyráběn technologií NMOS, současné typy SAB80C515 a SAB80C535 jsou vyráběny technologií CMOS a mohou pracovat i s krystalem 16 MHz (verze SAB 80C515-16/80C535-16).

8.4 Siemens SAB 80C517/80C537

Jednočipový mikročip Siemens SAB 80C517/80C537 [7] je ještě výraznější modifikací mikročipů 8051 než SAB80C515/80C535. Jeho struktura také vychází ze základního typu 8051, programy vytvořené pro 8051 můžeme na SAB 80C517/80C537 použít.

Paměť RAM mikročipů SAB 80C517/80C537 je větší o 128 B (celková kapacita 256 B). Verze SAB 80C517 má paměť ROM programovanou maskou při výrobě (vnějším signálem \overline{EA} ji lze odpojit), její kapacita je oproti 8051 o 4 KB větší (celkových 8 KB), verze SAB 80C537 vnitřní paměť neposkytuje (ROM-less verze). Jediný adresační registr DPTR procesoru 8051 byl nahrazen osmicí registrů, výběr právě aktivního registru DPTR je řízen přidaným registrem DPSEL (92_H). Více adresačních registrů DPTR podstatně zefektivňuje práci s vnější pamětí RAM. Výpočetní kapacita původní aritmetické jednotky, nedostatečná pro aplikace vyžadující složitější výpočty, byla podstatně zvýšena doplněním „koprocesoru“ pro 16-bitové násobení a dělení a pro 32-bitové posuvy a normalizaci. Uvedené operace potřebují pro výpočet čtyři až šest instrukčních cyklů (4-6 μs pro krystal 12 MHz).

Mikročip může pracovat ve třech úsporných módech: s osminou základní frekvence (Slow-Down Mode), s vypnutým procesorem, ale pracujícími perifériemi (Idle Mode) a s vypnutým oscilátorem (Power-Down Mode). Mikročip SAB 80C517/80C537 jsou vyráběny technologií CMOS a mohou pracovat i s krystalem 16 MHz (verze SAB 80C517-16/80C537-16).



Obrázek 8.11: Struktura jednočipového mikročipového SAB 80C517/80C537

Podstatně byl rozšířen soubor periferních obvodů (obr.8.11). Ke čtyřem osmibitovým portům

nut ($\overline{PE}/SWD=0$). Nulová hodnota signálu při běhu programu dovoluje programovaný přechod do stavu Idle Mode, Slow-Down Mode nebo Power-Down Mode. Informace o startu a restartu procesoru je vnějším obvodům k dispozici na výstupu \overline{RO} (Reset Out). Dalším hlídacím obvodem je Oscilator Watchdog, který vyvolá restart procesoru při výpadku oscilátoru (snížení jeho frekvence pod 300 kHz). Obvod lze blokovat nulovou úrovní signálu OWE (Oscilator Watchdog Enable).

Přidání řady periférií si vyžádalo podstatnou úpravu přerušovacího systému. Přerušovací systém mikropočítače SAB 80C517/80C537 reaguje na sedm vnitřních a osm vnějších zdrojů přerušení, generuje 14 vektorů a pracuje se čtyřmi úrovněmi přerušení.

Rozšíření periferních obvodů proti mikropočítači SAB 80C515/80C535 si pochopitelně vyžádalo větší pouzdro, mikropočítač SAB 80C517/80C537 používá pouzdro PLLC-84 (obr.8.12).

8.5 Dallas DS80C320

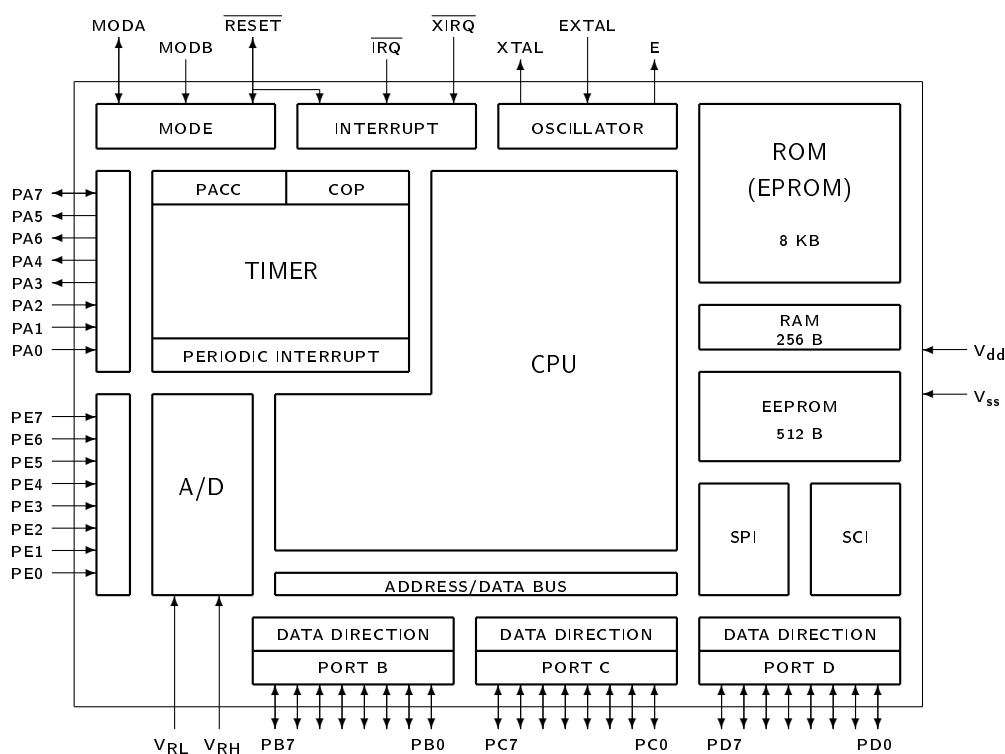
Zajímavou modifikací mikropočítače 80C32 je jednočipový mikropočítač Dallas 80C320. Modernizace se týká nejslabších míst procesoru, DS80C320 na provedení instrukce o délce jedné slabiky potřebuje pouze čtyři periody hodinového signálu. U delších instrukcí není zkrácení tak výrazné, přesto lze počítat s celkovým 2.5-násobným zrychlením výpočtu. Procesor běžně pracuje s krystalem 25 MHz, jeho rychlost by pak odpovídala rychlosti 80C32 s krystalem 62.5 MHz.

Procesor je doplněn o druhý registr DPTR, který podstatně zjednodušuje a zrychluje práci s vnější pamětí. Krátký cyklus vnější sběrnice (160 ns) lze pro pomalé vnější obvody rozhraní programově prodloužit až na 1.12 μ s, případně ho zkrátit na 80 ns.

Je rozšířen soubor obvodů rozhraní, přidané sériové rozhraní je podobné původnímu. Je doplněn obvod Watchdog a detektor poklesu napájecího napětí a dalších pět vstupů vnějšího přerušení. Obvod je přímo použitelný jako náhrada 80C32 v pouzdrech DIL-40 a PLCC-44 ve starších konstrukcích (má shodné rozložení vývodů), s krystalem 25 MHz má spotřebu 30 mA.

8.6 Motorola MC68HC11

Mikročítač MC68HC11 [8] je reprezentantem firmy Motorola ve třídě osmibitových jednočipových mikročítačů. Základem jeho struktury je osmibitový procesor, který vznikl postupným vývojem z historického typu MC6800. Procesor je doplněn o malou vnitřní paměť RAM a paměť ROM, EPROM (v provedení s mazacím okénkem nebo v levném provedení OTP) a/nebo EEPROM, časovač se skupinou záchytných a komparačních registrů, jednoduchý čítač událostí (PACC), periodický generátor přerušení, generátor PWM, hlídací obvod Watchdog (COP), firemní sériovou sběrnici SPI, běžné asynchronní sériové rozhraní SCI, osmivstupový osmibitový A/D převodník a prioritní přerušovací systém.



Obrázek 8.13: Struktura jednočipového mikročítače 68HC11

Na rozdíl od řady MC68HC05, která je charakteristická širokým spektrem modifikací lišících se použitými periferními obvody, jsou si mikročítače základní řady MC68HC11 značně podobné. Všechny zahrnují soubor periférií, který postačí pro většinu běžných aplikací, odlišnosti najdeme hlavně v kapacitách pamětí. Pro informaci si uvedeme přehled nejpoužívanějších prvků řady:

Označení	ROM	EEPROM	RAM	EPROM verze	Pouzdro
68HC11A0	0	0	256		48P,52FN
68HC11A1	0	512	256		48P,52FN
68HC11A8	8K	512	256	68HC711E9	48P,52FN
68HC11E0	0	0	512		52FN
68HC11E1	0	512	512		52FN
68HC11E9	12K	512	512	68HC711E9	52FN
68HC11D0	0	0	192		40P,44FN
68HC11D3	4K	0	192	68HC711D3	40P,44FN

Označení pouzder: 40P = DIP-40, 48P = DIP-48, 44FN = PLCC-44, 52FN = PLCC-52

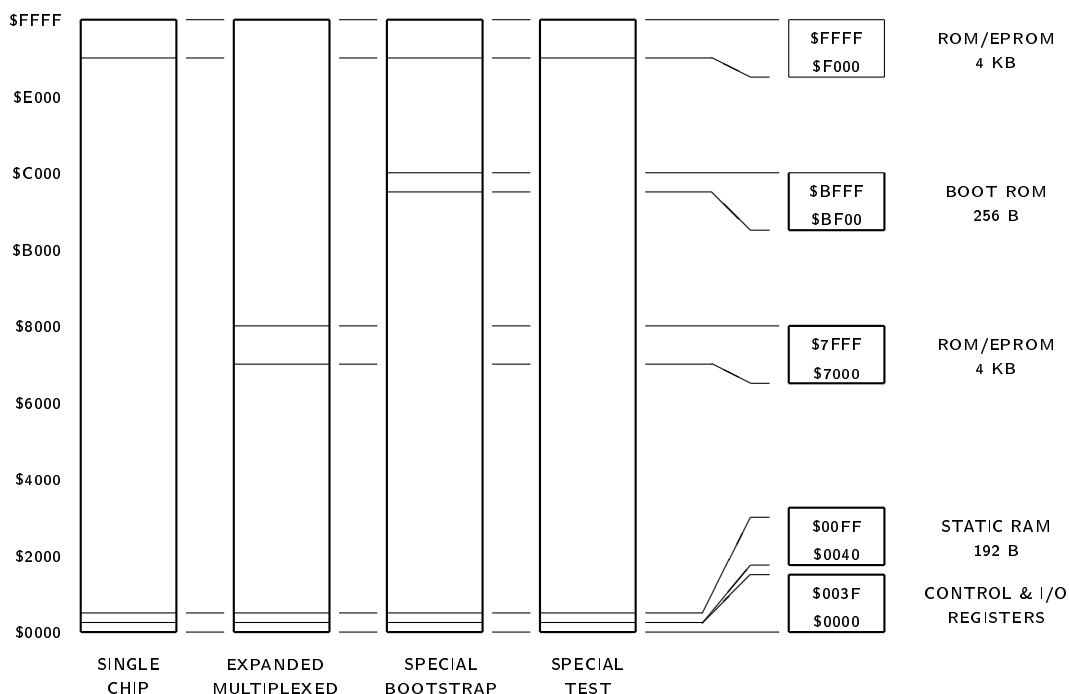
Řadu E lze považovat za standard, který nahrazuje starší řadu A, řada D neobsahuje A/D převodník a je vhodná pro jednoduché aplikace, u kterých je podstatná nízká cena.

Signálem na vstupech MODA a MODB při resetu (náběžné hraně signálu RESET) lze nastavit počáteční hodnoty bitů SMOD a MDA v konfiguračním registru RESET (\$003C) a mikropočítač tak uvést do jednoho ze čtyř možných režimů práce:

RESET	MODA	MODB	Režim práce	SMOD	MDA
1	0	1	Single-Chip	0	0
1	1	1	Expanded-Multiplexed	0	1
1	0	0	Bootstrap	1	0
1	1	0	Test	1	1
0	0	0	EPROM Emulation		

V režimu Single-Chip pracuje mikropočítač pouze s vnitřní pamětí programu a dat (není vytvářena vnější sběrnice), k dispozici aplikaci jsou všechny porty mikropočítače. V režimu Expanded-Multiplexed jsou porty B a C využity pro vytvoření vnější sběrnice. Data a nižší řády adresy jsou multiplexovány na portu C, pro vyšší řády adresy je využit port B. Multiplex je řízen signálem AS. V režimu Bootstrap je odstartován zaváděcí program z vnitřní paměti ROM, který zavede do paměti RAM aplikační program ze sériového rozhraní a po zavedení ho spustí. Režim Bootstrap je využíván pro diagnostiku a pro procesorem řízené programování vnitřní paměti EPROM. Režim Test je určen pro testování výrobcem.

Poslední řádek tabulky neoznačuje režim práce (nejsou uvedeny hodnoty bitů SMOD a MDA), nulová úroveň signálů na vstupech RESET, MODA a MODB dovolí programovat vnitřní paměť EPROM běžným programátorem tak, jako by se jednalo o běžnou paměť EPROM.



Obrázek 8.14: Paměťový prostor jednočipového mikropočítače 68HC11

Mikropočítač 68HC11 má jediný paměťový prostor o délce 64 KB, do tohoto prostoru je mapována vnitřní paměť ROM/EPROM, zaváděč v paměti BOOT ROM (pro režim Bootstrap), paměť RAM a řídicí a I/O registry (obr.8.14). Neobsazené oblasti paměťového prostoru jsou

v režimech Extended-Multiplexed a Test k dispozici na vnější sběrnici tvořené vývody portů B a C, kolize jsou řešeny ve prospěch vnitřních oblastí.

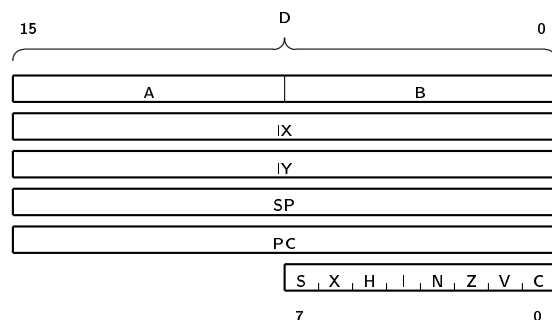
Vnitřní paměť ROM/EPROM je umístěna v adresním prostoru pevně, umístění paměti RAM a registrů je možné po resetu nastavit podle potřeby aplikace. Pozice oblasti RAM a oblasti registrů (nejvyšší čtyři bity) je určena obsahem registru INIT (\$003D) a zápisem do tohoto registru ji lze také změnit. K zápisu musí v normálních režimech dojít do 64 sběrnicevých cyklů po resetu, pak lze registr INIT pouze číst. V režimech Bootstrap a Test lze s registrem INIT pracovat bez omezení.

Processor

Procesor pracuje se dvěma rovnocennými střadači A a B, které jsou výjimečně využívány jako jediný šestnáctibitový registr D. Procesor pracuje s jediným adresním prostorem o délce 64 KB, programový čítač PC má délku 16 bitů, stejně jako ukazatel zásobníku SP. Přístup k datům zjednodušují dva šestnáctibitové indexační registry IX a IY, použití prvého z nich je výrazně efektivnější (instrukce pracující s registrem IY jsou delší).

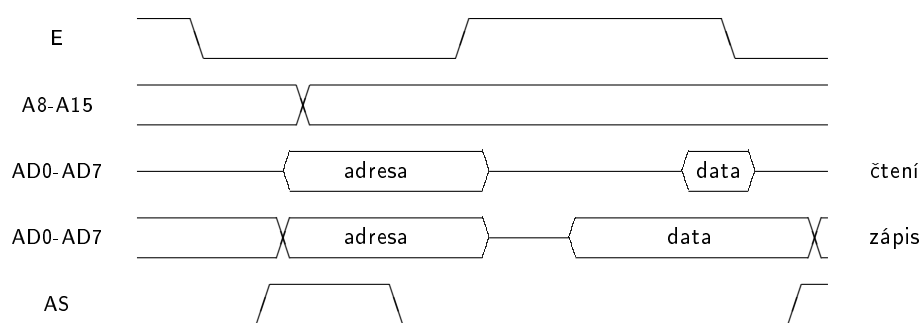
Stavový registr procesoru CCR (Condition Code Register) ukládá následující příznaky:

- S - Stop Disable, bit je nastavován programem, nastavený bit způsobí, že procesor interpretuje instrukci STOP jako NOP,
- X - X Interrupt Mask, bit je nastavován signály $\overline{\text{RESET}}$ a $\overline{\text{XIRQ}}$ a nulován instrukcemi TAP a RTI,
- H - Half-Carry, přenos mezi čtvrtým a pátým bitem, využíván pro BCD aritmetiku,
- I - Interrupt Mask, bit maskující maskovatelná přerušení, vnitřní i vnější,
- N - Negative, indikuje negativní výsledek poslední aritmetické, logické operace nebo bitové operace,
- Z - Zero, indikuje nulový výsledek poslední aritmetické, logické a bitové operace, nebo přesunu,
- V - Overflow, indikuje přetečení při aritmetické operaci,
- C - Carry, indikuje přenos při poslední aritmetické operaci nebo posuvu.



Obrázek 8.15: Registry procesoru mikropočítače 68HC11

Procesor je řízen vestavěným oscilátorem s vnějším krystalem, který pracuje na čtyřnásobku kmitočtu sběrnice. Běžný kmitočet krystalu je 8 MHz, kmitočet sběrnice je pak 2 MHz. Lze použít i vnější hodinový signál, při nižším hodinovém kmitočtu pochopitelně klesá spotřeba. Ukázkou signálů při čtení a zápisu na vnější sběrnici v režimu Extended-Multiplexed uvádí obr.8.16.

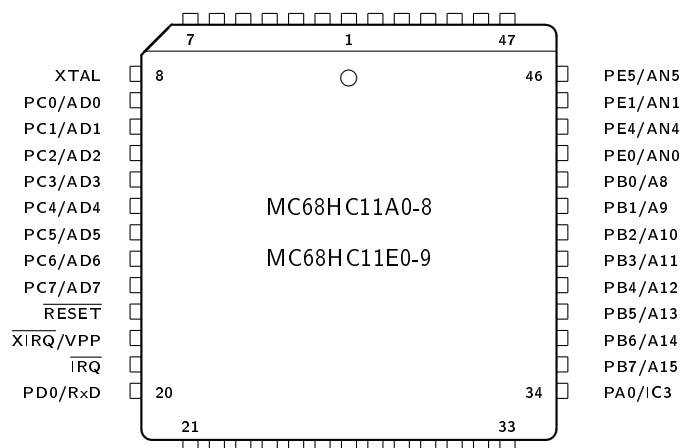


Obrázek 8.16: Signály vnější sběrnice mikro počítače 68HC11

Fáze sběrnicevého cyklu je indikována signálem E, který je charakteristický pro všechny procesory a mikro počítače Motorola. Multiplex nižších řádů adresy a dat je řízen signálem AS (Address Strobe).

Procesor lze programem převést do stavu WAIT (čekání na vnější událost, vnitřní aktivity jsou zastaveny), nebo do stavu STOP (procesor je zastaven). Typické hodnoty spotřeby jsou: v běžném provozu (5V, 2MHz) 15 (Single-Chip) až 25 mA (Expanded-Multiplexed), ve stavu WAIT 6 (Single-Chip) až 10 mA (Expanded-Multiplexed), ve stavu STOP 100 μ A.

Mikro počítač je vyráběn v pouzdech PLCC, redukovaná řada D a starší řada A i v pouzdech DIP (viz str.8.6). Jako příklad si uvedeme rozložení vývodů pouzdra PLCC-52 mikro počítače MC68HC11E0-9.



Obrázek 8.17: Rozložení vývodů pouzdra jednočipového mikro počítače MC68HC11E9

Instrukční soubor

Instrukční soubor má 145 instrukcí a dovoluje široké využití adresních módů. Základní přehled instrukcí uvádí tab.8.2.

Instrukce 68HC11 mohou využívat přímé operandy (Immediate Mode - I), adresovat v rámci stránky o velikosti 256 B na adresách \$0000 - \$00FF (Direct Mode - D) nebo v rámci celého adresního prostoru (Extended Mode - E). Indexaci slouží registry IX a IY, indexace používá osmibitový doplněk (v rozsahu 0 až 255). Podmíněné skoky používají relativní adresaci (Relative Mode - R), osmibitový doplněk je v rozsahu -128 až 127. Řada instrukcí má operand určený implicitně (Inherent Mode) operačním kódem.

Pole Adresace informuje o možnostech adresace operandu, jednopísmenové zkratky byly vysvětleny v předcházejícím odstavci. Sloupec označený znakem - přísluší implicitnímu operandu, další sloupce ostatním adresním módům. Číslo v příslušném sloupci pole Adresace znamená, že použití daného adresního módu je možné, hodnota udává délku instrukce. Tabulka instrukcí je uspořádána tak, že informace společné pro více instrukcí jsou spojené do jednoho políčka (např. informace o změnách v příznacích). Pro úsporu místa je u instrukcí skoku sloupec odpovídající relativní adresaci (R) na místě sloupce pro přímou osmibitovou adresaci (I) ostatních instrukcí. Instrukce manipulující s bity mají vedle hlavního operandu masku mm (instrukce BCLR, BSET), případně ještě relativní adresu (instrukce BRCLR, BRSET).

V tabulce jsou použity následující symboly:

A,B,D,IX,IY,SP,PC,CCR	- registry procesoru,
S,X,H,I,N,Z,V,C	- bity stavového slova CCR,
M	- adresovaná osmibitová slabika,
M:M+1	- adresované šestnáctibitové slovo,
00:B	- obsah registru B doplněný nulami na šestnáctibitové slovo,
mm	- osmibitová konstanta - bitová maska,
Stk	- vrchol zásobníku.

MC68HC11 - Instrukce skoku

Kód	Operace	Adresace		Příznaky																
		-	R D E X Y	S	X	H	I	N	Z	V	C									
BCC	skok při $C = 0$			2																
BCS	skok při $C = 1$			2																
BEQ	skok při $Z = 1$			2																
BGE	skok při $N \oplus V = 0$			2																
BGT	skok při $Z + (N \oplus V) = 0$			2																
BHI	skok při $C + Z = 0$			2																
BHS	skok při $C = 0$			2																
BLE	skok při $Z + (N \oplus V) = 1$			2																
BLO	skok při $C = 1$			2																
BLS	skok při $C + Z = 1$			2																
BLT	skok při $N \oplus V = 1$			2																
BMI	skok při $N = 1$			2																
BNE	skok při $Z = 0$			2																
BPL	skok při $N = 0$			2																
BRA	nepodmíněný skok			2																
BRCLR	skok při $M \wedge mm = 0$			4		4	5													
BRN	Branch Never - NOP			2																
BRSET	skok při $\bar{M} \wedge mm = 0$			4		4	5													
BSR	volání podprogramu			2																
BVC	skok při $V = 0$			2																
BVS	skok při $V = 1$			2																
JMP	nepodmíněný skok				3	2	3													
JSR	volání podprogramu				2	3	2	3												
RTI	návrat z přerušeni			1																
RTS	návrat z podprogramu			1																
SWI	programové přerušeni			1																

MC68HC11 - Řízení procesoru

Kód	Operace	Adresace		Příznaky																	
		-	R D E X Y	S	X	H	I	N	Z	V	C										
NOP	NOP			1																	
STOP	zastavení hodin			1																	
TEST	test adresní sběrnice			1																	
WAI	čekání na přerušeni			2																	

Tabulka 8.2: Instrukční soubor mikro počítače 68HC11

Konfigurační registry

Mikropočítač 68HC11 je vybaven několika registry, které ovlivňují režim činnosti, funkci dohlížecích obvodů, funkci přerušovacího systému a umístění vnitřních oblastí (registrů a pamětí RAM a BOOT ROM) v adresním prostoru. V následujících odstavcích si tyto registry popíšeme, u každého registru si uvedeme jeho strukturu (základní a případně i alternativní jména jednotlivých bitů) a hodnotu, kterou v něm najdeme po resetu.

INIT - RAM & I/O Control Register

	0	0	0	0	0	0	0	0
\$003D	RAM3	RAM2	RAM1	RAM0	REG3	REG2	REG1	REG0

Registr INIT (\$003D) dovoluje nastavit nejvyšší řády adresy pro oblast registrů a pro vnitřní paměť RAM. Po resetu jsou obě tyto oblasti ve všech režimech umístěny od adresy \$0000, přičemž oblast registrů překrývá prvých 128 slabik vnitřní paměti RAM.

OPTION - Configuration Options Register

	0	0	0	1	0	0	0	0
\$0039	ADPU	CSEL	IRQE	DLY	CME	0	CR1	CR0

Registr OPTION (\$0039) definuje reakci přerušovacího systému na přerušení \overline{IRQ} a nastavuje parametry dohlížecích obvodů. Bit IRQE dovoluje přerušovacímu systému reagovat na hranu (IRQE=1) nebo na úroveň na vstupu přerušení \overline{IRQ} . Bit DLY předepisuje prodlevu 4096 cyklů sběrnice (period signálu E) po vzbuzení ze stavu STOP, důvodem je potřeba dát oscilátoru čas pro rozběh. Při použití vnějšího zdroje hodinového kmitočtu může být vynulován. Bitem CME lze zapnout monitor hodinového signálu, který při poklesu jeho frekvence pod 10 kHz vyvolá reset procesoru. Konečně bity CR1 a CR0 definují modul předřazeného děliče dohlížecího obvodu Watchdog. Bity IRQE, DLY, CR1 a CR0 lze měnit v režimech Single Mode a Expanded Multiplexed pouze během prvých 64 sběrnicových cyklů.

Bity ADPU a CSEL souvisí s A/D převodníkem mikropočítače. Bit ADPU (A/D Power Up) zapíná zdroj napětí (Charge Pump) pro A/D převodník, bit je nutné nastavit v předstihu před používáním převodníku. Nastavením bitu CSEL na jedničku je aktivován pomocný RC oscilátor nutný, pokud vlastní mikropočítač pracuje s frekvencí sběrnice pod 750 kHz (A/D převodník pracuje s přerozdělováním náboje a vyžaduje dodržení určité frekvence hodin, naproti tomu celá ostatní struktura mikropočítače je plně statická). Pomocný RC oscilátor má kmitočet zhruba 2 MHz. Bit CSEL současně přivádí signál pomocného oscilátoru na zdroj zvýšeného napětí pro programování paměti EEPROM.

HPRIO - High Priority I Interrupt Register

	*	*	*	*	0	1	0	1
\$003C	RBOOT	SMOD	MDA	IRVNE	PSEL3	PSEL2	PSEL1	PSEL0

Registr HPRIO (\$003C) informuje o režimu činnosti (a případně ho i dovolí měnit) a definuje zvýšení priorit přerušovacího systému. Bity SMOD a MDA jsou nastaveny po resetu na hodnoty definované napěťovými úrovněmi na vývodech MODA a MODB (viz str.87). Změna hodnoty bitu SMOD přichází v úvahu pouze v režimech Bootstrap a Test a to navíc pouze jeho vynulování (přechod do režimu Single Chip nebo Expanded Multiplexed). Bit MDA smí být měněn ve všech režimech, ale v režimech Single Chip a Expanded Multiplexed pouze jednou. Bit RBOOT připojuje do adresního prostoru paměť BOOT ROM (a je nastaven na jedničku při resetu v režimu Bootstrap), jeho změna je možná pouze v režimech Bootstrap a Test. Konečně, bit

IRVE zviditelňuje na vnější sběrnici hodnoty čtené ve vnitřních oblastech a má význam pro režim Test.

Bity PSEL3 - PSEL0 určují zdroj přerušeni maskovatelného přerušeni, který bude mít prioritu zvýšenu proti ostatním (význam bitů uvádí tab.8.3). Po resetu je v těchto bitech hodnota 0101, této hodnotě neodpovídá žádný zdroj přerušeni a nejvyšší prioritu má signál vnějšího přerušeni (vývod \overline{IRQ}).

CONFIG - Configuration Control Register

	0	0	0	0	0	*	*	0
\$003F	0	0	0	0	0	NOCOP	EPON	0

Registr CONFIG (\$003F) ovládá dohlížecí obvod Watchdog (COP - Computer Operating Properly) a přístup k vnitřní paměti EPROM. V režimech Single Mode a Expanded Multiplexed je obvod COP po resetu spuštěn (NOCOP=0) a může být programem pouze zastaven. V režimech Bootstrap a Test je po resetu zastaven (NOCOP=1) a lze ho volně ovládat. Vypršení čítače obvodu Watchdog vyvolá reset. Bit EPON je po resetu nastaven (EPON=1) a zpřístupňuje vnitřní paměť EPROM. V režimech Single Chip a Expanded Multiplexed lze pouze vynulovat a paměť EPROM tak odpojit, v režimech Bootstrap a Test lze bit EPON ovládat volně.

Poznamenejme, že paměť EPROM je umístěna po resetu na adresách \$F000 - \$FFFF u všech režimů kromě režimu Expanded Multiplexed, kde ji najdeme na adresách \$7000 - \$7FFF. Chceme-li v tomto režimu pracovat s pamětí EPROM na adresách \$F000 - \$FFFF, musíme mikropočítač odstartovat např. v režimu Single Mode a po resetu režim přepnout (nastavením bitu MDA v konfiguračním registru HPRI0 na jedničku).

PPROG - PROM Programming Control Register

	0	0	0	0	0	0	0	0
\$003B	MBE	0	ELAT	EXCOL	EXROW	0	0	PGM

Registr PPROG (\$003B) je používán při programování paměti EPROM programem mikropočítače. Programování vyžaduje připojení programovacího napětí $V_{PP}=12V$ na vývod \overline{XIRQ}/V_{PP} a nastavení bitu ELAT (EPROM Latch Control) na jedničku. Potom je potřeba zapsat programovanou hodnotu na patřičnou adresu (vnitřně: zapsat adresu a hodnotu do vyrovnávacích registrů obvodu programování) a nastavit bit PGM (EPROM Program Command) na jedničku na dobu dvou až čtyř milisekund. Po naprogramování všech žádaných buněk paměti EPROM je potřeba přepnout paměť do režimu čtení vynulováním obsahu registru PPROG.

Registry obvodů rozhraní

Mikropočítač 68HC11 je vybaven pěti osmibitovými porty. V režimech Single-Chip a Bootstrap jsou všechny k dispozici pro aplikaci, v režimu Extended-Multiplexed a v režimu Test jsou porty B, C a bity D6(AS) a D7(R/W) portu D využity pro vytvoření vnější sběrnice.

PORTA - Port A Data Register

	0	0	0	0	0	0	0	0
\$0000	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
	PAI	OC2	OC3	OC4	OC5 C4	IC1	IC2	IC3
	OC1	OC1	OC1	OC1	OC1			

Osmibitový port PORTA (vývody PA0 až PA7) zpřístupňuje vstupy a výstupu jednotky

časovače/čítače, jeho jednotlivé vývody lze ale jinak ovládat a číst z programu (pouze vývody PA3 a PA7 jsou obousměrné, vývody PA0 až PA2 jsou pouze vstupy a vývody PA4 až PA6 jsou pouze výstupy). U obousměrných vývodů PA3 a PA7 je pro jejich použití jako výstupů nutné nastavit příslušný bit v řídicím registru. Těmito bity jsou zde bity DDRA3 a DDRA7 v registru PACTL (\$0026).

PORTB - Port B Data Register

	0	0	0	0	0	0	0	0
\$0004	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
	A15	A14	A13	A12	A11	A10	A9	A8

DDRB - Port B Data Direction Register

	0	0	0	0	0	0	0	0
\$0006	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0

Osmibitový port PORTB (\$0004, vývody PA0 až PA7) je vedle své základní funkce jako obousměrné paralelní rozhraní využíván k vytvoření vnější sběrnice (adresní bity A8 - A15). Jednotlivé vývody lze konfigurovat jako vstupy nebo výstupy nastavením odpovídajících bitů v registru DDRB (\$0006), po resetu jsou konfigurovány jako vstupy.

PORTC - Port C Data Register

	0	0	0	0	0	0	0	0
\$0003	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0

DDRC - Port C Data Direction Register

	0	0	0	0	0	0	0	0
\$0007	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0

PIOC - Port C Control Register

	0	0	0	0	0	0	0	0
\$0002			CWOM					

Osmibitový port PORTC (\$0003, vývody PA0 až PA7) je vedle své základní funkce jako obousměrné paralelní rozhraní využíván k vytvoření multiplexované části vnější sběrnice (data D0 - D7 a adresní bity A0 - A7). Jednotlivé vývody lze konfigurovat jako vstupy nebo výstupy nastavením odpovídajících bitů v registru DDRC (\$0007), po resetu jsou konfigurovány jako vstupy. Při nastavení bitu CWOM (C Wired-Or Mode) na jedničku lze výstupy portu C provozovat jako otevřené kolektory.

PORTD - Port D Data Register

	0	0	0	0	0	0	0	0
\$0008	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	R/W	AS	SS	SCK	MOSI	MISO	TxD	RxD

DDRD - Port D Data Direction Register

	0	0	0	0	0	0	0	0
\$0009	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0

Osmibitový port PORTD (vývody PD0 až PD7) zpřístupňuje vstupy a výstupy sériové sběrnice

SPI a asynchronního řadiče SCI. Pro použití jednotlivých vývodů jako programovatelných výstupů je potřeba nastavit příslušný bit v konfiguračním registru DDRD (\$0009), po resetu jsou výstupy konfigurovány jako vstupy.

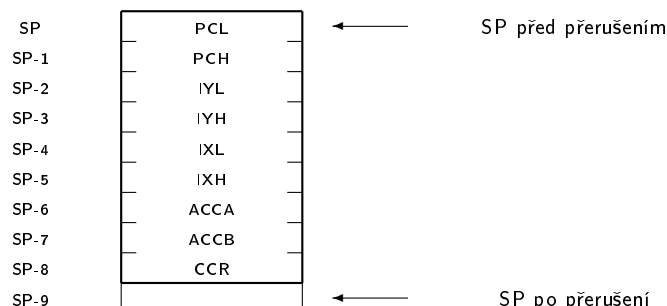
Přerušovací systém

Mikropočítač obsluhuje (vedle signálu $\overline{\text{RESET}}$ a dalších tří podmínek vedoucích na reset) sedmnáct signálových zdrojů přerušení a programové přerušení (instrukci SWI). Z toho patnáct zdrojů přerušení lze maskovat jednak globálně nastavením bitu I v příznakovém registru CCR na jedničku, jednak selektivně lokální maskou. Signál $\overline{\text{XIRQ}}$ (nemaskovatelné přerušení) je maskován bitem X v příznakovém registru CCR pouze, pokud tento bit nevynulujeme (bit pak je nastavován jen při vstupu do obslužné rutiny, programem ho již nelze nastavit a přerušení zakázat). Instrukce SWI je nemaskovatelná stejně jako přerušení vyvolané neznámým operačním kódem (to je standardní řešení u procesorů firmy Motorola, dovolující „rozšiřovat“ instrukční kód). Celkový přehled zdrojů přerušení (včetně příslušných masek), podmínek pro reset a identifikace přerušení v bitech PSEL3 - PSEL0 registru HPRIO uvádí tab.8.3.

Adresa vektoru	Zdroj	CCR maska	Lok. maska	PSEL3-0
FFC0,FFC1	rezerva	-	-	
-	rezerva	-	-	
FFD4,FFD7	rezerva	-	-	
FFD6,FFD7	SCI	I		0 1 0 0
	SCI - Receive Overrun		RIE	
	SCI - Receive Overrun		RIE	
	SCI - Idle Line		ILIE	
	SCI - převzetí znaku k odeslání		TIE	
	SCI - ukončení vysílání		TCIE	
FFD8,FFD9	SPI - ukončení přenosu	I	SPIE	0 0 1 1
FFDA,FFDB	PAI - hrana na vstupu	I	PAII	0 0 1 0
FFDC,FFDD	PAI - přetečení čítače	I	PAOVI	0 0 0 1
FFDE,FFDF	TIMER - přetečení čítače	I	TOI	0 0 0 0
FFE0,FFE1	TIMER - IC4/OC5	I	I4O5I	1 1 1 1
FFE2,FFE3	TIMER - shoda OC4	I	OC4I	1 1 1 0
FFE4,FFE5	TIMER - shoda OC3	I	OC3I	1 1 0 1
FFE6,FFE7	TIMER - shoda OC2	I	OC2I	1 1 0 0
FFE8,FFE9	TIMER - shoda OC1	I	OC1I	1 0 1 1
FFEA,FFEB	TIMER - zápis IC3	I	IC3I	1 0 1 0
FFEC,FFEC	TIMER - zápis IC2	I	IC2I	1 0 0 1
FFEE,FFEF	TIMER - zápis IC1	I	IC1I	1 0 0 0
FFF0,FFF1	RTI - přetečení čítače	I	RTII	0 1 1 1
FFF2,FFF3	$\overline{\text{IRQ}}$ - vnější přerušení	I	-	0 1 1 0
FFF4,FFF5	$\overline{\text{XIRQ}}$ - vnější přerušení	X	-	
FFF6,FFF7	instrukce SWI	-	-	
FFF8,FFF9	nepovolený Op. kód	-	-	
FFFA,FFFB	COP - vypršení limitu	-	NOCOP	
FFFC,FFFD	CM - výpadek hodin	-	CME	
FFFE,FFFF	$\overline{\text{RESET}}$	-	-	

Tabulka 8.3: Vektory a masky přerušení

Na rozdíl od mikropočítače 8051 mikropočítač 68HC11 uchovává při vstupu do obslužné rutiny automaticky řadu registrů (PC, IY, IX, ACCA, ACCB a CCR), které obnovuje při návratu z obslužné rutiny instrukcí RETI (obr.8.18).



Obrázek 8.18: Uchování registrů na zásobníku při přerušení

Přerušovací systém je prioritní, priorita jednotlivých zdrojů odpovídá pořadí vektorů (nepovolený operační kód má nejvyšší prioritu, sériová linka nejnižší). Prioritu libovolného maskovatelného přerušení lze zvýšit nad prioritu ostatních maskovatelných přerušení nastavením bitů PSEL0 až PSEL3 v konfiguračním registru HPRIO.

Práce se sníženým příkonem

Procesor lze programově převést do jednoho ze dvou režimů se sníženým příkonem - režimu STOP a režimu WAIT.

Do režimu STOP, ve kterém je zastaven oscilátor, mikropočítač převedeme instrukcí STOP při vynulovaném bitu S v příznakovém registru CCR. (Při bitu S nastaveném na jedničku funguje instrukce STOP jako NOP). Z režimu STOP lze mikropočítač vyvést pouze signálem $\overline{\text{RESET}}$, $\overline{\text{XIRQ}}$ nebo $\overline{\text{IRQ}}$ není-li toto přerušení maskováno jedničkovou hodnotou bitu I.

Do režimu WAIT převádí mikropočítač instrukce WAIT. Pokud není maskováno přerušení nebo je aktivován dohlížecí obvod Watchdog (COP), zůstává v činnosti oscilátor a všechny obvody rozhraní. V opačném případě je oscilátor zastaven a spotřeba je snížena na úroveň režimu STOP.

Programovatelný časovač

CFORC - Timer Compare Force Register

	0	0	0	0	0	0	0	0
\$000B	FOC1	FOC2	FOC3	FOC4	FOC5	0	0	0

OC1M - Output Compare 1 Mask Register

	0	0	0	0	0	0	0	0
\$000C	OC1M7	OC1M6	OC1M5	OC1M4	OC1M3	0	0	0

OC1D - Output Compare Data Register

	0	0	0	0	0	0	0	0
\$000D	OC1D7	OC1D6	OC1D5	OC1D4	OC1D3	0	0	0

TCNT - Timer Count Register

	0	0	0	0	0	0	0	0
\$000E	-	-	-	-	-	-	-	-
\$000F	-	-	-	-	-	-	-	-

TCTL1 - Timer Control Register 1

	0	0	0	0	0	0	0	0
\$00020	OM2	OL2	OM3	OL3	OM4	OL4	OM5	OL5

TCTL2 - Timer Control Register 2

	0	0	0	0	0	0	0	0
\$0021	EDG4B	EDG4A	EDG1B	EDG1A	EDG2B	EDG2A	EDG1B	EDG1A

TMSK1 - Timer Interrupt Mask Register 1

	0	0	0	0	0	0	0	0
\$00022	OC1I	OC2I	OC3I	OC4I	I4O5I	IC1I	IC2I	IC3I

TFLG1 - Timer Interrupt Flag Register 1

	0	0	0	0	0	0	0	0
\$00023	OC1F	OC2F	OC3F	OC4F	I4O5F	IC1F	IC2F	IC3F

TMSK2 - Timer Interrupt Mask Register 2

	0	0	0	0	0	0	0	0
\$00024	TOI	RTII	PAOVI	PAII	0	0	PR1	PR0

TFLG2 - Timer Interrupt Flag Register 2

	0	0	0	0	0	0	0	0
\$00025	TOF	RTIF	PAOVF	PAIF	0	0	0	0

Časovací jednotka mikropočítače 68HC11 je tvořena volně běžícím šestnáctibitovým čítačem TCNT (Timer Count Register), s předřazeným programovatelným děličem. Modul předřazeného děliče určují bity PR1 a PR0 (Timer Prescaler Select) v registru TMSK2 takto:

PR1	PR0	Dělicí poměr
0	0	1
0	1	4
1	0	8
1	1	16

Přetečení čítače TCNT je indikováno příznakem TOF (Timer Overflow) v registru TFLG2 a může vyvolat přerušení, pokud to dovoluje na jedničku nastavený bit TOI (Timer Overflow Interrupt Enable) v registru TMSK2.

Okamžitou hodnotu čítače TCNT lze zapsat do jednoho ze čtyř šestnáctibitových záchytných registrů (TIC1 (\$0010 a \$0011), TIC2 (\$0012 a \$0013), TIC3 (\$0014 a \$0015) a případně TI4O5 (\$001E a \$001F)). Podnětem k zápisu je splnění podmínky definované bity EDGiB a EDGiA (Input Capture Edge Control) v registru TCTL2 pro příslušný vývod. Touto podmínkou může být náběžná a/nebo sestupná hrana signálu na vývodech IC1, IC2, IC3 a případně IC4/OC5, přiřazení uvádí následující tabulka:

EDGiB	EDGiA	Funkce
0	0	-
0	1	Zápis při náběžné hraně
1	0	Zápis při sestupné hraně
1	1	Zápis při libovolné hraně

Zápis do záchytného registru je indikován nastavením bitu ICiF (nebo I4O5F, Input Capture Flag) v registru TFLG1. Zápis může vyvolat přerušení, pokud to dovolí nastavený bit ICiI (nebo I4O5I, Input Capture Interrupt).

Šestnáctibitové komparační registry (TOC1 (\$0016 a \$0017), TOC2 (\$0018 a \$0019), TOC3 (\$001A a \$001B), TOC4 (\$001C a \$001D) a případně TI4O5 (\$001E a \$001F)) jsou průběžně porovnávány s hodnotou čítače TCNT. Při shodě je nastaven bit OCiF (nebo I4O5F, Output Compare Flag) v registru TFLG1. Detekce shody může vyvolat přerušení, pokud to dovolí nastavený bit OCiI (nebo I4O5I, Output Compare Interrupt). Při shodě čítače s komparačními registry TOC2, TOC3, TOC4 a případně TI4O5 lze vyvolat akci na odpovídajícím vývodu na hodnotu předprogramovanou bity OMi (Output Mode) a OLi (Output Level) registr TCTL1 podle tabulky:

OMi	OLi	Funkce
0	0	-
0	1	Změna předchozí úrovně
1	0	Zápis nulové úrovně
1	1	Zápis jedničkové úrovně

Shodu na registru TOC1 můžeme využít pro změnu úrovně na libovolném z vývodů OC1, OC2, OC3, OC4 a případně IC4/OC5. Změnu povoluje nastavený bit OC1Mi (Output Compare Mask), napěťová úroveň na vývodu PAi je nastavena na hodnotu předprogramovanou v bitu OC1Di (Output Compare Data) registru OC1D. (Indexy zde odpovídají indexům vývodů portu PA, korespondenci indexu kanálu a indexu vývodu uvádí popis I/O portu PA.)

Čítač událostí a generátor periodického přerušení

PACTL - Pulse Accumulator Control Register

	0	0	0	0	0	0	0	0
\$00026	DDRA7	PAEN	PAMOD	PEDGE	DDRA3	I4/O5	RTR1	RTR2

PACNT - Pulse Accumulator Count Register

	0	0	0	0	0	0	0	0
\$00027	-	-	-	-	-	-	-	-

Čítač událostí (Pulse Accumulator) je osmibitový čítač dovolující počítat události (hrany) na vstupu PA7 nebo měřit délku intervalu definovaného pulsem na vstupu PA7 (klíčovým signálem je signál E/64). Mód činnosti a polarita čítaných hran nebo měřeného pulsu určují bity PAMOD a PEDGE registru PACTL.

PAMOD	PEDGE	Funkce
0	0	Čítání sestupných hran
0	1	Čítání náběžných hran
1	0	Čítání period E při jedničce na PA7
1	1	Čítání period E při nule na PA7

Bit DDRA7 určuje, zda bude vývod PA7 pracovat jako vstup (jednoduchý I/O vývod nebo vstup čítače událostí, DDRA7=0) nebo jako výstup (DDRA7=1). Jedničkovou hodnotou bitu PAEN čítač spouštíme, nulovou hodnotou ho můžeme zastavit.

Bit DDRA3 určuje funkci vývodu DDRA3, ten může fungovat jako vstup (DDRA3=0) nebo jako výstup (DDRA3=1). Bit I4/O5 konfiguruje registr TI4O5 jako komparační (I4/O5=0) nebo jako záchytný (I4/O5=1).

Konečně, bity RTR1 a RTR2 volíme jednu ze čtyř period periodického generátoru RTI (Real Time Interrupt). Dělicí poměry (vstupem je signál E) a periodu pro krystal 8 MHz uvádí následující tabulka:

RTR1	RTR2	Modul	Perioda
0	0	2^{13}	4.10 ms
0	1	2^{14}	8.19 ms
1	0	2^{15}	16.38 ms
1	1	2^{16}	32.77 ms

Asynchronní sériové rozhraní SCI

SCCR1 - Serial Communication Control Register 1

	-	-	0	0	0	0	0	0
\$0002C	R8	T8	0	M	WAKE	0	0	0

SCCR2 - Serial Communication Control Register 2

	0	0	0	0	0	0	0	0
\$0002D	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK

SCSR - Serial Communication Status Register

	1	1	0	0	0	0	0	0
\$0002E	TDRE	TC	RDRF	IDLE	OR	NF	FE	

SCDR - Serial Communication Data Register

	0	0	0	0	0	0	0	0
\$0002F								

BAUD - Baud Rate Register

	0	0	0	0	0	-	-	-
\$0002B	TCLR	0	SCP1	SCP0	RCKB	SCR2	SCR1	SCR0

Sériové rozhraní zajišťuje asynchronní přenos na vývodech PD0 a PD1 rozhraní PD. Bitem M (SCI Character Length) registru SCCR1 lze volit mezi osmibitovým přenosem (M=0, start bit, osm datových bitů a jeden stop bit) a devítibitovým přenosem (M=1, start bit, devět datových bitů a jeden stop bit). Osmibitový vysílaný znak zapisujeme do registru SCDR, z něj je znak přepsán do posuvného registru a odeslán. Pro devátý vysílaný bit je vyhrazena pozice T8 v registru SCCR1. Osmibitový přijatý znak je zapsán do registru SCDR, pro případný devátý bit je k dispozici pozice R8 registru SCCR1.

Pro využití sériového rozhraní ve sběrniciovém provozu (vícepočítačový systém) je důležité povolit selektivní reakci přerušovacího systému na přijímané zprávy. Řadič dovoluje volbu mezi dvěma postupy. Prvý je používán i u dalších mikro počítačů Motorola a opírá se o možnost

uspat řadič nastavením bitu RWU registru SCCR2 na jedničku. Bit RWU je nulován po příjmu 10 (M=0) nebo 11 (M=1) jedničkových bitů (podmínka Idle Line). Druhý postup je slučitelný s devítibitovou komunikací mikropočítačů 8051, nastavením bitu RWU lze řadič uspat do příjmu znaku s jedničkou v nejdůležitějším bitu (osmém pro M=0 nebo devátém pro M=1). Výběr postupu je řízen bitem WAKE (Wakeup Method Select), nule odpovídá první, jedničce druhý z uvedených postupů.

Jednotkový bit TE registru SCCR2 připojuje výstup posuvného registru vysílače na vývod PD1, jednotkový bit RE povoluje činnost přijímače. Další bity (jedničky) povolují generování přerušení při splnění následujících podmínek: TIE při uvolnění registru SCCR vysílače, TCIE při odeslání znaku, RIE při příjmu znaku nebo při chybě (Overrun), ILIE při detekci posloupnosti 10 (pro M=0) nebo 11 (pro M=1) jednotkových bitů.

Registr SCSR obsahuje stavové příznaky řadiče. Jednotkový bit TDRE (Transmit Data Register Empty) indikuje uvolnění registru SCCR (přepis znaku do posuvného registru), bit TC (Transmit Complete) indikuje ukončení vysílání znaku. Bit RDRF (Receive Data Register Full) indikuje příjem znaku, bit IDLE (Idle Line Detect) indikuje posloupnost 10 nebo 11 jednotkových bitů. Bit OR (overrun Error) indikuje situaci, kdy je přijat další znak, ale předchozí ještě nebyl převzat, bit NF (Noise Flag) indikuje změnu hodnoty v době odpovídající přijímanému bitu (včetně start a stop bitu). Konečně, bit FE (Framing Error) indikuje chybějící stop bit.

Hodnotou v registru BAUD lze volit přenosovou rychlost, sériový řadič má vlastní generátor přenosové rychlosti. Bity TCLR a RCKB slouží pro testování a lze je použít pouze v režimu test. Bity SCP1 a SCP0 definují modul předděliče, bity SCR2, SCR1 a SCR0 definují modul čítače generátoru. Následující tabulky uvádí použitelné dělicí poměry a odpovídající přenosové rychlosti, první uvádí frekvenci na výstupu předděliče pro krystal 8 MHz, druhá přenosové rychlosti odvozené z kmitočtu 9.6 kHz.

SCP1	SCP0	Modul	Frekvence
0	0	1	125 kHz
0	1	3	41.66 kHz
1	0	4	31.25 kHz
1	1	13	9.6 kHz

SCR2	SCR1	SCR0	Modul	Rychlost
0	0	0	1	9600 b/s
0	0	1	2	4800 b/s
0	1	0	4	2400 b/s
0	1	1	8	1200 b/s
1	0	0	16	600 b/s
1	0	1	32	300 b/s
1	1	0	64	150 b/s
1	1	1	128	75 b/s

Sériová sběrnice SPI

SPCR - Serial Peripheral Control Register

	0	0	0	0	0	-	-
\$00028	SPIE	SPE	DWOM	MSTR	CPOL	CPHA	SPR1 SPR0

SPSR - Serial Peripheral Status Register

	0	0	0	0	0	0	0	0
\$00029	SPIF	WCOL	0	MODF	0	0	0	0

SPDR - Serial Peripheral Data Register

	-	-	-	-	-	-	-	-
\$0002A	-	-	-	-	-	-	-	-

Bit MSTR registru SPCR dovoluje řadič konfigurovat jako řídicí (MSTR=1) nebo podřízený (MSTR=0), bit SPE aktivuje řadič a bit SPIE povoluje přerušeni po ukončení přenosu nebo při chybě. Bit CPOL definuje polaritu signálu SCK, bit CPHA definuje posuv datového signálu vůči hodinám (viz obr.4.13). Bity SPR1 a SPR1 (SPI Clock Rate Select) určují přenosovou rychlost. Kmitočet signálu SCK je pro krystal 8 MHz:

SCR1	SCR0	Modul	Frekvence
0	0	2	1.0 MHz
0	1	4	500 kHz
1	0	16	125 kHz
1	1	32	62.5 kHz

Ukončení přenosu indikuje jednotková hodnota bitu SPIF (Serial Peripheral Status Register) registru SPSR. Bit WCOL (Write Colision) indikuje chybu, zápis do posuvného registru jehož obsah ještě nebyl odeslán. Bit MODF (Mode Fault) indikuje kolizi v systému s více řadiči.

A/D převodník**ADCTL - A/D Control/Status Register**

	0	0	-	-	-	-	-	-
\$00030	CCF	-	SCAN	MULT	CD	CC	CB	CA

ADR1 - A/D Result Register 1

	0	0	0	0	0	0	0	0
\$00031	ADR17	ADR16	ADR14	ADR14	ADR13	ADR12	ADR11	ADR10

ADR2 - A/D Result Register 2

	0	0	0	0	0	0	0	0
\$00032	ADR27	ADR26	ADR24	ADR24	ADR23	ADR22	ADR21	ADR20

ADR3 - A/D Result Register 3

	0	0	0	0	0	0	0	0
\$00033	ADR37	ADR36	ADR34	ADR34	ADR33	ADR32	ADR31	ADR30

ADR4 - A/D Result Register 4

	0	0	0	0	0	0	0	0
\$00034	ADR47	ADR46	ADR44	ADR44	ADR43	ADR42	ADR41	ADR40

A/D převodník mikropočítače 68HC11 pracuje na principu vyrovnávání náboje metodou postupné aproximace. Režim převodníku je řízen obsahem registru ADCTL (\$0030), převod je realizován jako série čtyř měření, každé měření vyžaduje 32 sběrniceových cyklů, celá série

tedy 128 cyklů. Převod odstartujeme zápisem do registru ADCTL, ukončení převodu indikuje bit CCF (Conversion Complete Flag). Předčasný další zápis do registru ADCTL ukončí právě probíhající převod a odstartuje nový. Bitem SCAN (Continuous Scan Control) při jeho vynulování vyžádáme provedení jedné série čtyř měření a uložení výsledků do registrů ADR1 až ADR4, jedničkovou hodnotou bitu SCAN vyžádáme plynulý převod s cyklickým (round-robin) ukládáním výsledků měření do registrů ADR1 až ADR4. Nulová hodnota bitu MULT odpovídá sérii čtyř čtení na jediném kanálu, jedničková hodnota odpovídá čtení skupiny čtyř kanálů. Bity CD až CA řídí vstupní multiplexor, korespondenci s číslem kanálu a registrem výsledku uvádí tab.8.4.

CD	CC	CB	CA	Kanál	Výsledek
0	0	0	0	PE0	ADR1
0	0	0	1	PE1	ADR2
0	0	1	0	PE2	ADR3
0	0	1	1	PE3	ADR4
0	1	0	0	PE4	ADR1
0	1	0	1	PE5	ADR2
0	1	1	0	PE6	ADR3
0	1	1	1	PE7	ADR4
1	0	0	0	-	ADR1
1	0	0	1	-	ADR2
1	0	1	0	-	ADR3
1	0	1	1	-	ADR4
1	1	0	0	V_H	ADR1
1	1	0	1	V_L	ADR2
1	1	1	0	$V_H/2$	ADR3
1	1	1	1	-	ADR4

Tabulka 8.4: Výběr vstupního kanálu A/D převodníku

8.7 Hitachi H8/330

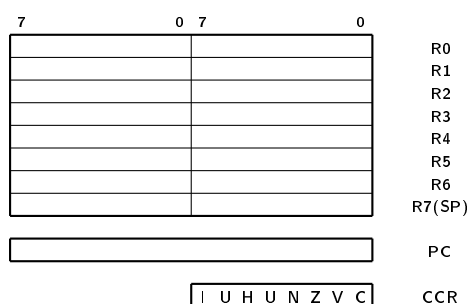
Necháme-li čtenáře porovnat přehled zde dosud uvedených osmibitových jednočipových mikropočítačů s výběrem jednočipových radičů a s výběrem šestnáctibitových jednočipových mikropočítačů v příslušných kapitolách, dojde k závěru, že procesory klasických osmibitových mikropočítačů se už dále nevyvíjejí, a že je pouze využíván potenciál zkušeností získaný prací s prvními osmibitovými procesory a s prvními osmibitovými jednočipovými mikropočítači.

Příkladem opaku, toho, že se i v této, značně stabilizované skupině projevuje vliv moderních přístupů k architektuře procesoru, může být jednočipový mikropočítač Hitachi H8/330 [9], který je přechodem mezi osmibitovými a šestnáctibitovými jednočipovými mikropočítači.

Mikropočítač Hitachi H8/330 je tvořen osmi/šestnácti-bitovým procesorem, jehož struktura je ovlivněna procesory RISC (firma ho uvádí jako „RISC-like“). Vnitřní paměť mikropočítače je tvořena pamětí programu PROM/ROM o kapacitě 16 kB a pamětí dat RAM o kapacitě 512 B. Do jediného paměťového prostoru o rozsahu 64 kB jsou mapovány i registry obvodů rozhraní a malá (16 B) dvouportová paměť RAM, která umožňuje efektivní předávání dat mezi spolupracujícími mikropočítači.

Obvody rozhraní jednočipového mikropočítače tvoří 58 I/O vývodů (z toho šestnáct je schopných přímo ovládat diody LED), osm dalších vývodů sloužících jako vstupy A/D převodníku lze použít pouze jako vstupy. Vývody I/O jsou sdruženy do devíti bran a jsou využívány i pro vytvoření vnější sběrnice. Konfigurace mikropočítače (Single-Chip, Expanded-1, Expanded-2, PROM-Programming) je definována signálem na dvou vývodech.

Sériové rozhraní může pracovat jako asynchronní nebo synchronní do rychlosti 1 Mb/s. Osmibitový převodník A/D má osm vstupů a je vybaven analogovou pamětí (obvod Sample/Hold) a čtyřmi registry vyrovnávací paměti. Jednotka časovačů se opírá o volně běžící šestnáctibitový čítač a má dva výstupy vybavené komparační logikou a čtyři vstupy vybavené záchytnými registry. Dva osmibitové čítače vybavené dvojicemi komparačních registrů lze použít pro čítání událostí nebo pro generování časových signálů, dva kanály obvodu PWM se opírají o dva další osmibitové čítače.



Obrázek 8.19: Registry procesoru jednočipového mikropočítače H8/330

Procesor pracuje s osmi šestnáctibitovými registry (jeden z nich, R7, adresuje zásobník v paměti RAM), lze však využívat i jejich osmibitové poloviny (obr.8.19). Registr příznaků dává procesorem nevyužívané bity U k dispozici aplikaci (podobně jako je tomu u 8051).

Instrukce, kterých obsahuje instrukční soubor pouze 57, mají délku od dvou do čtyř bytů. Pro adresaci operandů je k dispozici osm adresačních módů (hodnota přímo v instrukci, v registru, nepřímá adresace, indexace, nepřímá adresace s inkrementací nebo dekrementací registru, absolutní a relativní adresace, nepřímá adresace přes paměť pro instrukce skoku). Operace pracují nad bity, čtyřbitovými polovinami slabik (podpora BCD aritmetiky), osmibitovými slabikami a šestnáctibitovými slovy. Aritmetické operace nad polem registrů zahrnují i násobení 8*8 bitů a dělení 16/8 bity. Sečtení dvou operandů v registrech trvá procesoru 0.2 μ s, násobení nebo dělení 1.4 μ s. Velmi pružná je práce s jednotlivými bity. Bitové operace lze provádět nad libovolným registrem nebo buňkou paměti, bity lze adresovat i nepřímo. Instrukční soubor obsahuje i instrukci blokového přesunu, ta je schopná přenést až 64 kB.

Přerušovací systém reaguje na 9 vnějších (IRQ₀ - IRQ₇, NMI) a na 19 vnitřních zdrojů přerušení, každému zdroji přiřazuje jeden vektor, priorita přerušení je pevná.

Mikropočítač lze převést do režimů s malou spotřebou. V režimu „Sleep Mode“ je zastaven procesor, v režimu „Software Standby Mode“ i periférie. V obou případech je spuštění možné přerušením. Z režimu „Hardware Standby Mode“ se lze dostat pouze resetem, obsah paměti RAM je zachován (paměť je napájena).

Mikropočítač je vyráběn v pouzdrech QFP-80 s 80-ti vývody pro povrchovou montáž a v pouzdrech PLCC-84 s 84-mi vývody, rozložení vývodů pouzdra neuvádíme.

9. Šestnáctibitové jednočipové mikropočítače

Výpočetní výkon běžných osmibitových mikropočítačů je nedostatečný pro řadu zajímavých aplikací. Typickým příkladem je řízení vstřikování paliva a řízení zapalování spalovacích motorů nebo ochrana brzdového systému proti zablokování (systémy ABS). V těchto aplikacích je nepostačující i schopnost běžných struktur časovačů měřit časové parametry vstupních signálů a generovat výstupní signály s požadovanou přesností. Požadavky, kladené na tyto aplikace, vedly k rozvoji šestnáctibitových jednočipových mikropočítačů, jejich typickými představiteli jsou mikropočítače řady Intel 8096/80C196.

Snaha o co nejvyšší výpočetní výkon vede u šestnáctibitových mikropočítačů k volbě takových architektur procesoru, které minimalizují ztráty při přenosu mezi pracovními registry (nad nimiž lze provádět operace) a paměti RAM. Procesory Intel 8096/80C196 mají k dispozici soubor 256 pracovních registrů případně ještě s programově ovládaným okénkem (u typu 80C196KC), procesor Siemens 80C166 má k dispozici 1 kB paměti RAM, ke které lze přistupovat jako k registrům přes programově ovládané okénko o velikosti 32 B. Poměrně konzervativní návrh mikropočítače MC68HC16 vychází ze struktury procesoru MC68HC11, pouze jsou prodlouženy existující registry a přidáno několik dalších. Jednočipové mikropočítače řady Motorola 6833x se opírají o architekturu procesoru 68000 s jeho osmi 32-bitovými pracovními a osmi 24-bitovými adresačními registry. Operační jednotka šestnáctibitových jednočipových mikropočítačů je typicky vybavována násobičkou/děličkou, která je schopna realizovat šestnáctibitové násobení a dělení v jednotkách mikrosekund.

Obvody časovačů/čítačů mají větší počet záchytných a komparačních registrů připojitelných na více časových základů (volně běžících čítačů). U mikropočítače Intel 8096/80C196 se objevuje specializovaný procesor pro časové funkce označovaný jako HSI/HSO (High Speed Input/High Speed Output), který využívá paměť s organizací FIFO pro záznam vstupních událostí a paměť s asociativním výběrem pro plánování událostí výstupních.

Řízení některých obvodů rozhraní stojí cenný čas procesoru, navíc většinou v přerušovacích rutinách. Ušetřit lze autonomním řízením některých periférií, například A/D převodník mikropočítačů Motorola MC68HC16 (ale v omezené míře už i u MC68HC11) může změřit hodnoty na všech analogových vstupech bez zásahu procesoru. U mikropočítačů Intel 8096/80C196 je přenos dat mezi obvody rozhraní (A/D převodník, časový procesor) a paměti podporován řadičem PTS (Peripheral Transaction Server), který je obdobou obvodů přímého přístupu do paměti DMA.

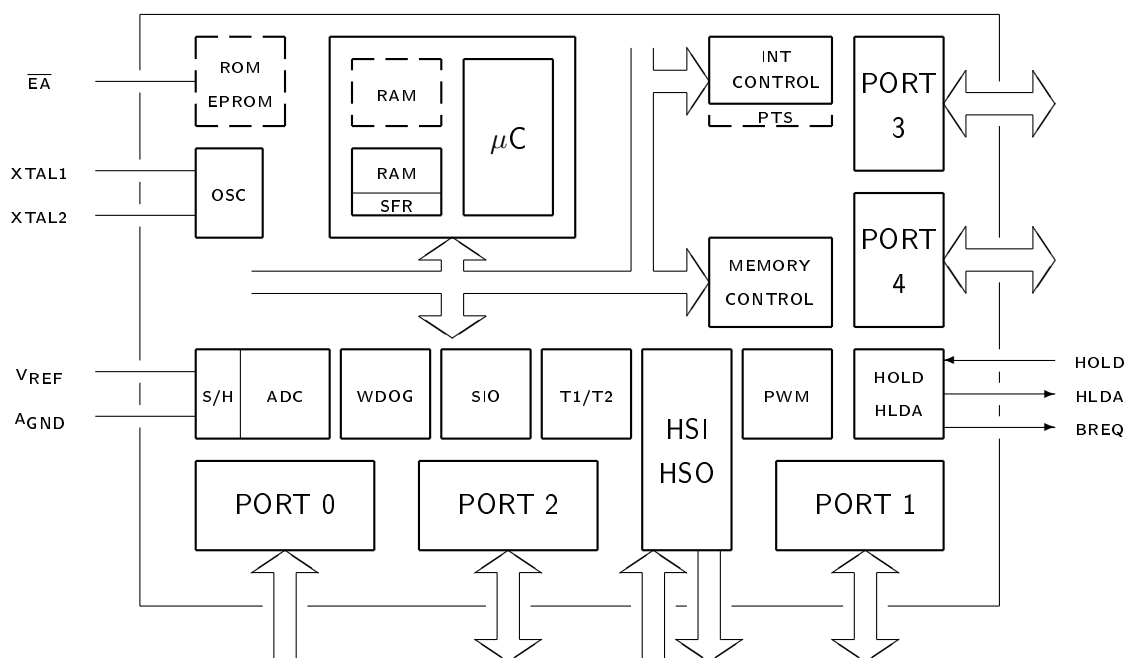
Výše uvedené skutečnosti jsou i důvodem, proč se v této třídě prakticky nesetkáme s využitím architektur procesorů dnes nejrozšířenějších osobních počítačů (s procesory 80x86). Struktura jejich procesoru má příliš velkou režii (práce s jediným střadačem, segmentace paměti), a v dnešních aplikacích typických pro šestnáctibitové jednočipové mikropočítače se příliš neprojevuje výhoda standardního vývojového prostředí.

V dalším textu si stručně popíšeme vnitřní strukturu rozšířenějších šestnáctibitových jednočipových mikropočítačů. Budou to Intel 80C196, Siemens 80C166, Motorola 68HC16 a Motorola 6833x.

9.1 Intel 80C196

Řada jednočipových šestnáctibitových mikropočítačů 80C196 [10] byla, podobně jako její předchůdce - řada 8096, navržena pro velmi rychlé aplikace reálného času v automobilech a pro výpočetně náročné aplikace, jakými jsou například řízení motorů v robotice.

Mikropočítač je vedle možnosti komunikovat s prostředím po pěti osmibitových paralelních portech vybaven standardním sériovým rozhraním, osmivstupovým desetibitovým nebo osmibitovým A/D převodníkem, zajímavými obvody pro měření času a generování signálů, až třemi kanály PWM generátoru a šestnáctibitovým čítačem obvodu Watchdog (obr.9.1). Vstupní port P0 slouží i jako osminásobný vstup A/D převodníku, porty P1 a P2 jsou obousměrné, přičemž port P2 je alternativně využit pro vstupy a výstupy sériového rozhraní, časovacích obvodů, obvodu Watchdog a přerušovacího řadiče.



Obrázek 9.1: Struktura mikropočítače Intel 80196

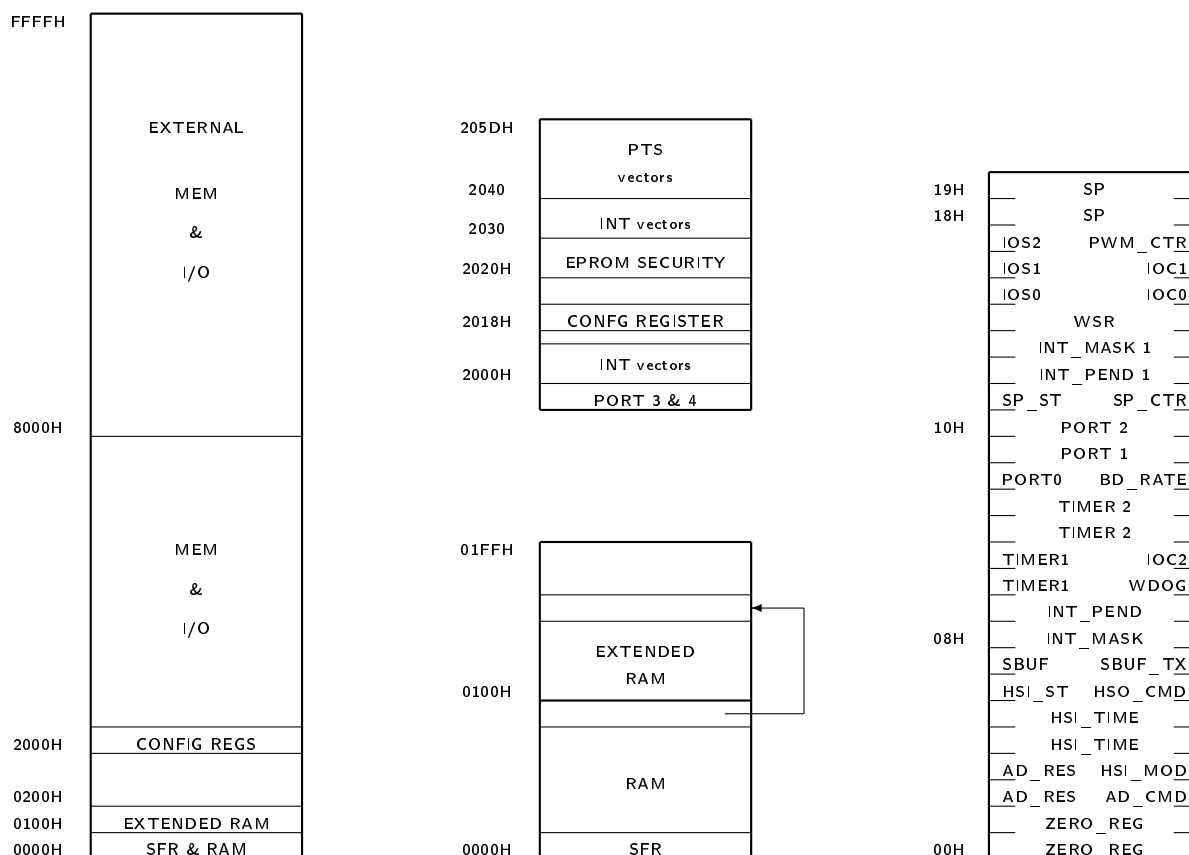
Porty P3 a P4 jsou použitelné jako obousměrné, běžně jsou využívány pro vytvoření multiplexované osmibitové nebo šestnáctibitové vnější sběrnice, která dokonce umožňuje předání řízení vnějšímu řadiči (signály HOLD/HLDA).

Jednotka časovačů/čítačů se opírá o dva šestnáctibitové čítače/časovače. Jeden z nich čítá strojové cykly ($1 \mu\text{s}$ při hodinovém kmitočtu 16 MHz), druhý může čítat buď strojové cykly, nebo změny na vstupech T2CLK nebo HSI.1 a je navíc doplněn záchytným registrem. Na tyto registry je připojen blok výstupů HSI (High Speed Input) s pamětí FIFO o kapacitě sedmi záznamů a blok výstupů HSO (High Speed Output) s „asociativní pamětí“ o kapacitě osmi požadavků. Podrobnější informace o struktuře jednotky časovačů/čítačů najde čtenář na straně 14.

Generátory PWM se opírají o osmibitový čítač, který je inkrementován v každém strojovém cyklu. Osmibitovou hodnotou v řídicím registru příslušného kanálu se zadává podíl periody, po který je na příslušném výstupu jednotková úroveň. Při požadavku na jinou základní periodu, než je 256-násobek (nebo i 128-násobek) strojového cyklu a přesnější rozlišení je nutno použít

jednotky časovačů/čítačů.

Desetibitový nebo osmibitový A/D převodník pracuje s postupnou aproximací a pro převod potřebuje 11.4 μ s při 16 MHz. Převodník je vybaven obvodem analogové paměti (Sample/Hold) a je mu předřazen osmivstupový multiplexer.



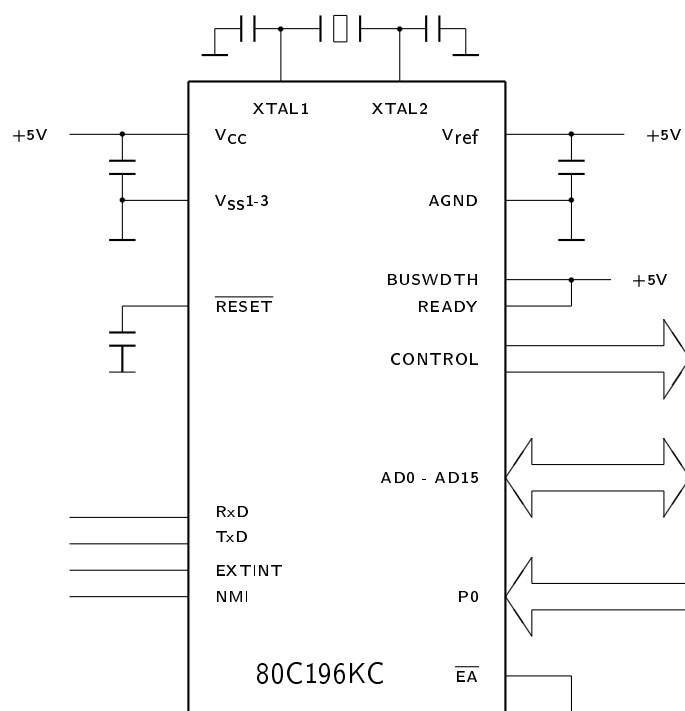
Obrázek 9.2: Paměťový prostor mikropočítače Intel 80196

Řadič přerušení reaguje na 28 zdrojů přerušení, generuje 15 vektorů pro maskovatelná přerušení a 3 vektory pro nemaskovatelné přerušení na vstupu NMI, instrukci TRAP a chybný operační kód, a pracuje s 16 úrovněmi přerušení.

V systému s více využívanými zdroji přerušení a požadovanou krátkou dobou odezvy je režie běžné přerušovací rutiny značně vysoká, vlastní obsluha přerušení přitom často spočívá pouze v uložení slabiky dat z obvodu rozhraní do paměti. V takových případech je užitečná podpora obvodu PTS (Peripheral Transaction Server) mikropočítače 80C196, jehož funkce je obdobou obvodu přímého přístupu do paměti DMA. Tento obvod dovoluje reagovat na zvolený výstup řadiče přerušení (odpovídá vektoru přerušení) přenosem jedné slabiky nebo slova, přenosem bloku slabik nebo slov, čtením výsledku A/D převodu a jeho novým spuštěním, čtením zvoleného počtu hodnot z paměti FIFO obvodu HSI nebo zápisem zvoleného počtu hodnot do asociativní paměti obvodu HSO.

Šestnáctibitový čítač obvodu Watchdog při svém přetečení vyvolá reset mikropočítače. Mikropočítač může být převeden do úsporných režimů Idle a Power-Down.

Aplikace využívající mikropočítač 80C196 nemusí být nutně konstrukčně složitá. Následující obrázek 9.3 uvádí strukturu minimálního systému s mikropočítačem 80C196KC:



Obrázek 9.3: Minimální systém s mikro počítačem Intel 80C196KC

Procesor mikro počítače 80196 na rozdíl od většiny osmibitových mikro počítačů nemá střadačovou architekturu, ale dovoluje použít libovolnou z 256 buněk z počátku paměti jako operand aritmetické nebo logické instrukce. V terminologii mikro počítače 80C196 je toto pole označováno jako soubor registrů (Register File), fyzicky je tvořeno 24 registry SFR (Special Function Registers), zbývajících 232 buněk obsazuje paměť RAM.

Soubor registrů je mapován na počátek jediného adresního prostoru mikro počítače, který má rozsah 64 kB. Od adresy 2000_H je umístěna vnitřní paměť ROM. Její začátek tvoří vektory přerušení, programový čítač je po resetu nastaven na 2080_H. Zbytek paměťového prostoru je k dispozici pro vnější paměť programu a dat a pro vnější periférie (obr.9.2).

Mikro počítač je vybaven rychlou násobičkou a děličkou, která je schopná vynásobit dvě šestnáctibitová čísla za 1.75 μ s (při hodinovém kmitočtu 16 MHz) a vydělit číslo dvaatřicetibitové šestnáctibitovým za 3 μ s (při 16 MHz).

Instrukční soubor procesoru 80C196 má 100 instrukcí, které pracují nad bity, slabikami a šestnáctibitovými a dvaatřicetibitovými slovy. Jejich přehled uvádí tab.9.1.

V tabulce jsou použity následující symboly:

D	adresovaná buňka paměti (slabika, slovo)
D, D+1	dvě po sobě následující slabiky v paměti
D, D+2	dvě po sobě následující slova v paměti
A	registr RAM
B	registr RAM
C	příznak přenosu
V	příznak přetečení
VT	pomocný příznak přetečení
ST	pomocný příznak pro zaokrouhlování

80196 - ALU operace a přesuny

Kód	Opd	Operace	Příznaky						Pozn.
			Z	N	C	V	VT	ST	
ADD/ADDB	2	$D \leftarrow D + A$	x	x	x	x	↑		
	3	$D \leftarrow B + A$							
ADDC/ADDCB	2	$D \leftarrow D + A + C$	↓	x	x	x	↑		
SUB/SUBB	2	$D \leftarrow D - A$	x	x	x	x	↑		
	3	$D \leftarrow B - A$							
SUBC/SUBCB	2	$D \leftarrow D - A + C - 1$	↓	x	x	x	↑		
CMP/CMPB	2	$D - A$	x	x	x	x	↑		
MUL/MULU	2	$D, D + 2 \leftarrow D * A$							
	3	$D, D + 2 \leftarrow B * A$							
MULB/MULUB	2	$D, D + 1 \leftarrow D * A$							
	3	$D, D + 1 \leftarrow B * A$							
DIVU	2	$D, D + 2 \leftarrow (D, D + 2) / A$				x	↑		
DIVUB	2	$D, D + 1 \leftarrow (D, D + 1) / A$				x	↑		
DIV	2	$D, D + 2 \leftarrow (D, D + 2) / A$				x	↑		
DIVB	2	$D, D + 1 \leftarrow (D, D + 1) / A$				x	↑		
NORML	2	<i>normalizace</i>	x	x	0				
EXT	1	$D \leftarrow D, D + 2 \leftarrow \text{Sign}(D)$	x	x	0	0			
EXTB	1	$D \leftarrow D, D + 1 \leftarrow \text{Sign}(D)$							
CLR/CLRB	1	$D \leftarrow 0$	1	0	0	0			
NEG/NEGB	1	$D \leftarrow 0 - D$	x	x	x	x	↑		
INC/INCB	1	$D \leftarrow D + 1$	x	x	x	x	↑		
DEC/DECB	1	$D \leftarrow D - 1$	x	x	x	x	↑		
AND/ANDB	2	$D \leftarrow D \wedge A$	x	x	0	0			
	3	$D \leftarrow B \wedge A$							
OR/ORB	2	$D \leftarrow D \vee A$	x	x	0	0			
XOR/XORB	2	$D \leftarrow D \oplus A$	x	x	0	0			
NOT/NOTB	1	$D \leftarrow \text{Not}(D)$	x	x	0	0			
SHL/SHLB	2	<i>logický posuv vlevo</i>	x	x	x	x	↑		
	SHLL		2						
SHR/SHRB	2	<i>logický posuv vpravo</i>	x	x	x	0	x		
	SHRL		2						
SHRA/SHRAB	2	<i>aritmetický posuv vpravo</i>	x	x	x	0	x		
	SHRAL		2						
SETC	0	$C \leftarrow 1$			1				
CLRC	0	$C \leftarrow 0$			0				
CLRVT	0	$VT \leftarrow 0$				0			
LD/LDB	2	$D \leftarrow A$							
ST/STB	2	$A \leftarrow D$							
LDBSE	2	$D \leftarrow A, D + 1 \leftarrow \text{Sign}(A)$							
LDBZE	2	$D \leftarrow A, D + 1 \leftarrow 0$							
PUSH	1	$SP \leftarrow SP - 2, (SP) \leftarrow A$							
POP	1	$A \leftarrow (SP), SP \leftarrow SP + 2$							
PUSHF	0	$SP \leftarrow SP - 2, (SP) \leftarrow PSW,$ $PSW \leftarrow 0000H, I \leftarrow 0$	0	0	0	0	0	0	
POPF	0	$PSW \leftarrow (SP), SP \leftarrow SP + 2,$ $I \leftarrow \uparrow$	x	x	x	x	x	x	

80196 - Skoky a řízení

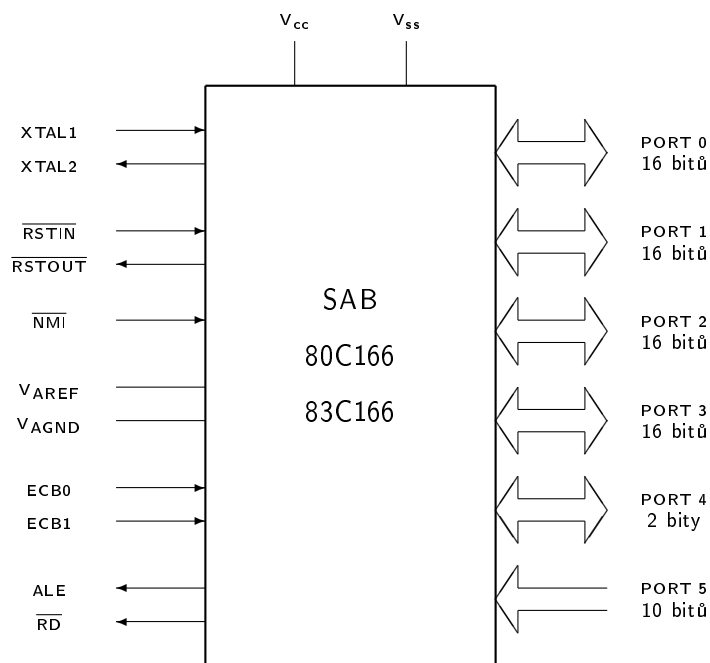
Kód	Opd	Operace	Příznaky						Pozn.
			Z	N	C	V	VT	ST	
SCALL	1	$SP \leftarrow SP - 2, (SP) \leftarrow PC,$ $PC \leftarrow PC + offset11$							
LCALL	1	$SP \leftarrow SP - 2, (SP) \leftarrow PC,$ $PC \leftarrow PC + offset16$							
RET	0	$PC \leftarrow (SP), SP \leftarrow SP + 2$							
SJMP	1	$PC \leftarrow PC + offset11$							
LJMP	1	$PC \leftarrow PC + offset16$							
BR	1	$PC \leftarrow (A), nepřímý$							
J	1	skok							
JC	1	skok při $C = 1$							
JNC	1	skok při $C = 0$							
JE	1	skok při $Z = 1$							
JNE	1	skok při $Z = 0$							
JGE	1	skok při $N = 0$							
JLT	1	skok při $N = 1$							
JGT	1	skok při $N = 0 \wedge Z = 0$							
JLE	1	skok při $N = 1 \vee Z = 1$							
JH	1	skok při $C = 1 \wedge Z = 0$							
JNH	1	skok při $C = 0 \vee Z = 1$							
JV	1	skok při $V = 1$							
JNV	1	skok při $V = 0$							
JVT	1	skok při $VT = 1, VT \leftarrow 0$					0		
JNVT	1	skok při $VT = 0, VT \leftarrow 0$					0		
JST	1	skok při $ST = 1$							
JNST	1	skok při $ST = 0$							
JBS	3	skok při bit = 1							
JBC	3	skok při bit = 0							
DJNZ	1	$D \leftarrow D - 1,$ if $D \neq 0$ $PC \leftarrow PC + offset8$							
RST	0	$PC \leftarrow 2080H$	0	0	0	0	0	0	
DI	0	$I \leftarrow 0$					0		
EI	0	$I \leftarrow 1$					0		
TRAP	1	$SP \leftarrow SP - 2, (SP) \leftarrow PC,$ $PC \leftarrow (2010H)$							
NOP	0	$PC \leftarrow PC + 1$					0		
SKIP	0	$PC \leftarrow PC + 2$					0		

Tabulka 9.1: Instrukční soubor mikroočítače 80196

9.2 Siemens 80C166

Velice zajímavě řešeným, ale bohužel méně známým, jednočipovým mikropočítačem je Siemens SAB80C166 [11]. Mikropočítač je velmi rychlý a je vybaven poměrně velkou vnitřní pamětí RAM. Jeho velmi dobře vyváženou sestavu obvodů rozhraní tvoří dvě skupiny časovačů/čítačů GPT1 a GPT2, jednotka časových funkcí CAPCOM, desetivstupový desetibitový A/D převodník, dvě sériová rozhraní, obvod Watchdog a pružný přerušovací systém doplněný o procesor PEC (Peripheral Event Controller). Mikropočítač 80C166 nemá vnitřní paměť ROM pro program, typ 83C166 je vybaven vnitřní pamětí ROM o kapacitě 8 kB.

Mikropočítač může pracovat v jedné ze čtyř možných konfigurací: bez vnější sběrnice (Single Chip Mode), s multiplexovanou sběrnicí o šířce datové sběrnice osm bitů nebo šestnáct bitů a konečně s nemultiplexovanou sběrnicí o šířce datové sběrnice šestnáct bitů. Konfiguraci mikropočítače definují elektrické úrovně na vývodech ECB0 a ECB1 při resetu mikropočítače.



Obrázek 9.4: Jednočipový mikropočítač SAB 83C166 v módu Single Chip

Při konfiguraci bez vnější sběrnice (obr.9.4, což přichází v úvahu pouze u typu 83C166, který má vnitřní paměť ROM) máme k dispozici čtyři paralelní šestnáctibitové brány P0 až P3, dvoubitovou bránu P4 a desetibitovou bránu P5, která je využitelná pouze pro vstupy.

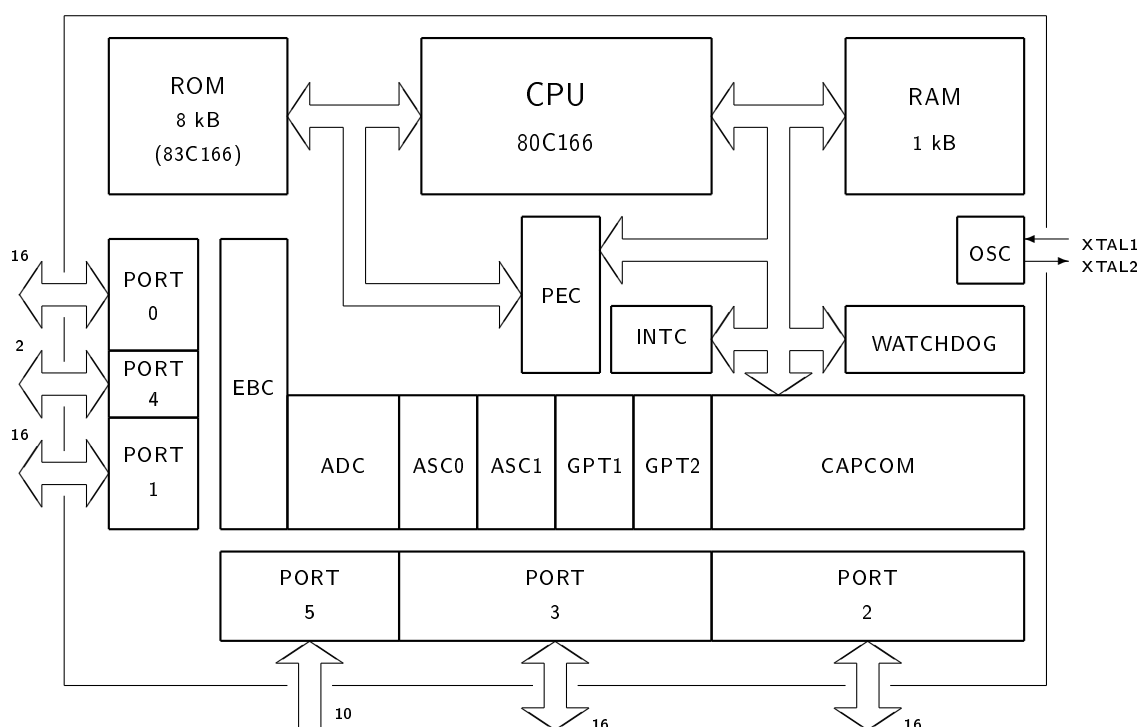
Brána P0 alternativně slouží jako multiplexovaná sběrnice adresy a dat nebo jako sběrnice dat pro nemultiplexovanou sběrnici. Brána P1 je využívána pro vytvoření sběrnice adresy pro nemultiplexovanou sběrnici. Nejvyšší řády adresy využívají dvoubitovou bránu P4. Brána P2 slouží alternativně jednotce časových funkcí CAPCOM, brána P3 zpřístupňuje vývody časovačů/čítašů, vstupy a výstupy sériových rozhraní a řídicí signály vnější sběrnice. Brána P5 slouží alternativně jako vstupy A/D převodníku.

Signál $\overline{\text{RSTIN}}$ dovoluje resetovat mikropočítač, výstup $\overline{\text{RSTOUT}}$ zůstává po resetu na nule do provedení instrukce EINIT. Signál ALE řídí multiplex adresy a dat v režimech multiplexované

vnější sběrnice, signál \overline{RD} je aktivován při čtení z vnější paměti programu nebo dat. Konečně, signál \overline{NMI} je vstupem nemaskovatelného přerušení.

Jednotka CAPCOM (obr.3.8 na str.14) je tvořena dvojicí šestnáctibitových čítačů T0 a T1, na které je připojeno pole šestnácti registrů, které mohou pracovat jako záchytné nebo jako komparační. Vstupním signálem čítače T0 a T1 může být buď vydělený kmitočet oscilátoru (nejvyšší rozlišení je 400 ns) nebo výstup čítače T6 jednotky GPT1. Čítač T0 lze navíc synchronizovat vnějším signálem T0IN/P3.0. Výstupy jednotky lze řídit jednotlivými komparačními registry, pro každý je možné definovat nastavení na jedničku, nulu nebo na změnu předchozí hodnoty. Lze dokonce využít dvou komparačních registrů pro řízení jednoho výstupu.

Jednotku univerzálních časovačů/čítačů tvoří pět šestnáctibitových čítačů rozdělených do dvou bloků: GPT1 a GPT2 (GPT - General Purpose Timer). Schéma propojení čítačů v blocích je takové, že dovoluje vytvořit řadu i dosti komplikovaných časových funkcí.



Obrázek 9.5: Vnitřní struktura mikropočítače SAB 80C166

Desetibitový analogově-číslíkový převodník má předřazen desetivstupový multiplexor a analogovou paměť (Sample/Hold). Je schopen realizovat jeden převod za $9.75 \mu\text{s}$. Lze ho naprogramovat na jednorázový nebo kontinuální převod jediného kanálu, nebo na jednorázový nebo kontinuální převod na zvolené skupině kanálů. Přenos mezi A/D převodníkem a pamětí může realizovat procesor PEC (Peripheral Event Controller).

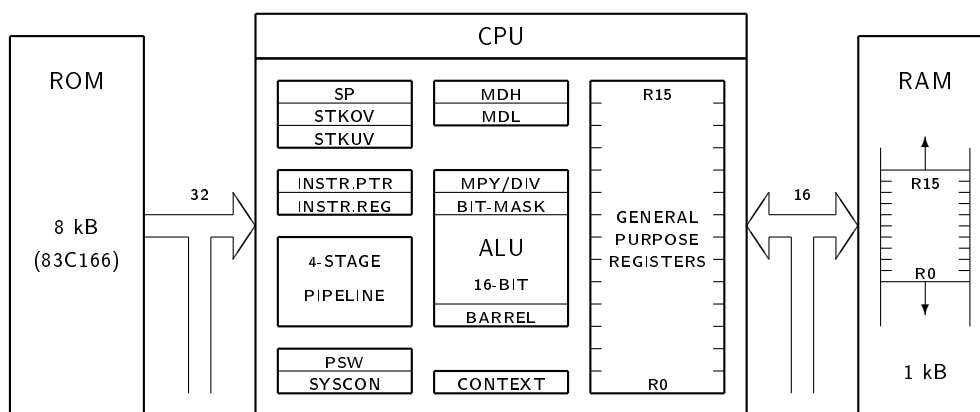
Mikropočítač je vybaven dvojicí sériových rozhraní ASC0 a ASC1, která jsou kompatibilní s rozhraním mikropočítače 8051. Dovolují práci až do 625 kb/s (v plném duplexu) nebo do 2.5 Mb/s (v poloduplexu). K dispozici jsou všechny módy činnosti, které najdeme u mikropočítače 8051, včetně devítibitové multiprocesorové komunikace. Navíc najdeme malá rozšíření. Sériová rozhraní jsou vybavena dvěma samostatnými generátory přenosové rychlosti.

Dohlízeč obvodu Watchdog je tvořen šestnáctibitovým čítačem, vstupní kmitočet je $f_{\text{osc}}/4$ nebo $f_{\text{osc}}/256$, významnějších osm bitů čítače lze nastavit. Při dočítání k nule je mikropočítač resetován, volbou vstupního kmitočtu a počáteční hodnoty čítače lze čas vypršení obvodu

Watchdog nastavit v rozmezí 25 μs až 420 ms. Čítač je automaticky startován po ukončení resetu (instrukce EINIT) a nelze ho vypnout.

Přerušovací systém dovoluje reagovat na 32 zdrojů přerušení ve vnitřních obvodech rozhraní, každému z nich lze přidělit jednu z šestnácti úrovní priority. Přerušení mohou být obsloužena běžnou přerušovací rutinou, rychlejší obsluhu dovolí osmikanálový procesor PEC (Peripheral Event Controller), který je obdobou systému DMA.

Kromě vnitřních zdrojů přerušení procesor reaguje na řadu výjimek. Patří sem nemaskovatelné vnější přerušení, překročení mezí zásobníku, nedefinovaný operační kód, nesprávná adresace a další. K dispozici je i programové přerušení (instrukce TRAP).



Obrázek 9.6: Struktura procesoru mikropočítače SAB 80C166

Procesor využívá řetězového zpracování instrukcí (pipelining), provedení typické instrukce odpovídá čtyři fáze. Při kmitočtu krystalu 40 MHz (hodiny procesoru 20 MHz) je každých 100 ns ukončeno zpracování jedné jednoduché instrukce (přesun, aritmetická nebo logická operace). Díky jednotce posuvů (Barrel Shifter) to platí i pro posuvy a rotace o více bitů. Násobení (16*16 bitů) potřebuje 500 ns, dělení (32/16) potřebuje 1 μs . Skokové instrukce vyžadují 200 ns (důvodem je latence skoku), zrychlení u opakovaných skoků (cykly) na 100 ns umožňuje paměť instrukce, která byla cílem posledního skoku, „Jump Cache“. Procesor má poměrně bohatý repertoár 78 instrukcí a možnost přímé a nepřímé adresace a konstanty jako operandu. Délka instrukce je jedna nebo čtyři slabiky.

Vnitřní paměť ROM (u mikropočítače 83C166) má kapacitu 8 kB, vnitřní paměť RAM má kapacitu 1 kB. Obě paměti a registry periférií, pro které je vyhrazeno 512 B, jsou mapovány do jediného paměťového prostoru o rozsahu 256 kB (struktura adresačních registrů dovoluje v budoucnosti rozšířit paměťový prostor až na 16 MB).

Do paměti RAM je adresováno okénko šestnácti šestnáctibitových registrů R0 - R15 (paměť RAM je realizována jako dvoubránová) adresované registrem CP (Context Pointer). Okénka se mohou překrývat a být použita pro efektivní předávání parametrů podprogramům. Poloviny registrů lze pochopitelně používat samostatně (jako RL0 - RL7 a RH0 - RH7).

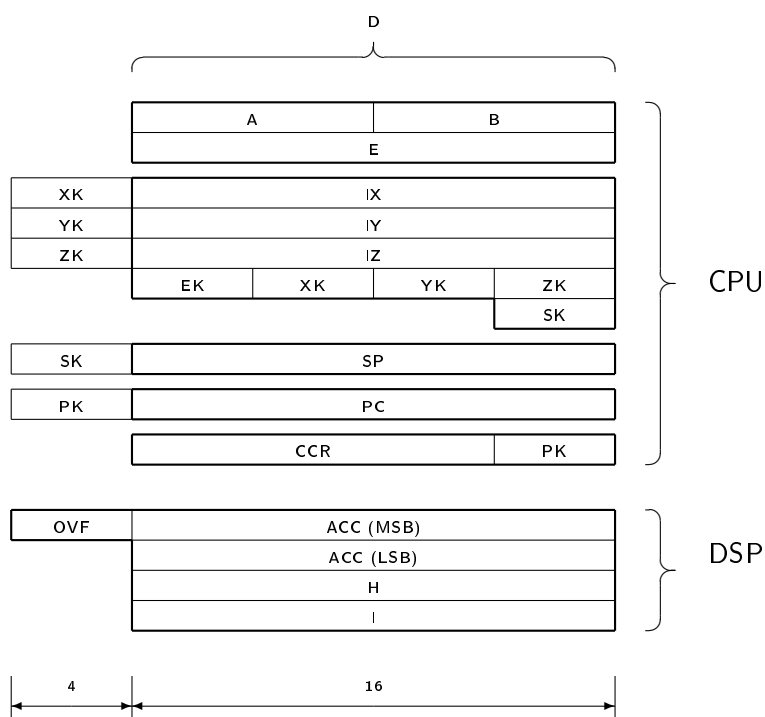
Mikropočítač 80C166 (83C166) je vyráběn technologií CMOS se šíří hradla 1.2 μm . Má poměrně velké množství vývodů a je pouzdrěn do pouzdra QFP-100.

9.3 Motorola MC68HC16

Šestnáctibitový jednočipový mikropočítač 68HC16 [12] je, pokud se jedná o instrukční kód, kompatibilním rozšířením osmibitového mikropočítače 68HC11. Struktura procesoru je doplněna o další indexační registr IZ, k šestnáctibitovému střadači D (složenému z osmibitových střadačů A a B) je doplněn další šestnáctibitový střadač E. Registry pro adresaci (IX, IY, přidaný IZ, SP a PC) jsou prodlouženy o čtyři bity, procesor dovoluje adresovat až 1 MB programu a 1 MB dat.

Šířka vnější sběrnice je šestnáct bitů, instrukční repertoár je doplněn o řadu šestnáctibitových instrukcí, ale k dispozici jsou i instrukce, které dovolují operovat s 32 bity. Výkon procesoru je až desetkrát vyšší než výkon procesoru 68HC11.

Procesor je doplněn o jednotku DSP (Digital Signal Processor) se dvěma šestnáctibitovými registry H a I, 36-bitovým střadačem ACC a se samostatnou násobičkou a sčítačkou. Výkon jednotky DSP lze srovnávat s výkonem slabších signálových procesorů.



Obrázek 9.7: Struktura procesoru mikropočítače 68HC16

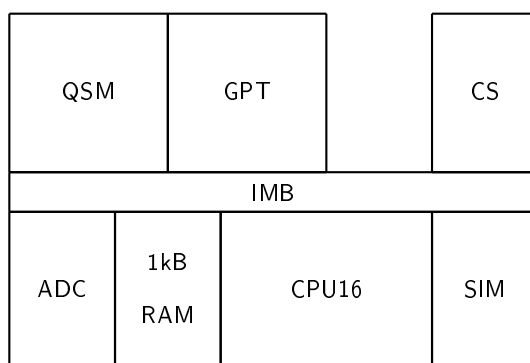
Mikropočítač je, vedle obvodů rozhraní, které jsou obdobou těch, které najdeme u 68HC11, vybaven obvodem vyvinutými pro dvaatřicetibitovou řadu 68300. Modul GPT (General Purpose Timer Module) obsahuje volně běžící šestnáctibitový čítač se třemi záchytnými a čtyřmi komparačními registry a s jedním registrem, který lze použít jako záchytný nebo komparační, dále osmibitový čítač událostí (Pulse Accumulator) a dva kanály PWM (Pulse-Width Modulation). Kromě této jednotky, která je obdobou čítačových obvodů 68HC11, jsou některé modifikace 68HC16 vybaveny velice výkonnou jednotkou čítačů/časovačů označovanou jako TPU (Time Processing Unit). Ta má šestnáct nezávisle programovatelných kanálů, jejich řízení má na starosti mikrořadič, jehož funkci lze definovat parametry v paměti RAM sdílené s procesorem.

Proti mikropočítači 68HC11 je podstatně rozšířen modul sériových přenosů. Jednotka označovaná jako QSM (Queued Serial Module) obsahuje asynchronní sériové rozhraní pracující do rychlosti 0.5 Mb/s a řadič sériové sběrnice SPI, který je vybaven frontou na až 16 požadavků a výstupy pro ovládání až čtyř periferních obvodů.

Desetibitový A/D převodník má osm vstupů, je poměrně rychlý (osmibitový převod zvládne za 8 μ s, desetibitový za 9 μ s) a je vybaven skupinou osmi výstupních registrů. Ty dovolují realizovat opakovaný převod na jednom, čtyřech nebo osmi kanálech bez zásahu procesoru, čas mezi měřeními lze definovat v několika krocích.

Připojování vnějších paměťových obvodů a obvodů rozhraní zjednodušuje pružně programovatelný adresní dekódér s dvanácti výstupy. Generátor periodického přerušení je podporou pro aplikace pracující v reálném čase. Spolehlivost aplikací výrazně zvyšuje dohlížecí systém Watchdog (COP - Computer Operates Properly) a monitor neočekávaných žádostí o přerušení (SIM - Spurious Interrupt Processor). Propojení procesoru s paměťmi a obvody rozhraní odpovídá řešení sběrnice řady 68300 (IMB - Inter-Module Bus).

Řadu 68HC16 tvoří několik typů, struktura nejjednoduššího z nich, jednočipového mikropočítače MC68HC16Z1, odpovídá obr.9.8.



Obrázek 9.8: Struktura jednočipového mikropočítače MC68HC16Z1

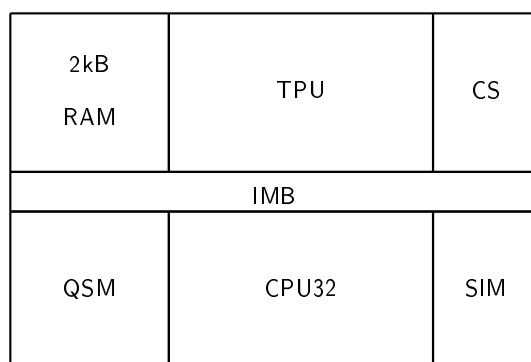
Typ MC68HC16Z2 má paměť RAM o kapacitě 2 kB a vnitřní paměť ROM o kapacitě 8 kB. Typ MC68HC16Y1 má navíc jednotku TPU a paměť ROM o kapacitě 48 kB, u typu MC68HC916Y1 je paměť ROM nahrazena pamětí FLASH EEPROM.

Obvody řady 68HC16 mají poměrně velký počet vývodů a jsou pouzdrěny do pouzder QFP-120, QFP-132 a QFP-160.

9.4 Motorola MC68332

Podobně, jako jsme ukončovali kapitolu týkající se osmibitových mikroočítačů mikroočítačem Hitachi H8/330, který by bylo vhodnější radit mezi mikroočítače šestnáctibitové, tak i nyní uzavřeme přehled šestnáctibitových mikroočítačů mikroočítačem, který by pro strukturu svého procesoru mohl patřit mezi mikroočítače dvaatřicetibitové.

Jednočipový mikroočítač MC68332 je nejznámějším členem řady mikroočítačů MC68300. Vnitřní struktura procesoru mikroočítače MC68332 je 32-bitová (32-bitové registry), šířka vnitřní i vnější datové sběrnice se však omezuje na šestnáct bitů.



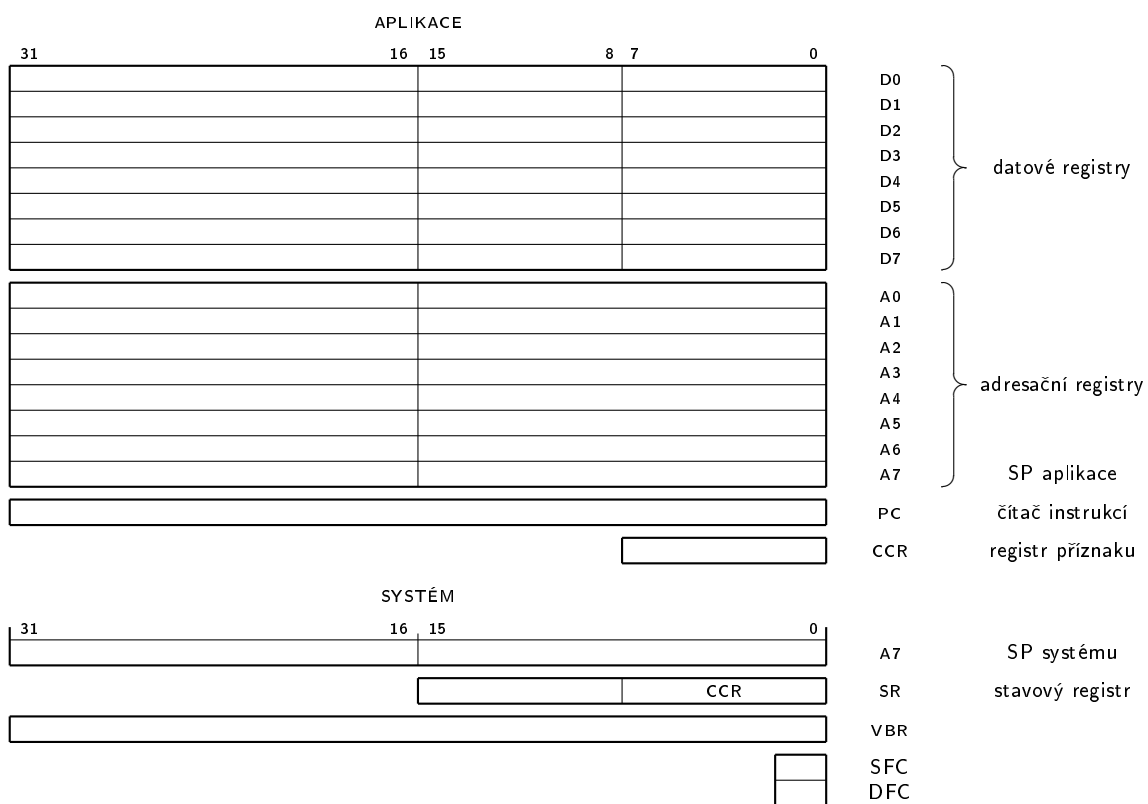
Obrázek 9.9: Struktura mikroočítače Motorola MC68332

Mikroočítač je tvořen (obr.9.9) 32-bitovým procesorem označovaným jako CPU-32, vnitřní pamětí RAM s kapacitou 2 kB, procesorem časových funkcí TPU (Time Processing Unit) a komunikačním modulem QSM (Queued Serial Module). Vzájemné propojení zajišťuje synchronní multiprocessorová vnitřní sběrnice IMB (Inter-Module Bus). Šířka je šestnáct bitů dat a čtyřicet bitů adresy. Moduly TPU a QSM realizují řadu funkcí autonomně, bez zásahu procesoru. Vytvoření vnější sběrnice, programovatelné generování signálů výběru pro vnější obvody rozhraní, zpracování signálů vnějšího přerušení a generování hodin pro vnější obvody zajišťuje modul EBI (External Bus Interface).

Procesor CPU-32 vychází z řady řady procesorů Motorola 68000 (nejblíže je mu procesor Motorola 68010). Opírá se o skupinu osmi univerzálních datových a osmi adresačních registrů o délce 32 bitů (obr.9.10). Možnost práce v chráněném systémovém režimu podstatně zvyšuje spolehlivost aplikací.

Procesor časových funkcí obsluhuje šestnáct kanálů vybavených záchytnými registry, komparační logikou a programovatelnými výstupními obvody. Komunikační modul zajišťuje asynchronní a synchronní sériovou komunikaci, je vybaven podpůrnými časovači a pamětí FIFO.

V současné době zahrnuje rodina mikroočítačů Motorola MC68300 následující prvky: MC68330 (CPU32), MC68331 (CPU32, GPT, QSM), MC68332 (CPU32, 2kB RAM, TPU, QSM) a MC68340 (CPU32, 2 čítače/časovače, 2 kanály UART, 2 kanály DMA). Mikroočítače jsou pouzřeny do pouzder QFP-132, QFP-144 a QFP-160.



Obrázek 9.10: Registry procesoru mikropočítače Motorola MC68332

10. Dvaatřicetibitové mikropočítače

Šestnáctibitové jednočipové mikropočítače pokrývají současné potřeby řízení v reálném čase. Oblast aplikací, na kterou již však nestačí jednak výpočetní kapacitou, jednak rozsahem použitelné paměti, je zpracování zvukového a obrazového signálu (případně signálů ultrazvukových a rentgenových snímačů ve zdravotnictví, radarových systémů v dopravě a vojenství). Řada těchto aplikací je doménou signálových procesorů vybavených výkonnou aritmetickou jednotkou. Jiné vyžadují provést množství operací nad rozsáhlými daty a to je typická doména 32-bitových mikroprocesorů a mikropočítačů.

Podobně, jako jsme uváděli aplikace v automobilech jako typické pro šestnáctibitové jednočipové mikropočítače, typickou aplikací pro mikropočítače 32-bitové je řízení kopírek a laserových tiskáren. Tyto aplikace kladou vysoké nároky na rychlost procesoru, musí pracovat s velkým pamětovým prostorem. Typickou úlohou je převod popisu stránky v jazyce Postscript na bitovou mapu odpovídající rozlišení mezi 300 bpi (běžné laserové tiskárny) a 1200 bpi (laserové tiskárny kvalitou odpovídající fotosazbě). Vzhledem k nemožnosti vyhnout se externí paměti RAM a k ne příliš vysokým typickým požadavkům na obvody rozhraní není tlak na výrobu specializovaných 32-bitových mikropočítačů příliš silný, v aplikacích jsou dnes využívány běžné 32-bitové mikroprocesory.

Nejčastěji se v jednoúčelových aplikacích setkáme s mikroprocesory Intel 80960, LSI Logic LR33000 a AMD 29050. Jedná se o prvky s velmi moderní vnitřní strukturou, zde si uvedeme pouze jejich nejvýraznější rysy.

Procesory řady Intel 80960 mají superskalární architekturu, při třístupňovém překrývání lze využít vedle základní celočíselné ALU ještě operační jednotku pro celočíselné násobení a dělení, operační jednotku pro plovoucí řádovou čárku a obvody pro zkrácení latence skoků. Procesor pracuje s blokem šestnácti registrů, které jsou přepisovány do vnitřní paměti RAM o kapacitě 1 KB při volání podprogramů. Instrukční cache se stupněm asociativity 2 má kapacitu 512 B až 1 KB. Některé z členů řady jsou vybaveny správou paměti a obvody DMA.

Procesor LSI Logic LR33000 vznikl modifikací procesorů MIPS R2000 a R3000. Zpracované instrukce je rozloženo do pěti fází, které se překrývají. Procesor pracuje s 32 registry, je vybaven pamětí cache pro data o kapacitě 1 KB a pamětí cache pro instrukce o kapacitě 8 KB. Procesor je doplněn o trojici čítačů, z nichž jeden podporuje vnější dynamickou paměť.

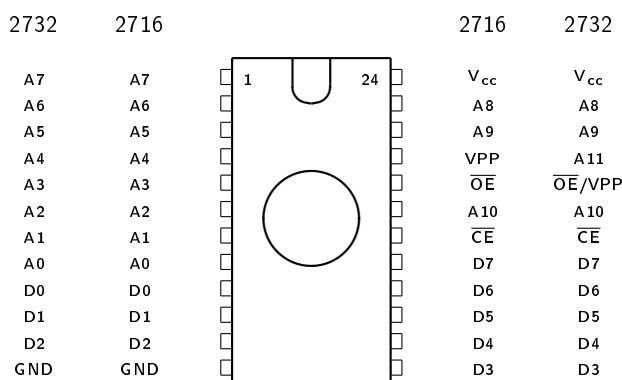
Konečně, často používaný procesor AMD Am29050 má čtyřfázové překrývání. Je vybaven jednotkou pro plovoucí řádovou čárku. Pracuje se souborem 192 registrů, které efektivně přepíná při volání podprogramů nebo při změně kontextu. Není vybaven klasickou pamětí cache, ale má paměť cache pro instrukce, které byly cílem skoku.

Omezujícím faktorem při výběru současných 32-bitových mikroprocesorů pro vestavné aplikace je nutnost vytvoření velmi rychlé a dosti komplikované sběrnice a paměťového subsystému. K tomu abychom mohli mluvit o skutečných jednočipových mikropočítačích také chybí efektivní obvody rozhraní, které jsme mohli poznat u osmibitových a šestnáctibitových mikropočítačů (čítače a časovače, moderní převodníky A/D).

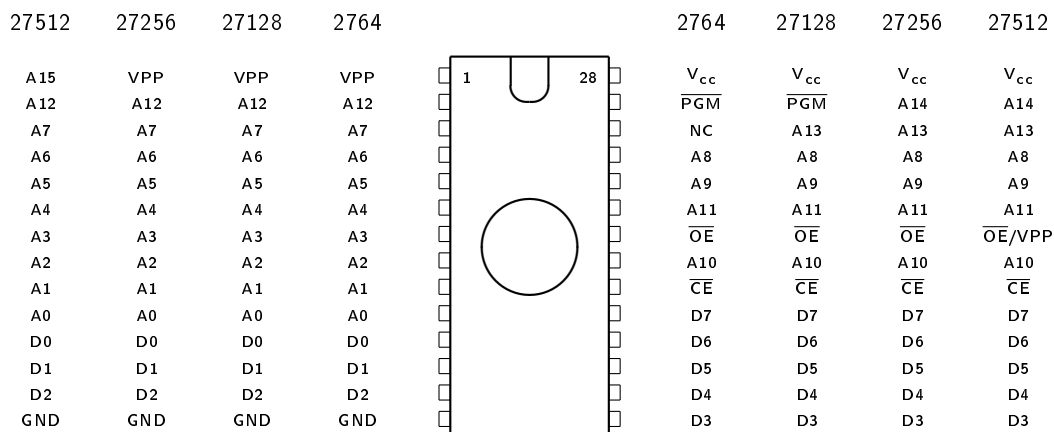
serverů pro konstrukci hlavní paměti, se pro jejich větší spotřebu (nutnost obnovování zapsané informace) v aplikacích jednočipových mikropočítačů prakticky nesetkáme.

11.2 Paměti EPROM

Paměti EPROM (Erasable Programmable Read-Only Memory) jsou pravidelně používány pro uložení programu v aplikacích mikropočítačů. Přehled nejběžnějších osmibitových typů a rozložení jejich vývodů uvádí obr.11.3 až obr.11.5.



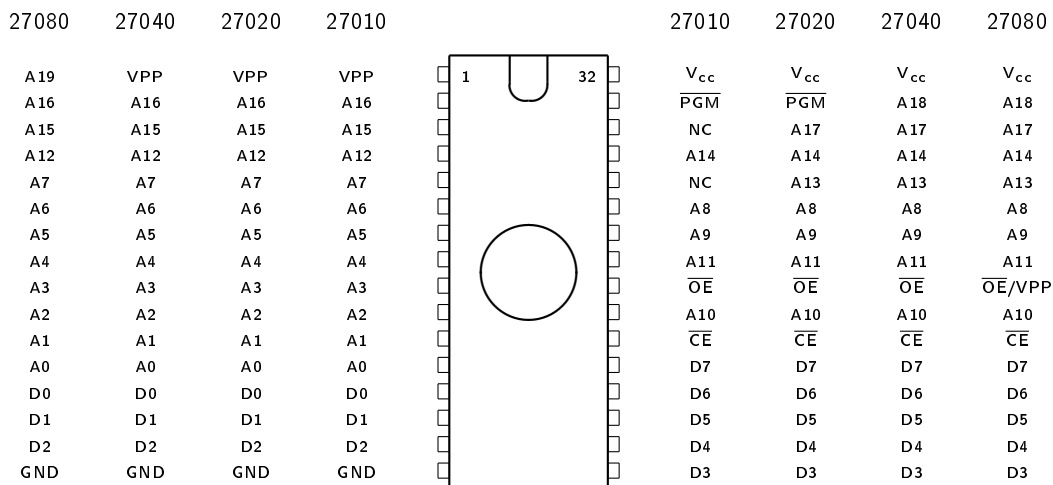
Obrázek 11.3: Rozložení vývodů paměti EPROM v pouzdrech DIP-24



Obrázek 11.4: Rozložení vývodů paměti EPROM v pouzdrech DIP-28

Signály $A_0 - A_{\max}$ adresují paměťovou buňku, vývody $D_0 - D_7$ slouží jako výstupy dat při čtení a vstupy při programování. Signál \overline{CE} aktivuje paměťový obvod, slouží k jeho výběru, signál \overline{OE} otvírá výstupní budiče obvodu.

Paměťové buňky paměti EPROM jsou tvořeny tranzistory MOSFET ovládanými nábojem zanechaným na jejich izolovaném hradle při programování. Náboj na hradle lze smazat na všech buňkách paměti současně několikaminutovým působením ultrafialového světla (vlnová délka 2537 Å), obvody proto musí být vybaveny okénkem z materiálu, který toto záření propustí. Pokud je podstatné udržení co nejnižší ceny a nepředpokládáme opakované programování (a tedy nepředpokládáme mazání), můžeme použít levnější obvody pouzdřené do plastových pouzder bez okénka označované jako OTP (One-Time Programmable).



Obrázek 11.5: Rozložení vývodů paměti EPROM v pouzdrech DIP-32

Při programování smazané buňky (obsah 0FF_{H}) je u obvodů 2716 - 27128 a u obvodů 27010 - 27020 potřeba přivést na vývod V_{PP} určené programovací napětí (v současné době převážně 12.5 V) a případně zvýšit napětí U_{CC} na 6 V (Fast-Programming Algorithm). Vlastní naprogramování paměťové slabiky adresované vývody $A_0 - A_{\text{max}}$ na hodnotu danou hodnotami signálů na vývodech $D_0 - D_7$ na vybraném obvodu ($\overline{\text{CE}}=0, \overline{\text{OE}}=0$) zajistí krátký impuls (pod 1 ms) nebo série impulsů na vývodu $\overline{\text{PGM}}$. U obvodů 27256 - 27512 a 27040 - 27080, které vývod $\overline{\text{PGM}}$ nemají, je programování řízeno pulsem nebo sérií pulsů na vývodu $\overline{\text{CE}}$.

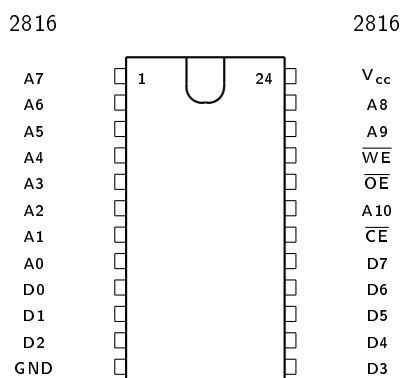
Protože se programovací postupy (! ale i programovací napětí !) u jednotlivých typů různých výrobců liší, musí být programátory paměti EPROM schopné identifikovat výrobce a konkrétní typ. Identifikaci výrobce ($A_0=0$) a konkrétního typu ($A_0=1$) lze číst při přivedení programovacího napětí na vývod A_9 .

11.3 Paměti EEPROM

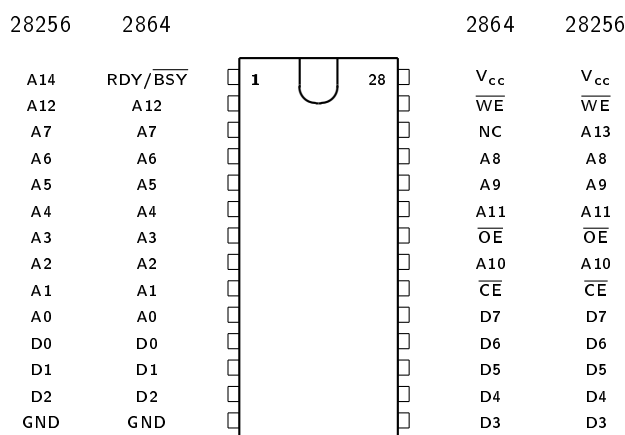
Mazání paměti EPROM většinou vyžaduje vyjmutí paměťového obvodu ze zařízení a časově náročné vystavení ultrafialovému světlu, navíc na dobu často o dost delší než je nezbytné nutné (mazačky se současnou kontrolou obsahu jsou výjimkou). Obvody EEPROM (Electrically-Erasable Programmable Read-Only Memory) jsou vhodnou alternativou tam, kde potřebujeme cykl programování zkrátit, nebo kde vyžadujeme snadnou výměnu (modifikaci) programu v aplikaci. Na rozdíl od paměti EPROM mazaných ultrafialovým světlem a paměti FLASH mazaných elektricky jako celek, je u paměti EEPROM možné (vedle kompletního smazání) i přepisování hodnot jednotlivých slabik.

Paměti EEPROM jsou zajímavé nejen pro uložení programu nebo jeho části, ale i pro uložení malého počtu parametrů. Paměti EEPROM pro uložení programu mají větší kapacitu (nad 2 kB), jsou rozložením vývodů obdobou paměti EPROM a vyžadují připojení na běžnou vnější paralelní sběrnici. Paměti EEPROM pro uložení parametrů mají obvykle kapacitu do 1 kB a jsou (vyhneme-li se historickým prvkům) doplňovány sériovým rozhraním. Pro komunikaci s mikropočítačem vyžadují pouze dva až tři signálové vodiče. Mohou pak být pouzřeny do pouzder s minimálním počtem vývodů. V dalším textu je budeme označovat jako „sériové“ paměti EEPROM.

Jako příklad paměti EEPROM s paralelním rozhraním si uvedeme typy 2816 (2 kB), 2864 (8 kB) a 28256 (32 kB).



Obrázek 11.6: Rozložení vývodů paměti EEPROM v pouzdrech DIP-24



Obrázek 11.7: Rozložení vývodů paměti EEPROM v pouzdrech DIP-28

Signály $A_0 - A_{\max}$ adresují konkrétní slabiku, vývody $D_0 - D_7$ slouží jako výstupy a vstupy dat. Výstup dat z aktivovaného obvodu ($\overline{CE}=0$) při čtení povoluje signál \overline{OE} ($\overline{OE}=0$). Zápis dat na adresovanou slabiku aktivovaného obvodu ($\overline{CE}=0$) je startován signálem \overline{WE} ($\overline{WE}=0$). Takové ovládání připomíná zápis do paměti RAM, u paměti EEPROM však pouze startuje autonomní vnitřní automat, který se postará o smazání starého obsahu paměti a o její přeprogramování. Mikropočítač může během této doby pokračovat ve výpočtu programu, ukončení zápisu do buňky EEPROM zjistíme čtením programované hodnoty (Data Pooling, nejméně významný bit je do ukončení zápisu negován), nebo čtením signálu na zvláštním vývodu RDY/\overline{BSY} .

Sériové paměti EEPROM mají zpravidla menší kapacitu, typickými představiteli jsou obvody 93C46 (128 B), 93C56 (256 B) a 93C66 (512 B) pro sériovou sběrnici MicroWire a 24C01 (128 B), 24C02 (256 B) a 24C04 (512 B) pro sériovou sběrnici I²C.

Obvody určené pro sběrnici MicroWire lze konfigurovat pro délku slova 8 bitů ($ORG=0$) nebo 16 bitů ($ORG=1$), signál CS aktivuje paměťový obvod ($CS=1$). Pověly, adresy a zapisovaná data vstupují sériově po vodiči DI, čtená data vystupují na vodiči DO. Sériový přenos je řízen hodinami CLK. Vlastní zápis do buňky paměti je startován po ukončení přenosu dat. Ukončení zápisu lze testovat opakovaným výběrem obvodu ($CS=1$), obvod odpovídá výstupem $DO=0$ až do ukončení zápisu. Sériové paměti shora uvedených typů dovolují následující funkce:



Obrázek 11.8: Rozložení vývodů sériových pamětí EEPROM v pouzdrech DIP-8

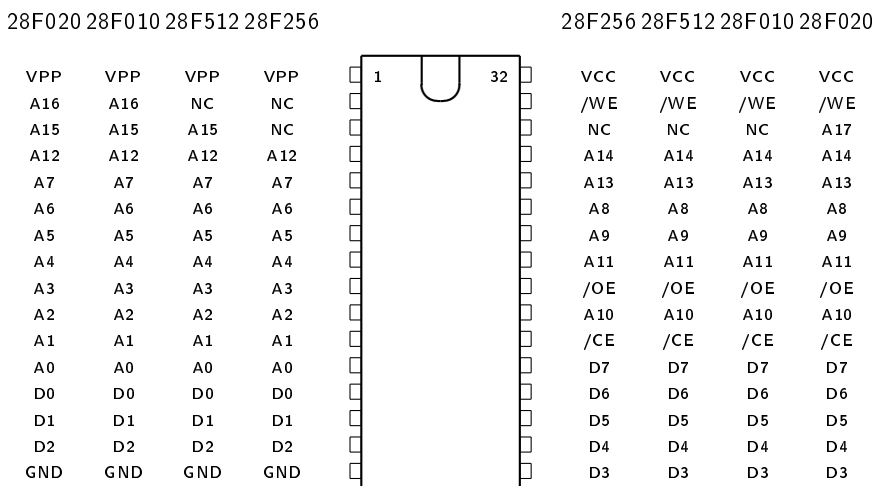
- READ - čtení adresované buňky paměti,
- EWEN - povolení zápisu do paměti,
- EWDS - zákaz zápisu do paměti,
- WRITE - zápis adresované buňky,
- WRITEALL - zápis do všech buněk paměti,
- ERASE - smazání obsahu adresované buňky paměti,
- ERASEALL - smazání všech buněk paměti.

Obvody určené pro sběrnici I²C jsou řízeny signály SCL (hodiny) a SDA (data). Napětovými úrovněmi na vývodech A0, A1 a A2 lze rozlišit více paměťových obvodů ovládaných jedinou sběrnicí I²C. Zápis a čtení odpovídá protokolu sběrnice I²C. Jediným příkazem lze zapisovat jednotlivé slabiky (Byte-Program Mode) nebo bloky dat (Page-Program Mode). Po ukončení přenosu je odstartován vlastní zápis do paměti řízený vnitřním řadičem, během zápisu obvod neodpovídá na adresaci, negeneruje prvek ACK na vodiči SDA.

11.4 Paměti Flash EEPROM

Alternativou k pamětem EEPROM pro větší kapacitu jsou paměti označované jako Flash (EEPROM). Na rozdíl od pamětí EEPROM je lze programovat jako běžné paměti v programátorech nebo zvláštním postupem přímo v zařízení.

Kapacita běžně vyráběných pamětí Flash je v současnosti v rozsahu od 32 kB do 256 kB. Paměti jsou dodávány v pouzdrech DIP, PLCC a TSOP, ale také jako paměťové karty PCMCIA s kapacitami 1, 2 a 4 MB. Obrázek 11.9 uvádí rozložení vývodů pamětí FLASH v pouzdrech DIP.



Obrázek 11.9: Rozložení vývodů pamětí Flash EEPROM v pouzdrech DIP-32

Čtení paměti Flash plně odpovídá čtení běžných pamětí EPROM a EEPROM. Mazání je elektrické, využívá Fowler-Nordheimova tunelového jevu. Smazání paměti vyžaduje typicky jednu sekundu. Pro mazání a programování je potřeba (na rozdíl od paměti EEPROM) přivést na vývod V_{PP} programovací napětí (standardně 12 V, novějším obvodům postačí 5 V).

Přivedení programovacího napětí zpřístupňuje příkazový registr, další postup mazání. Programování a verifikační čtení je řízeno příkazy, které zadává mikropočítač po datových vodičích do tohoto registru (na rozdíl od moderních pamětí EEPROM, kde je zápis řízen vestavěným autotatem). Příkaz ke smazání paměti je dvoubytový, je následován prodlevou 10ms a ověřením, že všechny buňky paměti byly smazány. Není-li paměť smazána dokonale, je nutné postup opakovat. Navíc, před zahájením mazání (uvedení všech buněk do stavu $0FF_H$) musí být všechny buňky paměti naprogramovány na hodnotu 00_H (pro rovnoměrné rozložení mazacího proudu). Příkaz zápisu je také dvoubytový, po něm následuje prodleva 10 μs a ověření, že buňka byla naprogramována správně.

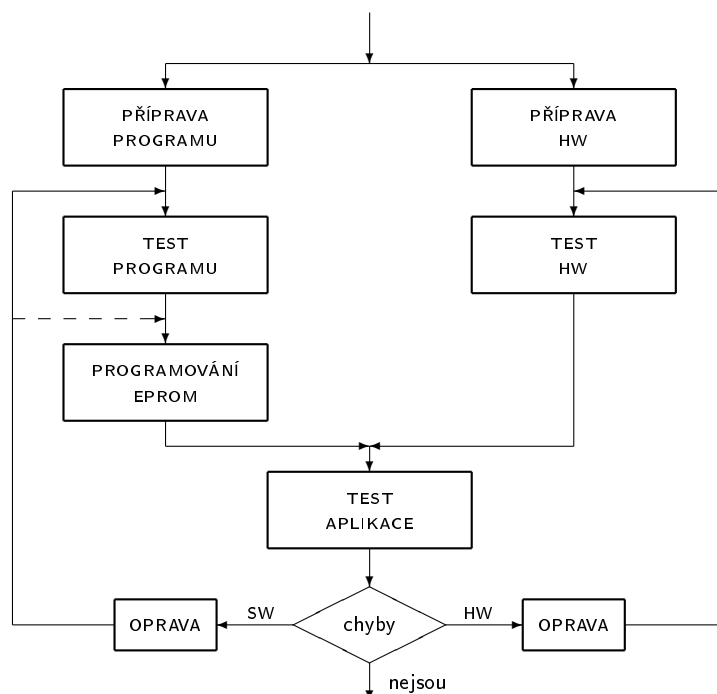
12. Vývojové prostředky

V okamžiku, kdy máme připravené technické prostředky aplikačního systému a programy přeložené a předběžně otestované na vývojovém systému, nastává poměrně obtížná fáze — *integrace* technických a programových částí. Je nutné odstranit chyby, které mají svůj původ v nepřesné specifikaci, v nepřesné informaci programátora o funkci technických prostředků a v nepřesné představě technika o možnostech a požadavcích aplikačního programu. Složitost této fáze zvyšuje i to, že technické prostředky nebyly dokonale otestovány a doladění programu probíhá často na funkčním vzoru, jehož spolehlivost není vždy nejvyšší, a který není pro ladění vybaven.

Doladění aplikačního systému si lze podstatně usnadnit, můžeme-li se opřít o zařízení, jako jsou *emulátory paměti*, *logické analyzátory* a *obvodové emulátory*. Funkci těchto zařízení a jejich použití při ladění aplikačních systémů je věnována následující kapitola.

12.1 Ladění bez podpůrných prostředků

Nejprimitivnějším postupem při přípravě programu aplikace je jeho částečné odladění na vývojovém systému (pod tímto označením zde rozumíme systém s mikropočítačem odpovídajícím cílové aplikaci) nebo na simulátoru mikropočítače, bez vazby na vnější prostředí. Simulačně ověřený program pak na programátoru vložíme do paměti EPROM a aplikaci otestujeme. Po analýze chybového chování upravíme program a případně i technické prostředky a postup opakujeme do odstranění zjistitelných chyb (obr.12.1).

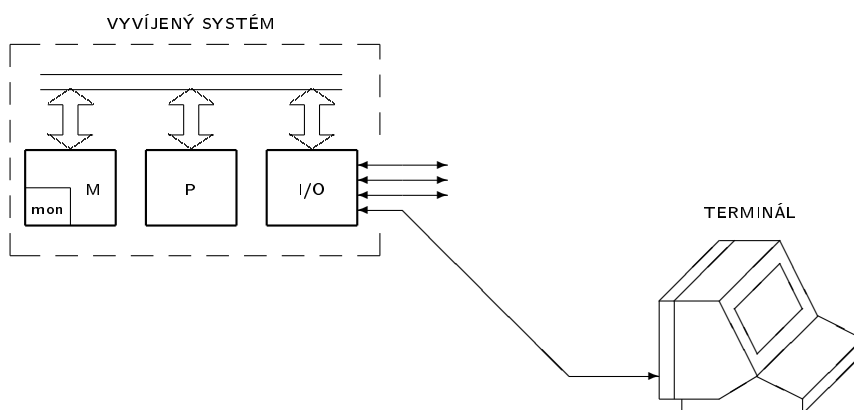


Obrázek 12.1: Příprava programu pro aplikační systém

Uvedený postup lze použít u jednoduchých aplikací (jednoduché regulátory, sběr analogových dat), s jejichž konstrukcí máme tolik zkušeností, abychom případné chyby technických prostředků nebo programu mohli lokalizovat pouze na základě sledování vnějšího chování

(výstupy akčních členů, průběhy signálů v dostupných bodech zjišťované logickou sondou, osciloskopem nebo logickým analyzátozem).

U složitějších aplikací je při doladování systému zpravidla potřebné sledovat běh programu, hodnoty proměnných a jejich změny v průběhu výpočtu. Toho lze nejsnadněji dosáhnout doplněním vyvíjeného systému o terminál (nebo jeho emulátor na osobním počítači) a jednoduchý ladící program — *monitor* (obr.12.2). Monitor nám dovolí zjistit hodnoty proměnných v registrech procesoru a v paměti a případně je modifikovat, zjistit a modifikovat hodnoty v registrech periferních obvodů, zjistit adresu programového čítače v okamžiku přerušení aplikace a odstartovat výpočet aplikace od zvolené adresy. Běžná je i schopnost analýzy programu (jeho výpis v jazyce symbolických instrukcí) a, pokud to paměť programu dovoluje (RAM nebo EEPROM) i modifikace programu. Pro připojení terminálu (nebo jeho emulátoru na vývojovém pracovišti, kterým je obvykle osobní počítač) nám postačí i primitivní, programově obsluhované, sériové rozhraní. U aplikačních systémů vybavených jednoduchým alfanumerickým zobrazovačem a klávesnicí lze vnější terminál nahradit alternativním využitím těchto prvků. Potřebné rozšíření o monitor a terminál nás v řadě případů mnoho nestojí (často stačí pouze doplnit kód monitoru a rozhraní terminálu), pouze u jednoduchých řadičů se můžeme setkat s nepřekonatelnými omezeními.



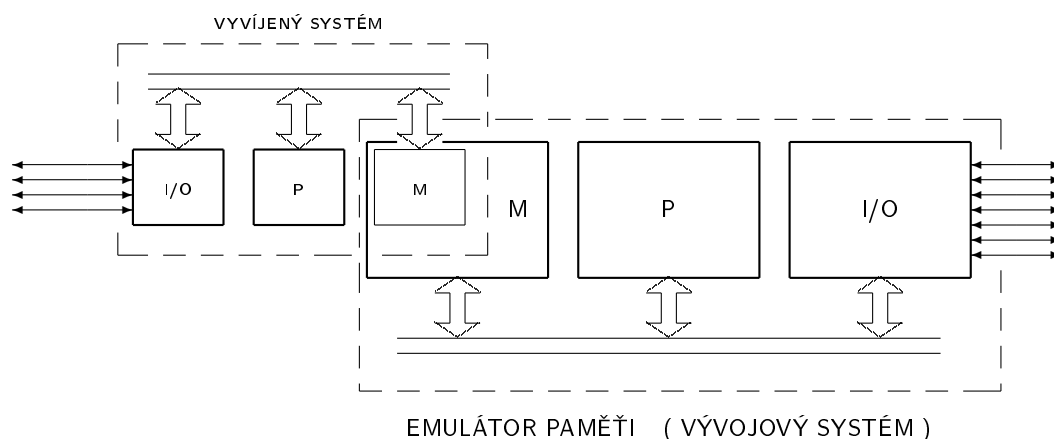
Obrázek 12.2: Doplnění aplikačního systému o vývojové prostředky

Ladění programu bez podpory, nebo pouze s podporou monitoru, je poměrně pomalé, po zjištění chyby v programu musíme přeprogramovat paměť programu (smazat a znovu naprogramovat paměť EPROM). Časové nároky lze snížit, pokud použijeme pružnější technologii paměti programu (EEPROM nebo Flash EEPROM), nebo pokud můžeme aplikační program umístit do paměti RAM a upravovat ho z připojeného terminálu nebo počítače. Poslední možnost ovšem obvykle znamená, že funkční vzor vyvíjeného zařízení je pro potřeby ladění vhodně doplněn.

12.2 Emulace paměti programu

Podstatně pružnějším prostředkem práce s aplikačním programem je náhrada vnější paměti programu aplikace pamětí specializovaného zařízení — *emulátoru paměti*. Funkci tohoto zařízení si můžeme znázornit obrázkem 12.3, na kterém je oblast programu vyvíjeného systému mapována na paměť emulátoru paměti. Tato oblast paměti je běžně realizována jako dvoubránová, lze ji naplnit a modifikovat z vývojového pracoviště a číst aplikačním systémem. Připojení je běžně realizováno kabelem zapojeným do objímky vnější paměti programu vyvíjeného systému.

Paměť emulátoru může být součástí paměti vývojového pracoviště (výjimečně) nebo součástí samostatného emulátoru paměti, který komunikuje s vývojovým pracovištěm po vhodném rozhraní. Procesor samostatného emulátoru bývá pro jednoduchost realizované funkce často nahrazován jednoduchou logikou.



Obrázek 12.3: Princip emulace paměti

Emulátor paměti dovoluje (obvykle) zpřístupnit pouze prostor programu, nedává přístup k proměnným. Jejich zpřístupnění lze dosáhnout doplněním aplikačního systému o monitor a terminál.

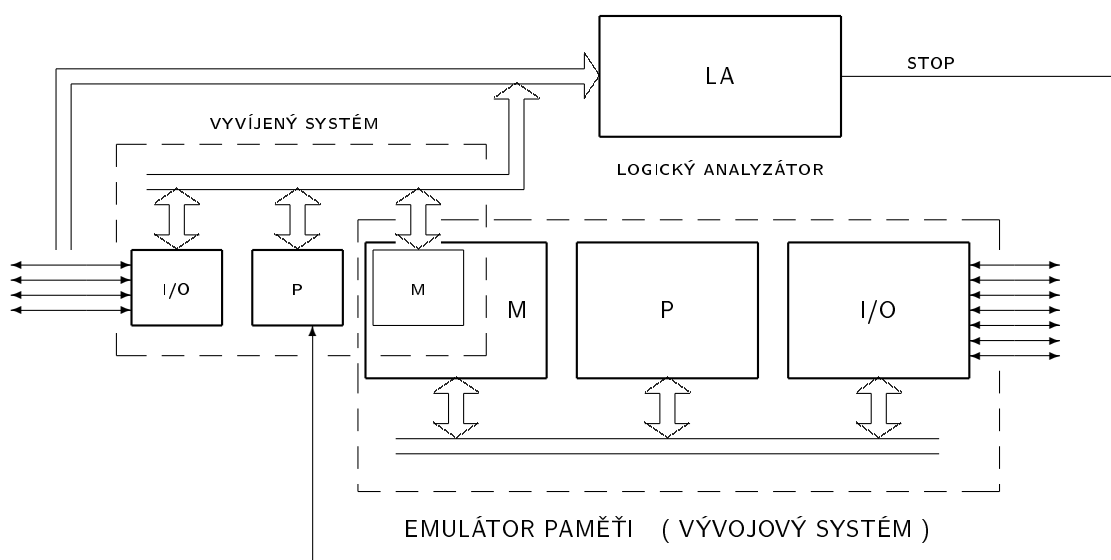
Výhodou emulátoru paměti je jeho snadná realizace jako doplňku osobního počítače. Jeho použití je však omezené na systémy, které dovolí připojení emulátoru do objímky vnější paměti programu (tedy na systémy s vnější sběrnici).

12.3 Sledování výpočtu, logická analýza

Užitečnost emulace paměti lze podstatně zvýšit, můžeme-li zastavovat výpočet aplikace při předvolené události, například při změně vstupního nebo výstupního signálu, nebo při přístupu procesoru ke zvolené buňce paměti programu nebo dat (při průchodu programem určeným místem a/nebo při přístupu ke zvolené proměnné). Toho lze nejsnadněji dosáhnout připojením logického analyzátoru k vyvíjenému systému (obr.12.4) a jeho nastavením takovým, že vyvolá ukončení záznamu do své *sledovací paměti*, popřípadě i *zastavení procesoru* aplikačního systému, při dosažení dané podmínky. Pro snadné připojení bývají logické analyzátory vybavovány sondami, které dovolují připojení analyzátoru na pouzdro mikropočítače případně na pouzdro vnější paměti.

Logický analyzátor, kromě schopnosti identifikovat podmínku zastavení a generovat signál zastavení pro aplikaci (STOP), ukládá sledované informace do sledovací paměti. Po zastavení tak máme k dispozici, kromě informace, že výpočet vyvolal předpokládanou akci, informaci o průběhu předcházejícího výpočtu ve formě záznamu řady hodnot sledovaných signálů ve sledovací paměti analyzátoru. Tyto hodnoty mohou být navíc kvalitním logickým analyzátořem prezentovány ve formě posloupnosti prováděných instrukcí. Vybavení aplikačního systému monitorem a terminálem dává pochopitelně možnost zjistit navíc podrobnosti o stavu proměnných aplikace v okamžiku zastavení.

Logický analyzátor může být v řadě případů nedostupný, možnost zastavení aplikačního programu při splnění zadané podmínky však může být pro ladění natolik zajímavá, že se vyplatí

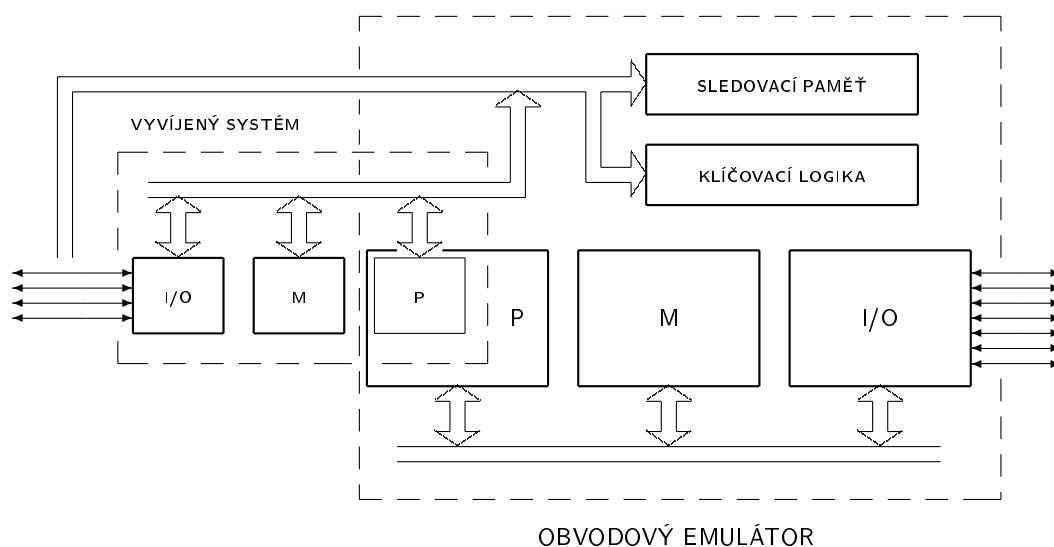


Obrázek 12.4: Sledování výpočtu logickým analyzátořem

připojit k vyvíjenému zařízení třeba i jednoduchý vícekanálový komparátor, který vygeneruje signál STOP.

12.4 Obvodová emulace

Schéma na obr.12.4 zahrnuje prakticky všechny podstatné prvky, které jsou základem *obvodového emulátoru*, označovaného častěji jako *emulátor procesoru* nebo *emulátor mikroprocesoru*. Obvodový emulátor umožňuje emulaci paměti programu ale i dat, zahrnuje programovatelnou klíčovací logiku, sledovací paměť a má schopnost analýzy stavu programu po jeho zastavení.



Obrázek 12.5: Princip obvodového emulátoru

Podstatnou odlišností od předchozího schématu je rozhraní mezi obvodovým emulátorem

Literatura

- [1] Motorola: *MC68HC05 HCMOS Single Chip Micro Programmers Reference Guide*. 1989.
- [2] SGS-Thompson: *ST6210/6215/6220/6225 Databook*. 1993.
- [3] Microchip: .
- [4] Intel: *8-bit Embedded Controller Handbook*. 1989.
- [5] Philips: *8051-based 8-bit Microcontrollers*. 1993.
- [6] Siemens: *SAB80C515/535 User's Manual*. 1990.
- [7] Siemens: *SAB80C517/537 User's Manual*. 1991.
- [8] Motorola: *MC68HC11 Reference Manual*. 1990.
- [9] Hitachi: *H8/330 Hardware Manual*.
- [10] Intel: *16-bit Embedded Controllers*. 1991.
- [11] Siemens: *SAB80C166 User's Manual*.
- [12] Motorola: *MC68HC16 Family CPU16 Reference Manual*. 1992.
- [13] Hintz K.J., Tabak D.: *Microcontrollers: Architecture, Implementation, and Programming*. McGraw-Hill, 1992.
- [14] Ayala K.J.: *The 8051 Microcontroller Architecture, Programming, and Applications*. West Publishing, St.Paul 1991.
- [15] Anceau F.: *The Architecture of Microprocessors*. Addison-Wesley, 1986.
- [16] Gansler J.G.: *The Art of Programming Embedded Systems*. Academic Press, 1992.

Rejstřík

čítač/časovač 8

- časovací procesor 13
- generátor periodického přerušení 12
- komparační registr 11
- obvod reálného času 15
- přednastavení 9
- čítač událostí 12
- Watchdog 17
- záchytný registr 10

Dallas DS80C320 84

emulátor

- paměti 127
- procesoru 129

Hitachi H8/330 104

Intel 8051 59

- paralelní rozhraní 6
- registry procesoru 61
- registry SFR 66
- strojový cyklus 61
- vnější sběrnice 58

Intel 80C196 107

- jednotka HSI 13
- jednotka HSO 13
- obvod PWM 20

logický analyzátor 128

Motorola MC68HC05 48

Motorola MC68HC11 85

- A/D převodník 103
- asynchronní sériové rozhraní 101
- čítač událostí 12, 100
- generátor periodického přerušení 12
- konfigurační registry 94
- programovatelný časovač 98
- registry rozhraní 95
- sběrnice SPI 31, 102
- vnější sběrnice 58, 87

Motorola MC68HC16 115

Motorola MC68332 117

obvod

- Power-Fail 18
- Power-On Reset 19
- PWM 20
- Watchdog 17

obvodový emulátor 129

paměti

- EEPROM 122
- EPROM 121
- Flash EEPROM 124

- RAM 120

paralelní rozhraní 6

Philips PCB80C552 71

- registry SFR 73
- Watchdog 18

PIC16C5x 54

protokol

- asynchronní sběrnice Intel 26
- asynchronní sběrnice Motorola 26
- Bitbus 27
- CAN 29
- Fieldbus 28
- Modbus 25

proudová smyčka 24

převodníky A/D 38

- ADC0831 39
- dvojí integrace 40
- integrační 40
- měření odporu 41
- postupná aproximace 38
- vyrovnávání náboje 41

převodníky D/A 42

- PWM generátor 43

rozhraní

- RS-232C 21, 22
- RS-422 23
- RS-423 22
- RS-485 24

sběrnice

- CAN 29
- Fieldbus 28
- I²C 34
- MicroWire 32
- SPI 31

Siemens SAB80166 112

- jednotka CAPCOM 14

Siemens SAB80515/535 80

- A/D převodník 38
- Watchdog 17

Siemens SAB80517/537 82

SGS-Thompson ST62 50

- čítač/časovač 8
- paralelní rozhraní 7

zobrazovač

- LCD 46
- LED 44
- LED - obvody M545x, M548x 45

doc. Ing. Jan Janeček, CSc.

PROJEKTOVÁNÍ MIKROPOČÍTAČOVÝCH SYSTÉMŮ

V roce 2022 vydalo České vysoké učení technické v Praze
První elektronické vydání

ISBN 978-80-01-06973-8