

Mikrokontroléry

ATMEL

s jádrem 8051

Ján Klúčik

Vojtech Fronc



TECHNICKÁ
LITERATURA
BEN

Publikace je určena především studentům a začátečníkům, kteří se rozhodli zabývat problematikou monolitických mikrokontrolérů ATMEL. Současně je snahou o zaplnění mezery na knižním trhu v této oblasti a možná přijde vhod i vyspělým pracovníkům v mikroprocesorové technice.

Vojtech Fronc, Ján Klůčik

MIKROKONTROLÉRY ATMEL s jádrem 8051

Bez předchozího písemného svolení nakladatelství nesmí být kterákoli část kopírována nebo rozmnožována jakoukoli formou (tisk, fotokopie, mikrofilm nebo jiný postup), zadána do informačního systému nebo přenášena v jiné formě či jinými prostředky.

Autoři a nakladatelství nepřejímají záruku za správnost tištěných materiálů. Předkládaná zapojení a informace jsou zveřejněny bez ohledu na případné patenty třetích osob. Nároky na odškodnění na základě změn, chyb nebo vynechání jsou zásadně vyloučeny.

Veškerá práva vyhrazena.

Ing. V. Fronc, Ing. J. Klůčik, 2001

ATMEL – ochranná známka firmy ATMEL Corporation

Nakladatelství BEN – technická literatura, Věšínova 5, Praha 10

Ing. V. Fronc, Ing. J. Klůčik: Mikrokontroléry ATMEL s jádrem 8051

BEN – technická literatura, Praha 2001

1. vydání

ISBN 978-80-7300-008-3 (tištěná kniha)

ISBN 978-80-7300-397-5 (elektronická kniha v PDF)

OBSAH

ÚVOD	5
ZÁKLADNÍ PŘEHLED A ARCHITEKTURA	6
1 ARCHITEKTURA MIKROKONTROLÉRŮ ATMEL	7
1.1 Paměťový podsystem	8
1.1.1 Paměť programu	8
1.1.2 Paměť dat	9
1.1.3 Způsoby adresace	13
1.2 Centrální procesorová jednotka	14
1.3 Paralelní vstupy a výstupy	16
1.3.1 Struktura a činnost	16
1.3.2 Ošetření vstupů	19
1.4 Komunikace s externí pamětí	19
1.4.1 Externí paměť programu	20
1.4.2 Externí paměť dat	21
1.5 Sériové a paralelní EEPROM paměti	24
1.5.1 Sériové paměti EEPROM	24
1.5.2 Paralelní paměti EEPROM	27
1.6 Čítače/časovače	28
1.6.1 Čítače 0, 1	28
1.6.2 Čítač/časovač 2 (pro AT89C52...)	32
1.7 Sériový vstup/výstup	35
1.7.1 Režimy činnosti sériového kanálu	35
1.7.2 Nastavování režimů činnosti sériového kanálu	42
1.7.3 Multiprocesorová komunikace	43
1.7.4 Nastavení přenosové rychlosti čítačem 1	43
1.7.5 Použití čítače 2 jako generátoru přenosové rychlosti	44
1.8 Přerušovací systém	45
1.8.1 Obsluha přerušení	48
1.8.2 Krokování programu	49

1.9	Inicializace mikrokontroléru	50
1.10	Napájení a zálohování	52
1.11	Provoz se sníženým příkonem	54
1.12	Hodinový generátor	56
1.13	Vnitřně obvodová emulace	58
2	STRUKTURA MIKROKONTROLÉRŮ ATMEL	59
2.1	Základní typy	59
2.2	AT89C51	59
2.3	Mikrokontroléry AT89C1051, AT89C2051, AT89C4051	70
2.4	Mikrokontroléry AT89C52, AT89S8252, AT89S53, AT89C55	78
2.5	Programovatelný hlídací časovač (WDT)	85
2.6	Rozhraní sériového styku (SPI)	86
3	INSTRUKČNÍ SOUBOR	91
3.1	Přehled instrukcí mikrokontrolérů ATMEL	92
4	PROGRAMOVÁNÍ MIKROKONTROLÉRŮ ATMEL	109
5	SOFTWAREVÉ PROSTŘEDKY A LADĚNÍ PROGRAMU	111
	SEZNAM OBRÁZKŮ	113
	SEZNAM TABULEK	115
	REJSTŘÍK	116
	LITERATURA	118

ÚVOD

Mikroprocesory způsobily ve vývoji elektroniky doslova revoluci stejně, jako v minulosti první tranzistor. Vývoj postupoval od nejjednodušších dvoubitových, až ke dnešním 32bitovým resp. 64bitovým. Na rozdíl od procesorů určených pro aplikace v osobních počítačích se dynamicky rozvíjí speciální kategorie mikroprocesorů, se snahou o sloučení více funkcí na jeden čip, určených pro aplikace v oblasti řízení, měření, předzpracování dat a jiné. Obvodům, které mají na čipu integrovánu paměť programu, paměť dat, hodinový oscilátor, sériový kanál, vstupně-výstupní obvody a mnohé jiné funkce, jako A/D, D/A převodník, Watch Dog aj. již přináležejí název mikrokontrolérů, nebo jednočipové mikropočítače. Na počátku byla populární řada 8035 od fy. Intel, a později řada 8051. Díky vynikajícím vlastnostem byly, a ještě i nyní jsou tyto obvody používány ve velice širokém rozsahu. Mikrokontroléry produkují mnozí světoví výrobci elektronických součástek, přičemž někteří klonovali základní „INTELOvskou řadu“ a jiní postupovali nezávisle, vzhledem k původní řadě nekompatibilně. Tím je trh elektronických součástek obohacen o mnohé nové druhy mikrokontrolérů různých výrobců. U nás nejnámějšími byly mikrokontroléry vytvořené na základě řady 8035 a 8051 firmy INTEL.

Nabídka široké škály velice zajímavých vlastností láká kromě nových zájemců o mikroprocesorovou techniku i klasické uživatele k experimentování, studování nové problematiky, pořizování nových, nákladově poměrně náročných vývojových prostředků, a realizaci svých záměrů prostřednictvím úplně nových mikrokontrolérů, které jim skýtají to, co řada 8051 nemohla poskytnout.

Posláním této knížky je upozornit odbornou veřejnost na produkty firmy Atmel, která neustále vyvíjí mikrokontroléry kompatibilní s INTELOvskou řadou. Navzdory tomu, že firmy dodavatelů nabízejí tyto prvky a v literatuře se objevily již mnohé aplikační články, mnozí uživatelé nejsou si vědomi toho, že můžou vzít takový AT89C51, vložit ho místo klasického 8051 a on bude bez problémů pracovat. Jedná se o prvky firmy ATMEL, která má široký sortiment a za velice přijatelné ceny. V nabídce jsou mikrokontroléry různého výkonu, jak provedení DIP, tak i pro povrchovou montáž. Zajímavé jsou mezi nimi například typy 20pinové pro jednodušší aplikace.

A skutečně velice důležitá je možnost bez problémů využít vývojové prostředky pro řadu 8051. Pro ty, co již pracovali s prvky řady 8051 je to s ohledem na zkušenosti značná úspora nákladů. Navíc, aplikačních zapojení pro tyto mikrokontroléry je v literatuře mnoho.

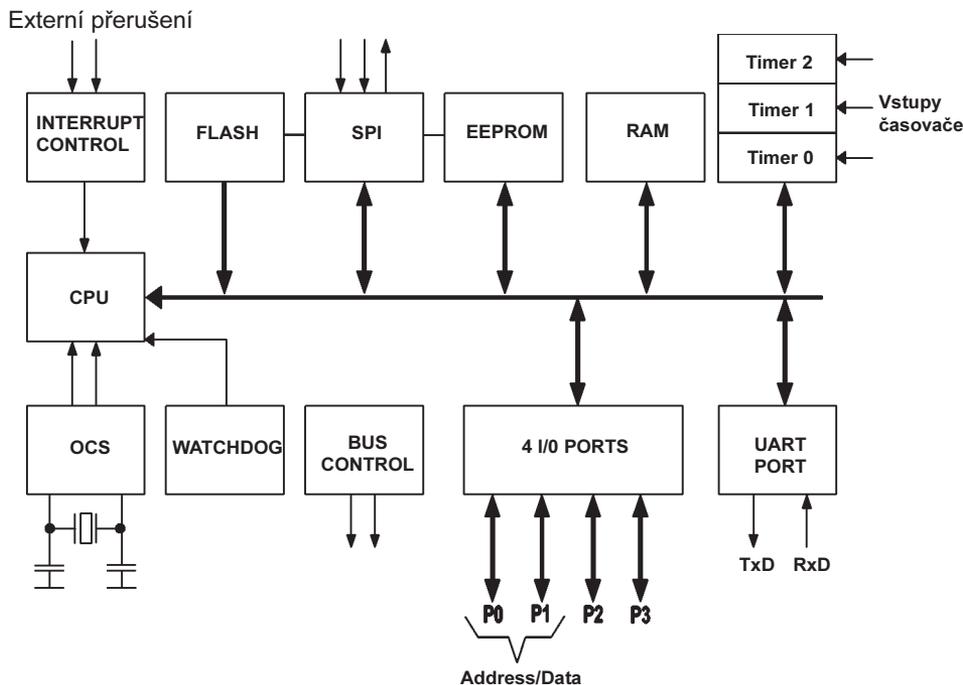
ZÁKLADNÍ PŘEHLED MIKROKONTROLÉRŮ ATMEL S JÁDREM 8051

TYP	AT89C1051	AT89C2051	AT89C4051	AT89C51	AT89C52	AT89S53	AT89C55	AT89S8252	AT89S4D12
Paměť RAM	64	128	128	128	256	256	256	256	256
Paměť Flash EPROM	1k	2k	4k	4k	8k	12k	20k	8k	128k
Paměť EEPROM	-	-	-	-	-	-	-	2k	-
Kmitočet MHz	0 – 24	0 – 24	0 – 24	0 – 24	0 – 24	0 – 24	0 – 33	0 – 24	12
IO linky	15	15	15	32	32	32	32	32	5
Počet časovačů	2	2	2	2	3	3	3	3	-
Zdroje přerušení	6	6	6	6	9	8	6	9	-
Analogový komparátor	●	●	●	-	-	-	-	-	-
SPI	-	-	-	-	-	●	-	●	●
Módy IDLE, PD	●	●	●	●	●	●	●	●	-
Dual Data Pointer	-	-	-	-	-	●	-	●	●
Programovatelný Watch Dog	-	-	-	-	-	●	-	●	-
Power Off Flag	-	-	-	-	-	●	-	●	-
Technologie CMOS	●	●	●	●	●	●	●	●	●

Přehled a vlastnosti jednotlivých typů mikrokontrolérů

1 ARCHITEKTURA MIKROKONTROLÉRŮ ATMEL

Architektura mikrokontrolérů ATMEL je znázorněna na obr. 1



Obr. 1 Blokové schéma mikrokontrolérů ATMEL

Řada AT89C sestává z těchto částí:

- 8bitová centrální procesorová jednotka (CPU);
- paměť programu (FLASH EPROM);
- datová paměť (RAM);
- paralelní programovatelné V/V obvody (4 I/O PORTS);
- blok čítačů/časovačů (TIMER0, TIMER1, ...);
- obvody časování (OSC);
- sériový V/V obvod (UART PORT);
- přerušovací podsystem (INTERRUPT CONTROL).

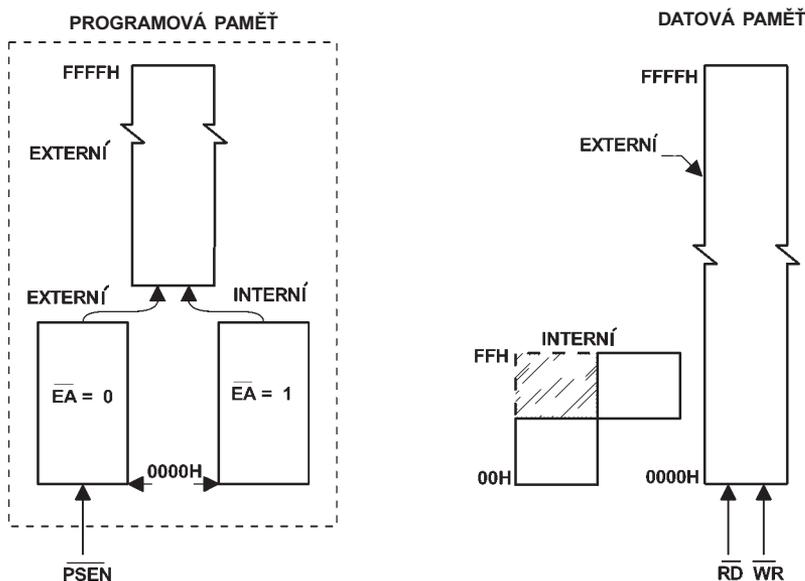
V řadě AT89S přibyly následující části:

- sériová paměť (EEPROM);
- řadič sériové sběrnice (SPI);
- hlídač činnosti (Watch Dog).

Jednotlivé části jsou propojeny interní 8bitovou sběrnici, která slouží pro přenos dat a instrukcí.

1.1 PAMĚŤOVÝ PODSYSTÉM

Všechny mikrokontroléry firmy ATMEL mají oddělen adresový prostor programu a adresový prostor dat jak je znázorněno na *obr. 2*. Jedná se tedy o Harwardské uspořádání.



Obr. 2 Architektura paměti

Toto oddělení poskytuje přístup k datům interní paměti dat prostřednictvím 8bitové adresace, čímž se zrychluje ukládání a manipulace s 8bitovou centrální procesorovou jednotkou (CPU). Navzdory tomu je možné adresovat paměť dat nepřímo, prostřednictvím 16bitového adresového ukazatele. Paměť programu je v provedení ROM, resp. FLASH EPROM. Adresový prostor paměti programu má velikost 64 kB (*obr. 2*). Výběrový impuls z externí paměti programu je nazván PSEN (Program Store Enable). Adresový prostor externí datové paměti má stejnou kapacitu 64 kB a je přímo adresovatelná v externím paměťovém prostoru. CPU generuje čtecí a zápisové impulsy, které povolují přístup k datům.

Adresový prostor externí paměti programu a externí paměti dat se překrývá a je možné ho odlišit použitím signálů RD a PSEN pro čtení. Odlišení provedeme volbou typu použité instrukce pro přístup k datům.

1.1.1 PAMĚŤ PROGRAMU

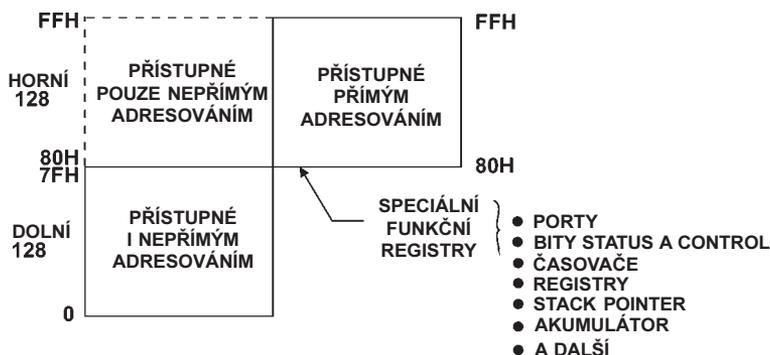
Paměť programu se dělí na interní o velikosti 1 až 20 kB v závislosti na typu mikrokontroléru (viz kapitola 2) a externí, jestli je použita. Pokud se realizuje paměť jako externí, je nutné připojit vstup EA = 0. Častěji se využívá jen interní paměť programu FLASH EPROM a signál EA se

standardně připojuje na hodnotu $\overline{EA} = 1$. Na nulté stránce programové paměti jsou rezervovány vektorové adresy obsluhy jednotlivých zdrojů přerušení (viz kap. Přerušovací systém).

Množina adres paměti programu je adresována 16bitovým čítačem instrukcí (Program Counter) – PC, který adresuje interní paměť programu v rozsahu platných adres dle kapacity interní paměti FLASH EPROM a externí paměť programu v rozsahu vyšších adres do FFFFH, přičemž je jeho obsah přenášen na výstupy portů P0 a P2.

1.1.2 PAMĚŤ DAT

Paměť dat mikrokontrolérů ATMEL je možno rozdělit na interní a externí. Interní paměť dat typu RAM má rozsah 64, 128 nebo 256 bytů. Mapa interní paměti RAM je na obr. 3.



Obr. 3 Interní datová paměť RAM

Interní datová paměť má paměťový prostor rozdělen na tři bloky:

- LOWER** – Dolní 128 (nižší část interní RAM);
- UPPER** – Horní 128 (vyšší část interní RAM);
- SPECIAL FUNCTION REGISTERS** – SFR (oblast speciálních registrů).

Adresování paměťového prostoru interní paměti je vždy 8bitové, což umožňuje adresovat 256 bytů. Oproti tomu prostřednictvím adresových módů se může přistupovat až k 384 bytům interní paměti RAM. Přímé adresování ovládá paměťový blok horní 128 (FFH-7FH) a nepřímé adresování paměťový blok SFR (FFH-7FH). Obsazují tedy stejný adresový prostor, ale fyzicky jsou odděleny. Přístup do paměťového bloku RAM Dolní 128 je realizován jak přímou, tak nepřímou adresací.

Interní paměť RAM Dolní 128 (tab. 1) obsahuje čtyři banky registrů označených RB0 až RB3 a každá banka registrů obsahuje 8 registrů R0 až R7. Uvedené banky registrů jsou umístěné na adresách 00H až 1FH. Oblast 16 následujících adres 20H až 2FH je bitově adresovatelná oblast paměti RAM, ve které je možné pomocí bitové adresace adresovat každý ze 128 bitů zvlášť (0–127). Instrukční sada obsahuje celou řadu instrukcí pro práci s bitově přístupnou oblastí paměti RAM.

Zůstávající oblast interní RAM tj. adresy 30H až 7FH jsou určeny na univerzální použití.

Skupina speciálních registrů SFR (Special Function Registers) je umístěna na adresách 80H až F0H jak je zobrazeno v tab. 2 – Mapa SFR. Obsahuje 21 (27) speciálních registrů.

HEX	(MSB)								(LSB)	DEC
FFH										255
:										:
:										:
7FH										127
:										:
:										:
3FH										63
:										:
:										:
32H										50
31H										49
30H										48
2FH	7F	7E	7D	7C	7B	7A	79	78	47	
2EH	77	76	75	74	73	72	71	70	46	
2DH	6F	6E	6D	6C	6B	6A	69	68	45	
2CH	67	66	65	64	63	62	61	60	44	
2BH	5F	5E	5D	5C	5B	5A	59	58	43	
2AH	57	56	55	54	53	52	51	50	42	
29H	4F	4E	4D	4C	4B	4A	49	48	41	
28H	47	46	45	44	43	42	41	40	40	
27H	3F	3E	3D	3C	3B	3A	39	38	39	
26H	37	36	35	34	33	32	31	30	38	
25H	2F	2E	2D	2C	2B	2A	29	28	37	
24H	27	26	25	24	23	22	21	20	36	
23H	1F	1E	1D	1C	1B	1A	19	18	35	
22H	17	16	15	14	13	12	11	10	34	
21H	0F	0E	0D	0C	0B	0A	09	08	33	
20H	07	06	05	04	03	02	01	00	32	
1FH	Banka registrů 3									31
18H										24
17H	Banka registrů 2									23
10H										16
0FH	Banka registrů 1									15
08H										8
07H	Banka registrů 0									7
00H										0

Tab. 1 Nižších 128 bytů interní RAM

bytová adresa	(MSB)							(LSB)	SFR	
0F0H	F7	F6	F5	F4	F3	F2	F1	F0	B	
	B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0		
0E0H	E7	E6	E5	E4	E3	E2	E1	E0	ACC	
	ACC.7	ACC.6	ACC.5	ACC.4	ACC.3	ACC.2	ACC.1	ACC.0		
0D5H	SPIE SPE DORD MSTR CPOL CPHA SPR1 SPR0								SPCR	
0D0H	D7	D6	D5	D4	D3	D2	D1	D0	PSW	
	CY	AC	F0	RS1	RS0	OV	-	P		
0CDH	Registr čítače 2 - horní byte								TH2	
0CCH	Registr čítače 2 - dolní byte								TL2	
0CBH	Záchytný registr čítače 2 - horní byte								RCAP2H	
0CAH	Záchytný registr čítače 2 - dolní byte								RCAP2L	
0C9H							T2OE	DCEN	T2MOD	
	CF	CE	CD	CC	CB	CA	C8	C8		
0C8H	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2	T2CON	
0B8H			BD	BC	BB	BA	B9	B8	IP	
			PT2	PS	PT1	PX1	PT0	PX0		
0B0H	B7	B6	B5	B4	B3	B2	B1	B0	P3	
	P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.3	P3.0		
	RD	WR	T1	T0	INT1	INT0	TxD	RxD		
0AAH	SPIF	WCOL								SPSR
0A8H	AF	AE	AD	AC	AB	AA	A9	A8	IE	
	EA		ET2	ES	ET1	EX1	ET0	EX0		
0A0H	A7	A6	A5	A4	A3	A2	A1	A0	P2	
	P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0		
	Adr15	Adr14	Adr13	Adr12	Adr11	Adr10	Adr9	Adr8		
99H	Příjmací/vysílací registr sériového kanálu								SBUF	
98H	9F	9E	9D	9C	9B	9A	99	98	SCON	
	SM0	SM1	SM2	REN	TB8	RB8	TI	RI		
96H	PS2	PS1	PS0	EEMWE	EEMEN	DPS	WDTRST	WDTEN	WCON	
90H	97	96	95	94	93	92	91	90	P1	
	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0		
	SCK	MISO	MOSI	SS			T2EX	T2		
8DH	Registr čítače 1 - horní byte								TH1	
8CH	Registr čítače 0 - horní byte								TH0	
8BH	Registr čítače 1 - dolní byte								TL1	
8AH	Registr čítače 0 - dolní byte								TL0	
89H	BF	BE	BD	BC	BB	BA	B9	B8	TMOD	
	1G	1C/T	1M1	1M0	0G	0C/T	0M1	0M0		
88H	8F	8E	8D	8C	8B	8A	89	88	TCON	
	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0		
87H	SMOD			POF	GF1	GF0	PD	IDL	PCON	
86H	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0	SPDR	
85H	Bázový registr DPTR1 - horní byte								DP1H	
84H	Bázový registr DPTR1 - dolní byte								DP1L	
83H	Bázový registr DPTR0 - horní byte								DP0H	
82H	Bázový registr DPTR0 - dolní byte								DP0L	
81H	Ukazovatel zásobníkové paměti								SP	
80H	87	86	85	84	83	82	81	80	P0	
	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0		
	D7/Adr7	D6/Adr6	D5/Adr5	D4/Adr4	D3/Adr3	D2/Adr2	D1/Adr1	D0/Adr0		

Tab. 2 Mapa SFR (speciálních funkčních registrů)

Střadač (Accumulator) – ACC, A

Základní pracovní registr centrální procesorové jednotky CPU, který vždy obsahuje operand instrukce a do něhož se ukládá výsledek operace. Střadač je součástí RAM (adresa E0H), což umožňuje kromě standardních operací i bitové operace (ACC.7 – ACC.0).

B registr – B

Pracovní registr, který je používán při operacích dělení a násobení. Lze ho také použít jako normální pracovní registr.

Stavový registr (Program Status Word) – PSW

PSW registr obsahuje v každém okamžiku stav procesoru a výsledek výkonu předcházející instrukce.

Ukazatel zásobníku (Stack Pointer) – SP

Ukazatel zásobníku je adresový registr interní RAM ukazující na vrchol zásobníku, ve které je uložena návratová adresa po volání instrukcí PUSH, POP a CALL. SP je inkrementován před výkonem instrukcí PUSH a CALL a dekrementován po vykonání instrukcí POP a RET. Po inicializaci je SP stále na adrese 07H což znamená, že ukazuje na adresnou buňku 08H. Jeho umístění je možné programově změnit v oblasti adresace RAM.

Ukazatel dat (Data Pointer) – DPTR

Ukazatel dat je dvoubytový adresný registr, skládá se ze dvou 8bitových registrů DPH, DPL. Jeho funkcí je ukazovat na paměťovou buňku v datové paměti v rozsahu 0000H – FFFFH. Je přístupný jako dvojitý 16bitový registr i jako dva 8bitové registry zvlášť.

Čítač instrukcí (Program Counter) – PC

16bitový čítač instrukcí, není programově přístupný.

VV porty (Ports) – P0, P1, P2, P3

Vstupně-výstupní porty P0, P1, P2, P3 jsou v oblasti SFR reprezentovány jako registry se stejným označením P0, P1, P2, P3.

Registr sériového kanálu (Serial Data Buffer) – SBUF

Registr sériového kanálu – jsou ve skutečnosti dva oddělené registry – vysílací a přijímací. Jsou-li jsou data přesunuta do registru SBUF, procházejí vysílacím registrem. Inicializace přenosu se vykoná okamžitě. Přicházející data procházejí pouze přes tento registr (SBUF) a musí být z něj přečteny.

Registry čítačů a časovačů (Timer Registers) – [TH0, TL0], [TH1, TL1]

Registry čítačů a časovačů jsou párováné 16bitové registry. Podrobněji v kapitole 6.5 Čítače/časovače.

Řídící registry IP, IE, TMOD, TCON, SCON, PCON obsahují řídící a stavové bity pro přerušovací systém, čítače/časovače, sériový kanál a režimy se sníženým příkonem, které budou podrobněji popsány v následujících kapitolách.

Registry [TH2, TL2], [RCAP2H, RCAP2L], T2CON, T2MOD, SPSR, SPCR, SPDR, [DP0L, DP0H], [DP1L, DP1H], WCON jsou použity pouze ve struktuře procesorů AT89C..52, ..53 a ..55.

Obsahy některých (celkem 16) SFR jsou adresovatelné také bitově. Jsou to ty registry, jejichž adresy v RAM jsou ukončeny na 000B (tj. na 0H nebo 8H).

Interní paměť dat se může rozšířit externí pamětí dat s velikostí až 64 kB. Externí paměť dat lze adresovat 16bitovým ukazatelem DPTR (Data Pointer) nebo s pomocí nepřímé adresace prostřednictvím adresových registrů R0 a R1 a následným pohybem v jednom z 256 bloků externí paměti dat o 256 bytech v závislosti na nastavení adresy bloku (stav portu P2). Přístup do interní nebo externí paměti dat je rozlišen typem použité instrukce.

1.1.3 ZPŮSOBY ADRESACE

Adresovací módy jsou v mikrokontrolérech určeny typem instrukce a způsobem jejího vykonání. Je možné rozdělit na:

1. **Přímá adresace** – operand je specifikován 8bitovým adresovacím polem v instrukci (např. MOV A, BUNKA, kde platí přiřazení EQU BUNKA 30H).
Přímo adresovat je možno pouze interní RAM a SFR.
2. **Nepřímá adresace** – instrukce obsahuje adresovací registr R0 anebo R1, který obsahuje adresu paměťového místa nebo ukazatel dat.
(např. MOV @R0,#00H, MOVX @DPTR,A)
Nepřímo adresovat se může jak interní tak externí RAM.
3. **Registrová adresace** – banky registrů RB0 až RB3 s registry R0 až R7, R0'–R7', ... Dolní tři bity operačního kódu instrukce určují přímo jeden z těchto registrů v závislosti na zvolené bance registrů v PSW.
(např. MOV R7,A)
4. **Adresace specifikovaného registru** – některé instrukce pracují přímo se specifikovaným registrem např. střadačem, jehož operační podkód instrukce přímo obsahuje.
(např. DAA, DIV AB)
5. **Konstanta** – následuje za operačním kódem instrukce.
(např. MOV R7,#100)
6. **Indexová adresace** – paměť programu je přístupná pouze prostřednictvím indexové adresace. Tento adresový mód je určen na čtení datových tabulek v paměti programu. 16bitový registr (DPTR nebo PC) ukazuje na počátek tabulky.

Adresa záznamu v tabulce vložená v paměti programu je vypočtena přičtením obsahu střadače k hodnotě ukazatele.

(např. `MOVC A,@A+DPTR`, `MOVC A,@A+PC`)

1.2 CENTRÁLNÍ PROCESOROVÁ JEDNOTKA

Centrální procesorová jednotka (CPU) sestává ze dvou základních částí:

- aritmeticko-logické jednotky;
- řídicí jednotky.

Aritmeticko-logická část obsahuje aritmeticko-logickou jednotku (ALU), dva dočasné registry TEMP1 a TEMP2 a střadač (ACC). Střadač obsahuje jeden z operandů a slouží na zapamatování výsledku operace, který se do střadače zapisuje přes interní sběrnici. Výsledek také ovlivňuje příznakové překlápěcí obvody v stavovém registru PSW (Program Status Word), který má tvar:

	(MSB)						(LSB)	
PSW	CY	AC	F0	RS1	RS0	OV	–	P

kde jednotlivé bity mají následující význam:

CY (Carry) – příznak přenosu, který se nastavuje při přeplněném střadači ACC, tj. když dochází u výkonu instrukce k přenosu z osmého do devátého bitu (`ACC.7 → ACC.8`), rovněž při instrukcích porovnání.

AC (Auxiliary Carry) – příznak pomocného přenosu, který se nastavuje při přenosu mezi slabiky střadače (`ACC.3 → ACC.4`). Využívá ho instrukce DAA (viz kap. 7).

F0 – příznakový bit, který je možné ovlivňovat programovými prostředky a lze ho využít k indikaci libovolné vnitřní nebo vnější události.

RS1, RS0 – volba pracovní banky registrů. Jejich význam je následující:

Banka registrů		RS1	RS0
RB0	banka registrů 0 (00H – 07H)	0	0
RB1	banka registrů 1 (08H – 0FH)	0	1
RB2	banka registrů 2 (10H – 17H)	1	0
RB3	banka registrů 3 (18H – 1FH)	1	1

Po resetu je vždy aktivní banka RB0.

Tab. 3 Banky pracovních registrů

OV – příznak přetečení, který se nastavuje při přenosu z 6. do 7. bitu ACC u aritmetických instrukcí s využitím 2. doplňku. Indikuje, že výsledek operace se znaménkem není možné umístit do 8 bitů.

P – příznak parity, který se nastavuje $P = 1$, když obsah střadače ACC má lichou paritu. Pro sudou paritu je $P = 0$. Když se přepíše obsah registru PSW, bit P zůstává nezměněn, protože vždy označuje paritu střadače ACC.

Součástí centrální procesorové jednotky jsou také speciální registry a to:

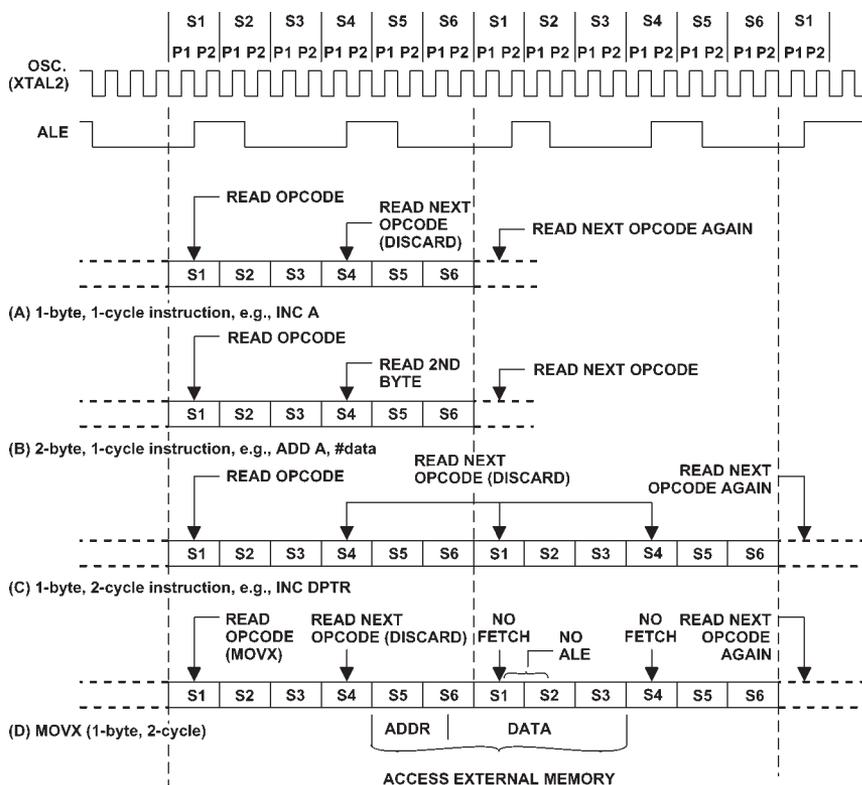
- 8bitový registr B, který se používá při instrukcích dělení a násobení;
- 8bitový ukazatel vrcholu zásobníku SP;
- 16bitový čítač instrukcí PC;
- 16bitový ukazatel dat DPTR (registrový pár [DPH, DPL]);

Řídící jednotka obsahuje obvody, které produkují synchronizační signály pro časování vnitřních činností mikrokontroléru a nezbytné synchronizační signály pro spolupráci s externími obvody.

Základní časování se dělí na instrukční a strojové cykly. Instrukční cyklus (čas potřebný na výběr a vykonání jedné instrukce) sestává z jednoho nebo dvou strojových cyklů. Každý strojový cyklus sestává ze šesti stavů označených S1 až S6. Trvání jednoho stavu je vymezeno dvěma periodami základního kmitočtu oscilátoru. Tedy jeden strojový cyklus má trvání 12 period oscilátoru nebo přesně 1 mikrosekundu, jestliže je kmitočet oscilátoru 12 MHz.

Uvedené časové relace jsou znázorněny na obr. 4, kde je také zobrazena realizace různých typů instrukcí.

Každý stav S1 až S6 je rozdělen na dvě fáze P1 a P2.



Obr. 4 Instrukční cyklus

Instrukce s délkou 1 byte, která se realizuje v průběhu jednoho strojového cyklu probíhá tak, že ve stavu S1 se načítá operační kód instrukce a instrukce se začíná vykonávat. Ve stavu S4 se formálně čte operační kód následující instrukce, je však mikrokontrolérem ignorován, takže následuje nové čtení operačního kódu následující instrukce ve stavu S1 dalšího strojového cyklu.

Instrukce o délce 2 byte, která vyžaduje dva strojové cykly se realizuje tak, že operační kód instrukce se načte ve stavu S1 prvního strojového cyklu instrukce se začíná vykonávat. Operační kód následující instrukce se formálně čte ve stavu S4 prvního strojového cyklu a ve stavu S1 a S4 druhého strojového cyklu. Akceptuje se pouze na počátku dalšího strojového cyklu.

Odlísným způsobem je vykonávána 1 bytová instrukce MOVX, která vyžaduje 2 strojové cykly. Tato instrukce vyžaduje vyslání adresy externí paměti na konci prvního strojového cyklu. Proto ve stavu S1 druhého strojového cyklu chybí signál ALE. Adresa externí paměti je generována ve stavu S4 prvního strojového cyklu.

Řídicí jednotka zabezpečuje také generování nebo zpracování řídicích signálů, které slouží k synchronizaci mikrokontroléru s externími obvody. Jsou to signály PSEN, ALE, EA a RST.

Jenom dvě instrukce (MUL – násobení, DIV – dělení) potřebují čtyři strojové cykly.

1.3 PARALELNÍ VSTUPY A VÝSTUPY

Mikrokontroléry ATMEL obsahují čtyři obousměrné vstupně-výstupní porty. Každý port obsahuje vyrovnávací paměť, kterou lze adresovat nejen jako registr skupiny SFR, ale i jako výstupní budič a vstupní hradlo, které umožňuje přenos informace z vývodů portu na interní sběrnici. Výstupní budiče portů P0 a P2 a vstupní hradla portu P0 se využívají při práci s externí pamětí. Portem P0 se přenášejí nižší bity adresy (A0-A7) a data (D0-D7), které se mají načíst (nebo zapsat) z externí paměti (do externí paměti). Portem P2 se přenášejí vyšší bity adresy (A9-A15) externí paměti, pokud je adresa víc jak osmibitová.

Všechny vývody portu P3 a některé z portu P1 mají kromě všeobecně vstupně-výstupního charakteru ještě alternativní význam, který je uveden v tab. 4.

Řídicí signály se na uvedených vývodech mohou aktivovat jenom za předpokladu, že odpovídající bit v příslušném registru SFR obsahuje „1“, jinak je výstup blokován úrovní „0“ bez ohledu na žádanou hodnotu interního resp. externího vstupního signálu.

1.3.1 STRUKTURA A ČINNOST

Struktura jednotlivých portů mikrokontroléru je znázorněna na obr. 5.

Z obr. 5 plyne, že výstupní budiče portů P0 a P2 se mohou připojit na vnitřní datovou sběrnici řídicím signálem CONTROL, což umožňuje vysílat na vývody portů P0 a P2 adresu externí paměti. Současně musí být ve vyrovnávací paměti portu P0 zapsána „1“, která pomocí výstupů Q zabezpečí uzavření dolního tranzistoru ve výstupním budiči a umožní načtení dat přes vývody portu P0 a vstupní hradla. Odlísným způsobem je řízen výstup na portu P3. Jestliže jsou ve vyrovnávací paměti portu P3 zapsány „1“, potom je stav výstup-

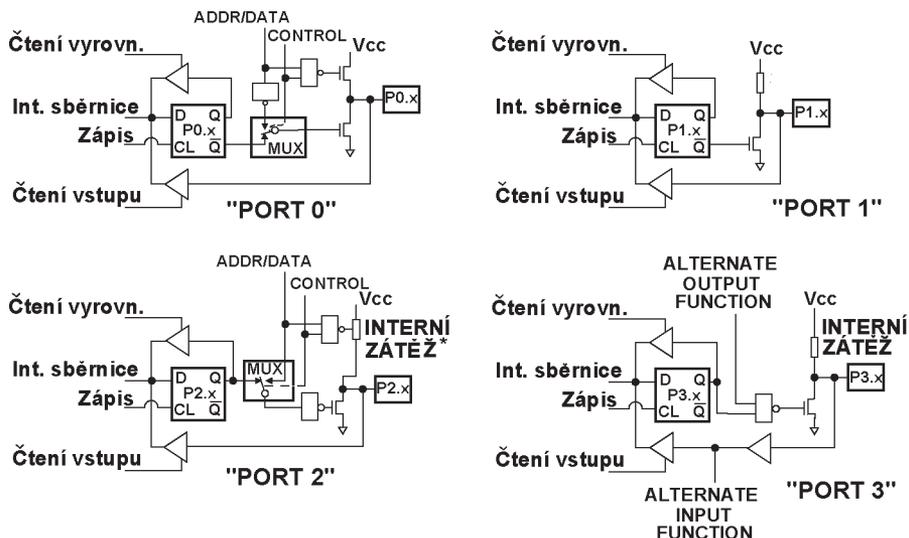
ního budiče ovlivňován řídicím signálem AOF (Alternable Output Function). Analogicky při přenosu dat z vývodů portu P3 na interní sběrnici je hodnota ovlivněná stavem řídicího signálu AIF (Alternable Input Function).

Vývod	Řídicí signál	Význam
P3.0	RxD	vstup sériového kanálu
P3.1	TxD	výstup sériového kanálu
P3.2	$\overline{\text{INT0}}$	vstup externího přerušení 0
P3.3	$\overline{\text{INT1}}$	vstup externího přerušení 1
P3.4	T0	externí vstup čítače 0
P3.5	T1	externí vstup čítače 1
P3.6	$\overline{\text{WR}}$	zápis do externí paměti dat
P3.7	$\overline{\text{RD}}$	čtení z externí paměti dat
P1.0	T2	externí vstup čítače 2
P1.1	T2EX	externí ovládání čítače 2
P1.4	$\overline{\text{SS}}$	výber pořízené jednotky při SPI ¹
P1.5	MOSI	výstup Master/vstup Slave při SPI ¹
P1.6	MISO	vstup Master/výstup Slave při SPI ¹
P1.7	SCK	výstup hodin Master/vstup hodin Slave ¹

T2 - pouze pro AT89C52 a výše

¹ - pouze pro AT89S53, AT89S8252

Tab. 4 Port P3



Obr. 5 Struktura jednotlivých portů mikrokontroléru

Každý z portů P0 až P3 je možno využít jak vstup resp. výstup. Pro všechny porty platí zásada, že je-li přes port realizován vstup dat, musí vyrovnávací paměť příslušného bitu obsahovat „1“, která zabezpečí uzavření spodního tranzistoru ve výstupním budiči. Porty P1, P2 a P3 mají výstupní budič stejný. Odlišně je konstruován port P0, jehož výstupní budič obsahuje spínaný zatěžovací tranzistor. Zatěžovací tranzistor je otevřen pouze při vysílání „1“ v čase přístupu na externí paměť, jindy je zavřen. Používá-li se port P0 jako výstupní, chovají se jeho výstupy jako výstupy s otevřeným kolektorem. Po zapsání „1“ do vyrovnávací paměti se uzavře také spodní tranzistor výstupního budiče a vývody portu P0 se nastaví do stavu vysoké impedance. Tenhle stav může být využit jako vysokoimpedanční vstup dat.

Výstupní budiče portů P1, P2 a P3 mohou ovládat 4LS TTL vstupy. Pokud jsou tyto porty použity jako vstupní, mohou být ovládané výstupy obvodů TTL nebo NMOS. Obě verze mikrokontrolérů HMOS a CHMOS mohou být ovládané též obvody s výstupními členy typu otevřený kolektor bez dalších rezistorů na napájecím napětí, což však zpomaluje přechod z log. 0 do log. 1.

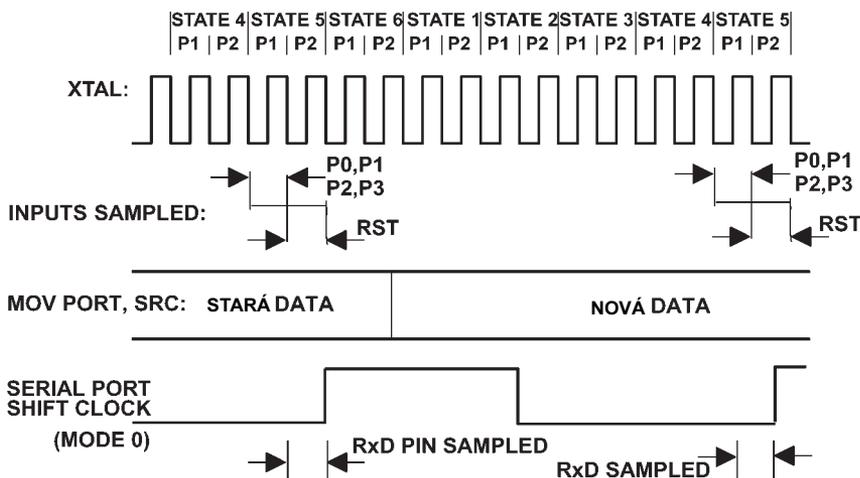
Jestliže se port P0 používá jako vstupně-výstupní, vyžaduje připojení vnějších zatěžovacích odporů, protože interní zatěžovací tranzistor je rozpojen.

Při použití portu P0 pro přenos adres resp. dat při komunikaci s externí paměti, může každým svým vývodem ovládat 8 LS TTL vstupů.

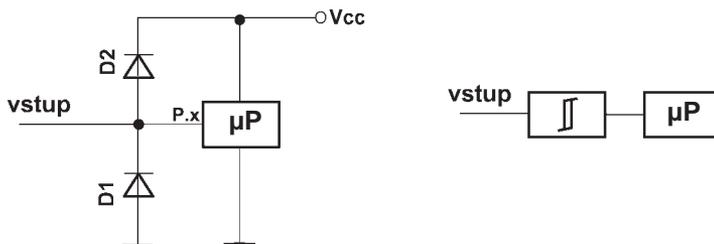
Když instrukce mění obsah na portu, nová hodnota přichází do vyrovnávací paměti v průběhu doby S6/P2 posledního strojového cyklu instrukce. Obsah vyrovnávací paměti je přenesen do výstupního buffru jenom v průběhu fáze P1 periody oscilátoru. V průběhu fáze P2 udržuje výstupní buffer hodnotu přenesenou ve fázi P1. Nová hodnota se tedy ve vyrovnávací paměti objeví na výstupu až po fázi P1, kterou je S1/P1 následujícího strojového cyklu. Časový průběh je znázorněn na obr. 6.

Jestliže se na portu P1, P2, P3 požaduje změna z 0 na 1, je v průběhu cyklů S1/P1 a S1/P2 zapnut interní zatěžovací odpor, což se projeví zvýšením přenosové rychlosti.

Tento interní zatěžovací rezistor může dodat stonásobně vyšší proud, nežli normální zatěžovací rezistor. Interní zatěžovací rezistory jsou tranzistory řízené polem, nikoli lineární rezistory. Zapojení jednotlivých vývodů portů P1 a P3 znázorňuje obr. 7.



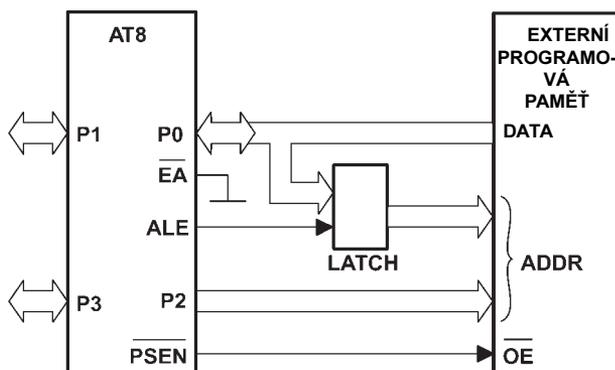
Obr. 6 Operace na V/V portu



Obr. 8 Ochrana vstupů mikrokontrolérů

1.4.1 EXTERNÍ PAMĚŤ PROGRAMU

Obr. 9 znázorňuje konfiguraci s externí pamětí programu. V tomto případě je 16 V/V linek portů P0 a P2 plně vyhrazených pro sběrnicové operace výběru programových dat. Port P0 představuje multiplexovanou adresně-datovou sběrnici.



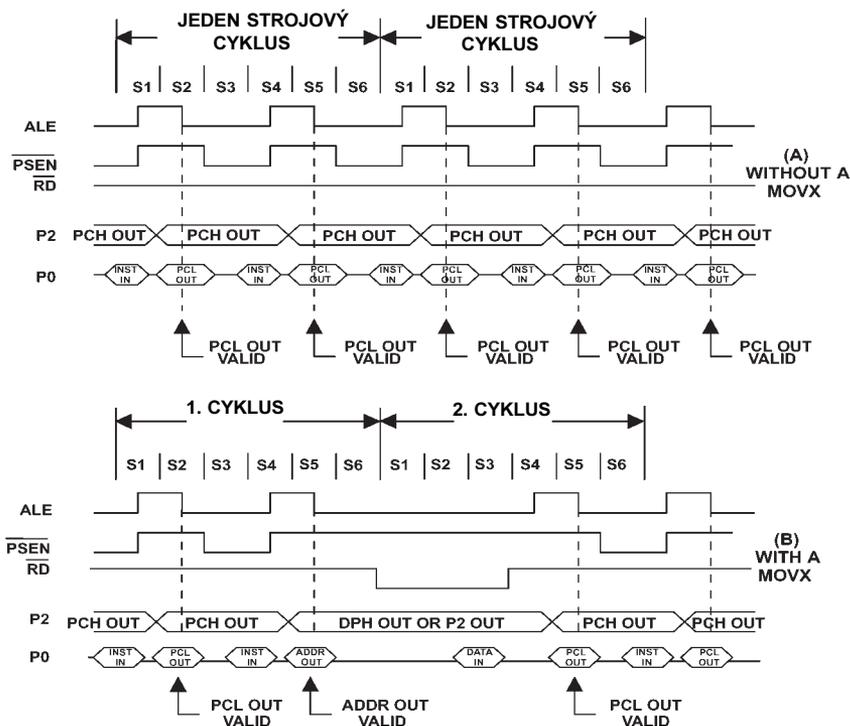
Obr. 9 Mikrokontrolér s externí pamětí programu

Externí paměť programu se adresuje 16bitovým čítačem instrukcí PC. Při výběru instrukce z externí paměti programu se portem P0 přenáší 8 nižších bitů adresy paměti (PCL) a portem P2 8 vyšších bitů adresy paměti (PCH). S ohledem na omezené trvání přenosu adresy portem P0 musí se tato adresa zapamatovat v externím registru adresy (LATCH). Přepis adresy do tohoto registru se realizuje sestupnou hranou signálu ALE, který je generován dvakrát v průběhu každého strojového cyklu. Na aktivování externí paměti programu slouží signál PSEN. Tento signál se také aktivuje dvakrát v průběhu každého strojového cyklu (vyjma instrukce MOVX) bez ohledu na to, zdali se v průběhu strojového cyklu čte jeden nebo dva byty. Časování signálu je znázorněno na obr. 10.

Přístup k externí paměti programu lze také ovlivňovat signálem \overline{EA} . Je-li signál $\overline{EA} = 0$, řídicí signály \overline{PSEN} a \overline{ALE} jsou generovány vždy a nezávisle na adrese paměti programu.

Je-li $\overline{EA} = 1$, signál \overline{PSEN} je generován jenom tehdy, když adresa paměti programu je větší než poslední adresa interní FLASH EPROM (např. FLASH EPROM = 4 kB je poslední adresa je 0FFFH – pak v rozsahu adres 1FFFH – FFFFH se adresují už místa z externí paměti programu).

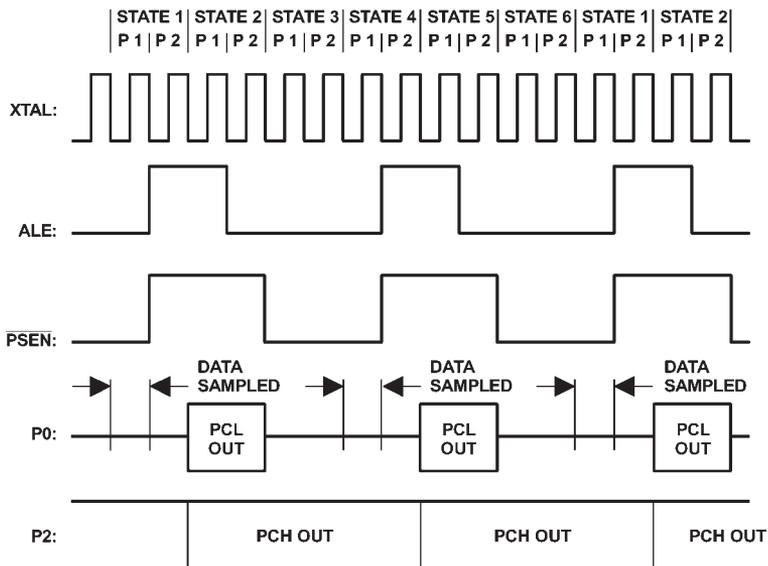
Obr. 12 znázorňuje konfiguraci s 2kB externí RAM. V tomto případě je program vykonáván z interní paměti FLASH EPROM. Port 0 pracuje v multiplexu adresy/data pro přístup do RAM. Adresace je rozšířena o 3 adresové linky portu 2 (adresy A10-A8 = P2.2, P2.1, P2.0). Linky P2.7-P2.3 jsou plně funkční V/V linky. V případě plné 2bytové adresace tj. A15-A0 je port 2 využit jako adresový a není možno jej použít jako V/V port.



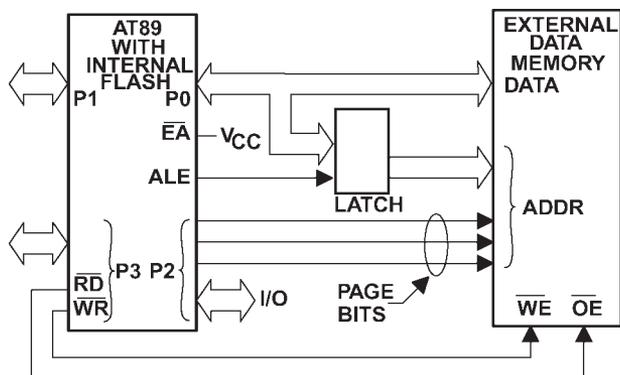
Obr. 10 Časování při výkonu programu z externí paměti

1.4.2 EXTERNÍ PAMĚŤ DAT

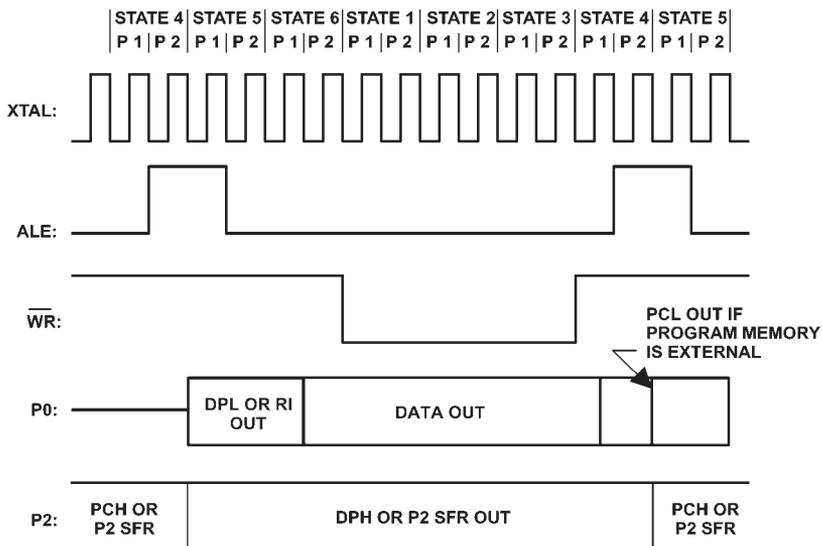
Externí paměť dat se adresuje 16bitovým ukazatelem dat DPTR. Přístup do této části paměti je možno realizovat instrukcí MOVX A, @DPTR a MOVX @DPTR, A nebo instrukcí MOVX A, @Ri a MOVX @Ri, A využitelnou pro přístup do spodní oblasti externí RAM, která je adresovatelná 8 bity, případně v závislosti od nastavení portu P2 pro pohyb v externí RAM po 256 bytových stránkách.



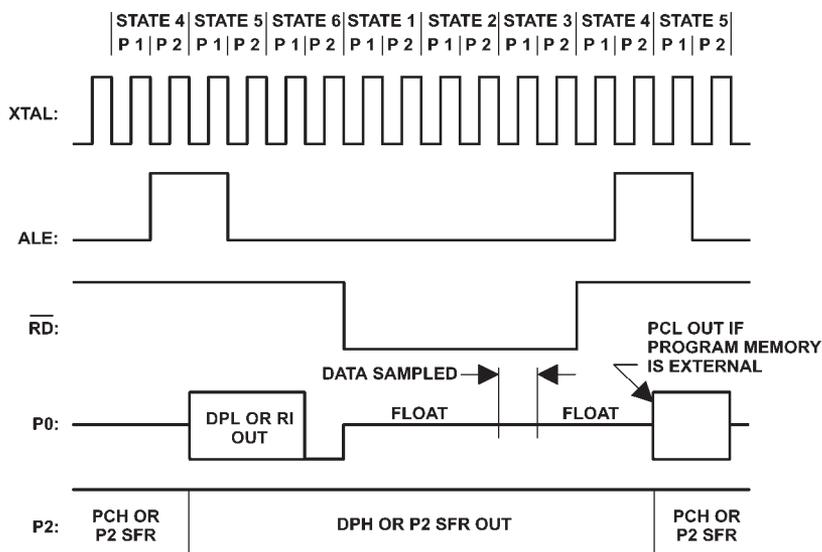
Obr. 11 Časování při čtení z externí paměti programu



Obr. 12 Mikrokontrolér s externí paměť dat



Obr. 13 Časování při zápisu do externí paměti dat



Obr. 14 Časování při čtení z externí paměti dat

1.5 SÉRIOVÉ A PARALELNÍ EEPROM PAMĚTI

1.5.1 SÉRIOVÉ PAMĚTI EEPROM

Uchovávají data i v případě výpadku napájecího napětí. Mají schopnost přeprogramování v systému přímo v aplikaci. Nabízeny jsou verze s připojením pomocí 2, 3, 4 vodičů a přes SPI Bus. Všechny verze mají velikosti od 1 Kb do 1 Mb. Dle typu lze provést do 10⁶ mazacích/zápisových cyklů. Firma ATMEL nabízí v tomto směru ideální řešení pro ukládání dat s minimálními hardwarovými nároky a v přijatelných cenových relacích. Zákazník si rovněž vybere z různých typů pouzření počínaje DIP přes verze SMD.

Příklady značení jsou následující:

(podtržený znak označuje vlastnost)

AT24xxx – paměti s 2vodičovým připojením

AT34xxx – paměti s 2vodičovým připojením s ochranou dat

AT93xxx – paměti s 3vodičovým připojením

AT59xxx – paměti s 4vodičovým připojením

AT25xxx – paměti s připojením přes SPI Bus v módu 0 a 3

Rozsah napájecích napětí pro všechny typy je od 1,8 do 5 V a značí se např.:

AT24Cxxx – 5 V napájení (4,5 – 5,5 V)

AT24LVxxx – 3,3 V napájení (2,7 – 5,5 V)

AT24BVxxx – 3 V napájení (baterie) – (2,7 – 3,6 V, některé typy od 1,8 V)

Bateriové verze snižují požadavky na příkon téměř o 80 %.

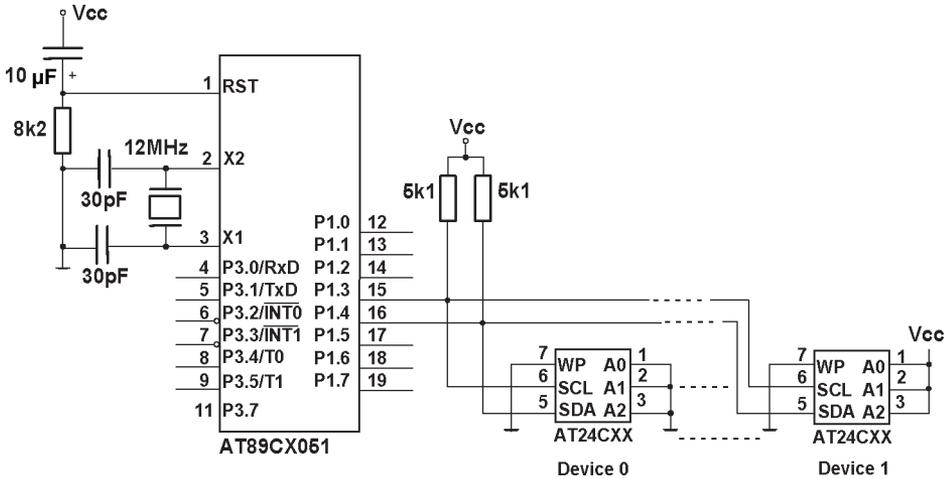
Tyto údaje je nutno ověřovat dle katalogových údajů, protože se v detailech u některých typů liší. Základní přehled typů sériových pamětí je uveden v tab. 5.

Pro informaci je možné uvést základní parametry paměti AT24C164:

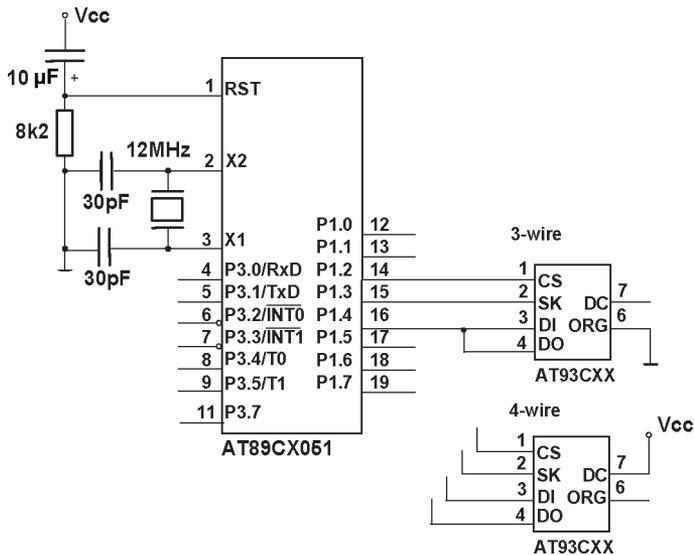
- jsou vyráběny ve verzích pro napájecí napětí od 1,8 do 5,5 V;
- interní organizace je 2048 × 8 (16 k);
- 2vodičový sériový interfejs;
- na vstupech Schmittovy obvody;
- reverzní přenosový protokol pro data;
- kaskádovatelnost umožňuje rozšířit kapacitu;
- 100 kHz (1,8; 2,5; 2,7 V) a 400 kHz (5 V) kompatibilita;
- zbytkový proud ve stavu klidu je na úrovni mikroampérů v závislosti na velikosti napájecího napětí;
- vývod pro hardwarovou ochranu dat proti přepisu;
- mód zápisu po stránkách 16 bytů;
- přístupní zápis části stránky;
- zápisový cyklus generován vnitřně (max 10 ms);
- až 1 milión zápisových cyklů;

- ochrana dat 100 let;
- odolnost proti statické elektřině >3000 V;
- určeno i pro automobilový průmysl s rozšířeným teplotním rozsahem;
- pouzdrění 8pin JEDEC SOIC a 8pin PDIP.

Způsob připojení paměti je znázorněn na obr. 15



Obr. 15 Připojení sériové EEPROM v kaskádě (I²C BUS)



Obr. 16 Připojení sériové EEPROM 3vodičové, 4vodičové

TYP	KAPACITA	ORGANIZACE	NAPÁJENÍ	POPIS
AT24C01	1k	128 × 8	1,8 ÷ 5,5 V	2vS
AT24C02	2k	256 × 8	1,8 ÷ 5,5 V	2vS,WP
AT24C04	4k	512 × 8	1,8 ÷ 5,5 V	2vS,WP
AT24C08	8k	1028 × 8	1,8 ÷ 5,5 V	2vS
AT24C16	16k	2048 × 8	1,8 ÷ 5,5 V	2vS,WP
AT24C32	32k	4096 × 8	1,8 ÷ 5,5 V	2vS,CAS
AT24C64	64k	8192 × 8	1,8 ÷ 5,5 V	2vS,CAS
AT24C128	128k	16384 × 8	1,8 ÷ 5,5 V	2vS,CAS
AT24C256	256k	32768 × 8	1,8 ÷ 5,5 V	2vS,CAS
AT24C512	512k	65536 × 8	1,8 ÷ 5,5 V	2vS,CAS
AT34C02	2k	256 × 8	1,8 ÷ 5,5 V	2vS
AT93C46	1k	64 × 16/128 × 8	1,8 ÷ 5,5 V	3vS
AT93C56	2k	128 × 16/256 × 8	2,5 ÷ 5,5 V	3vS
AT93C66	4k	256 × 16/512 × 8	2,5 ÷ 5,5 V	3vS
AT93C86	16k	1024 × 16/2048 × 8	2,7 ÷ 5,5 V	3vS
AT59C11	1k	64×16/128 × 8	2,5 ÷ 5,5 V	4vS
AT59C22	2k	128 × 6/256 × 8	2,5 ÷ 5,5 V	4vS
AT59C13	4k	256 × 16/512 × 8	2,5 ÷ 5,5 V	4vS
AT25010	1k	128 × 8	1,8 ÷ 5,5 V	SPI
AT25020	2k	256 × 8	1,8 ÷ 5,5 V	SPI
AT25040	4k	512 × 8	1,8 ÷ 5,5 V	SPI
AT25080	8k	1024 × 8	1,8 ÷ 5,5 V	SPI
AT25160	16k	2048 × 8	1,8 ÷ 5,5 V	SPI
AT25320	32k	4096 × 8	1,8 ÷ 5,5 V	SPI
AT25640	64k	8192 × 8	1,8 ÷ 5,5 V	SPI
AT25128	128k	16384 × 8	1,8 ÷ 5,5 V	SPI
AT25256	256k	32768 × 8	1,8 ÷ 5,5 V	SPI
AT25HP256	256k	32768 × 8	1,8 ÷ 5,5 V	SPI
AT25HP512	512k	65536 × 8	1,8 ÷ 5,5 V	SPI
AT25P1024	1M	131072 × 8	1,8 ÷ 5,5 V	SPI

POZNÁMKY:

2vS, 3vS, 4vS – 2, 3, 4vodičová sběrnice
 WP – ochrana dat
 CAS – kaskadovatelná

Tab. 5 Typy sériových a paralelních pamětí EEPROM

Sériové paměti nabízejí množství výhod oproti paralelním tam, kde není potřeba velice rychlého ukládání většího objemu dat. Vyžadují menší plochu, šetří vývody pro jiné funkce. Jsou tedy obzvláště vhodné pro spojení s mikrokontroléry s malým počtem výstupních vývodů jako např. AT89C1051 nebo AT89C2051.

Příklady zapojení dalších typů 3 a 4 vodičových sériových pamětí jsou uvedeny na obr. 16.

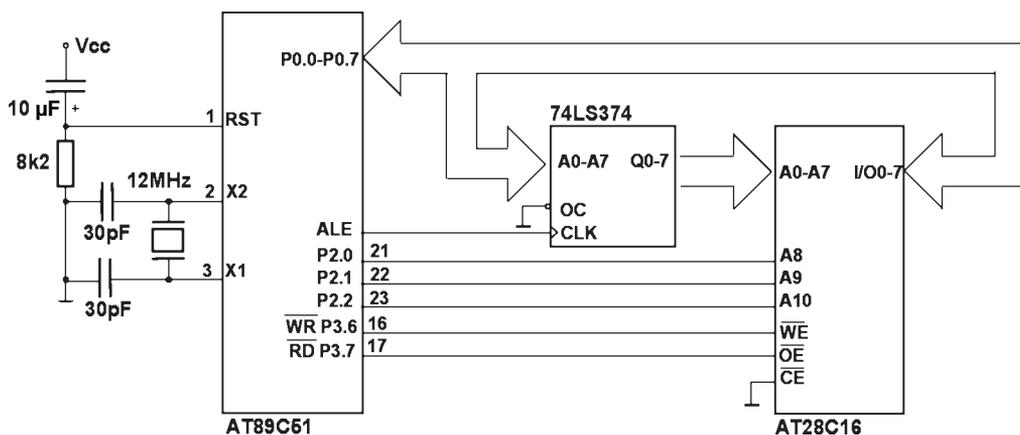
Další zajímavou skupinou jsou paměti se sériovým rozhraním SPI jsou vyráběny pod označením AT25xxx – viz tab. 5. Jejich vlastnosti jsou podobné předchozím typům, navíc mají schopnost spolupráce při vyšším hodinovém kmitočtu. U těchto typů je garantováno uschování dat 100 let, 1 milion zápisových cyklů a také volitelná ochrana dat pro 1/4, 1/2 resp. celého pole paměti.

1.5.2 PARALELNÍ PAMĚTI EEPROM

Mají označení AT28xxx. Jsou vyrobeny na bázi CMOS technologie, vyznačující se nízkým příkonem s možností až 10 000 zápisových cyklů, s ochranou dat 10 let. Z hlediska napájení jsou jak 5V, tak i verze 3,3V a pro bateriové napájení 3 V s příslušným znakovým označením C, LV, BV. Rovněž kapacity těchto pamětí jsou obdobné jako u sériových pamětí. Jejich výhodou je hlavně výrazně rychlejší přístup k datům. K propojení však vyžadují více vodičů a také převážně větší plochu na desce plošných spojů.

Např. AT28BV256 je paměť 256k s organizací 32 k × 8 pro bateriové napájení od 2,7 do 3,6 V, s dobou přístupu 200 ns, počtem zápisů 10 000, garancí udržení dat 10 let a ostatními vlastnostmi obdobně jako u sériových pamětí.

Pro informaci je zde uveden způsob připojení paralelního typu paměti:



Obr. 17 Připojení paralelní paměti EEPROM

1.6 ČÍTAČE/ČASOVAČE

Mikrokontroléry ATMEL mají dva (tři) 16bitové čítače/časovače, jejichž obsah je přístupný pomocí SFR registrových párů [TH0, TL0], [TH1, TL1], [TH2, TL2]. Všechny čítače mohou pracovat jako interní časovače nebo jako čítače vnějších událostí.

Nejdříve si vysvětlíme význam pojmu časovač a čítač pojmu procesoru. Je to vlastně 16bitový registr, který počítá na základě svého externího nastavení časové události od hodinového oscilátoru procesoru nebo od externího zdroje signálu, který je v našem případě realizován vstupem T0, T1. Jde-li o čítač externích událostí nazýváme tento registr „čítač“, jde-li o čítač period hodinového oscilátoru procesoru nazýváme ho „časovač“.

Jestliže čítač pracuje v režimu časovače, inkrementuje se jeho obsah v každém strojovém cyklu tvořením 12 periodami hodinových impulzů. Kmitočet oscilátoru se tedy dělí v poměru 1 : 12. V režimu čítače vnějších událostí se obsah čítače inkrementuje sestupnou hranou impulzu na vstupu T_i ($i = 0, 1, 2$). V tomto režimu se vstupní signál vzorkuje ve druhé polovině stavu S5P2 každého strojového cyklu. Pokud se v jednom strojovém cyklu otestuje vstup v „1“ a v následujícím v „0“, inkrementuje se obsah čítače. Nová hodnota se v registru čítače zaznamená v průběhu S3P1 následujícího strojového cyklu. Protože na zjištění hrany vstupního impulzu jsou zapotřebí 2 strojové cykly, tj. 24 period hodinových impulzů, nejvyšší kmitočet vstupních impulzů může být 1/24 kmitočet oscilátoru. Zároveň minimální délka úrovně „0“ resp. „1“ na vstupu čítače musí být 12 period hodinových impulzů, v opačném případě změna stavu na vstupu nemusí být zachycena.

Čítače/časovače 0 a 1 lze nastavit na čtyři režimy činnosti:

1. Režim 0 – 13bitový čítač/časovač;
2. Režim 1 – 16bitový čítač/časovač;
3. Režim 2 – 8bitový čítač/časovač s přednastavením;
4. Režim 3 – 8bitový čítač/časovač s dělením.

Čítač/časovač 2 v mikrokontrolérech AT89C52 ... má tři režimy činnosti:

1. zachytávací čítač/časovač;
2. čítač/časovač s automatickým přednastavením (vpřed a vzad);
3. generátor přenosové rychlosti.

Nastavení režimů činnosti čítačů 0 resp. 1 určuje obsah speciálního registru (SFR) s označením TMOD. Režim činnosti čítače 2 určuje obsah speciálního registru (SFR) s označením T2CON. Vlastní čítače se programově spouští nebo zastavují nastavením nebo nulováním bitu TR_i v registru TCON.

1.6.1 ČÍTAČE 0, 1

Tyto čítače obsahují všechny typy mikrokontrolérů ATMEL. Význam jednotlivých bitů registru TMOD je následující:

GATE – vnitřní signál na povolení činnosti čítače/časovače v součinnosti s externím vstupem INT a vnitřním signálem TR_i ($i = 0, 1$).

Při GATE = 1, je povolena činnost, jestli $INT_i = TR_i = 1$.
 Při GATE = 0, je povolena činnost, jestli $TR_i = 1$.

TMOD	(MSB)				(LSB)			
	GATE	C/T	M1	M0	GATE	C/T	M1	M0
	Čítač 1				Čítač 0			

C/T – volba funkce čítače nebo časovače.

Ak C/T = 0, pracuje čítač v režimu časovače.

Ak C/T = 1, pracuje čítač v režimu čítače vnějších událostí.

M1, M0 – volba režimů činnosti čítačů podle následující tabulky:

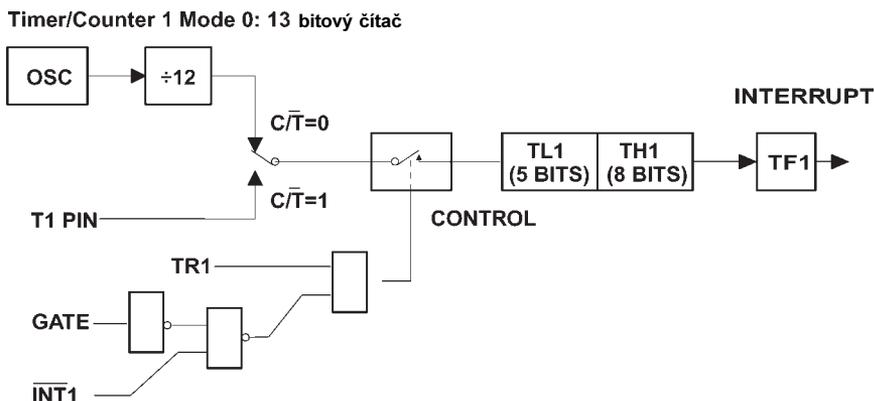
Režim	M0	M1
0	0	0
1	0	1
2	1	0
3	1	1

Tab. 6 Režimy čítačů/časovačů

Režim 0 pro čítače 0 a 1 (13bitový čítač/časovač)

V tomto režimu pracuje čítač jako 13bitový čítač/časovač, který obsahuje 8 bitů čítače TH1 a 5 bitů čítače TL1. Vyšší tři bity nejsou využity. Při přepnutí čítače, tj. při přechodu z obsahu všech „1“ na „0“ se nastavuje příznakový překlápěcí obvod TF1 v registru TCON. Na vstup čítače se přivádí vstupní impulzy, je-li sepnut sériový spínač, který se ovládá vnitřními signály TR1, GATE a úrovní na vstupu externího přerušení INT1 (viz obr. 18).

Režim čítače je ovládán vnitřním signálem C/T, který se podobně jako signál TR1 odvozuje od příslušných bitů v registru SFR TCON. Má následující tvar:



Obr. 18 Režim 0 pro čítače 0 a 1

Význam jednotlivých bitů registru TCON:

- TF1** – příznakový bit stavu čítače 1, který indikuje přeplnění čítače 1. Nastavuje se automaticky při přetečení čítače 1 a nuluje se automaticky přechodem na vektorovou adresu obsluhy přerušeni od čítače 1 při obsluze přerušeni mikrokontrolérem.

TCON	(MSB)				(LSB)			
	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
	Čítač 1		Čítač 0		Čítač 1		Čítač 0	

- TR1** – start/stop čítače 1. Je-li TR1 = 1, čítač čítá.
- TF0** – příznakový bit stavu čítače 0, analogický k TF1.
- TR0** – start/stop čítače 0. Je-li TR0 = 1, čítač čítá.
- IE1** – příznak žádosti o přerušeni ze vstupu INT1. Nastavuje sa automaticky aktivní hranou na uvedeném vstupu a nuluje se při obsluze přerušeni – programem.
- IT1** – řídicí bit typu přerušeni INT1. Je-li IT1 = 1, žádost o přerušeni vzniká při sestupné hraně impulsu na vstupu INT1, IT1 = 1. Je-li IT1 = 0, žádost o přerušeni vzniká při aktivní úrovni „0“ na vstupu INT1.
- IE0** – příznak žádosti o přerušeni ze vstupu INT1 významově analogický jako IE1.
- IT0** – řídicí bit typu přerušeni INT0 významově analogický jako IT1.

Stejně jako čítač 1 pracuje také čítač 0. Záměnou indexu 1 za 0, platí stejné vlastnosti pro čítač 0.

Režim 1 pro čítače 0 a 1 (16bitový čítač/časovač)

Režim 1 se shoduje s režimem 0 s výjimkou, že čítače TL_i a TH_i jsou plně využity, tedy čítač pracuje jako 16bitový čítač/časovač (viz obr. 19).

Režim 2 pro čítače 0 a 1 (8bitový čítač/časovač s přednastavením)

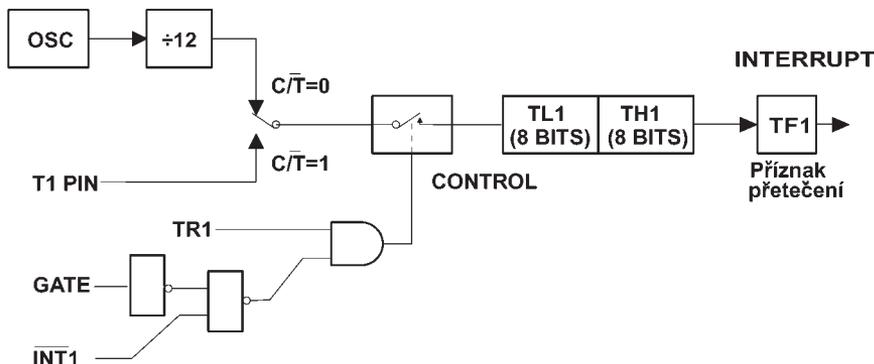
V tomto režimu je čítač tvořen 8bitovým čítačem TL_i, který je možno naplnit obsahem TH_i automaticky po přeplnění čítače TL_i. Blokové schéma činnosti čítače 1 v režimu 2 je na obr. 19. Přeplněním čítače se kromě přepisu obsahu TH_i do TL_i nastavuje příznakový bit TF1. Po přepisu TH_i do TL_i se obsah TH_i nemění. V tomto režimu se čítače 1 a 2 chovají stejně (viz obr. 20).

Režim 3 pro čítače 0 a 1 (8bitový čítač/časovač s dělením)

V tomto režimu může pracovat pouze čítač 0. Čítač 1 se chová tak, jako když je TR1 = 0. Čítač 0 se v režimu 3 rozdělí na dva samostatné 8bitové čítače TL0 a TH0, kterých výstupy ovládají příznakové bity TF0 a TF1. Čítač TH0 je připojen na interní hodinové impulsy a využívá signál TR1 a TF1, takže v režimu 3 vyvolává TH0 přerušeni tak, jako by

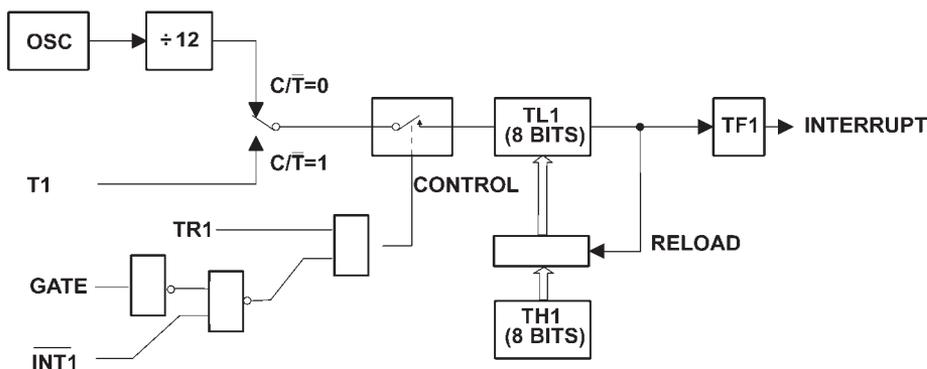
bylo generováno čítačem 1. Řízení čítače TL0 se realizuje signály INT0, GATE0, TR0 resp. T0 (viz obr. 20). Režim 3 je určen pro aplikace, které vyžadují dva samostatné 8bitové čítače/časovače. Pracuje-li čítač 0 v režimu 3, může se čítač 1 blokovat vstupem čítače 0 do režimu 3, resp. uvolnit po skončení režimu 3, případně využít jako generátor přenosové rychlosti, tedy v aplikaci, která nevyžaduje generování přerušení (viz obr. 21).

Timer/Counter 1 Mode 1: 16bitový čítač

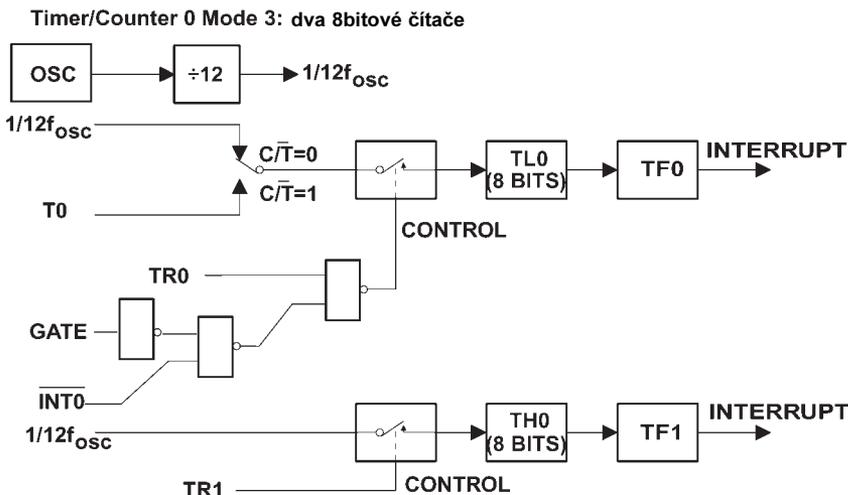


Obr. 19 Režim 1 pro čítače 0 a 1

Timer/Counter 1 Mode 2: 8bitový čítač



Obr. 20 Režim 2 pro čítače 0 a 1



Obr. 21 Režim 3 pro čítače 0 a 1

1.6.2 ČÍTAČ/ČASOVAČ 2 (pro AT89C52...)

Je to vysoce výkonný přírůstek. V oblasti SFR RAM přibýlo šest speciálních registrů: registrový pár časovače [TL2, TH2], řídicí registry T2CON, T2MOD a zachytávací registry RCAP2L, RCAP2H. Čítač/časovač 2 může pracovat podobně jako čítač/časovač 0 a 1 tj. jako časovač nebo čítač vnějších událostí v závislosti na nastavení bitu C/T2 v registru T2CON.

	(MSB)							(LSB)
T2CON	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2

Význam jednotlivých bitů registru T2CON:

- TF2** – příznakový bit stavu čítače 2, který indikuje přeplnění čítače 2. Nastavuje se automaticky při přetečení čítače 1. Musí se nulovat programově. Tento příznakový bit se nenastavuje jestliže RCLK = 1 nebo TCLK = 1.
- EXF2** – externí příznakový bit, který se nastaví přechodem T2EX z 1 do 0 při EXEN2 = 1. Je-li přerušování povoleno, nastavení EXF2 do 1 způsobí automaticky přechod na vektorovou adresu obsluhy přerušování. Musí se nulovat programově.
- RCLK** – příznakový bit přijímače sériového kanálu. Je-li nastaven, pak umožňuje použít přetečení čítače 2 na taktování příjmu na sériovém kanálu v režimech činnosti 1 a 3, pokud časovač 1 zabezpečuje přenosovou rychlost. Nastavení RCLK = 0 umožňuje použít taktování sériového kanálu časovačem 1.
- TCLK** – příznakový bit vysílače. Je-li nastaven, pak umožňuje použít přetečení čítače 2 na taktování vysílání na sériovém kanálu v režimech činnosti 1 a 3, jestliže časovač 1 zabezpečuje přenosovou rychlost. Nastavení TCLK = 0 umožňuje použít taktování sériového kanálu časovačem 1.

EXEN2 – příznak povolení pro vstup T2EX. Je-li EXEN2 = 1 a není časovač 2 využit jako generátor přenosové rychlosti, pak umožňuje zachycení obsahu registrů TH2 a TL2 v záchytných registrech nebo (v režimu s přednastavením) zápis do registrů TH2 a TL2. Při EXEN2 = 0 časovač 2 ignoruje události na vývodu T2EX.

TR2 – start/stop čítače 2. Když TR1 = 1, čítač čítá.

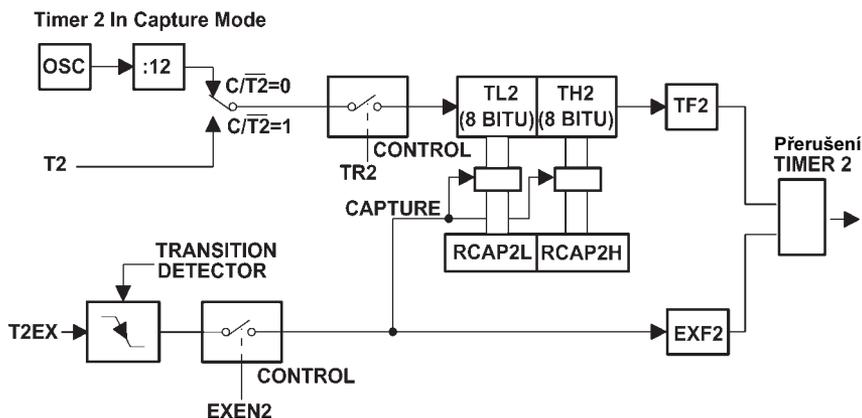
C/T2 – volba funkce čítače nebo časovače. Je-li C/T2 = 0, pracuje čítač v režimu časovače.

Je-li C/T2 = 1, pracuje čítač v režimu čítače vnějších událostí. Čítač vnějších událostí reaguje na sestupnou hranu na vstupu T2.

CP/RL2 – příznakový bit režimu zachycení (Capture), nebo přednastavení (Reload). Je-li nastaven režim zachycení, způsobí sestupná hrana na vstupu T2EX, když EXEN2 = 1. Jestliže je roven nule, přednastavení způsobí sestupná hrana na vstupu T2EX, když EXEN2 = 1. Je-li RCLK = 1 nebo TCLK = 1 pro sériový kanál je tento bit ignorován a časovač 2 je automaticky při přetečení nastaven na 16bitovou hodnotu uloženou v registrech RCAP2L, RCAP2H.

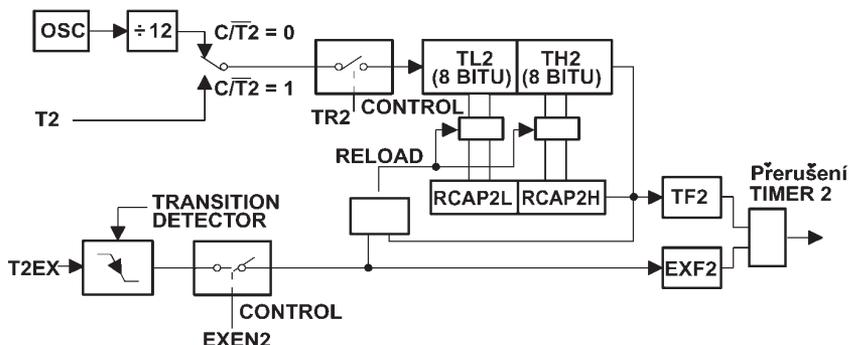
V záchytném režimu činnosti (*obr. 22*) má bit EXEN2 dva významy. V případě EXEN2 = 0 čítač/časovač je 16bitovým čítačem nebo časovačem, jehož přetečení nastaví příznak TF2. EXEN2 = 1 vykonává stejnou činnost, ale sestupná hrana na vstupu T2EX způsobí, že aktuální obsah registrů TL2, TH2 je zachycen do registrů RCAP2L, RCAP2H. Navíc se nastaví příznakový bit záhytu EXF2, který může stejně jako příznakový bit přetečení TF2 generovat přerušení. Oba příznaky přetečení je nutno nulovat programově, neboť nejsou nulovány automaticky jako TF0 a TF1 u čítačů 0 a 1 přechodem na obsluhu přerušení. Je zřejmé, že vstup T2EX lze využít i jako další vstup vnějšího přerušení.

V režimu čítač/časovač s automatickým přednastavením (*obr. 23 a obr. 24*) má bit EXEN2 také dva významy. Jeli EXEN2 = 0, pak přeplnění čítače/časovače nastaví příznak TF2 a opět přednastaví jeho hodnotu přepsáním obsahu registrů [TL2, TH2] z registrů RCAP2L a RCAP2H, které jsou nastaveny programově. EXEN2 = 1 vykonává shodnou činnost, ale sestupná hrana na vstupu T2EX také opět přednastaví jeho hodnotu přepsáním obsahu registrů [TL2, TH2] z registrů RCAP2L a RCAP2H a současně nastaví příznakový obvod EXF2.

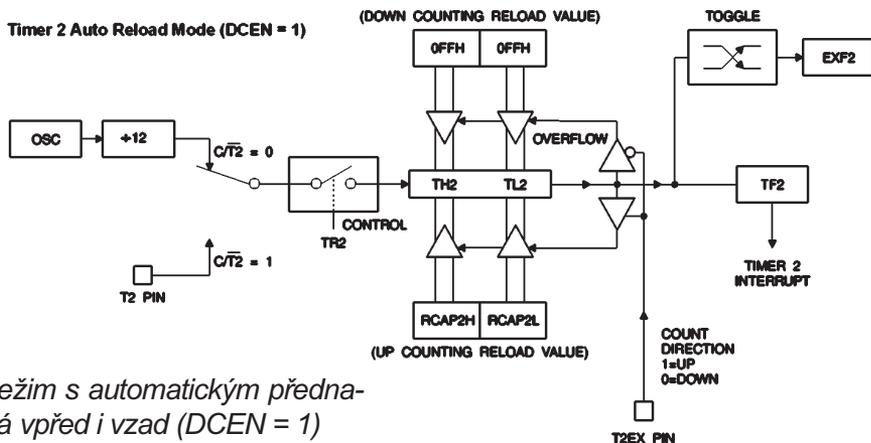


Obr. 22 Záchytný režim časovače 2

Timer 2 in Auto-Reload Mode (DCEN = 0)

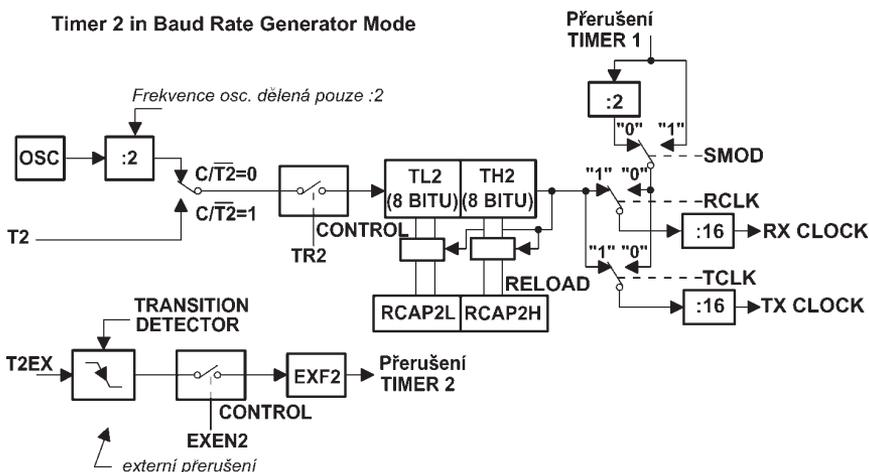


Obr. 23 Režim s automatickým přednastavením – čítá jen vpřed (DCEN = 0)



Obr. 24 Režim s automatickým přednastavením – čítá vpřed i vzad (DCEN = 1)

Timer 2 in Baud Rate Generator Mode



Obr. 25 Režim generátoru přenosové rychlosti

Režim generátoru přenosové rychlosti (*obr. 25*) je ovládán bity RCLK nebo TCLK. Tento režim bude popsán podrobně ve spojitosti se sériovým kanálem.

1.7 SÉRIOVÝ VSTUP/VÝSTUP

Sériový kanál je plně duplexní, tj. přijímač a vysílač mohou pracovat zcela samostatně. Přijímač a vysílač jsou vybaveny vyrovnávací pamětí SBUF, která je součástí skupiny speciálních registrů SFR. Zápisem údajů do paměti SBUF se plní vysílač. Údaje přijímané přijímačem se zapisují do vyrovnávací paměti SBUF, přičemž, když se nestihnou přečíst tyto údaje před zápisem nových, jsou staré údaje ztraceny.

1.7.1 REŽIMY ČINNOSTI SÉRIOVÉHO KANÁLU

Sériový kanál může pracovat ve čtyřech režimech:

Režim 0 – obr. 26

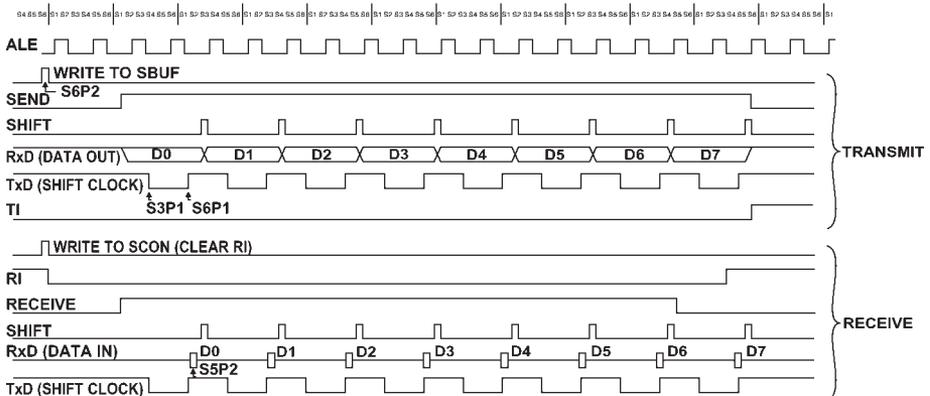
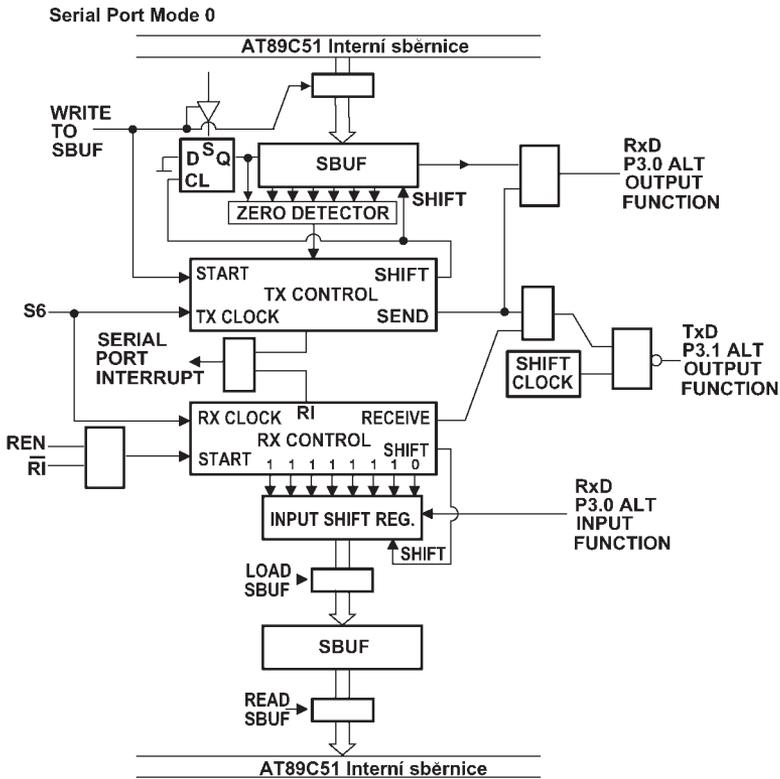
Údaje v sériovém tvaru vstupují a vystupují přes vodič RxD. Na výstupu TxD se přenášejí synchronizační impulzy. Přenášejí se 8bitové údaje, přičemž jako první se přenáší bit LSB. Přenosová rychlost je pevná a rovná se 1/12 kmitočtu oscilátoru. Na výstupu TxD se pro každé údajové slovo objeví 8hodinových impulzů, jejichž sestupná hrana určuje okamžik platnosti údajů. Údaje jsou platné pokud TxD = 0.

Zápisem bytu do registru SBUF se v S6/P2 nastaví log. 1 na 9. pozici posuvného registru a oznámí řídicímu bloku pro vysílání (TX Control) začátek přenosu.

Vnitřní časování je takové, že uplyne jeden celý strojový cyklus mezi zápisem do SBUF a aktivací vysílače (SEND). Výstup vysílače (SEND) je pro posuvný registr alternativní funkcí výstupu P3.0 a taktovací signál (SHIFT CLOCK) je alternativní funkcí výstupu P3.1. Taktovací signál (SHIFT CLOCK) je v log. 0 v době S3, S4 a S5 v každém strojovém cyklu a na úrovni log. 1 v době S6, S1 a S2. V S6/P2 strojového cyklu, jestliže je vysílač (SEND) aktivní, obsah posuvného registru je posunut napravo. Po každém vysunutí jednoho bitu napravo se do registru z levé strany plní log. 0. Když se vysune poslední bit (MSB), log. 1, která byla vložena na 9. pozici je nalevo od MSB a všechny ostatní pozice jsou nulové. Podmínka (ZERO DETECTOR) signalizuje bloku pro vysílání (TX Control), že má vykonat poslední posun, deaktivovat vysílač (SEND) a nastavit TI. Tyto dvě poslední akce nastanou v čase S1/P1 desátého strojového cyklu po zápisu do SBUF.

Příjem se inicializuje za podmínky REN = 1 a R1 = 0. V čase S6/P2 následujícího strojového cyklu blok pro příjem (RX Control) naplní posuvný registr příjmu (INPUT SHIFT REGISTER) hodnotou 11111110B a aktivuje příjem. Příjem zabezpečují hodiny (SHIFT CLOCK), které jsou alternativní funkcí výstupu P3.1. Hodiny (SHIFT CLOCK) vykonávají příjem v čase S3/P1 a S6/P1 každého strojového cyklu. V čase S6/P2, kdy je příjem aktivován, obsah příjmového posuvného registru je posunut vlevo. Bity přicházející zprava, jsou ovzorkovány hodnotami vstupu P3.1 v čase S5/P2 v každém strojovém cyklu. Zprava přicházející bity vysouvají log. 1 nalevo. Když se před příjmem vložená log. 0 dostane na nejkrajnější pozici vlevo v posuvném registru, blok pro příjem (RX Control) dostává signál, že se má vykonat poslední posun a naplnit registr SBUF.

V čase S1/P1 10. strojového cyklu po zápisu do registru SCON, kterým byl RI vynulován, je příjem ukončen a bit RI nastaven.



Obr. 26 Sériový kanál – režim 0

Režim 1 – obr. 27

V tomto režimu se přenášejí 10bitová slova na výstup TxD a vstupují přes vývod RxD. Slovo začíná startovacím bitem, za kterým následuje 8 datových bitů (opět bit LSB jako první) a jeden stop bit. Při příjmu se stop bit zapisuje do RB8 v speciálním registru SCON. Přenosová rychlost je volitelná.

V tomto režimu je opět přenos inicializován instrukcí, ve které je cílovým registrem SBUF. Zápisem bytu do registru SBUF se ve S6/P2 nastaví log. 1 na 9. pozici posuvného registru a oznámí řídicímu bloku pro vysílání (TX Control) začátek přenosu. Ve skutečnosti se přenos začíná v čase S1/P1 strojového cyklu, který následuje po přeplnění 16bitového čítače. Tím je taktování přenosu bitu dané čítačem s 16. modulem a ne vlastním zápisem do SBUF.

Vysílání začíná aktivací vysílače (SEND), který vloží na TxD startovací bit. Aktivací signálu DATA se vysílající bity přenesou do posuvného registru. První takt se objeví následně po uplynutí času jednoho bitu. Po každém vysunutí jednoho bitu napravo se do registru zleva plní log. 0. Když se vysune poslední bit (MSB), log. 1, která byla vložena na 9. pozici je vlevo od MSB a všechny ostatní pozice jsou nulové. Tato podmínka (ZERO DETECTOR) signalizuje bloku pro vysílání (TX Control), že má vykonat poslední posun, deaktivovat vysílač (SEND) a nastavit TI. To nastane po desátém přeplnění 16bitového čítače po zápisu bytu do registru SBUF.

Příjem se realizuje sestupní hranou na vstupu RxD. Pro ten účel je vstup RxD vzorkován 16× nastavenou přenosovou rychlostí. Je-li přenos detekován, 16bitový čítač je okamžitě nulován a do vstupního posuvného registru je vložena hodnota 11111111B. Vynulování 16bitového čítače zabrání jeho přeplnění v čase příjmu bitů. 16 stavů čítače dělí každý čas jednoho bitu 16. V 17., 18. a 19. stavu čítače času každého bitu je vzorkována hodnota na vstupu RxD. Přijata je ta hodnota, která se objeví alespoň ve dvou vzorcích. To eliminuje případné rušení. Aby byly potlačeny falešné bity, není-li hodnota prvního vzorku nulová, přijímací obvod je vynulován a jednotka čeká další sestupnou hranu. Je-li start bit platný, je vsunut do posuvného registru a přistoupí se na příjem zbytku.

Zprava přicházející bity vysouvají log. 1 vlevo. Posunem startovacího bitu na nejzazší pozici vlevo v 9bitovém posuvném registru se jednotka RX Control dostává do stavu příjmu posledního bitu. Jednotka RX Control provede poslední posun, naplní registr SBUF, bit RB8 a zároveň nastaví příznak přerušení při příjmu RI. To se vykoná tehdy a jenom tehdy když následující podmínky jsou splněny současně v době posledního taktu pro posuvný registr:

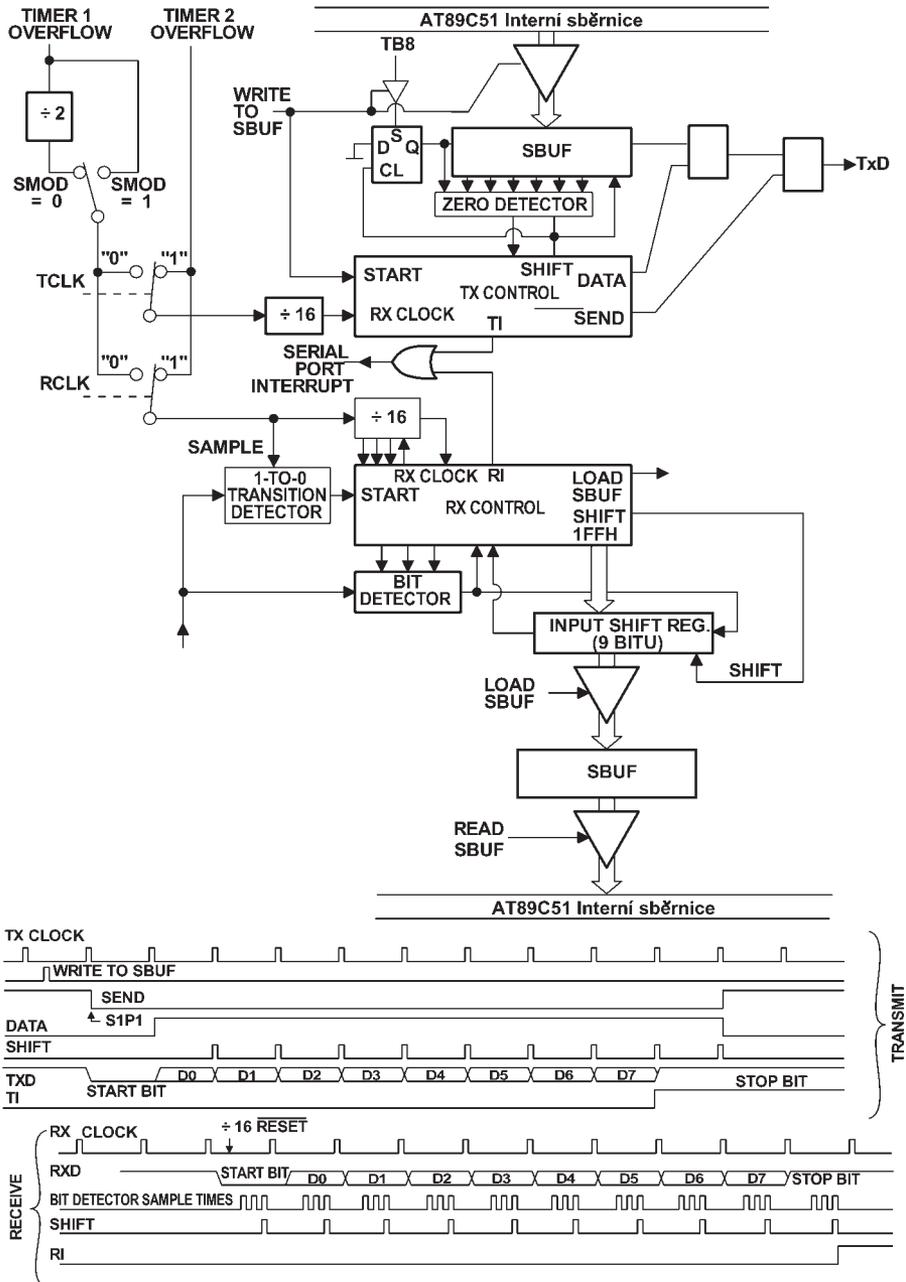
1. RI = 0.
2. SM2 = 0 nebo přicházející stop bit = 1.

Když některá z těchto podmínek není splněna, je celý přijímaný vzorek nenapravitelně ztracen a RI není nastaven. Jestliže jsou splněny obě podmínky, je stop bit vložena do RB8 a 8 datových bitů se přesune do registru SBUF a příznak RI je nastaven. V této době, nezávisle na tom, jestli byly podmínky splněny, testuje jednotka další sestupnou hranu na vstupu RxD.

Režim 2 – obr. 28

V režimu 2 se přenášejí 11bitová slova na výstupu TxD a vstupují přes vývod RxD. Slovo začíná startovacím bitem, za kterým následuje 8 datových bitů, programovatelný 9. bit a jeden stop bit. Vysílání 9. programovatelného bitu se realizuje přes bit TB8 v registru SCON, který

Serial Port Mode 1. TCLK, RCLK and Timer 2 are Present In the AT89C52 Only.



Obr. 27 Sériový kanál – režim 1

je možné programově ovládat. Uvedený bit je použitelný i jako paritní bit. Při vysílání se do RB8 zapíše paritní bit P ze stavového registru PSW. Při příjmu se programovatelný 9. bit zapisuje do RB8 v registru SCON a dá se současně s převzetím obsahu přijímacího registru testovat. Stop bit není zapisován na žádné místo. Rychlost přenosu se programuje jako 1/32 nebo 1/64 kmitočtu oscilátoru pomocí bitu SMOD v registru PCON.

Obr. 28 a obr. 29 znázorňují funkční schéma sériového kanálu pro režim činnosti 2 a 3. Příjímací část je stejná jako u režimu 1. Vysílací část se odlišuje pouze 9bitovým vysílacím registrem.

V tomto režimu je znovu přenos inicializován instrukcí, ve které je cílovým registrem SBUF. Zápisem bytu do registru SBUF a přesunem RB8 na 9. pozici posuvného registru, vzkáže řídicímu bloku pro vysílání (TX Control) začátek přenosu. Přenos se ve skutečnosti začíná v čase S1/P1 strojového cyklu, který následuje po přeplnění 16bitového čítače. Tím je taktování přenosu bitu dané čítačem s šestnáctkovým zbytkem po dělení 16 a ne vlastním zápisem do SBUF.

Vysílání začíná aktivací vysílače (SEND), který vloží na TxD startovací bit. Aktivací signálu DATA se vysílající bity přenesou do posuvného registru. První takt se objeví pak po uplynutí času jednoho bitu. Po každém vysunutí jednoho bitu vpravo se do registru zleva plní log. 0. Když se vysune poslední bit TB8, stop bit je vlevo od MSB a všechny ostatní pozice jsou nulové. Tato podmínka (ZERO DETECTOR) signalizuje bloku pro vysílání (TX Control), že má vykonat poslední posun, deaktivovat vysílač (SEND) a nastavit TI. To nastane po jedenáctém přeplnění 16bitového čítače po zápisu bytu do registru SBUF.

Příjem se realizuje sestupnou hranou na vstupu RxD. Pro tento účel je vstup RxD vzorkován 16× nastavenou přenosovou rychlostí. Je-li přenos detekován, 16bitový čítač je okamžitě nulován a do vstupního posuvného registru je vsunuta hodnota 11111111B. Vynulování 16bitového čítače zamezí jeho přeplnění v čase přijímání bitů. 16 stavů čítače dělí každý čas jednoho bitu 16. V 17., 18. a 19. stavu čítače času každého bitu je vzorkována hodnota na vstupu RxD. Akceptována je hodnota, objevující se alespoň ve dvou vzorcích. To eliminuje případné rušení. Aby byly potlačeny falešné bity, je-li hodnota prvního vzorku nenulová, přijímací obvod je vynulován a jednotka čeká další sestupnou hranu. V případě platného start bitu, se vsune do posuvného registru a přistoupí se na příjem zbytku.

Zprava přicházející bity vysouvají log. 1 vlevo. Posunem startovacího bitu na nejzazší pozici vlevo v 9bitovém posuvném registru se jednotka RX Control dostává do stavu příjmu posledního bitu. Jednotka RX Control provede poslední posun, naplní registr SBUF, bit RB8 a zároveň nastaví příznak přerušení při příjmu RI. Vykoná se to jen při současném splnění následujících podmínek v době posledního taktu pro posuvný registr:

1. RI = 0,
2. SM2 = 0 nebo přijímaný 9. bit = 1.

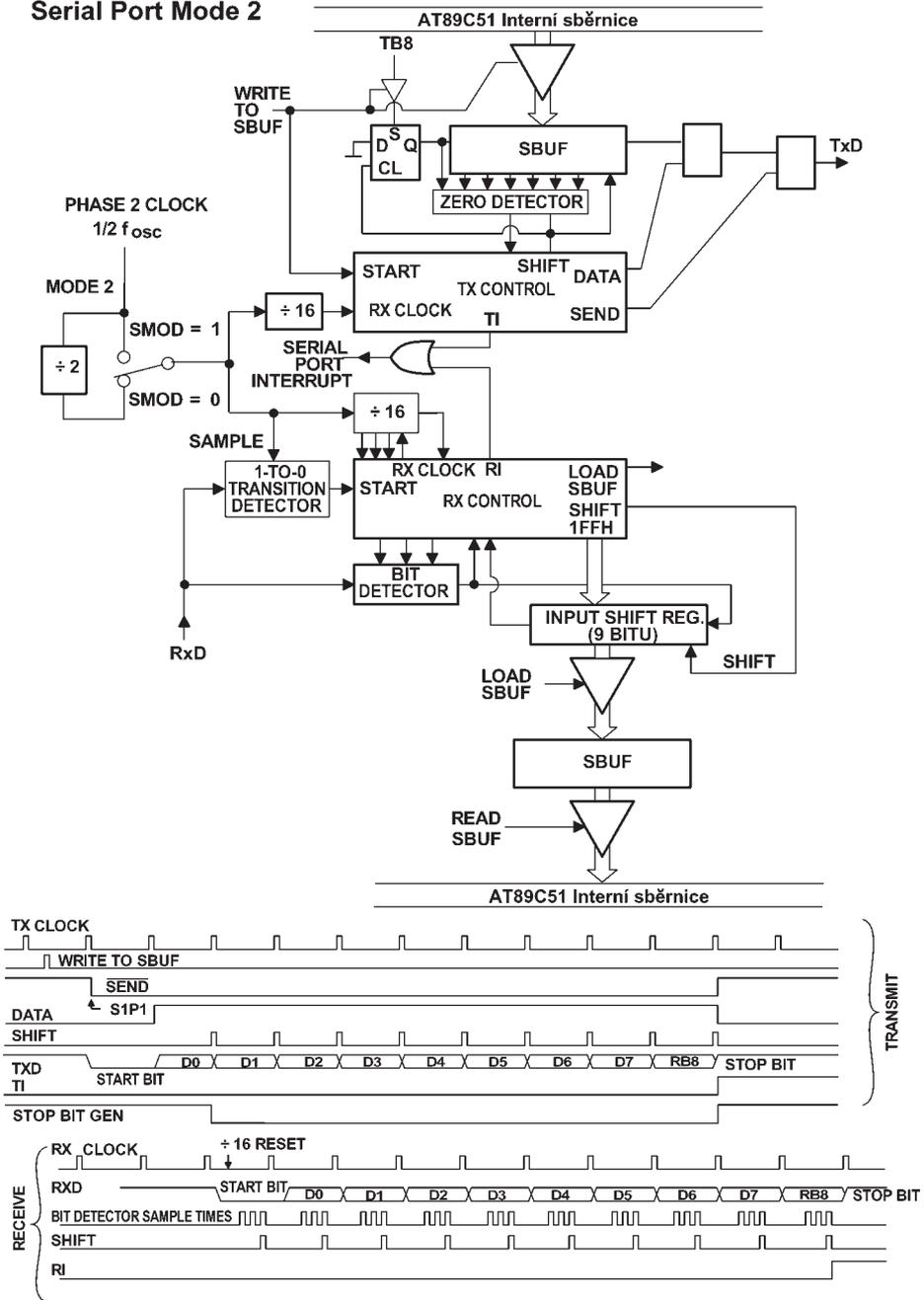
Nesplněním některé z těchto podmínek, je celý přijímaný vzorek nenapravitelně ztracen a RI není nastaven. Splněním obou podmínek je 9. bit vložen do RB8 a 8 datových bitů se přesune do registru SBUF a příznak RI je nastaven. V této době, nezávisle na tom, zda byly podmínky splněny, testuje jednotka další sestupnou hranu na vstupu RxD.

Hodnota přijímaného stop bitu neovlivňuje SBUF, RB8 ani RI.

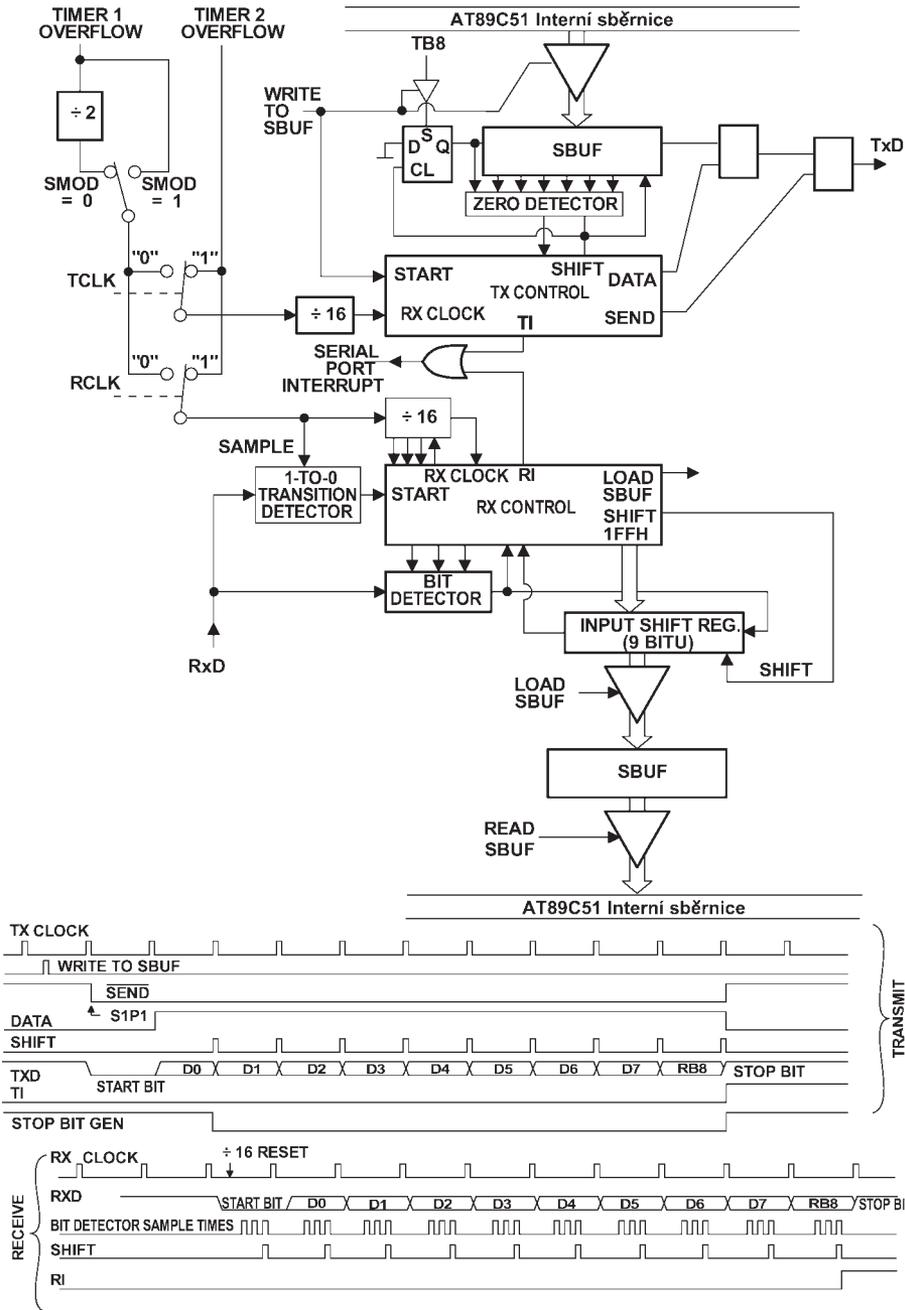
Režim 3 – obr. 29

V režimu 3 se rovněž přenáší 11bitová slova na výstupu TxD a vstupují přes vývod RxD. Slovo začíná startovacím bitem, za kterým následuje 8 datových bitů, programova-

Serial Port Mode 2



Obr. 28 Sériový kanál – režim 2



Obr. 29 Sériový kanál – režim 3

telný 9. bit a jeden stop bit. Vysílání 9. programovatelného bitu se realizuje přes programově ovladatelný bit TB8 v registru SCON. Uvedený bit se může použít jako paritní bit. Při vysílání se do RB8 zapíše paritní bit P ze stavového registru PSW. Při příjmu se programovatelný 9. bit zapisuje do RB8 v registru SCON a může se současně s převzetím obsahu přijímacího registru testovat. Stop bit se nezapisuje vůbec. Přenosová rychlost je volitelná.

Pro všechny režimy činnosti je sériový přenos inicializován instrukcí, která pracuje s registrem SBUF jako cílovým registrem. Příjem je v režimu 0 inicializován za podmínky RI = 0 a REN = 1. V ostatních režimech je příjem inicializován přijetím start bitu, je-li REN = 1.

1.7.2 NASTAVOVÁNÍ REŽIMŮ ČINNOSTI SÉRIOVÉHO KANÁLU

Nastavení režimů sériového kanálu se realizuje zápisem stavového slova do řídicího registru SCON, který má následující tvar:



Jednotlivé bity SM0, SM1 mají význam dle tabulky:

Režim	SM0	SM1
0	0	0
1	0	1
2	1	0
3	1	1

Tab. 7 Režimy sériového kanálu

- SM2** – povolení multiprocessorové komunikace v režimech 2 a 3, je-li SM2 = 1. Pro režim 0 má být SM2 = 0.
- REN** – povolení sériového příjmu dat. Nastavuje se programovými prostředky po ukončení činnosti přijímače.
- TB8** – definování 9. bitu, který se vysílá v režimu 2 a 3. Nastavuje se programovými prostředky.
- RB8** – v režimu 2. a 3. Je to 9. bit přijatého datového slova. V režimu 1, je-li SM2 = 0 je RB8 rovný přijatému stop bitu. V režimu 0 se RB8 neuzivá.
- TI** – příznakový bit přerušení od vysílače. Nastavuje se automaticky v době vysílání 8. bitu v režimu 0 a při vysílání stop bitu ve všech ostatních režimech. Musí se programově nulovat.
- RI** – příznakový bit přerušení od přijímače. Nastavuje se automaticky při příjmu 8. bitu v režimu 0 nebo po příjmu první poloviny stop bitu v ostatních režimech (vyjma případu, je-li SM2 = 1). Musí se programově nulovat.

1.7.3 MULTIPROCESOROVÁ KOMUNIKACE

Režimy 2 a 3 mají specifické možnosti, které se dají využít při multiprocesorové komunikaci. V těchto režimech se totiž 9. bit přijímaného slova zapisuje do bitu RB8 v registru SCON. Za tímto bitem se přijímá stop bit. Sériový kanál se může naprogramovat tak, že přerušení od sériového kanálu se generuje jen, když $RB8 = 1$. Tento režim je možné nastavit bitem $SM2 = 1$ v registru SCON a využít v multiprocesorové komunikaci typu „MASTER-SLAVE“. V uvedené komunikaci je definován adresový byte tak, že v 9. bitu (TB8) je povinně zapsána „1“ na rozdíl od datových, které mají v 9. bitu povinně zapsanou „0“. Mikrokontrolér typu „MASTER“ vysílá tedy nejdříve adresové slovo, jehož 9. bit se zapisuje do bitu RB8 v registru SCON mikrokontroléru „SLAVE“ a vyvolá tak při nastaveném bitu $SM2 = 1$ přerušení činnosti všech mikrokontrolérů „SLAVE“. Tyto mikrokontroléry mohou pak porovnat vyslanou adresu se svou přidělenou adresou a při shodě adres přejít do režimu příjmu datových slov, které mají v 9. bitu povinně zapsanou „0“, tedy i při nastaveném bitu $SM2 = 1$ nevyvolávají přerušení činnosti mikrokontrolérů typu „SLAVE“. Ostatní mikrokontroléry „SLAVE“ budou následující datové byty (9. bit = 0) ignorovat. Bit $SM2$ nemá vůbec vliv v režimu 0, ale může být použit na ověření platnosti stop bitu v režimu 1. V režimu 1, je-li $SM2 = 1$, není přerušení v důsledku příjmu aktivováno, pokud není stop bit přijat platně.

1.7.4 NASTAVENÍ PŘENOSOVÉ RYCHLOSTI ČÍTAČEM 1

Přenosová rychlost sériové komunikace je závislá na zvoleném režimu činnosti. V režimu 0 je přenosová rychlost pevná a je dána vztahem

$$v_0 = \text{kmitočet oscilátoru}/12$$

Přenosová rychlost v režimu 2 je závislá na nastavení bitu SMOD v SFR registru PCON. Je-li $SMOD = 0$, dělí se kmitočet oscilátoru v poměru 1 : 64, je-li $SMOD = 1$, dělí se kmitočet oscilátoru v poměru 1 : 32. Přenosová rychlost v režimu 2 se vyjadřuje vztahem

$$v_2 = (2^{SMOD} \times f_{osc})/64$$

V režimech 1 a 3 se přenosová rychlost nastavuje čítačem 1 a je popsána vztahem

$$v_{1,3} = (2^{SMOD} \times [\text{rychlost plnění čítače 1}])/32$$

V mikrokontrolérech AT89C52... je přenosová rychlost určena čítačem 1 nebo čítačem 2 popřípadě oběma (jeden pro příjem, druhý pro vysílání).

Pro takový způsob použití čítače 1 se musí přerušení maskovat při jeho přeplnění.

Čítač 1 může pracovat jako čítač nebo časovač v libovolném režimu. V převážné většině aplikací se používá v režimu s předvolbou, který se nastaví tak, aby horní 4 bity registru TMOD byli rovné „0010B“. Pak je přenosová rychlost rovna (samoplňací režim)

$$v_{1,3} = (2^{SMOD} / 32) \times (f_{osc} / (12 \times 256 - (TH1)))$$

K dosažení velice malé přenosové rychlosti se pro čítač 1 používá režim 1, ve kterém pracuje čítač jako 16bitový registr a přerušení od čítače 1 při přeplnění se využije na opakované nastavování předvolby registrů čítače.

Nastavení normalizovaných hodnot přenosových rychlostí a potřebné podmínky jsou uvedeny v tab. 8.

1.7.5 POUŽITÍ ČÍTAČE 2 JAKO GENERÁTORU PŘENOSOVÉ RYCHLOSTI

V mikrokontrolérech AT89C52... se čítač 2 stává generátorem přenosové rychlosti použitím bitů TCLK a RCLK v registru T2CON. Za těchto podmínek může být přenosová rychlost pro vysílání a příjem současně různá. Nastavením RCLK nebo TCLK uvede čítač 2 do režimu generátoru přenosové rychlosti.

Režim generátoru přenosové rychlosti se podobá režimu čítače s přednastavením a přetečení v TH2 registru způsobí znova přednastavení 16bitové hodnoty z registrů RCAP2L a RCAP2H, které byly nastaveny programově. V tomto případě je přenosová rychlost v režimu 1 a 3 určena přeplněním čítače 2 a je dána vztahem

$$v_{1,3} = (\text{rychlost plnění čítače 2})/16$$

Čítač 2 může pracovat jako čítač nebo časovač. Ve většině typických aplikací je použit jako časovač (C/T2 = 0). Čítač se inkrementuje každý strojový cyklus, tedy 1/12 kmitočtu oscilátoru, ale obsluha čítače je jiná, je-li čítač 2 použit jako generátor přenosové rychlosti. Tehdy je čítač inkrementován v každém strojovém cyklu. V tomto případě je přenosová rychlost dána vztahem

$$v_{1,3} = f_{osc}/(32 \times [65536 - (RCAP2H, RCAP2L)])$$

kde RCAP2H, RCAP2L je obsah záchytných registrů reprezentován 16bitovým kladným celým číslem.

Obr. 25 je schéma činnosti čítače 2 jako generátoru přenosové rychlosti za předpokladu nastavení RCLK + TCLK = 1 v registru T2CON. Přeplnění registru TH2 nenastaví příznak TF2 a tedy nevyvolá přerušení. Proto přerušení od čítače 2 nemůže být zakázáno, když je čítač 2 v tomto režimu. Je-li EXEN2 nastaven, pak sestupnou hrana na vstupu T2EX nastaví příznak EXF2, avšak nezpůsobí to již opětovné nastavení čítače ze záchytných registrů RCAP2H, RCAP2L. Takto může být vstup T2EX použit jak externí požadavek na přerušení. Když čítač 2 počítá (TR2 = 1), programátor by nemohl přečíst nebo zapsat z nebo do registrů

Přenosová rychlost	f _{osc} [MHz]	SMOD	Čítač 1		
			C/T	Režim	Předvolba
Režim 0: Max 1 MHz	12	x	x	x	x
Režim 2: Max 375 K	12	1	x	x	x
Režim 1,3: Max 62.5	12	1	0	2	FFH
19.2K	11.059	1	0	2	FDH
9.6K	11.059	0	0	2	FDH
4.8K	11.059	0	0	2	FAH
2.4K	11.059	0	0	2	F4H
1.2K	11.059	0	0	2	F8H
137.5K	11.059	0	0	2	1DH
110	6	0	0	2	72H
110	12	0	0	1	FEEBH

Tab. 8 Přenosové rychlosti

TH2 a TL2. Za těchto podmínek čítač 2 je inkrementován v každém strojovém cyklu a výsledky čítání a zápisu by byly nepřesné. RCAP registry mohou být přečteny, avšak nemůžou být přepsány, protože zápis může způsobit kolizi přednastavení a způsobit chybu v přednastavení. Čítač 2 je nutné zastavit ($TR2 = 0$) dříve, než přistoupíme k registrům [TH2, TL2] nebo RCAP2H, RCAP2L.

1.8 PŘERUŠOVACÍ SYSTÉM

V mikrokontrolérech ATMEL existuje pět (šest) zdrojů přerušení: dva externí vstupy, dva (tři) přerušení od čítačů/časovačů jeden od sériového kanálu. Jednotlivé zdroje přerušení jsou znázorněny na *obr. 30* a *obr. 31*.

Externí zdroje přerušení se přivádějí na vstupy $\overline{INT0}$ a $\overline{INT1}$, které lze nastavit tak, že jsou úroňové (reagují na úroveň „0“) nebo reagují na sestupnou hranu vstupního impulsu. Volba charakteru vstupů $\overline{INT0}$ a $\overline{INT1}$ se realizuje pomocí bitů IT0 a IT1 v registru TCON. Přerušení od vstupů $\overline{INT0}$ a $\overline{INT1}$ nastavují příznaky IE0 a IE1 v registru TCON. Nastavené bity těchto příznaků se automaticky nulují po obsluze žádosti o přerušení. Pokud je vstup $\overline{INT0}$ resp. $\overline{INT1}$ nastaven jako úroňový je nutné, aby vnější zdroj přerušení zrušil svou žádost dříve, než skončí obslužný podprogram jeho žádosti o přerušení, protože by se obslužná procedura přerušení zavolala hned po opuštění svého předchozího volání, takže by nezbyl čas na obsluhu jiných instrukcí mimo obsluhu tohoto přerušení. Při nastavení vstupu $\overline{INT0}$ resp. $\overline{INT1}$ tak, že sestupnou hranu vstupního signálu, bity těchto příznaků přerušení se automaticky nulují po obsluze žádosti.

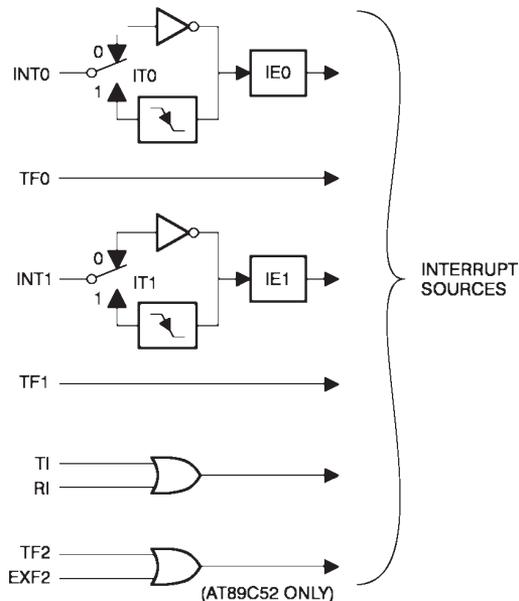
Čítače 0 a 1 generují žádosti o přerušení vnitřními signály TF0 a TF1 při splnění různých podmínek po nastaveném režimu činnosti. Žádosti o přerušení se nulují automaticky při přechodu na jejich obslužný podprogram. Sériový kanál vytváří součtový signál žádosti o přerušení od přijímače RI a vysílače TI. Nulování těchto žádosti o přerušení se musí realizovat v programu.

Čítač 2 vytváří v mikrokontrolérech 89C252, 89C53, 89C55 také součtový signál žádosti o přerušení ze vstupu externího přerušení EXF2 a z příznaku přeplnění čítače 2 TF2. Ani jeden z příznaků uvedených přerušení se nenuluje automaticky, proto se musí nulovat programovými prostředky. Všechny příznakové bity přerušení se mohou nulovat resp. nastavovat programově se shodným výsledkem, jako kdyby byly ovládány technickými prostředky. Každá žádost o přerušení se může maskovat obsahem speciálního registru IE ve skupině SFR, přičemž pomocí bitu EA možno maskovat všechny přerušení. Registr IE má tvar:

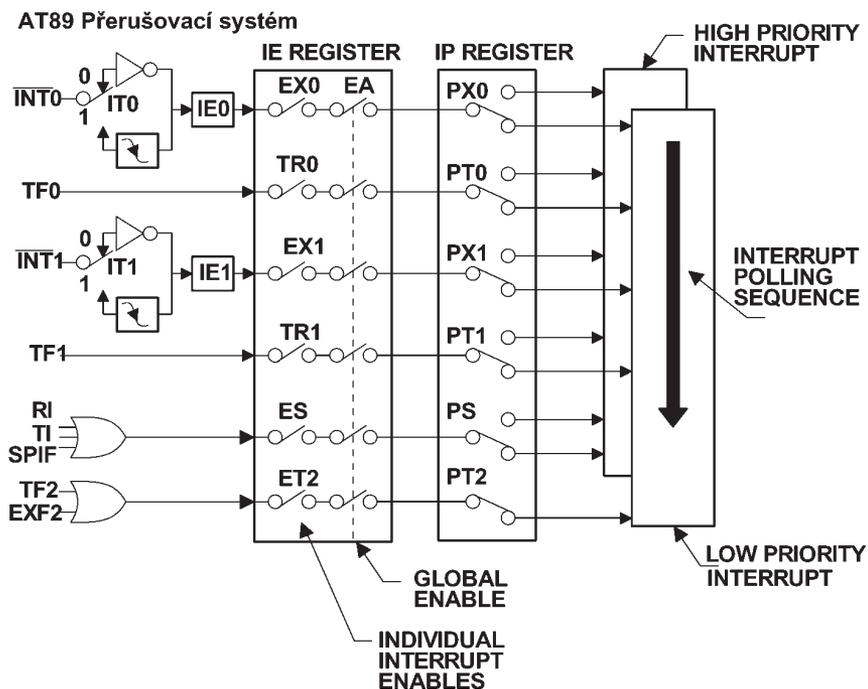
	(MSB)							(LSB)
IE	EA	-	ET2	ES	ET1	EX1	ET0	EX0

- EA** – povolení resp. zákaz všech typů přerušení. Je-li EA = 1, pak se mohou jednotlivé zdroje přerušení maskovat individuálně příslušným bitem v registru IE. Je-li EA = 0, pak jsou všechny zdroje přerušení zakázána.
- ET2** – maskování přerušení od čítače 2 (AT89C52...).
- ES** – maskování přerušení od sériového kanálu.

Interrupt Sources



Obr. 30 Zdroje přerušení



Obr. 31 Zdroje přerušení dle priority

ET1 – maskování přerušení od čítače 1;

EX1 – maskování přerušení od vstupu externího přerušení $\overline{\text{INT1}}$;

ET0 – maskování přerušení od čítače 0;

EX0 – maskování přerušení od vstupu externího přerušení $\overline{\text{INT0}}$.

Nejrychlejší možná odezva nastane v posledním strojovém cyklu C2 instrukce, která není RETI, nebo instrukcí změny registrů IE a IP.

Mimo maskování jednotlivých žádostí o přerušení, se jim může přidělit jedna ze dvou úrovní priority nastavením resp. nulováním příslušného bitu ve speciálním registru IP. Po resetu mikrokontroléru je registr IP nulový, což přidělí všem zdrojům přerušení nižší priority. Obsluha přerušení s nižší úrovní priority může být přerušena přerušením s vyšší úrovní priority, ale nikdy ne s úrovní nižší. Obsluha přerušení s vyšší úrovní priority nemůže být přerušena nikdy. Jestliže dva zdroje přerušení s rozdílnou úrovní priority jsou přijaty současně, bude akceptován požadavek s vyšší úrovní priority. Když dva zdroje přerušení se shodnou úrovní priority jsou přijaty současně, vnitřní definice pořadí určí, který požadavek bude akceptován (viz tab. 9). V rámci každé ze dvou úrovní priorit jsou jednotlivým žádostem o přerušení přiřazeny pevně stanovené priority podle tab. 9.

Zdroj přerušení	Priority
IE0	nejvyšší
TF0	
IE1	
TF1	
RI + TI + SPIF	
TF2 + EXF2	nejnižší

Tab. 9 Priority přerušení

Vyšší úroveň priority se zvolí nastavením příslušného bitu v IP, který má tvar:

	(MSB)							(LSB)
IP	-	-	PT2	PS	PT1	PX1	PT0	PX0

Jednotlivé bity mají následující význam:

PT2 – úroveň priority čítače 2;

PS – úroveň priority sériového kanálu;

PT1 – úroveň priority čítače 1;

PX1 – úroveň priority externího přerušení $\overline{\text{INT1}}$;

PT0 – úroveň priority čítače 0;

PX0 – úroveň priority externího přerušení $\overline{\text{INT0}}$.

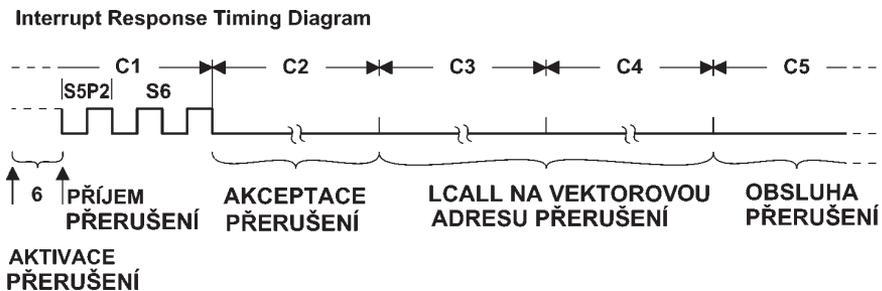
1.8.1 OBSLUHA PŘERUŠENÍ

Žádosti o přerušení se vzorkují ve stavu S5/P2 v každém strojovém cyklu a vyhodnocují se v následujícím strojovém cyklu. Přerušovací cyklus čítače/časovače 2 je jiný. Pokud byl jeden z jeho příznaků nastaven před stavem S5/P2, přerušení vygeneruje instrukci LCALL, jestli-že není blokována některou z následujících podmínek:

1. Přerušení je shodné nebo nižší úrovně priority jako u obsluhy, která probíhá.
2. Aktuální kontrolní cyklus rozhodování není posledním cyklem výkonu instrukce probíhající obsluhy.
3. Vykonávaná instrukce je RETI nebo instrukce zápisu do registrů IE a IP.

Každá z těchto podmínek bude blokovat generování instrukce LCALL pro obsluhu přerušení. Podmínka č. 2 zabezpečí, že právě probíhající instrukce bude vykonána před přechodem na obsluhu nového přerušení. Podmínka č. 3 zabezpečí, že je-li vykonávána instrukce RETI nebo instrukce přístupu do registrů IE a IP, pak nejméně jedna instrukce bude vykonána před přechodem na obsluhu přerušení.

Kontrolní cyklus rozhodování se opakuje v každém strojovém cyklu s hodnotou, která byla přítomna ve stavu S5/P2 předcházejícího strojového cyklu. Když není aktivní příznak přerušení obslužen, protože jedna nebo více blokovacích podmínek je aktivních, přestože jsou blokovací podmínky zrušeny, zakázané přerušení nebude obsluženo. Jinak řečeno: „skutečnost, že příznak přerušení byl aktivní, avšak obslužen nebyl, se neumí zapamatovat“. Každý kontrolní cyklus rozhodování se vykonává znovu. Všimněte si, že když přerušení s vyšší úrovní priority je aktivní před stavem S5/P2 strojového cyklu s označením C3 na obr. 32, pak ve shodě s vyšší prioritou bude obsluženo v době cyklů C5 a C6 bez ohledu na to, že nějaká instrukce v obsluze přerušení s nižší prioritou byla vykonávána. Tak mikrokontrolér připouští, aby požadavek o přerušení byl vykonán generováním instrukce LCALL na příslušnou obsluhu.



Obr. 32 Průběh odezvy na přerušení

Obsluha žádostí o přerušení se realizuje po každém instrukčním cyklu generováním instrukce LCALL, která uloží obsah čítače instrukcí PC do zásobníku („stack“ – adresován ve speciálním registru SP standardně na hodnotu 07H) a naplní vektorovou adresou příslušného zdroje přerušení podle tab. 10:

Přerušení	Zdroj přerušení	Vektorová adresa	Bit v SFR
Externí 0	IE0	0003H	TCON.1
Čítač 0	TF0	000BH	TCON.5
Externí 1	IE1	0013H	TCON.3
Čítač 1	TF1	001EH	TCON.7
Sériový kanál	RI + TI + SPIF	0023H	SCON.0,SCON.1,SPSR.7
Čítač 2	TF2 + EXF2	002BH	T2CON.7,T2CON.6
Reset	RST	0000H	-

Tab. 10 Vektory přerušení

Instrukce LCALL neodkládá obsah registru PSW do zásobníku, takže při obsluze přerušení je nevyhnutelné nejdříve uložit i obsah dalších registrů, které se při obsluze přerušení budou používat. Na konci obslužného podprogramu je potřebné obsahy těchto registrů „obnovit“.

Obslužný podprogram se musí ukončit instrukcí RETI. Tato instrukce obnoví obsah čítače programu (PC) uloženého v zásobníku (SP) a indikuje konec obsluhy přerušení. Jestliže by se na ukončení podprogramu použila instrukce RET, nastane návrat do hlavního programu, ale přerušovací systém je stále blokován, protože se neindikuje konec obsluhy přerušení, jako při instrukci RETI.

Vstupy externích žádostí o přerušení se mohou programovat tak, že reagují na sestupnou hranu – vstupního signálu nebo na jeho úroveň v „0“ nastavením, popřípadě nulovým bitu IT1 resp. IT0 ve speciálním registru TCON.

Doba generování externí žádosti o přerušení je rozdílná, v závislosti na typu žádosti. Při žádosti s aktivním týlem je potřebné, aby nejméně v době jednoho strojového cyklu byl vstupní signál na úrovni „1“ a pak nejméně po dobu dalšího strojového cyklu na úrovni „0“. Bity IE0 resp. IE1 se automaticky nulují po přechodu na obslužný podprogram. Při žádosti o přerušení s aktivní úrovní vstupního signálu je potřebné udržet úroveň „0“ na vstupu až do té doby, pokud se přerušení nezačne obsluhovat. V opačném případě nebude žádost o přerušení akceptována.

1.8.2 KROKOVÁNÍ PROGRAMU

Mikrokontroléry ATMEL nemají možnost krokování programu po instrukcích technickými prostředky. Krokování se může realizovat jenom programově.

Jednou z možností je využití vlastnosti, že obsluha přerušení té samé úrovně se může realizovat až po výkonu instrukce RETI a po výkonu nejméně jedné další instrukce.

Tedy, když se přejde na obsluhu přerušení, nemůžeme do obslužného podprogramu znovu vstoupit, pokud se nevykoná alespoň jedna instrukce přerušného programu.

Program, který využívá externí přerušení INT0 na krokování má tvar:

Adresa	Strojový kód	Assembler	Význam
0003	30,B2,FD	S0:JNB P3.2, S0	čekání na zrušení INT0
0006	20,B2,FD	S1:JB P3.2, S1	čekání na aktivaci INT0
0009	32	RETI	návrat do přerušného programu a vykonání jedné instrukce

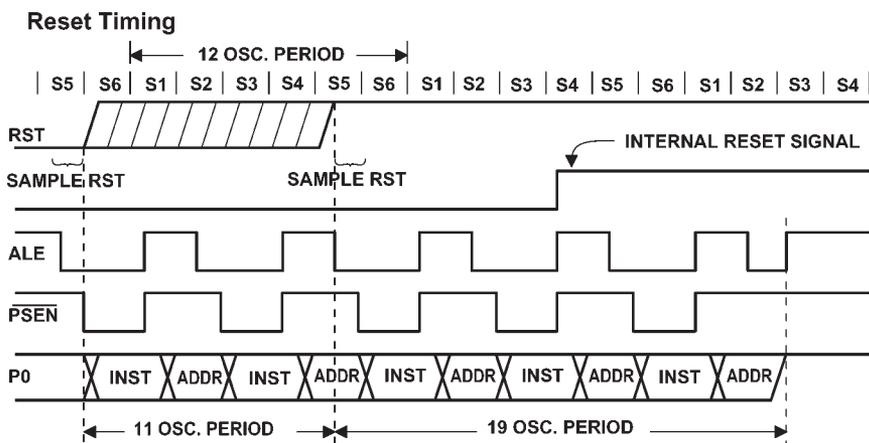
Nastavíme-li $\overline{\text{INT0}} = 0$ na vývodu P3.2, přejde se po akceptování této žádosti na vektorovou adresu 0003, kde se nachází první instrukce krokovacího podprogramu, která realizuje čekací smyčku, tj. čas dokud se nezruší žádost $\overline{\text{INT0}}$. Po zrušení žádosti ($\overline{\text{INT0}} = 1$) se přejde na další instrukci, která realizuje čekací smyčku, pokud se $\overline{\text{INT0}}$ nezaktivizuje ($\overline{\text{INT0}} = 0$). Potom se vykoná instrukce RETI, která zabezpečí návrat do přerušenoého programu, vykoná se jediná instrukce a protože $\overline{\text{INT0}} = 0$ přejde se opět na adresu 0003. Doba, po kterou je $\overline{\text{INT0}} = 1$ je doba trvání v jednom kroku.

1.9 INICIALIZACE MIKROKONTROLÉRU

Na inicializaci mikrokontroléru je k dispozici vstup RST (reset), na který je připojen interní Schmittův klopný obvod. Proto lze na tento vstup připojit i spojitě se měnící napětí, tj. např. napětí na kondenzátoru při automatické inicializaci po připojení napájecího napětí.

Reset mikrokontroléru proběhne, bude-li signál RST = 1 alespoň po dobu dvou strojových cyklů (24 period hodinových impulzů oscilátoru). Samozřejmě pokud oscilátor běží. V době inicializace se nastaví ALE a PSEN do vstupního stavu. Vnitřní inicializace se vykoná v době druhého strojového cyklu, ve které je RST = 1 a opakuje se pokud RST = 0. Vnitřní inicializace ovlivní obsah registrů následovně:

Obsah interní paměti RAM se resetem neovlivní a po zapnutí napájecího napětí je v ní náhodný obsah (v aplikacích, kde na tom záleží je potřeba ji vynulovat programově po resetu). Časový průběh resetu je na obr. 33.



Obr. 33 RESET – časování

Obsah jednotlivých speciálních registrů SFR po inicializaci mikrokontroléru je znázorněn v tab. 11.

Registr SFR	Adresa v SFR	Obsah po resetu
PC	-	0000H
ACC	E0H	00H
B	F0H	00H
PSW	D0H	00H
SP	81H	07H
DPTR0 [DPH0,DPL0]	83H,82H	0000H
DPTR1 [DPH1,DPR1]	85H,84H	0000H
P0	80H	FFH
P1	90H	FFH
P2	A0H	FFH
P3	B0H	FFH
IP	B8H	XX000000B
IE	A8H	0X000000B
TMOD	89H	00H
T2MOD	C9H	XXXXXX00B
TCON	88H	00H
T2CON	C8H	00H
TH0	8CH	00H
TL0	8AH	00H
TH1	8DH	00H
TL1	8BH	00H
TH2	CDH	00H
TL2	CEH	00H
RCAP2H	CBH	00H
RCAP2L	CAH	00H
SCON	98H	00H
SBUF	99H	náhodný
SPSR	AAH	00XXXXXXB
SPDR	86H	náhodný
SPCR	D5H	000001XXB
WCON / WMCON	96H	00000010B
PCON	87H	0XXX0000B

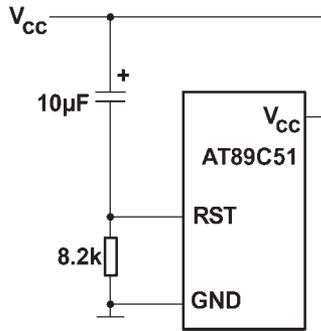
Tab. 11 Registry SFR po inicializaci

Vstup RST je vzorkován v době stavu S5/P2 každého strojového cyklu. Po dobu 19 period oscilátoru po přivedení log. 1 na vstup RST jsou na vývodech portu udržovány aktuální hodnoty tzn., že externí resetovací signál může být aplikován od 19. po 31. periodu oscilátoru.

Zapojení resetovacího obvodu je na obr. 34.

Pro verze CMOS lze rezistor vynechat, protože vstup RST má vnitřní zatěžovací rezistor. Tehdy musí být hodnota předřazené kapacity zmenšena na úroveň 1 μF .

Power-On Reset Circuit



Obr. 34 Resetovací obvod

Po připojení napájení V_{CC} drží obvod vstup RST ve vysoké úrovni po dobu, která je závislá na hodnotě připojené kapacity a rychlosti jakou bylo napájení přivedeno. Abychom si byli jisti, že RESET bude platný, musí být RST držen ve vysoké úrovni déle než dva strojové cykly.

Náběh napájecího napětí V_{CC} trvá přibližně 10 ms. Startovací čas oscilátoru je závislý na jeho kmitočtu. Např. pro 10 MHz krystal je asi 1 ms. Pro 1 MHz krystal je to asi 10 ms.

Pro daný obvod rychlejší pokles V_{CC} na 0 způsobí, že napětí na RST spadne na nulovou hodnotu okamžitě. Avšak toto napětí je interně omezeno a nezpůsobí poškození zařízení.

Přivedení výstupů ALE a PSEN do log. 0 pokud je reset aktivní, může způsobit, že zařízení se dostane do neurčitěho stavu.

Přivedení napájecího napětí V_{CC} bez vykonání platného RESETu způsobí, že CPU mikrokontroléru začíná vykonávat program z náhodného místa v paměti programu, protože programové počítadlo PC nebylo spolehlivě vynulováno a jeho obsah by byl náhodný.

1.10 NAPÁJENÍ A ZÁLOHOVÁNÍ

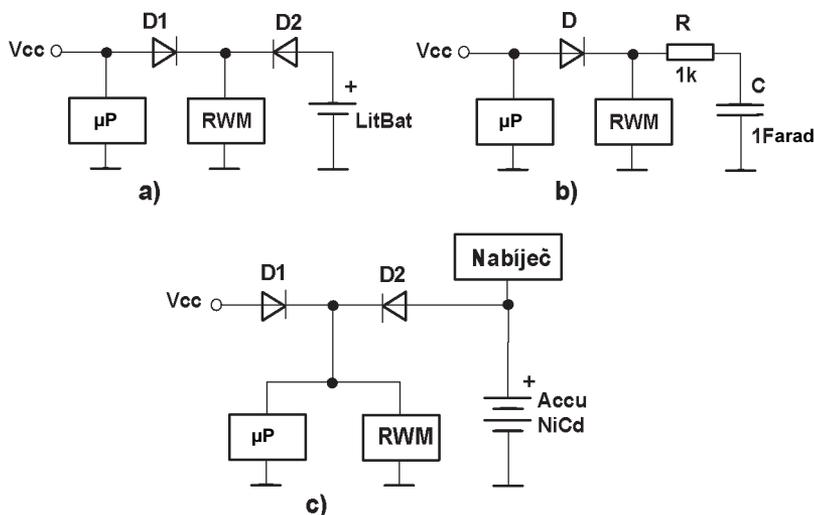
Způsob napájení mikroprocesorových systémů je velice důležitý. Má výrazný vliv na činnost obvodové soustavy jak při kontinuálním napájení tak i při jeho přerušení. Mikroprocesory ATMEL standardně pracují s napájecím napětím od 2,7 do 6 V. Velikost napětí má vliv na maximální hodinový kmitočet, jenž může být použit pro danou aplikaci. Zpravidla čím vyšší napětí, tím vyšší kmitočet. Skutečně velice zajímavou se stává otázka způsobu napájení z hlediska zálohování dat hlavně při poruchách v energetické síti. Např. u mikroprocesorových systémů používaných v měřicích a regulačních systémech se často vyžaduje nutnost zachování dat i při výpadku napájení a korektní pokračování v činnosti po obnově napájení. Data jsou totiž tou nejdůležitější částí z mnoha informací, které jsou uloženy v paměťových obvodech. Jedná se zejména o uložení dat v paměti RWM.

Jeden z nejjednodušších způsobů jak zálohovat napájení paměti je znázorněn na obr. 35a. Při normální činnosti je RWM napájena přes diodu D1 a v případě výpadku sítě přes diodu D2 z litiové baterie s ohledem na to, že paměti mají velice nízkou spotřebu. V případě nutnosti zálohovat data jen omezenou dobu můžeme použít vysokokapacitní kondenzátor 1F, kterým můžeme zálohovat napájení pro RWM několik hodin, obr. 35b. Kondenzátor je nabíjen přes diodu a rezistor R. V případě ztráty napájecího napětí zálohuje napájení pro paměť RWM. Dioda D v tomto případě brání vybití kondenzátoru přes ostatní obvody v zapojení.

Jestliže potřebujeme zálohovat také napájení mikrokontroléru, je vhodné využít zapojení s nabíjením zálohovacího článku, viz obr. 35c. V tomto případě mikro počítač při výpadku síťového napájení pokračuje v normální činnosti, avšak je napájen ze zálohového zdroje, který musí být dostatečně dimenzován vzhledem k vyššímu odběru. Může to být nikel-kadmiový článek (NiCd), který je při normální činnosti trvale dobíjen.

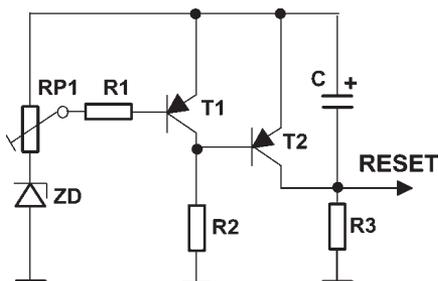
Způsobů zapojení jak zálohovat napájení je mnoho. V odborné literatuře zaměřené na mikroprocesorovou techniku jsou tato zapojení často uváděna.

Zálohování dat při přerušení napájení je možné výrazně zjednodušit využitím NV-SRAM paměti, které uchovávají data i při ztrátě napájení. Jsou to např. paměti EEPROM.



Obr. 35 Způsoby zálohování napájení

Velice důležitou činností při přerušení napájení, nebo poklesu napájecího napětí je zabezpečení korektního obnovení činnosti chodu mikroprocesoru. Na obr. 36 je znázorněn hlídací obvod podpětí, u kterého se v důsledku poklesu napájecího napětí uzavře tranzistor T1 a otevře tranzistor T2, čímž je na výstupu generován kladný resetovací impuls.



Obr. 36 Hlídací obvod podpětí

V současné době je možné s výhodou použít specializované integrované obvody tzv. hlídače podpětí pod označením (např. TL 7702 až 12), které nabízejí mnozí výrobci obvodů. Mimo jiné nabízejí i tzv. Watch Dog obvody, které generují resetovací signál i při poruše činnosti mikrokontroléru. Ten v korektním režimu generuje na některém z vývodů pravidelně impulzy, jež vstupují do obvodu Watch Dog a které při nesprávné činnosti mají jiný charakter středy nebo zaniknou, v důsledku čehož Watch Dog generuje signál Reset.

Výrobci nabízejí tyto obvody pod označením např. MAX1232, TC1232, resp. DC1232, popřípadě MAX690 nebo ADM690.

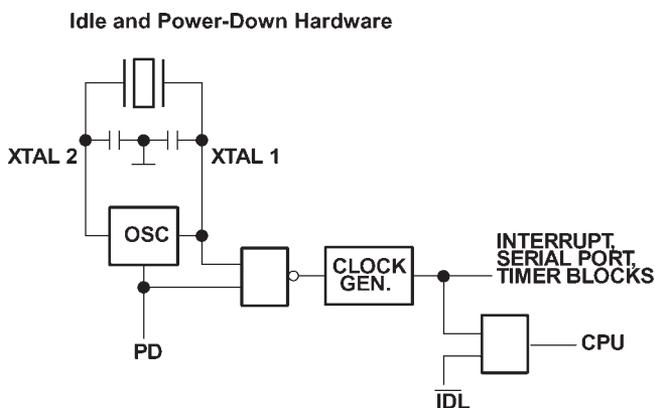
1.11 PROVOZ SE SNÍŽENÝM PŘÍKONEM

V některých aplikacích, kde je spotřeba energie limitujícím faktorem, mohou mikrokontroléry ATMEL pracovat v režimu se sníženým příkonem. Pro mikrokontroléry vyrobené technologií CHMOS je provoz se sníženým příkonem typickým rysem.

Snížení příkonu mikrokontrolérů se dosahuje tak, že napájecí napětí V_{CC} se odpojí, přičemž zapamatování obsahu interní RAM zabezpečuje napájecí napětí připojené na vstup RST. Využití této vlastnosti umožňuje při zjištění poklesu napájecího napětí V_{CC} generovat přerušeni a přenést důležité obsahy registrů do interní paměti RAM a povolit napájení ze záložního zdroje na vstupu RST. Po obnově napájecího napětí V_{CC} je možné mikrokontrolér inicializovat signálem RST = 1 a tak obnovit jeho činnost.

U mikrokontrolérů CMOS jsou možné ještě dva režimy se sníženým příkonem. Odlišují se rozsahem chráněné oblasti a způsobem inicializace.

Zapojení vnitřních obvodů pro tyto režimy činnosti znázorňuje obr. 37.



Obr. 37 Režimy IDLE a POWER DOWN

Provoz se sníženým příkonem se řídí podle čtyř spodních bitů registru PCON, který má následující tvar:

	<i>(MSB)</i>							<i>(LSB)</i>	
PCON	SMOD	-	-	-	GF1	GF0	PD	IDL	

- SMOD** – Příznak dvojnásobné přenosové rychlosti. Je-li SMOD = 1, čítač T1 bude použit jako generátor přenosové rychlosti a sériový kanál pracuje v režimech 1, 2 nebo 3.
- GF1** – příznakový bit určující, zda přerušení bylo generováno v provozu se sníženým příkonem;
- GF0** – příznakový bit určující, zda přerušení bylo generováno v normálním provozu;
- PD** – příznakový bit POWER DOWN režimu;
- IDL** – příznakový bit IDLE režimu.

Ostatní bity registru nejsou využity.

Režim IDLE (IDL = 1) se vyznačuje tím, že vnitřní generátor hodinových impulzů se odpojí od CPU, ale zůstává připojen pro obvody přerušení od čítačů a sériového kanálu. Stav CPU, tj. obsahy registrů ACC, SP, PC, PSW a všech dalších registrů, se zachová. Vývody vstupně-výstupních portů zůstávají na té logické úrovni na jaké byly, byl-li mikrokontrolér uveden do režimu se sníženým příkonem. Signály ALE a PSEN se drží na úrovni „1“.

Uvedení mikrokontroléru do režimu IDLE se zabezpečuje nastavením bitu IDL = 1 v registru PCON. Instrukce, která vykoná nastavení bitu IDL musí být poslední instrukcí před vstupem do režimu se sníženým příkonem.

Ukončení režimu IDLE se sníženým příkonem se může provést dvěma způsoby:

1. První způsob se realizuje generováním libovolného typu přerušení, které je odmaskováno. Akceptováním žádosti o přerušení se nuluje bit IDL v registru PCON a režim se sníženým příkonem je ukončen. Po dokončení obsluhy přerušení a provedení instrukce RETI bude program pokračovat výkonem instrukce, která následuje za instrukcí zajišťující vstup do režimu se sníženým příkonem. Na rozlišení toho, jestliže přerušení bylo generováno v normálním provozu nebo v režimu se sníženým příkonem slouží dva příznakové bity GF0 a GF1 ve speciálním registru PCON. Např. instrukce, která aktivuje režim IDLE může také nastavit jeden nebo oba příznaky. Když je IDLE režim ukončen, obslužný podprogram může příznaky testovat.
2. Druhý způsob ukončení režimu IDLE se sníženým příkonem se realizuje generováním signálu RST = 1 hardwarovým resetem. V tomto případě CPU vrátí výkon programu na místo, odkud byl do režimu se sníženým příkonem uveden, tj. na výkon následující instrukce. Vnitřní hardwarová struktura zabrání přístupu do interní RAM, ale přístup na vývody portů není chráněn. Na eliminaci změn na vývodech portů je potřeba, aby instrukce, která následuje za instrukcí, co vyvolala režim IDLE, nebyla instrukcí zápisu na port nebo do externí RAM.

Režim POWER DOWN (PD = 1) je druhý možný režim se sníženým příkonem. V tomto režimu se zastaví vnitřní oscilátor a tím se vyřadí všechny funkce. Obsah interní RAM a registrů skupiny SFR se nemění. Na vývodech vstupně-výstupních portů se udržuje stav nastavený obsahem jim příslušných registrů ve skupině SFR, přičemž signály ALE a PSEN jsou udržovány na úrovni „0“

Ukončení tohoto režimu se sníženým příkonem je možné jen generováním signálu RST = 1, který inicializuje obsahy všech registrů skupiny SFR (viz tab. 10), ale nemění obsah interní paměti RAM. V případě, že pro pokračování práce je nutno zachovat obsahy někte-

rých registrů skupiny SFR, je nevyhnutelné jejich obsah zálohovat před uvedením mikrokontroléru do stavu se sníženým příkonem a po RESETU jejich obsahy aktualizovat.

Snížení napájecího napětí V_{CC} (typicky až na 2 V) by se mělo stát až po vstupu do režimu se sníženým příkonem. Při ukončení režimu se sníženým příkonem signál RST lze generovat až po zvýšení napájecího napětí V_{CC} na původní hodnotu. Délka signálu RST = 1 musí být větší než 2 strojové cykly (24 period oscilátoru), protože jen po této době se opětovně spouští oscilátor. Běžně je potřebné generovat signál RST alespoň po dobu 10 mikrosekund, což je doba, ve které se oscilátor bezpečně stabilizuje.

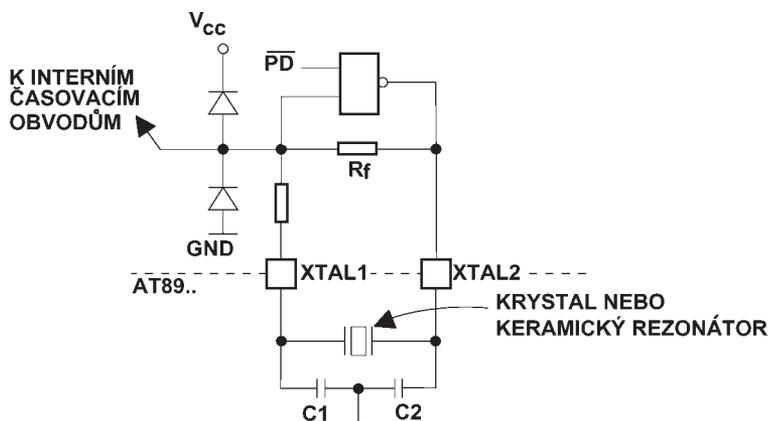
Při projektování aplikací se sníženým příkonem je potřebné také přihlížet na možnost výběru vhodného mikrokontroléru a současně navrhovat jeho oscilátor jako kompromis mezi jeho příkonem a vlastní činností.

1.12 HODINOVÝ GENERÁTOR

Mikrokontroléry ATMEL obsahují ve své struktuře Pierceův oscilátor, který se nastavuje externími obvody. Jednou z možností je použití krystalu připojeného na vývody XTAL1 a XTAL2 podle obr. 38.

Hodinový generátor synchronizuje jednotlivé stavy strojových cyklů a strojové cykly v instrukčních cyklech.

Při použití krystalu je potřebné stabilizovat kmity pomocí dvou keramických kondenzátorů (cca 30 pF) jak ukazuje obr. 38. Použití vhodného krystalu je dáno požadovaným kmitočtem, jeho tolerancí a požadovaným tepelným rozsahem činnosti zařízení.



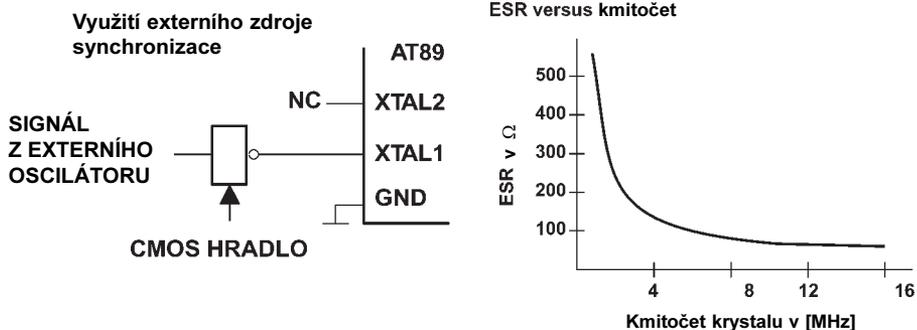
Obr. 38 Zapojení oscilátoru

Kromě krystalu lze použít keramický rezonátor nebo na časovou přesnost méně náročných aplikací i indukčnost.

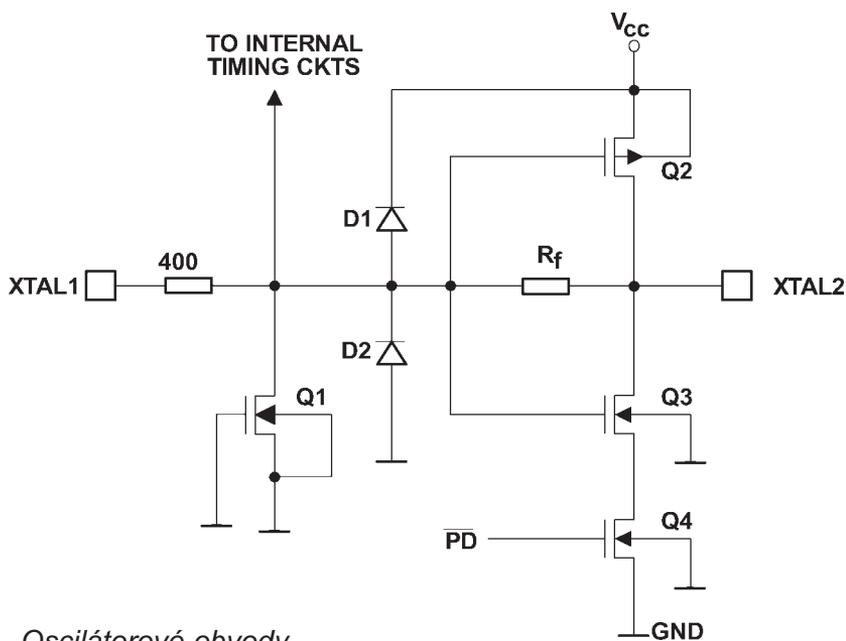
Při použití keramického rezonátoru jsou kapacity C1 a C2 nepatrně vyšší (cca 47 pF). Výrobce keramických rezonátorů obvykle doporučuje vlastní hodnoty těchto kapacit.

Na řízení interního oscilátoru je možno také použít externí hodinové impulzy TTL přivedené na vstup XTAL1, přičemž vstup XTAL2 není připojen, jak znázorňuje obr. 39.

Vnitřní struktura oscilátorového obvodu je znázorněna na obr. 40.



Obr. 39 Externí oscilátor



Obr. 40 Oscilátorové obvody

1.13 VNITŘNĚ OBVODOVÁ EMULACE

Mikrokontroléry ATMEL povolují uživateli testovat a ladit aplikaci bez potřeby vyjmout mikrokontrolér ze zapojení. Do tohoto režimu činnosti tzv. ONCE režim („on-circuit emulation“) je možné uvést mikrokontrolér následovně:

1. Přivést vývod ALE do log. 0 v době resetu a $\overline{\text{PSEN}}$ podržet na úrovni log. 1.
2. Podržet ALE na log. 0, pokud je RST deaktivován.

Pokud je mikrokontrolér v režimu vnitřně obvodové emulace, vývody portu P0 přejdou do třetího stavu a vývody ostatních portů stejně jako ALE a $\overline{\text{PSEN}}$ jsou slabé log. 1. Oscilátor obvodu zůstává v činnosti. Zároveň v tomto režimu mikrokontroléru se může prostřednictvím připojené koncovky emulátoru nebo testeru řídit popřípadě testovat aplikace. Návrat do normálního režimu zajistí platný RESET.

2.1 ZÁKLADNÍ TYPY

V současné době se vyrábí množství typů mikrokontrolérů ATMEL funkčně plně kompatibilních s řadou 8051 firmy INTEL. Jednotlivé typy se liší hlavně řešením paměťového pod-systému a některými funkcemi. Dále jsou tyto obvody vyráběné v různém provedení podle pracovního rozsahu teplot, napájecího napětí a pouzdra. Mezi nejznámější z těchto mikro-počítačů patří typy AT89C51, což jsou vlastně obvody analogické k typům 87C51, ale místo vnitřní paměti typu EPROM mají integrovanou vnitřní paměť typu FLASH EPROM o velikosti 4 kB. Její výhodou je možnost přepsání obsahu elektrickým signálem až tisíckrát bez nutnosti mazat ultrafialovým světlem. Navíc obsah této vnitřní paměti je možno přepsat i přímo v aplikaci při 5V napájení a tak modifikovat funkce daného systému při jeho normální činnosti. Typ AT89C51 je tedy funkčně plně kompatibilní s řadou obvodů 8051 s jediným rozdílem a to ve způsobu naprogramování vnitřní paměti typu FLASH EPROM.

2.2 AT89C51

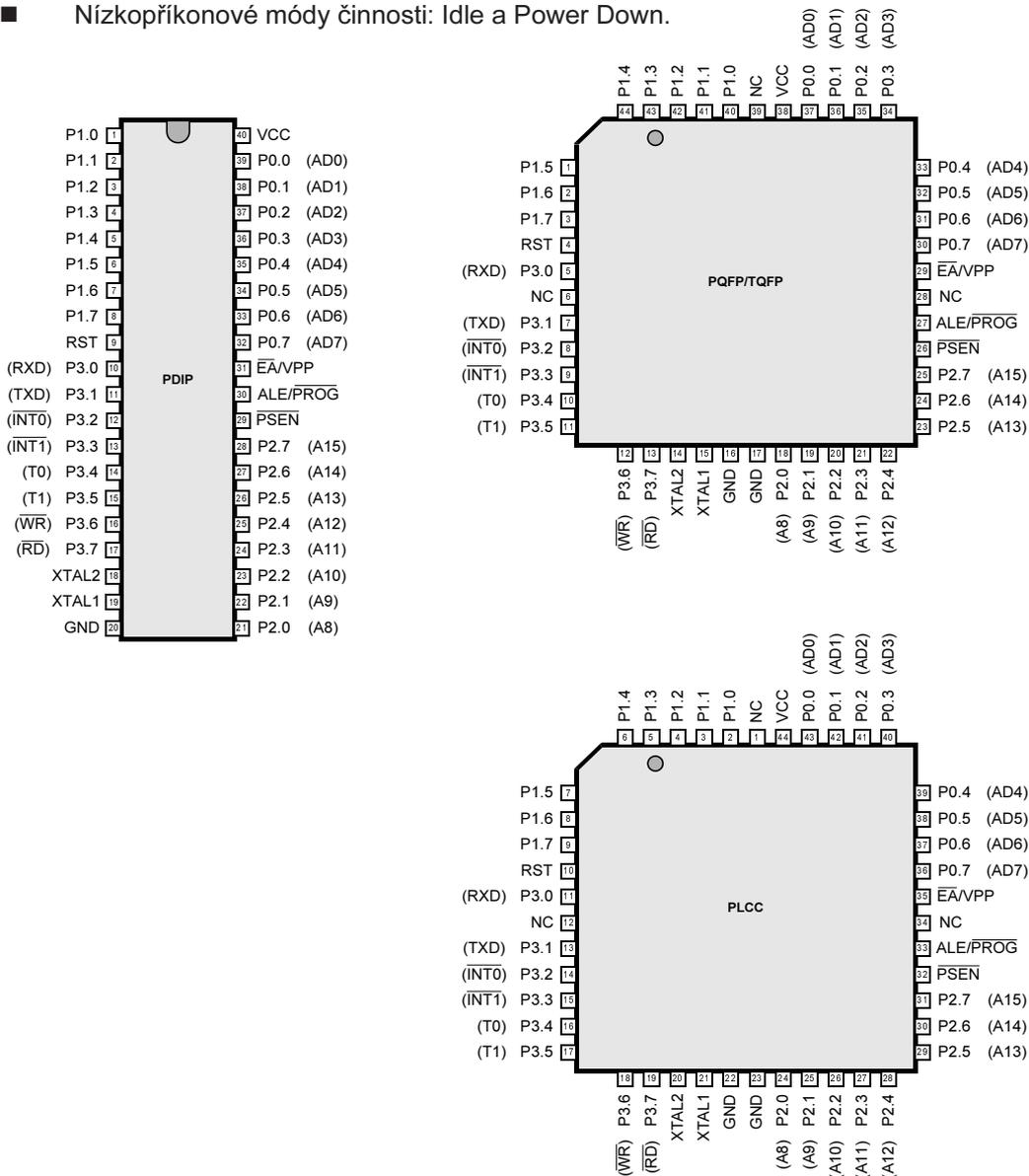
V této části knihy je podrobněji popsán základní typ z řady mikrokontrolérů AT89C51 plně kompatibilní s 8051 firmy INTEL, známým naší široké odborné veřejnosti. Tento mikrokontrolér se vyznačuje nízkým příkonem, vysokým výkonem, 8bitovou architekturou s elektricky přeprogramovatelnou pamětí typu FLASH EPROM s možností její přeprogramování i přímo v aplikaci. Je vyroben na bázi technologie energeticky nezávislých prvků o vysoké hustotě integrace a je kompatibilní s vývodů i s instrukční sadou průmyslového standardu MCS-51™. Integrovaní 8bitového CPU s pamětí FLASH EPROM na jednom čipu dává uživateli výkonný, vysoce flexibilní a levný prvek pro množství aplikací v oblasti řízení, v jednoúčelových zařízeních apod.

POPIS VLASTNOSTÍ

Základní a nejznámější typ mikrokontroléru ATMEL je AT89C51 (*obr. 41*). Je to obvod, který je vyroben na bázi CMOS technologie s těmito základními technickými parametry:

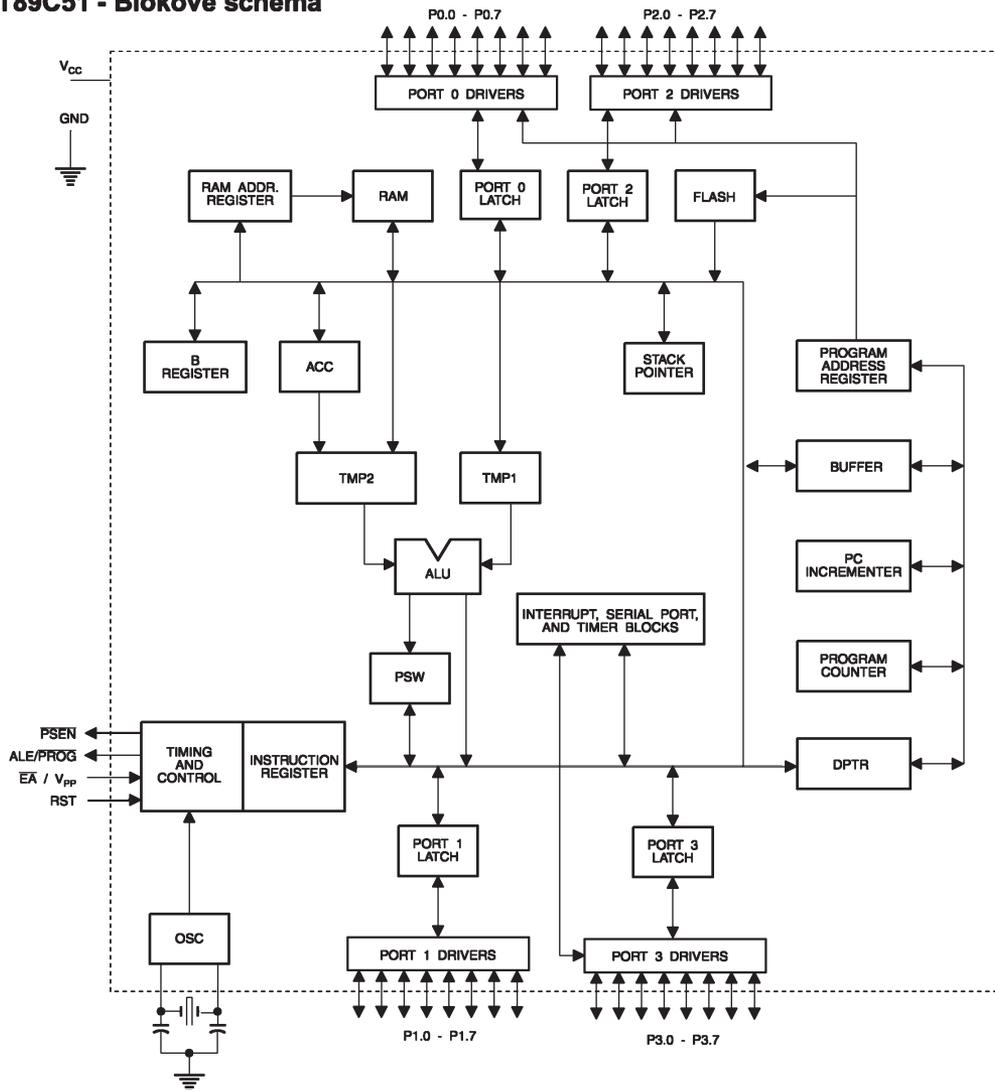
- Plně kompatibilní s produkty MCS-51 standardu.
- 4 kB elektricky přeprogramovatelné paměti typu FLASH EPROM se schopností 1000 zapisovacích/mazacích cyklů.
- Napájecí napětí 2,7 až 6 V.
- Plně statická činnost: 0 Hz až 24 MHz.
- Tříúrovňové uzamknutí paměti programu.
- Interní RAM 128 × 8 bitů.

- 32 programovatelných I/O linek.
- Dva 16bitové časovače/čítače.
- Šest zdrojů přerušení.
- Programovatelný sériový kanál.
- Nízkopříkonové módy činnosti: Idle a Power Down.



Obr. 41 Pouzdření mikrokontroléru AT89C51

AT89C51 - Blokové schéma



Obr. 42 Blokové schéma mikrokontroléru AT89C51

AT89C51 má tyto následující standardní vlastnosti:

Blokové schéma je na obr. 42

Paměť 4 Kbyte FLASH EPROM, 128 byte RAM, 32 I/O linek, dva 16bitové časovače/čítače, pět maskovatelných zdrojů přerušení, plně duplexní sériový kanál, oscilátor a hodinový obvod na čipu. AT89C51 je navržen s plně statickou logikou pro práci od nulového hodinového kmitočtu a má dva softwarově volitelné příkonově úsporné režimy. Režim *Idle* zastavuje CPU, přičemž RAM, časovače/čítače, sériový port a přerušovací systém zůstá-

vají funkční. Režim *Power Down* zachovává obsah paměti RAM, ale zastavuje oscilátor, čímž znepřístupňuje všechny další funkce na čipu až do hardwarového resetu.

Popis vývodů mikrokontroléru:

V_{cc} – napájecí napětí.

GND – zem.

Port 0 – je 8bitový obousměrný I/O port s otevřeným kolektorem. Na každý vývod je možno připojit 8 TTL vstupy. Při zapsání log. 1 na port 0 mohou být vývody využity jako vysokoimpedanční vstupy. Port 0 může být též konfigurován jako multiplexovaná adresové/datová sběrnice nižších významových bitů pro přístup do externí programové resp. datové paměti. Port zároveň přijímá kódy v době programování interní paměti FLASH EPROM a přenáší kódy v čase verifikace programu.

Port 1 – je 8bitový obousměrný I/O port. Výstupy portu mohou budit 4 TTL vstupy. Jsou-li na vývody portu zapsány log. 1, vývody portu jsou drženy vnitřními rezistory na vysoké napěťové úrovni a mohou být použity jako vstupy. Jestliže se na vývod portu externě přivede log. 0, chová se vnitřní zatěžovací rezistor jako zdroj proudu (I_{IL}). Port 1 zároveň přijímá spodní bity adresy (A0 až A7) při programování a verifikaci vnitřní paměti FLASH EPROM.

Port 2 – je 8bitový obousměrný I/O port s vnitřními zatěžovacími rezistory. Výstupy portu 2 jsou zatížitelné 4 TTL vstupy. Když jsou na vývody portu zapsány log. 1, vývody portu jsou drženy vnitřními rezistory na horní napěťové úrovni a mohou být použity jako vstupy. Jestliže se na vývod portu externě přivede log. 0, chová se vnitřní zatěžovací rezistor jako zdroj proudu (I_{IL}). Port 2 vysílá horní adresové bity v době přístupu do externí programové resp. datové paměti v případě 16bitového adresování pomocí instrukce (MOVX @ DPTR). Stav portu 2 je dán obsahem speciálního funkčního registru P2 v případě 8bitové adresace využíváním instrukce (MOVX @ Ri). Naopak, port 2 přijímá vyšší adresové bity a některé řídicí signály v době programování a verifikace interní paměti FLASH EPROM. Port 1 zároveň přijímá horní bity adresy (A8 až A11) při programování a verifikaci vnitřní paměti FLASH EPROM.

Port 3 – je 8bitový obousměrný I/O port s vnitřními zatěžovacími rezistory. Výstupy portu 2 jsou zatížitelné 4 TTL vstupy. Jsou-li na vývody portu zapsány log. 1, vývody portu jsou drženy vnitřními rezistory na horní napěťové úrovni a mohou být použity jako vstupy. Jestliže se na vývod portu externě přivede log. 0, chová se vnitřní zatěžovací rezistor jako zdroj proudu (I_{IL}).

Port 3 se liší obhospodařením více alternativních funkcí, jak je popsáno v tab. 4. Port 3 rovněž přijímá řídicí signály pro programování a verifikaci vnitřní paměti FLASH EPROM.

RST – resetovací vstup. Log. 1 v době dvou strojových cyklů resetuje systém.

ALE/PROG – (Address Latch Enable) umožňuje zápis dolních bitů adresy do záchytného registru v době přístupu do externí paměti. Tento vývod je rovněž vstupem pro programovací pulz (PROG) při programování interní FLASH EPROM paměti. Kmitočet ALE je konstantní a rovný 1/6 kmitočtu oscilátoru a může být použit na externí časování. Avšak pozor, protože jeden pulz ALE je v době přístupu do externí paměti vynechán. Výstup ALE může být blokován nastavením LSB bitu na adrese 8EH v SFR vyjma přístupu na externí paměť (MOVX nebo MOVC).

$\overline{\text{PSEN}}$ – (Program Store Enable) je strobovací signál pro čítání obsahu programové externí paměti. Jestliže je kód přenášen z externí programové paměti, je PSEN aktivován dvakrát v každém strojovém cyklu, mimo dvou aktivací PSEN, které jsou vynechány v čase každého přístupu do externí paměti dat.

$\overline{\text{EA}}$ V_{PP} – vývod EA (External Acces Enable) musí být připojen na zem pro příjem kódu z externí paměti programu v rozsahu 0000H až FFFFH. Jestliže je při programování interní paměti programu FLASH EPROM nastaven uzamykací bit LB1, EA bude interně přepojen na RESET. Pro výkon programu z vnitřní paměti FLASH EPROM se $\overline{\text{EA}}$ připojí na V_{CC} .

Vývod V_{PP} je taky vstupem pro programovací napětí při programování interní paměti programu FLASH EPROM.

XTAL1, XTAL2 – vstupy oscilátoru.

PROGRAMOVÁNÍ INTERNÍ PAMĚTI FLASH EPROM

AT89C51 je normálně nabízen s vymazanou interní paměti programu FLASH EPROM (tj. obsah je FFH) a připravenou na programování. Programovací rozhraní dovoluje napájení s vyšším 12V nebo nízkým 5V napětím. Programování interní paměti napětím 5 V se používá pro programování AT89C51 v uživatelském systému, zatím co 12V programování je kompatibilní s obvyklými programovacími technikami.

Dodávané mikrokontroléry AT89C51 mají režim programování interní paměti programu FLASH EPROM vyznačen na pouzdře.

Pozor při objednávání – trvejte na dodání podle vašich požadavků.

Způsob označování obvodů na vrchní straně pouzdra je popsán v tab. 12.

	$V_{\text{pp}} = 12\text{V}$	$V_{\text{pp}} = 5\text{V}$
Vrchní strana pouzdra	AT89C51 xxxx yyww	AT89C51 xxxx-5 yyww
Signatura	(030H) = 1EH (031H) = 51H (032H) = FFH	(030H) = 1EH (031H) = 51H (032H) = 05H

Tab. 12 Značení pouzder

Kódová část paměti AT89C51 je programována byte po byte v každém programovacím módu. Pro programování libovolného bytu do paměti FLASH EPROM, musí být předtím její celý obsah vymazán.

PROGRAMOVÁNÍ OCHRANNÝCH UZAMYKACÍCH BITŮ

Každý vývojář má zájem chránit své výrobky před okopírováním. Mikroprocesorové aplikace se oproti řešením s klasickými číslicovými obvody vyznačují výrazně jednodušším zapojením a často není problém udělat hardwarovou kopii a zkopírovat program, bez kterého je zařízení nefunkční. Proto mají obvody ATMEL zabudovanou několikastupňovou ochranu proti okopírování obsahu interní paměti FLASH EPROM. Na čipu jsou tři uzamy-

kací bity (LB1, LB2, LB3), které nemusí (N), ale mohou (P) být naprogramovány pro získání dalších níže popsaných vlastností. Je-li naprogramován bit LB1, je logická úroveň na vývodu \overline{EA} získána a zapamatována v době resetu. Jestliže je mikrokontrolér připojen na napájecí napětí bez resetování, vnitřní paměť se inicializuje do nahodilého stavu a drží si ho pokud není aktivován RESET. Je nutné, aby uložená hodnota \overline{EA} byla v souladu s aktuální logickou úrovní tohoto vývodu pro správnou činnost procesoru.

	Uzamykací bity			Typ ochrany
	LB1	LB2	LB3	
1	N	N	N	Bez ochrany uzamčením
2	P	N	N	Instrukce MOVC vykonávané z externí paměti programu nad obsahem interní FLASH jsou blokovány. EA je vzorkováno a nastaveno na Reset, a další programování FLASH je blokováno.
3	P	P	N	Tak jako mód 2, je blokována i verifikace
4	P	P	P	Tak jako mód 3, je blokována i činnost z externí paměti programu

Tab. 13 Režimy ochrany mikrokontrolérů

ALGORITMUS PROGRAMOVÁNÍ INTERNÍ PAMĚTI FLASH EPROM AT89C51

Před programováním AT89C51 musí být připojeny adresové a datové signály, stejně jako řídicí signály podle zapojení na obr. 43.

Jednotlivé režimy programování interní paměti programu FLASH EPROM jsou zřejmé z tab. 14.

Je nutné dodržet následující posloupnost kroků:

1. Vložit na adresové linky (P1 a P2.0 až P2.3) požadovanou adresu paměti.
2. Vložit datový byte na datové linky (P0).
3. Přivést správnou kombinaci řídicích signálů (P2.6, P2.7, P3.6 a P3.7).
4. Zvednout \overline{EA} /Vpp na 12 V pro „vysokonapěťový“ programovací režim.

Mód	RST	\overline{PSEN}	$\overline{ALE}/$ PROG	\overline{EA} /Vpp	P2.6	P2.7	P3.6	P3.7
Zápis dat	H	L	←	H/12V	L	H	H	H
Čítání dat	H	L	H	H	L	L	H	H
Zápis do zámku	Bit-1	H	←	H/12V	H	H	H	H
	Bit-2	H	←	H/12V	H	H	L	L
	Bit-3	H	L	←	H/12V	H	L	L
Mazání čipu	H	L	← (1)	H/12V	H	L	L	L
Čítání Bytu SIGNATURA	H	L	H	H	L	L	L	L

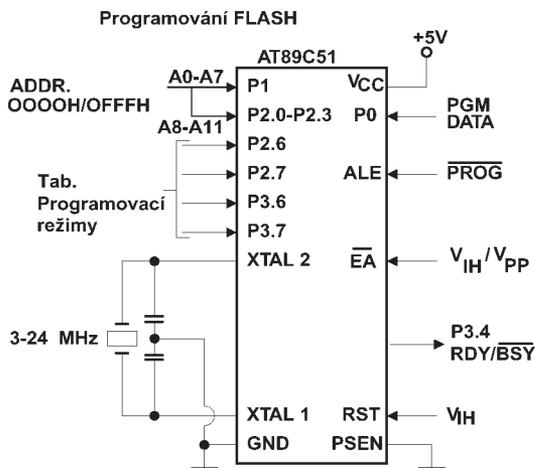
POZNÁMKY: ¹ Mazání čipu vyžaduje PROG impuls s délkou 10 ms.

Tab. 14 Režimy programování AT89C51

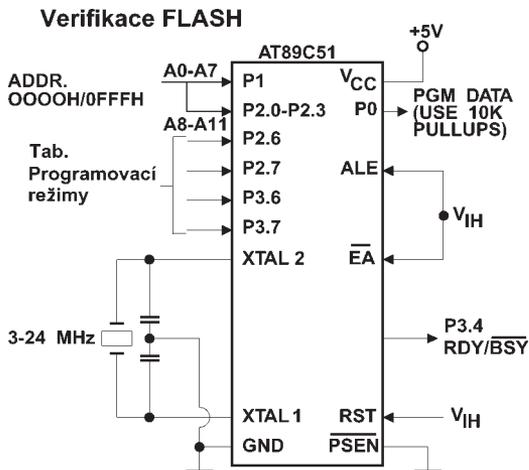
5. Injektovat impuls do ALE/ $\overline{\text{PROG}}$ pro naprogramování datového bytu nebo Lock bitu do FLASH EPROM paměti. Cyklus zápisu bytu je časován vnitřně a trvá typicky 1,5 ms.

Opakujte kroky 1 až 5 se změnou adres a dat pro požadovanou oblast paměti, resp. po konec programovaného datového souboru.

Data Polling AT89C51 je vlastnost, jenž se vyznačuje výzvou na indikaci konce zápisového cyklu. V průběhu výkonu zápisového cyklu bude na P0.7 vývodu komplementární doplněk. Poté co byl zápisový cyklus ukončen, jsou správná data na všech výstupech a může začít další zápisový cyklus. Testováním této výzvy je možno optimalizovat programování paměti FLASH EPROM.



Obr. 43 Schéma programování AT89C51



Obr. 44 Schéma verifikace AT89C51

Ready/ $\overline{\text{BSY}}$ – Postup programování jednotlivého bytu může být monitorován také prostřednictvím výstupního signálu $\text{RDY}/\overline{\text{BSY}}$. Vývod P3.4 tj. $\text{RDY}/\overline{\text{BSY}}$ je po přivedení signálu $\overline{\text{PROG}}$ stažen dolů a v průběhu programování indikuje stav **BUSY** – obsazen. Po ukončení programování se P3.4 $\text{RDY}/\overline{\text{BSY}}$ vrací zpět nahoru a indikuje stav **READY** – připraven. Tento **BUSY** stav trvá typicky 2 ms a je rozhodujícím pro rychlost programování.

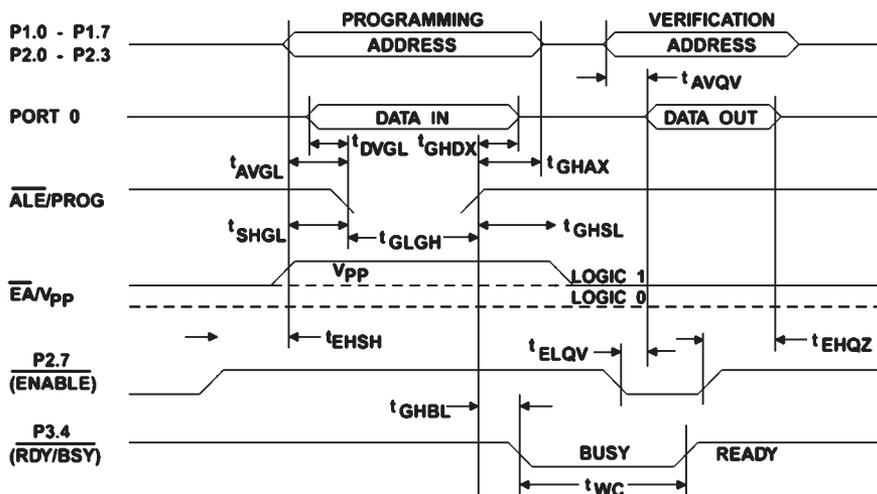
Program Verify (Verifikace programu) – obr. 44. Jestliže nebyly naprogramovány uzamykací bity, nebo byli naprogramováni pouze uzamykací bit LB1, mohou být naprogramována data opětovně čtena. Nastavení uzamykacích bitů LB1-LB3 nelze zjistit přímo. Zjistí se to explicitně tím, že jsou blokovány příslušné funkce.

Chip Erase (Mazání čipu). Celý obsah paměti je vymazán elektricky využitím správné kombinace řídicích signálů a podržením $\text{ALE}/\overline{\text{PROG}}$ na úrovni log. 0 po dobu 10 ms. Do všech paměťových buněk jsou zapsány logické jedničky. Před libovolným dalším přeprogramováním paměti musí být vykonáno mazání.

Reading the Signature Bytes (Čtení značkových Bytů). Značkové byty se čtou tím samým způsobem jako při normální verifikaci paměti na adresách 30H, 31H a 32H s tím, že vývody P3.6 a P3.7 musí být nastaveny na úroveň log. 0. Čtené hodnoty jsou:

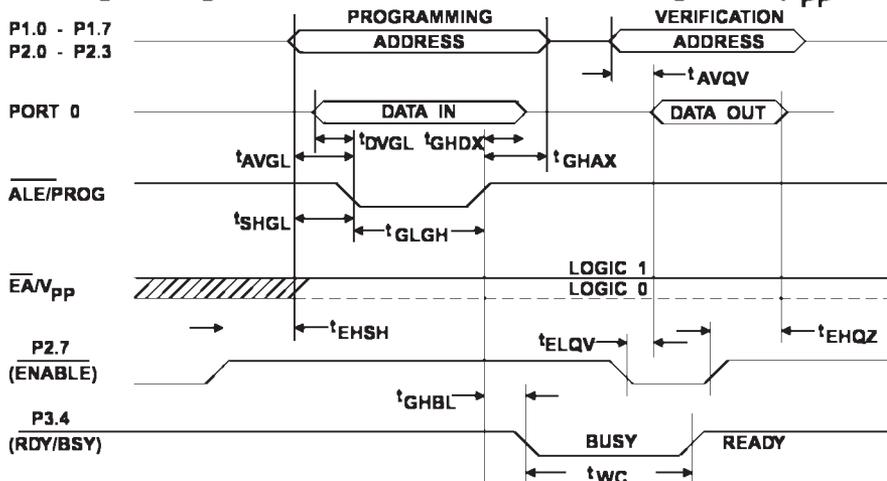
- (30H) = 1EH indikuje výrobce ATMEL;
- (31H) = 51H indikuje 89C51;
- (32H) = FFH indikuje 12 V programování;
- (32H) = 05H indikuje 5V programování.

Flash Programming and Verification Waveforms - High Voltage Mode ($V_{pp}=12V$)



Obr. 45 Průběhy signálů při programování AT89C51 (12 V)

Flash Programming and Verification Waveforms - Low Voltage Mode ($V_{PP} = 5V$)



Obr. 46 Průběhy signálů při programování AT89C51 (5 V)

Symbol	Parametr	Min	Max	Jednotka
$V_{PP}^{(1)}$	Programovací napětí	11.5	12.5	V
$I_{PP}^{(1)}$	Programovací proud		1.0	mA
$1/t_{CLCL}$	Kmitočet oscilátoru	3	24	MHz
t_{AVGL}	Předstih adresy před PROG	$48t_{CLCL}$		
t_{GHAX}	Držení adresy po PROG	$48t_{CLCL}$		
t_{DVGL}	Předstih dat před PROG	$48t_{CLCL}$		
t_{GHDX}	Držení dat po PROG	$48t_{CLCL}$		
t_{EHSH}	Předstih P2.7 (\overline{ENABLE}) před V_{PP}	$48t_{CLCL}$		
t_{SHGL}	Předstih V_{PP} před PROG	10		μs
t_{GHSL}	Držení V_{PP} po PROG	10		μs
t_{GLGH}	Trvání PROG	1	110	μs
t_{AVQV}	Platnost adresy před daty		$48t_{CLCL}$	
t_{ELQV}	Zpoždění dat po \overline{ENABLE}		$48t_{CLCL}$	
t_{EHQZ}	Platnost dat po \overline{ENABLE}	0	$48t_{CLCL}$	
t_{GHBL}	Zpoždění BUSY po PROG		1.0	μs
t_{WC}	Trvání BUSY – čas zápisu Bytu		2.0	ms

Poznámka: ¹ Použito jen při 12V programovacím módu.

Tab. 15 Parametry programování AT89C51
($T_A = 0^\circ C$ až $70^\circ C$, $V_{CC} = 5,0 V \pm 10\%$)

Rozsah pracovních teplot	$-55^\circ C$ až $+125^\circ C$
Teplota pro skladování	$-65^\circ C$ až $+150^\circ C$
Napětí na libovolném vývodu	$-1,0 V$ až $+7,0 V$ (vůči zemi)
Maximální pracovní napětí	6,6 V
Maximální výstupní proud	15,0 mA

Tab. 16 Mezní parametry AT89C51

PARAMETRY A CHARAKTERISTIKY

UPOZORNĚNÍ: Překročení uvedených mezních hodnot může způsobit trvalé zničení. Obdobně, činnost prvku po delší dobu na úrovni mezních hodnot může mít za následek snížení spolehlivosti.

Symbol	Parametr	Podmínky	Min	Max	Jednotka
V_{IL}	Vstupní napětí LOW	(bez EA)	-0.5	$0.2 V_{CC} - 0.1$	V
V_{IL1}	Vstupní napětí LOW (\overline{EA})		-0.5	$0.2 V_{CC} - 0.3$	V
V_{IH}	Vstupní napětí HIGH	(bez XTAL1, RST)	$0.2 V_{CC} + 0.9$	$V_{CC} + 0.5$	V
V_{IH1}	Vstupní napětí HIGH	(XTAL1, RST)	$0.7 V_{CC}$	$V_{CC} + 0.5$	
V_{OL}	Výstupní napětí LOW ⁽¹⁾ (Porty 1, 2, 3)	$I_{OL} = 1.6 \text{ mA}$		0.45	V
V_{OL1}	Výstupní napětí LOW ⁽¹⁾ (Port 0, ALE, PSEN)	$I_{OL} = 3.2 \text{ mA}$		0.45	v
V_{OH}	Výstupní napětí high ⁽¹⁾ (Porty 1, 2, 3, ALE, PSEN)	$I_{OH} = -60 \mu\text{A}$, $V_{CC} = 5 \text{ V} \pm 10 \%$	2.4		V
		$I_{OH} = -25 \mu\text{A}$	$0.75 V_{CC}$		V
		$I_{OH} = -10 \mu\text{A}$	$0.9 V_{CC}$		V
V_{OH1}	Výstupní napětí HIGH (Port 0 v externím BUS módu)	$I_{OH} = -800 \mu\text{A}$ $V_{CC} = 5 \text{ V} \pm 10 \%$	2.4		V
		$I_{OH} = -300 \mu\text{A}$	$0.75 V_{CC}$		V
		$I_{OH} = -80 \mu\text{A}$	$0.9 V_{CC}$		V
I_{IL}	Vstupní proud log. 0 (Porty 1, 2, 3)	$V_{IN} = 0.45 \text{ V}$		-50	μA
I_{TL}	Proud přechodu log. 1 na log. 0 (Porty 1,2,3)	$V_{IN} = 2 \text{ V}$ $V_{CC} = 5 \text{ V} \pm 10 \%$		-650	μA
I_{LI}	Vstupní svodový proud (Port 0, EA)	$0.45 < V_{IN} < V_{CC}$		$\pm 10 \%$	μA
RRST	Interní zatěžovací rezistor pro RESET		50	300	K Ω
C_{IO}	Kapacita vývodu	Test. kmit. = 1MHz	$T_A = 25 \text{ }^\circ\text{C}$	10	pF
I_{CC}	Napájecí proud	Aktivní mód, 12MHz		20	mA
		Idle mód, 12MHz		5	mA
	Režim sníženého příkonu ⁽²⁾	$V_{CC} = 6 \text{ V}$		100	μA
		$V_{CC} = 3 \text{ V}$		40	μA

POZNÁMKY:

1. Proud I_{OL} musí být externě limitován následovně:

Maximální I_{OL} na každý výstup portu:		10 mA
Maximální I_{OL} na celý port:	Port 0:	26 mA
	Port 1, 2, 3:	15 mA
Sumární maximální proud I_{OL} pro všechny výstupy:		71 mA

2. Minimální napájecí napětí V_{CC} pro režim sníženého příkonu (Power Down) je 2 V.

Tab. 17 *Statické vlastnosti AT89C51*
 $T_A = -40 \text{ }^\circ\text{C}$ až $85 \text{ }^\circ\text{C}$, $V_{CC} = 5,0 \text{ V} \pm 20 \%$

Symbol	Parametr	12MHz oscilátor		16 až 24MHz oscilátor		Jednotka
		Min	Max	Min	Max	
t/t_{CLCL}	Kmitočet oscilátoru			0	24	MHz
t_{LHLL}	Trvání impulsu ALE	127		$2t_{CLCL}-40$		ns
t_{AVLL}	Předstih adresy před ALE	43		$t_{CLCL}-13$		ns
t_{LLAX}	Držení adresy po ALE	48		$t_{CLCL}-20$		ns
t_{LLIV}	Předstih ALE před platností instrukce		233		$4 t_{CLCL}-65$	ns
t_{LLPL}	Předstih ALE před \overline{PSEN}	43		$t_{CLCL}-13$		ns
t_{PLPH}	Trvání pulzu \overline{PSEN}	205		$3t_{CLCL}-20$		ns
t_{PLIV}	Předstih \overline{PSEN} před platností instrukce		145		$3t_{CLCL}-45$	ns
t_{PXIX}	Držení instrukce před \overline{PSEN}	0		0		ns
t_{PXIZ}	Přesah instrukce po \overline{PSEN}		59		$t_{CLCL}-10$	ns
t_{PXAV}	Zpoždění platné adresy po \overline{PSEN}	75		$t_{CLCL}-8$		ns
t_{AVIV}	Odstup adresy a instrukce		312		$5t_{CLCL}-55$	ns
t_{PLAZ}	Přesah adresy po \overline{PSEN}		10		10	ns
t_{RLRH}	Trvání RD	400		$6t_{CLCL}-100$		ns
t_{WLWH}	Trvání \overline{WR}	400		$6t_{CLCL}-100$		ns
t_{RLDV}	Předstih RD před platností dat		252		$5t_{CLCL}-90$	ns
t_{RHDX}	Držení dat po RD	0		0		ns
t_{RHDZ}	Přesah dat po RD		97		$2t_{CLCL}-28$	ns
t_{LLDV}	Předstih ALE před platností dat		517		$8t_{CLCL}-150$	ns
t_{AVDV}	Odstup adresy a dat		585		$9t_{CLCL}-165$	ns
t_{LLWL}	Předstih ALE před RD nebo \overline{WR}	200	300	$3t_{CLCL}-50$	$3t_{CLCL}+50$	ns
t_{AVWL}	Předstih adresy před RD nebo \overline{WR}	203		$4t_{CLCL}-75$		ns
t_{QVWX}	Platnost dat před \overline{WR}	23		$t_{CLCL}-20$		ns
t_{QVWH}	Platnost dat před koncem \overline{WR}	433		$7t_{CLCL}-120$		ns
t_{WHQX}	Držení dat po \overline{WR}	33		$t_{CLCL}-20$		ns
t_{RLAZ}	Přesah adresy po RD		0		0	ns
t_{WHLH}	Zpoždění ALE po RD nebo \overline{WR}	43	123	$t_{CLCL}-20$	$t_{CLCL}+25$	ns

Tab. 18 Dynamické vlastnosti AT89C51

AT89C51 je vývodově i instrukčně kompatibilní s řadou obvodů 8051. Jak bylo zmíněno dříve, mají tyto obvody nízkou spotřebu, napájecí napětí již od 2,7 V, plně statickou činnost, dva nízkopříkonové režimy např. pro aplikace s bateriovým napájením a možnost jednoduše a rychle změnit obsah paměti programu nejenom pomocí programátoru, ale i přímo v aplikaci při běžném 5V napájení. To jsou výborné vlastnosti, které tento mikrokontrolér předurčují jako vhodný stavební prvek do mnoha elektronických aplikací. Navíc bez problémů je možno tímto obvodem nahradit klasické mikrokontroléry 8751, 8051 i 8031. Vyrábějí se v pouzdech typu TQFP, PLCC a klasickém 40pinovém PDIP. Z hlediska pracovního rozsahu teplot jsou dodávány jak je uvedeno v tab. 19.

Kmitočet [MHz]	Napájecí napětí [V]	Značení	Pouzdro				Rozsah prac. teplot [°C]
			TQFP	PLCC	PDIP	PQFP	
12	4 až 6	AT89C51-12	AC	JC	PC	QC	0 až 70
12	4 až 6	AT89C51-12	AI	JI	PI	QI	-40 až 85
12	4 až 6	AT89C51-12	AA	JA	PA	QA	-40 až 105
16	4 až 6	AT89C51-16	AC	JC	PC	QC	0 až 70
16	4 až 6	AT89C51-16	AI	JI	PI	QI	-40 až 85
16	4 až 6	AT89C51-16	AA	JA	PA	QA	-40 až 105
20	4 až 6	AT89C51-20	AC	JC	PC	QC	0 až 70
20	4 až 6	AT89C51-20	AI	JI	PI	QI	-40 až 85
24	4 až 6	AT89C51-24	AC	JC	PC	QC	0 až 70
24	4 až 6	AT89C51-24	AI	JI	PI	QI	-40 až 85

Tab. 19 Značení pouzder AT89C51

2.3 MIKROKONTROLÉRY AT89C1051, AT89C2051, AT89C4051

Jako podmožinu a velice zajímavou skupinu mikrokontrolérů této řady je nutno uvést obvody AT89Cx051. Ty jsou sice méně výkonné než obvody AT89C5x, z hlediska programování jsou však plně kompatibilní s instrukční sadou průmyslového standardu MCS-51, a s ohledem na to, že jsou vyráběny v menších pouzdech tj. PDIP – 20pinovém (obr. 47), mají méně I/O linek a ještě několik malých odlišností.

Tyto mikrokontroléry mají dle typu (1051, 2051, 4051) následující vlastnosti:

Paměť programu (1, 2, 4 Kbyte FLASH EPROM), RAM (64, 128, 256 byte), 15 I/O linek, dva 16bitové časovače/čítače, šest zdrojů přerušení, plně duplexní sériový kanál, vestavěný analogový komparátor, přímé buzení LED (P1, P3), oscilátor a hodinový obvod na čipu. AT89C1051, 2051, 4051 jsou navrženy se statickou logikou pro práci od nulového hodinového kmitočtu do 24 MHz a mají dva softwarově volitelné úsporné módy příkonu IDLE, PD.

Nejjednodušším z nich je **AT89C1051**. Pro zajímavost jsou uvedeny rovněž jeho základní technické parametry:

- Plně kompatibilní se standardem MCS-51 produkty.
- 1 kB elektricky přeprogramovatelné paměti typu FLASH EPROM se schopností až 1000 zápisových/mazacích cyklů.
- Napájecí napětí 2,7 až 6 V.
- Plně statická činnost: 0 Hz až 24 MHz.
- Dvojúrovňové uzamknutí paměti programu.
- Interní RAM 64 × 8 bitů.

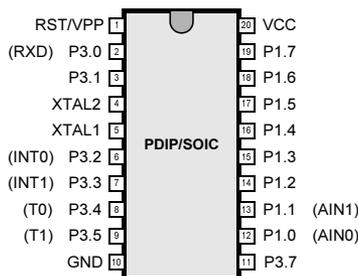
- 15 programovatelných I/O linek.
- Dva 16bitové časovače/čítače.
- Šest zdrojů přerušení.
- Programovatelný sériový kanál UART.
- Schopnost přímo budít LED.
- Analogový komparátor.
- Nízkopříkonové módy činnosti: Idle a Power Down.

AT89C1051, s ohledem na jeho malé rozměry, je to výkonný mikrokontrolér umožňující vysoce flexibilní, efektivní a levné řešení v mnoha aplikacích v průmyslu. Mimo uvedených parametrů má dvouúrovňovou architekturu přerušení a nejzajímavější část – precizní analogový komparátor. Rovněž má na čipu oscilátor a hodinový obvod. Pro co nejnižší spotřebu je konstruován se statickou logikou, čímž umožňuje snížit hodinový kmitočet až na nulu a podporuje tak jako všechny tyto mikrokontroléry softwarově dva nízkopříkonové módy (IDLE, PD). Idle mód zastavuje CPU, zatímco například RAM, čítače, sériový port a systém přerušení pracuje dále. Power Down mód zachovává obsah paměti RAM, ale zastaví oscilátor, čímž vypíná všechny čipové funkce až do následujícího hardwarového resetu.

Druhý zajímavý typ z těchto menších mikrokontrolérů je AT89C2051, který má větší paměť programu – 2 kB FLASH EPROM a paměť dat 128 × 8 bitů RAM. Pracuje v rozsahu

Kmitočet [MHz]	Napájecí napětí [V]	Značení	Pouzdro		Rozsah prac. teplot [°C]
			PDIP	SOIC	
12	3 až 6	AT89C1051-12	PC	SC	0 až 70
12	3 až 6	AT89C1051-12	PI	SI	-40 až 85
24	4 až 6	AT89C1051-24	PC	SC	0 až 70
24	4 až 6	AT89C1051-24	PI	SI	-40 až 85
12	2,7 až 6	AT89C2051-12	PC	SC	0 až 70
12	2,7 až 6	AT89C2051-12	PI	SI	-40 až 85
12	2,7 až 6	AT89C2051-12	PA	SA	-40 až 105
24	4 až 6	AT89C2051-24	PC	SC	0 až 70
24	4 až 6	AT89C2051-24	PI	SI	-40 až 85
12	3 až 6	AT89C4051-12	PC	SC	0 až 70
12	3 až 6	AT89C4051-12	PI	SI	-40 až 85
12	3 až 6	AT89C4051-12	PA	SA	-40 až 105
24	4 až 6	AT89C4051-24	PC	SC	0 až 70
24	4 až 6	AT89C4051-24	PI	SI	-40 až 85

Tab. 20 Značení pouzder AT89C1051, AT89C2051, AT89C4051



Obr. 47 Pouzdrění mikrokontrolérů AT89C1051, AT89C2051, AT89C4051

napájecích napětí od 2,7 do 6 V. Podle pracovního rozsahu teplot a pouzdrění jsou tyto mikrokontroléry vyráběny v modifikacích uvedených v tab. 20.

Posledním z těchto malých mikrokontrolérů je typ AT89C4051. Podobně jako předcházející typy je vyráběn v malém 20pinovém PDIP pouzdrě a také v pouzdrě SOIC. Má větší paměť programu – 4 kB. Pracuje v rozsahu napájecích napětí 3 až 6 V.

Popis vývodů mikrokontroléru:

Vcc – napájecí napětí.

GND – zem.

Port 1 – je 8bitový obousměrný I/O port s otevřeným kolektorem. Piny P1.2 až P1.7 jsou ošetřeny interními rezistory. Vývody P1.0 až P1.1 vyžadují externí zatěžovací rezistory. Vývody P1.0 až P1.1 jsou současně analogové vstupy komparátoru AIN0 (pozitivní vstup) a AIN1 (negativní vstup). Výstupy portu mají zatížitelnost 20 mA a mohou přímo budit např. LED displej. Při zápisu log. 1 na vývody portu 1 je port ve stavu vstupním. Jsou-li vývody P1.2 až P1.7 konfigurovány jako vstupní a drženy zvenčí na log. 0 je zatěžovací proud (I_{L}) určen interními rezistory.

Port 1 se využívá na přenos dat při programování a verifikaci interní FLASH EPROM paměti. Tento port rovněž přijímá kódy v době programování interní paměti FLASH EPROM a přenáší kódy v čase verifikace programu.

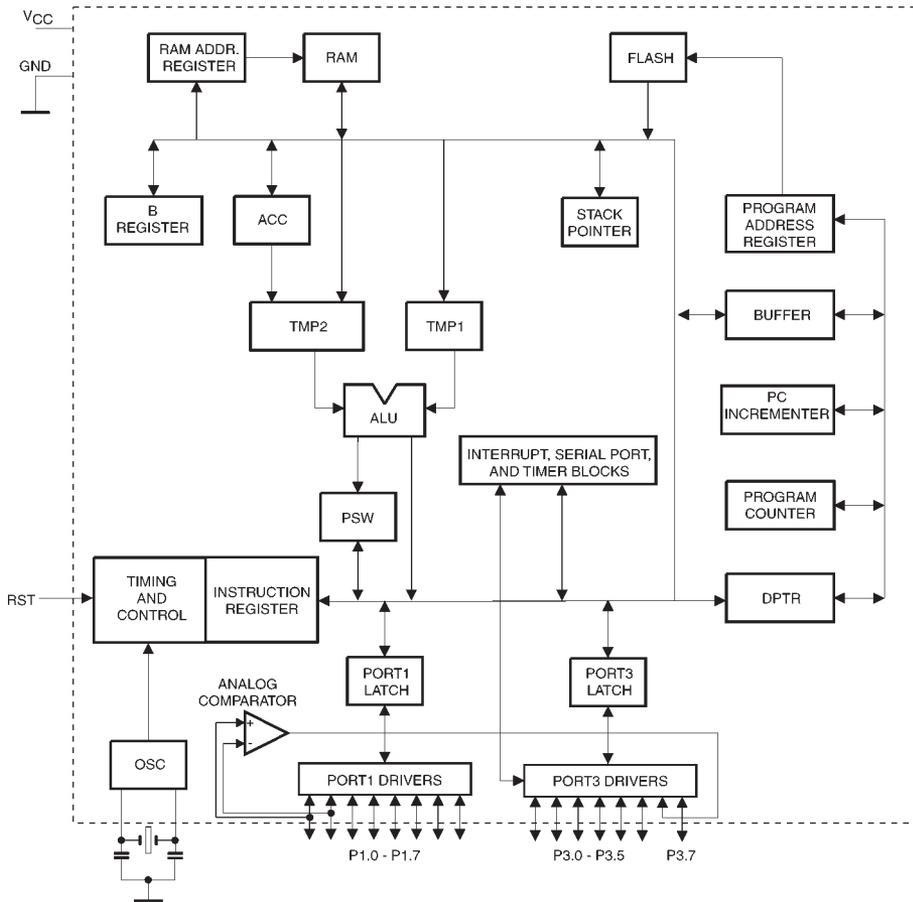
Port 3 – je 7bitový (P3.0 až P3.5 a P3.7) obousměrný I/O port s vnitřními zatěžovacími rezistory. Vývod P3.6 je využit interně, je totiž připojen k výstupu vestavěného analogového komparátoru a není přístupný jako běžná I/O linka. Výstupy portu jsou zatížitelné 20 mA a mohou přímo budit např. LED displej. Při zápisu log. 1 na vývody portu 1 je port ve stavu vstupním. Když jsou vývody P1.2 až P1.7 vstupní a drženy zvenčí na log. 0 bude zatěžovací proud (I_{L}) určen interními rezistory. Port 3 má ještě zabudovány alternativní funkce jako je to u typu AT89C51.

Port 3 navíc řídí režimy činnosti v době programování a verifikace interní paměti FLASH EPROM.

RST – resetovací vstup. Log. 1 po dobu dvou strojových cyklů resetuje systém.

XTAL1, XTAL2 – vstup oscilátoru.

Blokové schéma je na obr. 48.



Obr. 48 Blokové schéma mikrokontrolérů AT89C1051, AT89C2051, AT89C4051

Mikrokontroléry AT89C1051, AT89C2051, AT89C4051 mají oblast SFR pokrytou s omezením zřejmým podle mapy SFR na obr. 49.

Navzdory tomu, že tyto mikrokontroléry jsou levné a vysoce efektivní obvody, je nutno si uvědomit, že nestačí akceptovat hardwarové omezení dané strukturou a pouzdřením, ale uvědomit si i omezení softwarové.

1. Instrukce nepodmíněných skoků a volání podprogramů [LCALL; LJMP; ACALL; AJMP; SJMP; JMP @A+DPTR] jsou možné jenom v oblasti platné interní paměti programu (pro AT89C1051 – 000H-3FFH, pro AT89C2051 000H – 7FFH, pro AT89C4051 – 000H – FFFH)
2. Instrukce podmíněných skoků [CJNE; DJNZ; JB; JNB; JC; JNC; JBC; JZ; JNZ] mají stejné omezení jako v bodě 1.
3. Při práci v RAM paměti je třeba si uvědomit omezení dané její velikostí (64, 128 bytů).

4. Přístup do externí paměti programu není možný.
5. Přístup do externí paměti dat (instrukce MOVX) není možný. V případě potřeby je nutné volit konstrukci s externí pamětí dat na bázi sériových pamětí např. EEPROM.

0F8H								0FFH
0F0H	B 00000000							0F7H
0E8H								0EFH
0E0H	ACC 00000000							0E7H
0D8H								0DFH
0D0H	PSW 00000000							0D7H
0C8H								0CFH
0C0H								0C7H
0B8H	IP 00000000							0BFH
0B0H	P3 11111111							0B7H
0A8H	IE 00000000							0AFH
0A0H								0A7H
98H	SCON 00000000	SBUF XXXXXXXX						9FH
90H	P1 11111111							97H
88H	TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000		8FH
80H		SP 00000111	DPL 00000000	DPH 00000000			PCON 00000000	87H

Obr. 49 Mapa SFR AT89C1051, AT89C2051, AT89C4051

ALGORITMUS PROGRAMOVÁNÍ INTERNÍ PAMĚTI FLASH EPROM AT89C1051

Mikrokontrolér AT89C1051 je dodáván s 1k bytovou EPROM pamětí na čipu ve vymazaném stavu tj. připravenou k naprogramování. Obsahuje vnitřní adresový čítač, který je vždy resetován na 000H vzestupnou hranou na vstupu RST.

Postup při programování je následující:

1. Před programováním AT89C1051 musí být připojené adresové a datové bity, jako i řídicí signály podle zapojení na obr. 50.
2. Připojí se napájecí napětí.
3. RST a XTAL1 se připojí na zem.
4. RST nastavit na úroveň „H“.

5. Vývod 3.2 nastavit na úroveň „H“.
6. Použít příslušnou kombinaci logických úrovní na vývody P3.3, P3.4, P3.5, P3.7 pro výběr jednoho z programovacích režimů znázorněných v tabulce programování (tab. 21).
7. Nastavit kombinaci dat na P1.0 až P1.7 pro adresu 000H.
8. Na RST přivést 12 V pro aktivaci programování.
9. Na P3.2 přivést záporný impuls pro naprogramování obsahu paměti, resp. uzamykacích bitů. Zápisový cyklus je generován vnitřně.
10. Pro verifikaci naprogramovaných dat snížíme napětí na RST z 12 V na log. 1 a řídicí vývody P3.3, až P3.7 nastavíme na příslušné úrovně podle tab. 21. Výstupní data můžou být čteny na vývodech portu P1.
11. Pro naprogramování dat na následující adresu přivedeme znova impuls na vývod XTAL1 pro inkrementaci na vnitřním čítači adresy. Nastavíme nové data na vývody portu P1.
12. Opakujeme kroky 8 až 11 až po naprogramování celého obsahu vnitřní paměti.
13. Při vypnutí se nastaví vývod XTAL1 na log. 0 a RST na log. 0. Pak se odpojí napájecí napětí.

Data Polling AT89Cx051 je vlastnost, jenž se vyznačuje výzvou na indikaci konce zápisového cyklu. V průběhu výkonu zápisového cyklu bude na vývodu P1.7 komplementární doplněk. Poté co byl zápisový cyklus ukončen, jsou správná data na všech výstupech a může začít další zápisový cyklus. Testováním této výzvy je možno optimalizovat programování paměti FLASH EPROM.

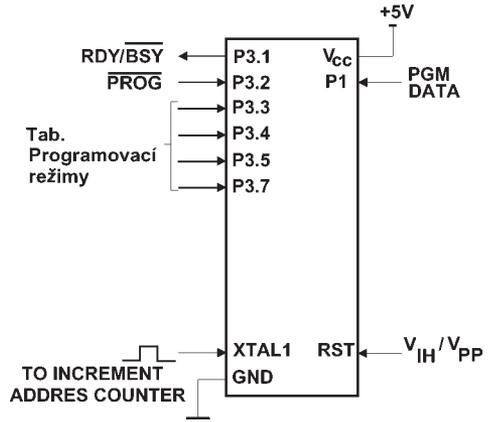
Ready/Busy : Postup programování jednotlivého bytu může být monitorován taky prostřednictvím výstupního signálu RDY/BSY. Vývod P3.1, tj. RDY/BSY, je po přivedení signálu PROG stažen dolů a v průběhu programování indikuje stav BUSY – obsazen. Po ukončení programování se P3.4 RDY/BSY vrací zpět nahoru a indikuje stav READY – připraven. Tento BUSY stav trvá typicky 2 ms a je rozhodujícím pro rychlost programování.

Program Verify (Verifikace programu) – obr. 51. Jestliže nebyly naprogramovány uzamykací bity, nebo byl-li naprogramován pouze uzamykací bit LB1, mohou být naprogramovaná data opětovně čtena.

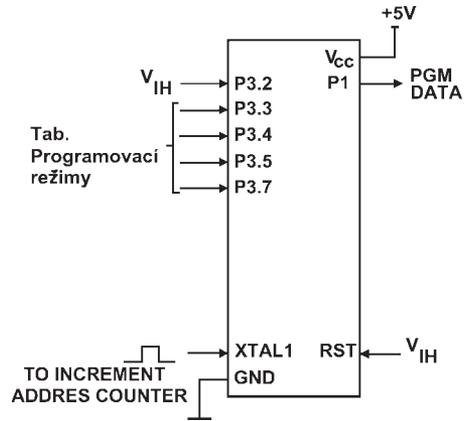
1. Resetovat interní adresový čítač na 000H změnou RST z log. 0 na log. 1.
2. Nastavit vhodnou kombinaci řídicích signálů pro čtení dat.
3. Číst data na vývodech portu P1.
4. Přivést impuls na vstup XTAL1 a opakovat bod 3. a 4.

Nastavení uzamykacích bitů LB1, LB2 nelze zjistit přímo. Zjistí se to explicitně tím, že jsou blokovány příslušné funkce.

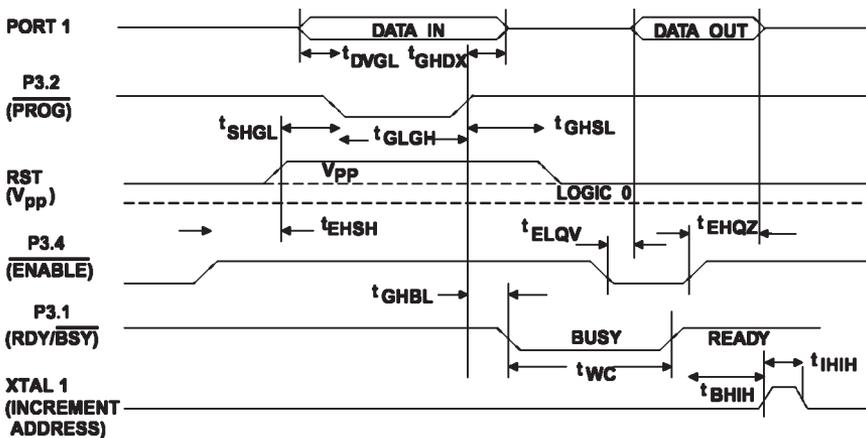
Chip Erase (Mazání čipu). Celý obsah paměti je vymazán elektricky využitím správné kombinace řídicích signálů a podržením PROG P3.2 na úrovni log. 0 po dobu 10 ms. Do všech paměťových buněk jsou zapsány logické jedničky. Před libovolným dalším přeprogramováním paměti musí být vykonáno mazání.



Obr. 50 Schéma programování
AT89C1051, AT89C2051, AT89C4051



Obr. 51 Schéma verifikace
AT89C1051, AT89C2051, AT89C4051



Obr. 52 Průběhy programování a verifikace AT89C1051, AT89C2051, AT89C4051

Reading the Signature Bytes (Čtení značkových Bytů). Značkové byty se čtou tím samým způsobem jako při normální verifikaci paměti na adresách 00H, 01H a 02H s tím, že vývody P3.5 a P3.7 musí být nastaveny na úroveň log. 0. Čtené hodnoty jsou:

(00H) = 1EH indikuje výrobce ATMEL,
 (01H) = 11H indikuje 89C1051,
 21H indikuje 89C2051,
 41H indikuje 89C4051.

Jednotlivé režimy programování interní paměti programu FLASH EPROM jsou zřejmé z tab. 21.

Mode	RST/V _{pp}	P3.2/ PROG	P3.3	P3.4	P3.5	P3.7
Zápis dat	12V	←	L	H	H	H
Čtení dat	H	H	L	L	H	H
Zápis do zámku	Bit-1	12V	←	H	H	H
	Bit-2	12V	←	H	H	L
Mazání čipu	12V	←	H	L	L	L
Čtení Byte SIGNATURA	H	H	L	L	L	L

Tab. 21 Režimy programování AT89C1051, AT89C2051, AT89C4051

Symbol	Parametr	Min	Max	Jednotka
V _{PP}	Programovací napětí	11.5	12.5	V
I _{PP}	Programovací proud		250	μA
t _{DVGL}	Předstih dat před $\overline{\text{PROG}}$	1.0		μs
t _{GHDX}	Držení dat po $\overline{\text{PROG}}$	1.0		μs
t _{EHSH}	Předstih ENABLE před V _{PP}	1.0		μs
t _{SHGL}	Předstih V _{PP} před $\overline{\text{PROG}}$	10		μs
t _{GHSL}	Držení V _{PP} po $\overline{\text{PROG}}$	10		μs
t _{GLGH}	Trvání $\overline{\text{PROG}}$	1	110	μs
t _{ELQV}	Předstih ENABLE před platností dat		1.0	μs
t _{EHQZ}	Data Float After ENABLE	0	1.0	μs
t _{GHBL}	Zpoždění BSY po $\overline{\text{PROG}}$		50	ns
t _{WC}	Trvání BSY – čas zápisu Bytu		2.0	ms
t _{BHIH}	Zpoždění inkrementálního impulzu po RDY	1.0		μs
t _{IHIL}	Trvání hodinových pulzů	200		ms

Tab. 22 Parametry programování AT89C1051, AT89C2051, AT89C4051

2.4 MIKROKONTROLÉRY AT89C52, AT89S8252, AT89S53, AT89C55

V standardu úspěšných řídicích mikroprocesorů sehrával důležitou úlohu procesor 8052, který byl realizován rozšířením 8051 o 128 bytů interní RAM a třetí čítač/časovač. Podobně je to i s mikrokontrolérem AT89C52, avšak v porovnání s AT89C51 má rozšířenou paměť programu FLASH EPROM na 8 KB a typ AT89S8252 má navíc vestavěnou EEPROM paměť 2 KB s možností 100 000 cyklů mazání/zápis pro uchování dat. Mikrokontrolér AT89S53 má paměť programu rozšířenou na 12 KB a AT89C55 na 20 KB.

Mikrokontroléry AT89C52 jsou proti AT89C51 výkonnější, mají větší paměť dat – 256 × 8 bitů, tři časovače/čítače, až 8 kB paměť FLASH EPROM. V ostatních parametrech se s předcházejícím typem shodují. Ještě výkonnější typ je AT89C55. Tento typ je v základních technických parametrech shodný s typem AT89C52, má však integrovanu výrazně větší paměť programu – 20 kB FLASH EPROM pro náročnější aplikace, pracuje až do 30 MHz a má devět úrovní přerušení.

Neméně zajímavým mikrokontrolérem v 40vývodovém pouzdře PDIP analogickým s obvody AT89C5x je mikrokontrolér **AT89S8252**, který má navíc několik dalších zajímavých funkcí. Proto pro informativní obraz o tomto obvodu je uveden přehled jeho základních technických parametrů:

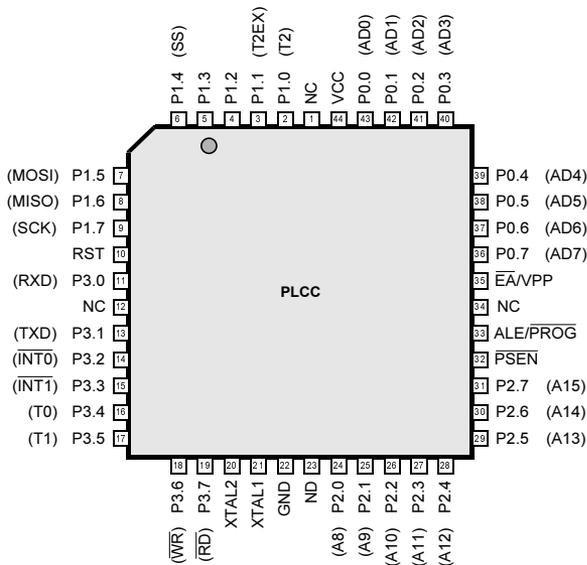
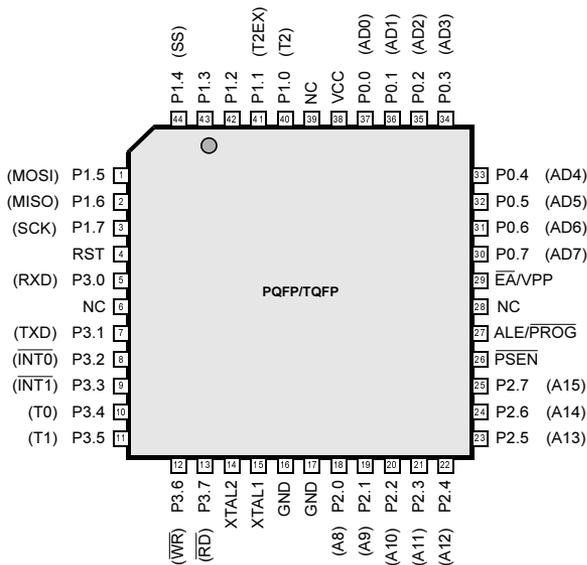
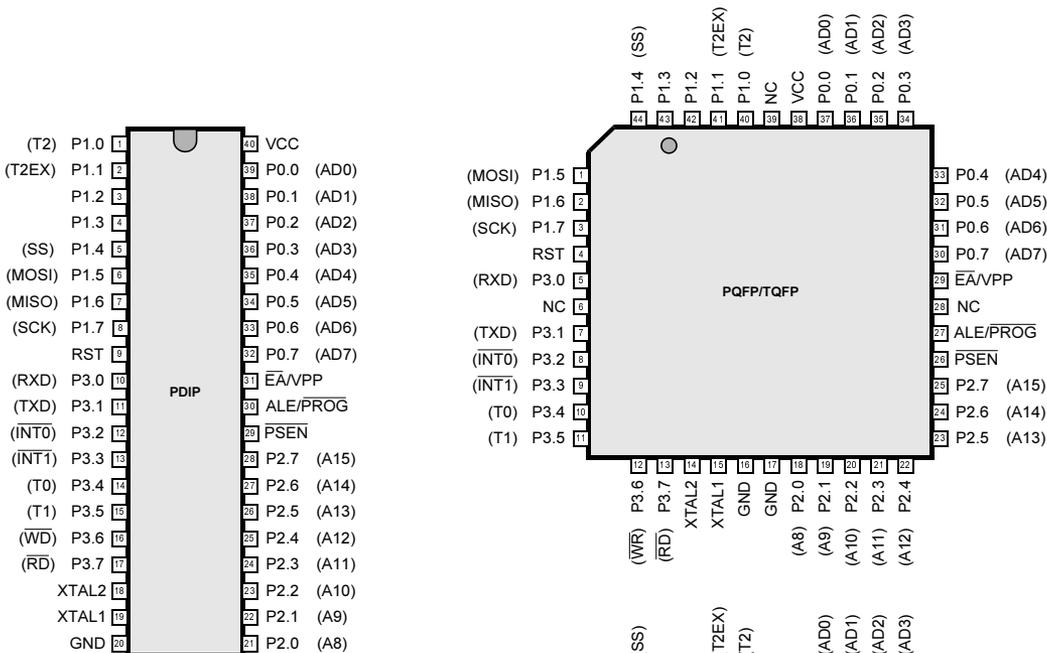
- Plně kompatibilní se standardy MCS-51.
- 8 kB paměti typu FLASH EPROM se schopností až 1000 zápisových/mazacích cyklů – SPI sériový interfejs pro natažení programu.
- 2 kB EEPROM (100 000 cyklů zápisu).
- Napájecí napětí 4 až 6 V.
- Plně statická činnost: 0 Hz až 24 MHz.
- Tříúrovňové uzamčení paměti programu.
- 256 × 8bitová interní RAM.
- 32 programovatelných I/O linek.
- Tři 16bitové časovače/čítače.
- Devět zdrojů přerušení.
- Programovatelný sériový kanál UART.
- SPI sériový interfejs.
- Nízkopříkonové módy činnosti: Idle a Power Down.
- Návrat z Power Down módu přes přerušení.
- Programovatelný hodinový obvod.
- Dva ukazatele dat.
- Příznak Power Off.

Je to skutečně mikrokontrolér s bohatou škálou funkcí a velice dobrým výkonem. Paměť EEPROM navíc integrovaná na čipu, zachovávající data i po vypnutí napájecího napětí, je výborně využitelná k zapisování dat, přičemž snese až 100 000 zápisových cyklů. Navíc výkonný program je možno modifikovat přímo v aplikaci nahráváním přes SPI sériový interfejs standardním programátorem pro energeticky nezávislé (NV-SRAM paměti). Kombinace všestranné 8bitové CPU a naprogramovatelné paměti FLASH EPROM na čipu dělá z AT89S8252 výkonný mikrokontrolér, umožňující vysoce flexibilní, laciné a efektivní řešení pro mnoho aplikací v průmyslu. Verze dle pouzdra a pracovního rozsahu teplot jsou uvedeny v tab. 23.

Posledním z popisovaných mikrokontrolérů je typ **AT89S53**. Na rozdíl od typu AT89S8252 nemá na čipu integrovanou paměť EEPROM, ale má větší paměť FLASH EPROM a to 12 kB. Obsah paměti programu může být také změněn přímo v aplikaci, což výrazně zvyhodňuje tento prvek při volbě v řešení různých problémů v automatizovaných systémech řízení a regulačních obvodech. Tyto obvody jsou podobně jako typy AT89S8252 dodávány

Kmitočet [MHz]	Napájecí napětí [V]	Značení	Pouzdro				Rozsah prac. teplot [°C]
			TQFP	PLCC	PDIP	PQFP	
12	4 až 6	AT89C52-12	AC	JC	PC	QC	0 až 70
12	4 až 6	AT89C52-12	AI	JI	PI	QI	-40 až 85
16	4 až 6	AT89C52-16	AC	JC	PC	QC	0 až 70
16	4 až 6	AT89C52-16	AI	JI	PI	QI	-40 až 85
20	4 až 6	AT89C52-20	AC	JC	PC	QC	0 až 70
20	4 až 6	AT89C52-20	AI	JI	PI	QI	-40 až 85
24	4 až 6	AT89C52-24	AC	JC	PC	QC	0 až 70
24	4 až 6	AT89C52-24	AI	JI	PI	QI	-40 až 85
16	4 až 6	AT89S53-16	AA	JA	PA	-	-40 až 105
24	4 až 6	AT89S53-24	AC	JC	PC	-	0 až 70
24	4 až 6	AT89S53-24	AI	JI	PI	-	-40 až 85
33	4,5 až 5,5	AT89S53-33	AC	JC	PC	-	0 až 70
16	4 až 6	AT89C55-16	AA	JA	PA	QA	-40 až 105
24	4 až 6	AT89SC55-24	AC	JC	PC	QC	0 až 70
24	4 až 6	AT89C55-24	AI	JI	PI	QI	-40 až 85
33	4,5 až 5,5	AT89C55-33	AC	JC	PC	QC	0 až 70
16	4 až 6	AT89S8252-16	AA	JA	PA	QA	-40 až 105
24	4 až 6	AT89S8252-24	AC	JC	PC	QC	0 až 70
24	4 až 6	AT89S8252-24	AI	JI	PI	QI	-40 až 85
33	4,5 až 5,5	AT89S8252-33	AC	JC	PC	QC	0 až 70

Tab. 23 Značení pouzder AT89C52, AT89S53, AT89C55, AT89S8252



**Signály SS, MOSI, MISO a SCK
jenom pro typy AT89S...
Adresové bity platné dle kapacity
FLASH mikrokontroléru.**

*Obr. 53 Pouzdření AT89C52,
AT89S53, AT89C55, AT89S8252*

v těch samých pouzdech mimo typu PQFP. Na *obr. 54* a *obr. 55* jsou schematicky zapojeny mikrokontroléry AT89x52, ..53, ..55, ..8252 v programovacím a verifikačním režimu.

Navzdory tomu, že paralelní programování už bylo v předchozích částech popsáno, je zde uvádíme také, protože pro mikrokontroléry s vnitřní EEPROM jsou patrné rozdíly především v adresování a v přístupu k oběma oblastem paměti (FLASH EPROM a EEPROM).

Je nutno si uvědomit, že v případě paralelního programování, obě (kódová i datová) oblasti paměti obsazují stejný souvislý paměťový prostor na adresách 0000H-1FFFH pro kódovou oblast paměti (FLASH) a adresy 2000H-27FFH pro datovou oblast paměti (EEPROM).

V paralelním programování nelze využít automatického mazání při zápisu jako je popsáno níže u sériového programování. Proto je nutno dodržet konvenci o smazání celé paměti pokud její obsah není prázdný tj. přímo u výrobce nastaven na hodnotu FFH pro veškerá paměťová místa.

Při paralelním programování je nutno dodržet následující posloupnost kroků:

1. Po přivedení napájení mezi Vcc a GND vložit na RST = log. 1 při zapojeném oscilátoru.
2. Přivést na vývod $\overline{\text{PSEN}}$ log. 0 při současném nastavení ALE = log. 1, $\overline{\text{EA}}$ = log. 1 a ostatní nevyužití vývody = log. 1.
3. Přivést správnou kombinaci řídicích signálů (P2.6, P2.7, P3.6 a P3.7).
4. Přivést adresu na vývody P1.0-P1.7 a P2.0-P2.5.
5. Přivést data na vývody P0.0-P0.7 při operaci zápisu.
6. Zvednout $\overline{\text{EA}}/V_{\text{pp}}$ na 12 V pro povolení programování (mazání nebo verifikaci) FLASH.
7. Injektovat impulz do ALE/ $\overline{\text{PROG}}$ pro naprogramování datového bytu nebo Lock bitu do FLASH EPROM paměti. Cyklus zápisu bytu je časován vnitřně a trvá typicky 1,5 ms.
8. Na verifikaci změnit bit P2.7 = 0 a načíst data přes vývody P0.0-P0.7

Opakujte kroky 3 až 8 se změnou adres a dat pro požadovanou oblast paměti, resp. po konec programovaného datového souboru.

Data/Polling je vlastnost, která indikuje konec zápisového cyklu. Pokus o čtení dat v průběhu výkonu zápisového cyklu v paralelním nebo sériovém programovacím režimu způsobí komplementaci na P0.7 vývodu nebo MSB bitu na sériovém výstupu MISO. Poté co byl zápisový cyklus ukončen, jsou správná data na všech výstupech a může začít další zápisový cyklus. Testováním této výzvy je možno optimalizovat programování paměti.

Ready/Busy – Postup programování jednotlivého bytu může být monitorován taky prostřednictvím výstupního signálu RDY/BSY. Vývod P3.4 tj. RDY/BSY je po přivedení signálu PROG stažen dolů a v průběhu programování indikuje stav BUSY – obsazen. Po ukončení programování se P3.4 RDY/BSY vrací zpět nahoru a indikuje stav READY – připraven. Tento BUSY stav trvá typicky 2 ms a je rozhodujícím pro rychlost programování.

Program Verify (Verifikace programu) – obr. 55. Jestliže nebyly naprogramovány uzamykací bity, nebo byl-li naprogramován pouze uzamykací bit LB1, mohou být naprogramovaná data opětovně čtena. Nastavení uzamykacích bitů LB1-LB3 nelze zjistit přímo. Zjistí se to explicitně tím, že jsou blokovány příslušné funkce.

Chip Erase (Mazání čipu). Celý obsah paměti FLASH EPROM i EEPROM je současně vymazán elektricky využitím správné kombinace řídicích signálů a podržením ALE/PROG na úrovni log. 0 po dobu 10 ms. Do všech paměťových buněk jsou zapsány logické jedničky. Před libovolným dalším přeprogramováním paměti musí být vykonáno mazání. V sériovém programování se čip maže spuštěním instrukce pro mazání. Výkon této instrukce je časován vnitřně a trvá přibližně 16 ms. Čtení v průběhu mazání z libovolného paměťového místa, vrací na výstupu nuly.

Reading the Signature Bytes (Čtení značkových bytů). Značkové byty se čtou tím samým způsobem jako při normální verifikaci paměti na adresách 30H, 31H s tím, že vývody P3.6 a P3.7 musí být nastaveny na úroveň log. 0. Čtené hodnoty jsou:

(30H) = 1EH indikuje výrobce ATMEL;

(31H) = 72H indikuje 89S8252;

Mód	RST	PSEN	ALE/PROG	\overline{EAV}_{pp}	P2.6	P2.7	P3.6	P3.7	Data I/O P0.7:0	Address P2.5:0 P1.7:0
Sériové programování	H	h ⁽¹⁾	h ⁽¹⁾	x						
Mazání	H	L	 ⁽²⁾	12V	H	L	L	L	X	X
Zápis(10K bajtu) paměti	H	L	 ⁽²⁾	12V	L	H	H	H	DIN	ADDR
Čtení(10K bajtu) paměti	H	L	H	12V	L	L	H	H	DOUT	ADDR
Zápis uzamykacích bitů	H	L	 ⁽²⁾	12V	H	L	H	L	DIN	X
Bit - 1									P0.7 = 0	X
Bit - 2									P0.6 = 0	X
Bit - 3									P0.5 = 0	X
Čtení uzamykacích bitů:	H	L	H	12V	H	H	L	L	DOUT	X
Bit - 1									⊙P0.2	X
Bit - 2									⊙P0.1	X
Bit - 3									⊙P0.0	X
Čtení kódu výrobce	H	L	H	12V	L	L	L	L	DOUT	30H
Čtení kódu mikrokontroléru	H	L	H	12V	L	L	L	L	DOUT	31H
Povolení sériového programování	H	L	 ⁽²⁾	12V	L	H	L	H	P0.0 = 0	X
Zákaz sériového programování	H	L	 ⁽²⁾	12V	L	H	L	H	P0.0 = 1	X
Čtení pojistky pro sériové programování	H	L	H	12V	H	H	L	H	⊙P0.0	X

POZNÁMKY:

1. „h“ = interní připojení na log. 1
2. Mazání a pojistka sériového programování vyžaduje pulz \overline{PROG} o délce 10 ms.
3. P3.4 = log. 0 indikuje po dobu programování $\overline{RDY/BSY}$.
4. „X“ = bez významu.

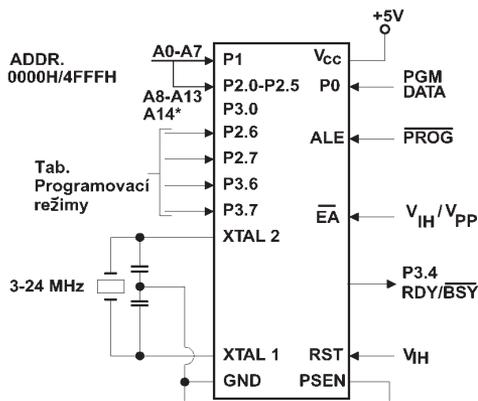
Tab. 24 Módy paralelního programování pro FLASH a EEPROM AT89S8252

Serial Programming Fuse (Sériová programovací pojistka) – blokuje sériové programování. Může být naprogramována pouze v paralelním programovacím režimu.

Serial Downloading (Sériové programování). Obě paměti FLASH (adresy 0000H po 1FFFH) i EEPROM (000H po 7FFFH) mohou být naprogramovány použitím sériové SPI sběrnice při $RST = V_{CC}$. Sériové rozhraní tvoří vývody mikrokontroléru SCK, MOSI a MISO. Instrukce povolení programování musí předcházet libovolnou instrukcí pro mazání, čtení nebo zápis. V sériovém programovacím módu je operace mazání s vnitřním časováním vykonaná automaticky a tedy není nutno vykonat mazání celého čipu, pokud nebyly naprogramovány uzamykací bity. Operace mazání čipu změni obsah všech paměťových míst na FFH. Taktování sériového přenosu je určeno externě přivedením taktovacích impulzů na vývod XTAL1 nebo jako 1/40 kmitočtu krystalového oscilátoru tj. při 24 MHz maximálně 600 kHz.

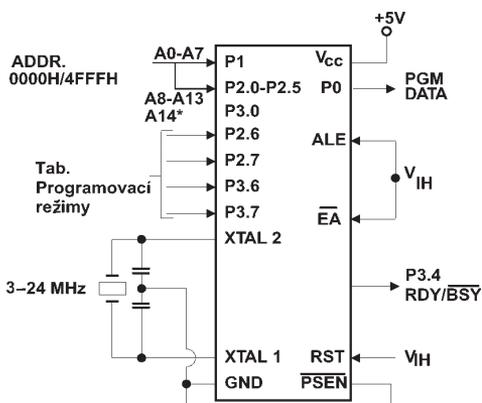
Sériové programování nejlépe charakterizuje následující posloupnost kroků:

1. Po provedení napájení mezi V_{CC} a GND vložit na $RST = \text{log. 1}$ při zapojeném oscilátoru.
2. Odeslat instrukci povolení sériového programování přes vývod MOSI/P1.5. Kmitočet taktování na vývodu SCK/P1.7 se vyžaduje menší než 1/40 kmitočtu oscilátoru.

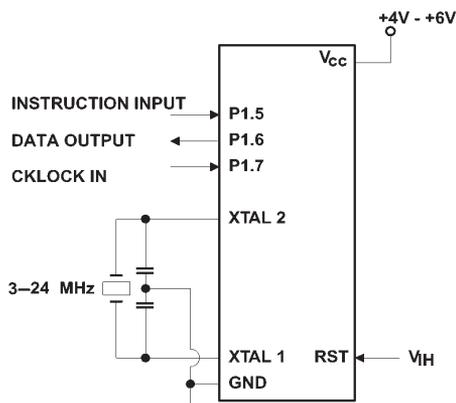


Adresa A14 (P3.0) není shodná s adresou A14 externí paměti P2.6

Obr. 54 Schéma paralelního programování paměti FLASH EPROM



Obr. 55 Schéma verifikace paměti FLASH EPROM



Obr. 56 Schéma sériového programování paměti AT89S8252

3. Kódová i datová oblast paměti se programuje byte po byte posláním adresy, dat i kódu instrukce pro zápis. Zvolené místo paměti je nejdříve (před zápisem) automaticky vymazáno. Vnitřně časován zápisový cyklus trvá typicky 2,5 ms při 5 V.
4. Libovolné místo paměti lze verifikovat použitím instrukce čtení, která vrací obsah dle vybrané adresy přes sériový výstup MISO/P1.6.
5. Po skončení programování je nutno pro běžný provoz nastavit RST = log. 0.

Instrukční sada pro sériové programování v 3bytovém protokolu je zobrazena v tab. 25.

Instrukce	Vstupní formát			Operace
	Byte 1	Byte 2	Byte 3	
Povolení programování	1010 1100	0101 0011	xxxx xxxx	povoluje programování po přechodu RST do 1.
Mazání paměti	1010 1100	xxxx x100	xxxx xxxx	mazání obou 8K a 2k paměťových polí
Čtení kódu	aaaa a001	low addr	xxxx xxxx	Čtení dat z paměti programu. Prvních 5 MSB bitů jsou vyšší adresové bity a nižší adresové bity jsou v druhém byte. Data jsou přístupná na výstupu MISO ve třetím bajtu.
Zápis kódu	aaaa a010	low addr	data in	Zápis dat na vybrané adrese do paměti programu. Prvních 5 bitů v prvním byte jsou MSB adresy spolu s druhým byte.
Čtení dat	00aa a101	low addr	xxxx xxxx	Čtení dat z vybrané adresy datové paměti. Data jsou přístupná na výstupu MISO po dobu třetího byte.
Zápis dat	00aa a110	low addr	data in	Zápis dat do datové paměti na vybrané adrese
Zápis uzamykacích bitů	1010 1100	0101 0011	xxxx xxxx	Nastavení LB1, LB2 nebo LB3 = „0“ na programování zamykacích bitů

POZNÁMKY:

1. Data Polling je užito pro indikaci konce zápisového cyklu, který je kratší než 2,5 ms při 5 V.
2. „aaaa“ – vyšší řád adresy
3. „X“ = bez významu.

Tab. 25 Instrukční sada sériového programování

2.5 PROGRAMOVATELNÝ HLÍDACÍ ČASOVAČ (WDT)

Programovatelný hlídací časovač pracuje s nezávislým vnitřním oscilátorem. Předdělička tvořena bity PS0, PS1, PS2 v speciálním registru WMCON nastavuje periodu činnosti časovače v rozsahu 16 ms až 2048 ms jak znázorňuje tab. 26.

Bity předděličky			Perioda (nominální) [ms] ±30 %
PS2	PS1	PS0	
0	0	0	16
0	0	1	32
0	1	0	64
0	1	1	128
1	0	0	256
1	0	1	512
1	1	0	1024
1	1	1	2048

při $V_{cc} = 5\text{ V}$

Tab. 26 Nastavení hlídacího časovače

Činnost časovače WDT se povoluje nastavením bitu WDTRST. Je-li činnost povolena, musí se časovač pravidelně resetovat nastavením bitu WDTRST před uplynutím nastavené periody. Když časovač WDT přeteče (bez předcházejícího resetu nebo zákazu činnosti), generuje se vnitřní impuls RST, který ziniculuje mikrokontrolér.



PS2, PS1, PS0 – Bity předděličky pro WDT časovač.

EEMWE – (EEPROM Data Memory Write Enable). U typů mikrokontrolérů s interní EEPROM na čipu bit povolující zápis do paměti při EEMWE = 1. Je-li EEMWE = 0, pak je zápis blokován. Uživatelský aplikační program by měl tento bit důsledně nastavovat, aby nedocházelo k mylnému (duplicitnímu) zápisu při použití instrukce MOVX.

EEMEN – (EEPROM Access Enable). Bit povolující přístup k paměti EEPROM. Je-li EEMEN = 1, pak instrukce MOVX se vykoná nad paměť EEPROM místo externí RAM. Je-li EEMEN = 0, pak instrukce MOVX se vykoná nad externí RAM.

DPS – (Data Pointer Select). Výběr ukazatele dat. Je-li DPS = 0, je vybrána první banka DPTR tj. registrový pár ukazatele dat [DPH0, DPL0]. Je-li DPS = 1, pak je vybrána druhá banka DPTR tj. registrový pár ukazatele dat [DPH1, DPL1].

WDTRST RDY/RST – (Watch Dog Timer Reset a EEPROM Ready/Busy Flag). Při každém zápisu log. 1 do bitu je vynulován hlídací časovač WDT. WDTRST je nastaven automaticky na log. 0 vykonáním následující instrukce. Bit WDTRST je pouze pro zápis.

Tento bit je současně příznakem zaneprázdnění EEPROM paměti. Je-li $\overline{RDY/BSY} = 1$, pak paměť EEPROM je připravena na zápis. V čase zápisu do EEPROM je bit $\overline{RDY/BSY} = 0$ a je automaticky vrácen do log. 1, pokud je proces zápisu (cca 10 ms) ukončen.

WDTEN – (Watch Dog Timer Enable). Příznak povolení činnosti časovače WDT. Je-li $WDTEN = 1$ je jeho činnost povolena, je-li $WDTEN = 0$, je činnost zakázána.

Interní EEPROM paměť je aktivována prostřednictvím bitu EEMEN v registru WMCOM. Její kapacita pro AT89S8252 je 2 KB v rozsahu adres 000H – 7FFH. Pro přístup k externí paměti dat je nevyhnutelné nulovat bit EEMEN. Zápis do EEPROM je podmíněn nastavením bitu EEMWE. Typická doba zápisu do EEPROM je 2,5 ms. Průběh zápisu do EEPROM lze sledovat testováním bitu $\overline{RDY/BSY}$. Navíc v době zápisu se nedoporučuje paměť ani číst, neboť vlastní zápis je ukončen nekorektně – obvykle je bit MBS zapsán chybně.

2.6 ROZHRANÍ SÉRIOVÉHO STYKU (SPI)

Rozhraní sériového styku (SPI – Serial Peripheral Interface) – umožňuje vysokorychlostní synchronní přenos dat mezi AT89S8252 a vnějšími zařízeními nebo mezi více mikrokontroléry AT89S8252 vzájemně. Základní vlastnosti SPI přenosu jsou:

- Plný duplex, třívodičový synchronní přenos dat.
- Operace podřízení (Master-Slave).
- 1,5MHz bitový kmitočet (max.).
- První vysílaný LSB nebo MSB.
- Čtyři programovatelné dávky bitů.



- Příznak ukončení přenosu.
- Ochranný příznak kolize zápisu .
- Probuzení z IDLE režimu (pouze pro Slave).

SPI řídicí registr má tvar:

SPIE – (SPI Interrupt Enable) Povolení přerušení od SPI. Tento bit v závislosti od příznaku ES v registru IE povoluje SPI přerušení. Je-li $SPIE = 1$ a $ES = 1$ je přerušení povoleno. Pro $SPIE = 0$ je SPI přerušení zakázáno.

SPE – (SPI Enable) Povolení činnosti SPI. Je-li $SPE = 1$ je SPI kanál povolen a bity SS, MOSI, MISO, SCK jsou připojeny na vývody P1.4, P1.5, P1.6, P1.7. Pro $SPE = 0$ je SPI zakázáno.

DORD – (Data Order) Řazení dat. Je-li $DORD = 1$, první je přenášen LSB. Je-li $DORD = 0$, pak je první přenášen MSB.

MSTR – (Master/Slave Select) Nastavení nadřazenosti/podřazenosti mikrokontroléru. $MSRT = 1$ pro Master, $MSTR = 0$ pro Slave.

CPOL – (Clock Polarity) Polarita hodin. Je-li $CPOL = 1$, pak je $SCK = 1$ při IDLE režimu. Je-li $CPOL = 0$, pak je SCK na straně Master = 0, když není přenos.

CPHA – (Clock Phase) Fáze hodin. Bit CPHA spolu s bitem CPOL řídí vztahy mezi hodiny a daty, mezi stranou Master a Slave.

SPR1, SPR0 – (SPI Clock Rate Select) Tyto dva bity řídí dávkový přenos na SCK na straně Master. Bity SPR1, SPR0 na straně Slave nemají žádný vliv. Vztahy mezi SCK a kmitočtu F_{osc} jsou v tab. 26.

SPR1	SPR0	SCK = F_{osc} dělená dělitelem
0	0	4
0	1	16
1	0	64
1	1	128

Tab. 27 Nastavení dávkového přenosu SPI

Stavový registr SPI:

	(MSB)							(LSB)
SPSR	SPIF	WCOL	-	-	-	-	-	-

SPIF – (SPI Interrupt Flag). Když je přenos kompletní je $SPIF = 1$ a je generováno přerušení při ($SPIE = 1$ a $ES = 1$). SPIF je nulován čtením SPI stavového registru SPSR.

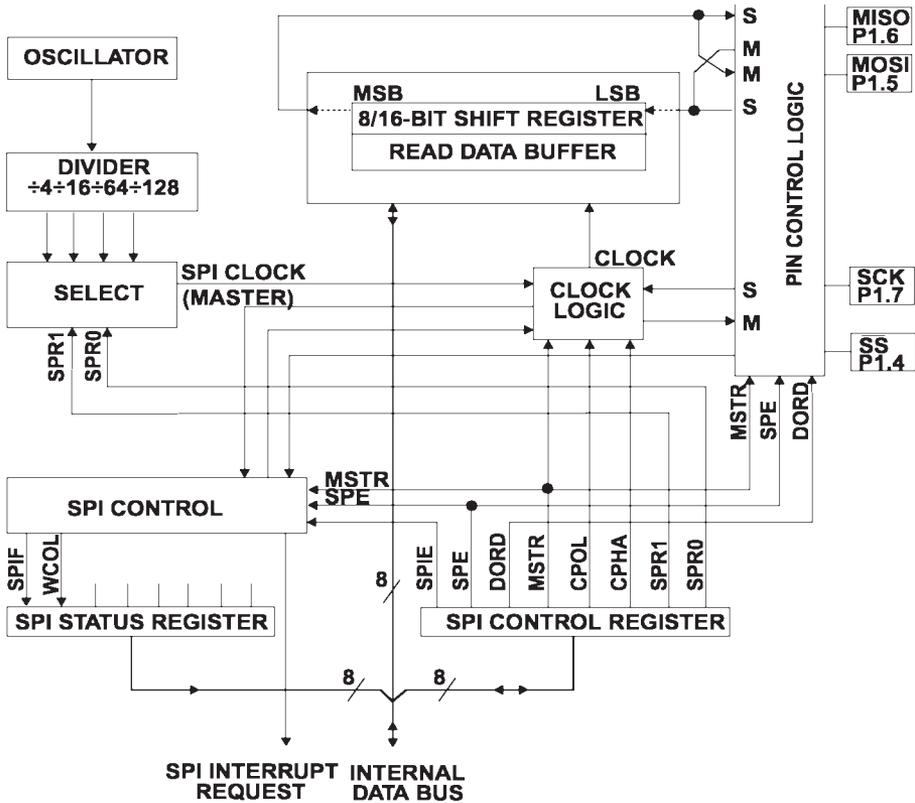
WCOL – (Write Collision Flag). WCOL je nastaven, je-li SPI datový registr přepsán v čase přenosu. V průběhu přenosu dat čtení z registru SPDR může být nekorektní. Případný zápis nemá na přenos žádný vliv. WCOL je vynulován přečtením stavového registru SPSR.

Blokové schéma činnosti SPI je na obr. 57.

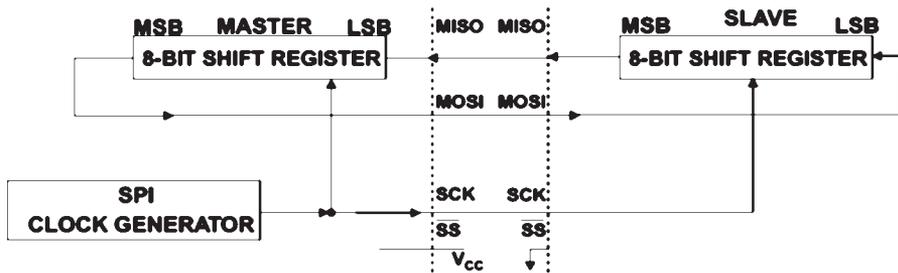
Datový registr SPI:

	(MSB)							(LSB)
SPDR	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0

Propojení mezi mikrokontroléry typu Master a Slave je na obr. 58. Vývod SCK je hodinový výstup pro mikrokontrolér Master a hodinový vstup pro mikrokontrolér Slave. Přenos startuje zápisem do SPI datového registru rozběhem SCI hodinového generátoru a zapisovaná data jsou přesouvána přes vývody MOSI obou mikrokontrolérů. Po přesunu jednoho bytu SPI se zastaví hodinový generátor s nastavením příznaku pro ukončení přenosu. Jsou-li příznakové bity SPIE (SPI interrupt enable) a ES (serial port interrupt enable) nastaveny, pak je přerušení požadováno. Výběrový vstup podřazeného mikrokontroléru Slave SS/P1.4 lze použít na individuální výběr podřazeného mikrokontroléru Slave. Je-li $SS/P1.4 = 1$, pak je SPI port deaktivován a vývod MOSI/P1.5 lze využít jako vstup.

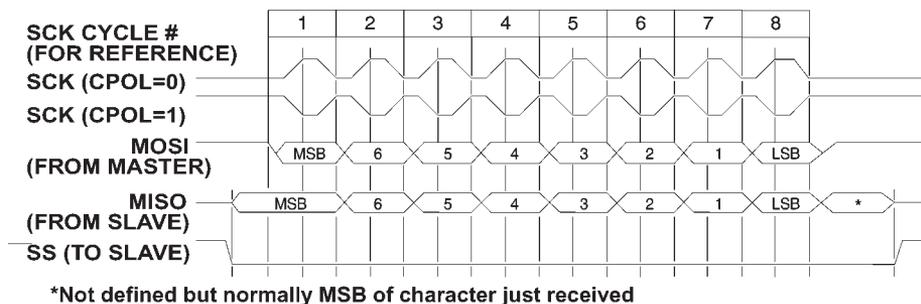


Obr. 57 SPI – blokové schéma

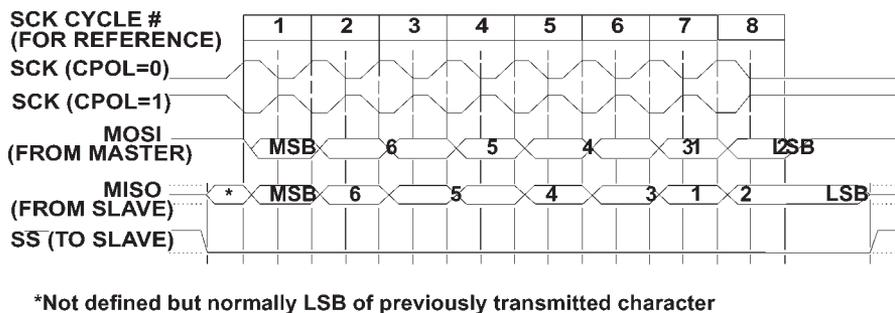


Obr. 58 SPI – propojení Master-Slave

Existují čtyři kombinace pro fázi a polaritu signálu SCK s ohledem na sériová data, které se určují řídicími bity CPHA a CPOL. Formáty přenosu dat jsou znázorněny na obr. 59 a obr. 60.



Obr. 59 SPI – formát přenosu pro CPHA = 0



Obr. 60 SPI – formát přenosu pro CPHA = 1

3 INSTRUKČNÍ SOUBOR

Soubor instrukcí řady obvodů 8051, tedy i mikrokontrolérů ATMEL obsahuje 111 instrukcí, z toho 49 jednobytových, 45 dvoubytových a 17 tříbytových. Formát instrukce se obvykle skládá z operačního kódu, adresy příjemce a zdroje dat.

Instrukce pro přenos dat

Tyto instrukce je možné rozdělit na instrukce, které realizují

- aritmetické;
- logické;
- s Booleovými proměnnými;
- pro přenos dat;
- pro větvení programu.

Univerzální přenosy typu MOV realizují přenos bitů resp. bytů (celých osmibitových slov) ze zdrojové adresy na cílovou adresu. Přenosy s pomocí instrukce PUSH přenášejí jeden Byte do zásobníku na adresu, kterou obsahuje SP registr. Podobně přenosy s pomocí instrukce POP přenášejí jeden byte ze zásobníku na cílovou adresu, přičemž se obsah SP registru po instrukci dekrementuje.

Přenosy přes střadač využívají hlavně instrukce XCH, XCHD, MOVX a MOVXC. Instrukce XCH realizuje výměnu obsahu zdrojové adresy a střadače ACC, instrukce XCHD realizuje výměnu nižší tetrády zdrojové adresy a ACC. Instrukce MOVX přenáší jeden Byte dat mezi vnější paměť dat a ACC. Paměť dat se adresuje registrem DPTR nebo registry R0 resp. R1. Instrukce MOVC přenáší jeden byte mezi vnější paměť programu a ACC.

Na přenos adresy se používá instrukce MOV DPTR, která naplní registr DPTR 16bitovou konstantou.

Žádná z instrukcí na přenos dat neovlivňuje obsah registru PSW.

Aritmetické instrukce

Mikrokontroléry ATMEL, kompatibilní s řadou obvodů 8051 mohou realizovat všechny čtyři základní aritmetické operace. V aritmetických operacích možno pracovat jen s 8bitovými čísly bez znaménka. Při využití příznaku OV mohou pracovat i s čísly v 2. doplňku.

Mezi instrukce na realizaci aritmetických operací patří instrukce na dekrementaci zdrojového operandu (instrukce INC), přičtení zdrojového operandu k ACC (instrukce ADD a ACC) a instrukce na desítkovou korekci výsledku (instrukce DA). Odečtení možno realizovat instrukcemi SUBB a dekrementování zdrojového operandu instrukcí DEC.

Instrukce MUL realizuje násobení 8bitových čísel bez znaménka, uložených v ACC a registru B, přičemž výsledek je uložen v ACC (nižší byte) a v registru B (vyšší byte).

Instrukce DIV realizuje dělení obsahu ACC obsahem registru B, přičemž celočíselná část výsledku je v ACC a desetinná část v registru B.

Logické instrukce

Logické instrukce mohou pracovat s jedním nebo dvěma operandy. Pokud instrukce pracuje s jedním operandem, je to obsah ACC. Při instrukcích se dvěma operandy se jeden z operandů nachází v ACC a druhý je v paměti dat. Logickými instrukcemi s dvěma operandy možno realizovat logické operace AND (instrukce ANL), OR (instrukce ORL) a Exclusive OR (instrukce XRL).

Instrukce na větvení programu

V této skupině instrukcí se nacházejí podmíněné i nepodmíněné skoky, které mohou být krátké v rozmezí jedné stránky paměti nebo dlouhé v rozmezí 64 k. Podobně se realizují i dva typy volání podprogramu, volání podprogramu absolutně (instrukce ACALL) a volání podprogramu v celém rozsahu paměti programu (instrukce LCALL). Mimo to je možné realizovat nepřímý skok, u kterého adresa skoku se určí obsahem registru DPTR, ke kterému se přičte obsah ACC. Na návrat z podprogramu se používají dvě instrukce: instrukce RETI při návratu z obslužného podprogramu přerušeni a RET při návratu ze všech ostatních typů podprogramů.

Podmíněné skoky jsou vázány na příznaky Z, NZ, C, NC a hodnoty adresovaných bitů (instrukce JB a JNB).

Na vytváření cyklů máme dva podmíněné skoky. První se realizuje instrukcí CJNE, která vykonává porovnání dvou operandů a při neshodě se realizuje skok. Druhý se realizuje instrukcí DJNZ, která dekrementuje obsah zdrojové adresy a skok se realizuje při jejím nenulovém obsahu. V instrukční sadě se rovněž nacházejí instrukce pro práci s porty, s čítači/časovači resp. pro práci s přerušeni.

3.1 PŘEHLED INSTRUKCÍ MIKROKONTROLÉRŮ ATMEL

Legenda

A	– obsah střadače
Rn	– obsah registru
(Ri)	– obsah nepřímo adresovaného paměťového místa /R0, R1/
C, AC, OV	– příznaky
PC	– programový čítač
direct	– obsah přímo adresovaného paměťového místa
#data	– osmibitová konstanta
#data16	– šestnáctibitová konstanta
rel	– relativní adresa
@DPTR	– ukazatel adresy – nepřímý přístup
bit	– 8bitová adresa bitu
add11	– 11bitová adresa programové paměti
add16	– 16bitová adresa programové paměti

Symbol			Mnemonika	
Kód:	1. operand	Hex	Popis instrukce	Byty/Cykly
	2. operand	Hex		Periody
	3. operand	Hex		Příznaky
Instrukce aritmetické				
ADD A, Rn			$A \leftarrow A + Rn$	
Kód:	1. 0010 1 r r r	28-F	Sčítání obsahu registru Rn a střadače	1/1
	2.			12
	3.			C,AC,OV
ADD A, direct			$A \leftarrow A + \text{direct}$	
Kód:	1. 0010 0100	25	Sčítání obsahu přímo adresovaného paměťového místa v RWM a střadače	2/1
	2. direct			12
	3.			C,AC,OV
ADD A, @ Ri			$A \leftarrow A + (Ri)$	
Kód:	1. 0010 011 i	26-7	Sčítání obsahu nepřímo adresovaného paměťového místa v RWM a střadače	1/1
	2.			12
	3.			CY,AC,OV
ADD A, # data			$A \leftarrow A + \text{data}$	
Kód:	1. 0010 0100	24	Sčítání konstanty a obsahu střadače	2/1
	2. data			12
	3.			C,AC,OV
ADDC A, Rn			$A \leftarrow A + Rn + C$	
Kód:	1. 0011 1 r r r	38-F	Sčítání obsahu registru Rn a střadače s přenosem	1/1
	2.			12
	3.			C,AC,OV
ADDC A, direct			$A \leftarrow A + \text{direct} + C$	
Kód:	1. 0011 0101	35	Sčítání obsahu přímo adresovaného paměťového místa v RWM a střadače s přenosem	2/1
	2. direct			12
	3.			C,AC,OV
ADDC A, @ Ri			$A \leftarrow A + (Ri) + C$	
Kód:	1. 0011 011 i	36-7	Sčítání obsahu nepřímo adresovaného paměťového místa v RWM a střadače s přenosem	1/1
	2.			12
	3.			C,AC,OV
ADDC A, # data			$A \leftarrow A + \text{data} + C$	
Kód:	1. 0011 0100	34	Sčítání konstanty a obsahu střadače s přenosem	2/1
	2. data			12
	3.			C,AC,OV
SUBB A, Rn			$A \leftarrow A - Rn - C \quad n = 0-7$	
Kód:	1. 1001 1 r r r	98-F	Odečtení obsahu registru Rn od střadače s půjčkou	1/1
	2.			12
	3.			C,AC,OV

Tab. 28 Instrukce mikrokontrolérů ATMEL

SUBB A, direct			$A \leftarrow A - \text{direct} - C$	
Kód:	1. 1001 0101	95	Odečtení obsahu přímo adresovaného paměťového místa v RWM a střadače	2/1
	2. direct			12
	3.			C,AC,OV
SUBB A, @ Ri			$A \leftarrow A - (Ri) - C$	
Kód:	1. 1001 011 i	96-7	Odečtení obsahu nepřímo adresovaného paměťového místa v RWM od střadače s vypůjčkou	1/1
	2.			12
	3.			C,AC,OV
SUBB A, # data			$A \leftarrow A - \text{data} - C$	
Kód:	1. 1001 0100	94	Odečtení konstanty od obsahu střadače s půjčkou	1/1
	2. data			12
	3.			C,AC,OV
INC A			$A \leftarrow A + 1$	
Kód:	1. 0000 0100	04	Inkrementace obsahu střadače	1/1
	2.			12
	3.			-
INC Rn			$Rn \leftarrow Rn + 1$	
Kód:	1. 0000 1 r r r	08-F	Inkrementace obsahu registru Rn	1/1
	2.			12
	3.			-
INC direct			$\text{direct} \leftarrow \text{direct} + 1$	
Kód:	1. 0000 0101	05	Inkrementace obsahu přímo adresovaného paměťového místa v RWM	2/1
	2. direct addr.			12
	3.			-
INC @ Ri			$Ri \leftarrow (Ri) + 1$	
Kód:	1. 0000 011 i	06-7	Inkrementace obsahu nepřímo adresovaného paměťového místa v RWM	1/1
	2.			12
	3.			-
DEC A			$A \leftarrow A - 1$	
Kód:	1. 0001 0100	14	Dekrementace obsahu střadače	1/1
	2.			12
	3.			-
DEC Rn			$Rn \leftarrow Rn - 1$	
Kód:	1. 0001 1 r r r	18-F	Dekrementace obsahu registru Rn	1/1
	2.			12
	3.			-

Tab. 28 (pokračování)

DEC direct			direct ← direct – 1	
Kód:	1. 0001 0101	15	Dekrementace obsahu přímo adresovaného paměťového místa v RWM	2/1
	2. direct addr.			12
	3.			–
DEC @ Ri			Ri ← (Ri) – 1	
Kód:	1. 0001 011r	16-7	Dekrementace obsahu nepřímě adresovaného paměťového místa v RWM	1
	2.			12
	3.			
INC DPTR			DPTR ← DPTR + 1	
Kód:	1. 1010 0011	A3	Inkrementace ukazatele DPTR	1/2
	2.			24
	3.			–
MUL AB				
Kód:	1. 1010 0100	A4	Násobení obsahu střadače a registru B	1/4
	2.			48
	3.			OV,CY
DIV AB				
Kód:	1. 1000 0100	84	Dělení obsahu střadače a registru B	1/4
	2.			48
	3.			C,OV
DA A				
Kód:	1. 1101 0100	D4	Desítková korekce obsahu střadače	1/1
	2.			12
	3.			C,AC

Instrukce logické				
ANL A, Rn			A ← A ∩ Rn	
Kód:	1. 0101 1 rrr	58-F	Logický součin AND obsahu registru a střadače	1/1
	2.			12
	3.			–
ANL A, direct			A ← A ∩ direct	
Kód:	1. 0101 0101	55	Logický součin AND obsahu přímo adresovaného paměťového místa v RWM a střadače	2/1
	2. direct			12
	3.			–

Tab. 28 (pokračování)

ANL A, @ Ri			$A \leftarrow A \cap (Ri)$	
Kód:	1. 0101 0111	56-7	Logický součin AND obsahu nepřímo adresovaného paměťového místa v RWM a střadače	1/1
	2.			12
	3.			
ANL A, # data			$A \leftarrow A \cap data$	
Kód:	1. 0101 0100	54	Logický součin AND konstanty a střadače	2/1
	2. data			12
	3.			–
ANL direct, A			$direct \leftarrow A \cap direct$	
Kód:	1. 0101 0010	52	Logický součin AND obsahu střadače a obsahu přímo adresovaného paměťového místa v RWM	2/1
	2. direct			12
	3.			–
ANL direct, # data			$direct \leftarrow direct \cap data$	
Kód:	1. 0101 0011	53	Logický součin AND konstanty a obsahu přímo adresovaného paměťového místa v RWM	3/2
	2. direct			24
	3. data			–
ORL A, Rn			$A \leftarrow A \cup Rn$	
Kód:	1. 0100 1 r r r	48-F	Logický součet OR obsahu střadače a obsahu registru Rn	1/1
	2.			12
	3.			–
ORL A, direct			$A \leftarrow A \cup direct$	
Kód:	1. 0100 0101	45	Logický součet OR obsahu střadače a obsahu přímo adresovaného paměťového místa v RWM	2/1
	2.			12
	3.			–
ORL A, @ Ri			$A \leftarrow A \cup (Ri) \quad i = 0, 1$	
Kód:	1. 0100 011 i	46-7	Logický součet OR obsahu střadače a obsahu nepřímo adresovaného paměťového místa v RWM	
	2.			12
	3.			
ORL A, # data			$A \leftarrow A \cup data$	
Kód:	1. 0100 0100	44	Logický součet OR obsahu střadače a konstanty	2/1
	2. data			12
	3.			–
ORL direct, A			$direct \leftarrow direct \cup A$	
Kód:	1. 0100 0010	42	Logický součet OR obsahu přímo adresovaného paměťového místa v RWM a střadače – výsledek v RWM	2/1
	2. direct			12
	3.			–

Tab. 28 (pokračování)

ORL direct, # data			direct ← direct ∪ data	
Kód:	1. 0100 0011	43	Logický součet OR obsahu přímo adresovaného paměťového místa v RWM a konstanty – výsledek v RWM	3/2
	2. direct			24
	3. data			–
XRL A, Rn			A ← A ⊕ Rn	
Kód:	1. 0110 1 rrr	68-F	EXCLUSIVE OR obsahu střadače a obsahu registru Rn	1/1
	2.			12
	3.			–
XRL A, direct			A ← A ⊕ direct	
Kód:	1. 0110 0101	65	EXCLUSIVE OR obsahu střadače a obsahu přímo adresovaného paměťového místa v RWM	2/1
	2. direct			12
	3.			–
XRL A, @ Ri			A ← A ⊕ (Ri)	
Kód:	1. 0110 011 i	66-7	EXCLUSIVE OR obsahu střadače a obsahu nepřímo adresovaného paměťového místa v RWM	1/1
	2.			12
	3.			–
XRL A, # data			A ← A ⊕ data	
Kód:	1. 0110 0100	64	EXCLUSIVE OR obsahu střadače a konstanty	2/1
	2. data			12
	3.			–
XRL direct, A			direct ← direct ⊕ A	
Kód:	1. 0110 0010	62	EXCLUSIVE OR obsahu přímo adresovaného paměťového místa v RWM a střadače – výsledek v RWM	2/1
	2. direct			12
	3.			–
XRL direct, # data			direct ← direct ⊕ data	
Kód:	1. 0110 0011	63	EXCLUSIVE OR obsahu přímo adresovaného paměťového místa v RWM a konstanty – výsledek v RWM	3/2
	2. direct			24
	3. data			–
CLR A			A ← 0	
Kód:	1. 1110 0100	E4	Nulování obsahu střadače	1/1
	2.			12
	3.			–
CPL A			A ← A_	
Kód:	1. 1111 0011	F4	Negace obsahu střadače	1/1
	2.			12
	3.			–

Tab. 28 (pokračování)

RL A				
Kód:	1. 0010 0011	23	Rotace obsahu střadače o jednu pozici vlevo	1/1
	2.			12
	3.			–
RLC A				
Kód:	1. 0011 0011	33	Rotace obsahu střadače o jednu pozici vlevo přes C	1/1
	2.			12
	3.			C
RR A				
Kód:	1. 0000 0011	03	Rotace obsahu střadače o jednu pozici vpravo	1/1
	2.			12
	3.			–
RRC A				
Kód:	1. 0001 0011	13	Rotace obsahu střadače o jednu pozici vpravo přes C	1/1
	2.			12
	3.			C
SWAP A				
Kód:	1. 1100 0100	C4	Výměna vyšších čtyř bitů s nižšími bity (shodné s rotací o čtyři bity vpravo nebo vlevo)	1/1
	2.			12
	3.			–

Instrukce pro přenos dat				
MOV A, Rn			$A \leftarrow R_n \quad n = 0 - 7$	
Kód:	1. 1110 1 r r r	E8-F	Přesun obsahu registru do střadače	1/1
	2.			12
	3.			–
MOV A, direct			$A \leftarrow \text{direct}$	
Kód:	1. 1110 0101	E5	Přesun obsahu přímo adresovaného paměťového místa v RWM do střadače	2/1
	2. direct			12
	3.			–
MOV A, @ Ri			$A \leftarrow (R_i) \quad i = 0, 1$	
Kód:	1. 1110 011 i	E6-7	Přesun obsahu nepřímo adresovaného paměťového místa v RWM do střadače	1/1
	2.			12
	3.			–

Tab. 28 (pokračování)

MOV A, # data			A ← data	
Kód:	1. 0111 0100	74	Naplnění střadače konstantou	2/1
	2. data			12
	3.			–
MOV Rn, A			Rn ← A n = 0 – 7	
Kód:	1. 1111 1 rrr	F8-F	Přesun obsahu střadače do registru Rn	1/1
	2.			12
	3.			–
MOV Rn, direct			Rn ← direct	
Kód:	1. 1010 1 rrr	A8-F	Přesun obsahu přímo adresovaného paměťového místa v RWM do registru Rn	2/2
	2.			24
	3.			–
MOV Rn, # data			Rn ← data n = 0 – 7	
Kód:	1. 0111 1 rrr	78-F	Naplnění registru konstantou	2/1
	2. data			12
	3.			–
MOV direct, A			direct ← A	
Kód:	1. 1111 0101	F5	Přesun obsahu střadače na přímo adresované paměťové místo v RWM	2/1
	2. direct			12
	3.			–
MOV direct, Rn			direct ← Rn	
Kód:	1. 1000 1 rrr	88-F	Přesun obsahu registru na přímo adresované paměťové místo v RWM	2/2
	2. direct			24
	3.			–
MOV direct, direct			direct ← direct	
Kód:	1. 1000 0101	85	Přesun obsahu přímo adresovaného paměť. místa na přímo adresované cílové paměť. místo v RWM	3/2
	2. direct			24
	3. direct			–
MOV direct, @ Ri			direct ← (Ri) i = 0, 1	
Kód:	1. 1000 011 i	86-7	Přesun obsahu nepřímého adresovaného paměťového místa v paměti RWM na přímo adresované paměťové místo v paměti RWM	2/2
	2. direct			24
	3.			–
MOV direct, # data			direct ← data	
Kód:	1. 0111 0101	75	Naplnění přímo adresovaného paměť. místa v RWM konstantou	3/2
	2. direct			24
	3. data			–

Tab. 28 (pokračování)

MOV @ Ri, A			$(Ri) \leftarrow A \quad i = 0, 1$	
Kód:	1. 1111 011 i	F6-7	Přesun obsahu střadače na nepřímoadresované paměťové místo v RWM	1/1
	2.			12
	3.			–
MOV @ Ri, direct			$(Ri) \leftarrow \text{direct}$	
Kód:	1. 1010 011 i	A6-7	Přesun obsahu přímo adresovaného paměťového místa v RWM na nepřímoadresované paměťové místo v RWM	2/2
	2. direct			24
	3.			–
MOV @ Ri, # data			$(Ri) \leftarrow \text{data} \quad i = 0, 1$	
Kód:	1. 0111 011 i	76-7	Naplnění obsahu nepřímoadresovaného paměťového místa v RWM konstantou	2/1
	2. data			12
	3.			–
MOV DPTR, # data 16			$DPTR \leftarrow \text{data } 16$	
Kód:	1. 1001 0000	90	Naplnění ukazatele DPTR 16bitovou konstantou	3/2
	2. data H			24
	3. data L			–
MOVC A, @ A + DPTR			$A \leftarrow (A + DPTR)$	
Kód:	1. 1001 0011	93	Přesun obsahu paměťového místa z externí paměti RWM, kterého je získána sčítáním obsahu střadače a DPTR do střadače	1/2
	2.			24
	3.			–
MOVC A, @ A + PC			$A \leftarrow (A + PC)$	
Kód:	1. 1000 0011	83	Přesun obsahu paměťového místa z externí paměti programu, kterého je získána sčítáním obsahu střadače a PC do střadače	1/2
	2.			24
	3.			–
MOVX A, @ Ri			$A \leftarrow (Ri) \quad i = 0, 1$	
Kód:	1. 1110 001 i	E2-3	Přesun obsahu nepřímoadresovaného paměťového z externí paměti RWM do střadače (jeho adresa je v registru 0 resp. 1)	1/2
	2.			24
	3.			–

Tab. 28 (pokračování)

MOVX A, @ DPTR			A ← (DPTR)	
Kód:	1. 1110 0000	E0	Přesun obsahu nepřímo adresovaného paměťového místa z externí paměti RWM do střadače (jeho adresa je v DPTR)	1/2
	2.			24
	3.			–
MOVX @ Ri, A			(Ri) ← A i = 0, 1	
Kód:	1. 1111 011 i	F6-7	Přesun obsahu střadače na nepřímo adresované paměťové místo v RWM (jeho adresa je v R0 resp. R1)	1/1
	2.			24
	3.			–
MOVX @ DPTR, A			(DPTR) ← A	
Kód:	1. 1111 0000	F0	Přesun obsahu střadače na paměťové místo v externí paměti RWM nepřímo adresované DPTR	1/2
	2.			24
	3.			–
PUSH direct			SP+1 ← direct	
Kód:	1. 1100 0000	C0	Zápis obsahu přímo adresovaného paměťového místa v RWM do zásobníku	2/2
	2. direct			24
	3.			–
POP direct			direct ← SP (-1)	
Kód:	1. 1101 000	D0	Přesun bytu ze zásobníku na přímo adresované paměťové místo v interní paměti RWM	2/2
	2. direct			12
	3.			–
XCH A, Rn			A X Rn	
Kód:	1. 1100 1 r r r	C8-F	Výměna obsahu registru Rn s obsahem střadače	1/1
	2.			12
	3.			–
XCH A, direct			A X direct	
Kód:	1. 1100 0101	C5	Výměna obsahu přímo adresovaného paměťového místa v RWM s obsahem střadače	2/1
	2. direct			12
	3.			–
XCH A, @ Ri			A X (Ri)	
Kód:	1. 1100 011 i	C6-7	Výměna obsahu nepřímo adresovaného paměťového místa v RWM s obsahem střadače	1/1
	2.			12
	3.			–
XCHD A, @ Ri			A_L X (Ri_L)	
Kód:	1. 1101 011 i	D6-7	Výměna čtyř nižších bitů nepřímo adresovaného paměť. místa v RWM se čtyřmi nižšími bity střadače	1/1
	2.			12
	3.			–

Tab. 28 (pokračování)

Instrukce s Booleovými proměnnými				
CLR C			$C \leftarrow 0$	
Kód:	1. 1100 0011		Nulování příznaku C	1/1
	2.	C3		12
	3.			C
CLR bit			$\text{bit} \leftarrow 0$	
Kód:	1. 1100 0010	C2	Nulování bitu bitově adresovaného paměťového místa v RWM	2/1
	2. bit adr.			12
	3.			–
SETB C			$C \leftarrow 1$	
Kód:	1. 1101 0011	D3	Nastavení příznaku C	1/1
	2.			12
	3.			C
SETB bit			$\text{bit} \leftarrow 1$	
Kód:	1. 1101 0010	D2	Nastavení bitu bitově adresovaného paměťového místa v RWM	2/1
	2.			12
	3.			–
CPL C			$C \leftarrow \bar{C}$	
Kód:	1. 1011 0011	B3	Negace příznaku C	1/1
	2.			12
	3.			C
CPL bit			$\text{bit} \leftarrow \bar{\text{bit}}$	
Kód:	1. 1011 0010	B2	Negace bitu bitově adresovaného paměťového místa v RWM	2/1
	2.			12
	3.			–
ANL C, bit			$C \leftarrow C \cap \text{bit}$	
Kód:	1. 1000 0010	82	Logický součin příznaku C s bitem bitově adresovaného paměťového místa v RWM	2/2
	2. bit			24
	3.			C
ANL C, / bit			$C \leftarrow C \cap \text{bit}_-$	
Kód:	1. 1011 0000	B0	Logický součin příznaku C s negovaným bitem bitově adresovaného paměťového místa v RWM	24
	2. bit			
	3.			
ORL C, bit			$C \leftarrow C \cup \text{bit}$	
Kód:	1. 0111 0010	72	Logický součet příznaku C s bitem bitově adresovaného paměťového místa v RWM	2/2
	2. bit adr.			24
	3.			C
ORL C, / bit			$C \leftarrow C \cup \text{bit}_-$	
Kód:	1. 1010 0000	A0	Logický součet příznaku C s negovaným bitem bitově adresovaného paměťového místa v RWM	2/2
	2. bit adr.			24
	3.			C

Tab. 28 (pokračování)

MOV C, bit			C ← bit	
Kód:	1. 1010 0010	A2	Přesun bitu bitově adresovaného paměťového místa v RWM do příznaku C	2/1
	2.			12
	3.			C
MOV bit, C			bit ← C	
Kód:	1. 1001 0010	92	Přesun příznaku C na bitově adresované paměťové místo v RWM	2/2
	2. bit adr.			24
	3.			–
JC rel			PC ← PC+2+rel	
Kód:	1. 0100 0000	40	Skok, je-li příznak C je nastaven	2/2
	2. rel			24
	3.			–

Instrukce pro větvení programu				
ACALL addr 11			PC ₁₀₋₀ ← adr11 {SP+1 ← PC ₇₋₀ SP+2 ← PC ₁₅₋₈ }	
Kód:	1. xxx10001	1-F1	Nepodmínené volání podprogramu uvnitř 2kB adresového prostoru	2/2
	2. adr ₇₋₀			24
	3.			–
LCALL addr 16			PC ← adr16 {SP+1 ← PC ₇₋₀ SP+2 ← PC ₁₅₋₈ }	
Kód:	1. 0001 0010	12	Volání podprogramu uvnitř 64kB adresového prostoru	3/2
	2. adr ₁₅₋₈			24
	3. adr ₇₋₀			–
RET			PC ₁₅₋₈ ← SP PC ₇₋₀ ← SP-1 {SP ← SP-2}	
Kód:	1. 0010 0010	22	Návrat z podprogramu	1/2
	2.			24
	3.			–
RETI			PC ₁₅₋₈ ← SP PC ₇₋₀ ← SP-1 {SP ← SP-2}	
Kód:	1. 0011 0010	32	Návrat z obslužného podprogramu přerušení	1/2
	2.			24
	3.			–
AJMP addr 11			PC ₁₀₋₀ ← adr11	
Kód:	1. xxx00001	2-E1	Skok uvnitř 2kB adresového prostoru	2/2
	2. adr ₇₋₀			24
	3.			–
LJMP addr 16			PC ← adr16	
Kód:	1. 0000 0010	02	Dlouhý skok uvnitř 64kB adresového prostoru	3/2
	2. adr ₁₅₋₈			24
	3. adr ₇₋₀			–
SJMP rel			PC ← PC + rel	
Kód:	1. 1000 0000	80	Krátký skok na relativní adresu	2/2
	2. rel			24
	3.			–

Tab. 28 (pokračování)

JMP @ A + DPTR			$PC \leftarrow (A + DPTR)$	
Kód:	1. 0111 0011	73	Skok na adresu, která je určena sčítáním obsahu střadače a DPTR	1/2
	2.			24
	3.			-
JZ rel			$PC \leftarrow PC + 2 + rel$	
Kód:	1. 0110 000	60	Skok, je-li je obsah střadače nulový	2/2
	2. rel			24
	3.			-

JNZ rel			$PC \leftarrow PC + 2 + rel$	
Kód:	1. 0111 000	70	Skok, je-li obsah střadače nenulový nulový	2/2
	2. rel			24
	3.			-
CJNE A, direct, rel			$PC \leftarrow PC + 3 + rel \{C \leftarrow 1 \text{ pre } (A) \langle \text{direct} \rangle\}$	
Kód:	1. 10110101	B5	Porovnání obsahu přímo adresovaného paměťového místa v RWM a skok s obsahem střadače a skok, když nejsou shodné	3/2
	2. direct			24
	3. rel			C
CJNE A, # data, rel			$PC \leftarrow PC + 3 + rel \{C \leftarrow 1 \text{ pre } (A) \langle \text{data} \rangle\}$	
Kód:	1. 1011 0100	B4	Porovnání konstanty s obsahem střadače a skok, když nejsou shodné	3/2
	2. data			24
	3. rel			C
CJNE Rn, # data, rel			$PC \leftarrow PC + 3 + rel \{C \leftarrow 1 \text{ pre } (Rn) \langle \text{data} \rangle\}$	
Kód:	1. 1011 1 r r r	B8-F	Porovnání konstanty s obsahem registru Rn a skok, když nejsou shodné	3/2
	2. data			24
	3. rel			C
CJNE @ Ri, # data, rel			$PC \leftarrow PC + 3 + rel \{C \leftarrow 1 \text{ pre } (Ri) \langle \text{data} \rangle\}$	
Kód:	1. 1011 011 i	B6-7	Porovnání konstanty s obsahem nepřímo adresovaného paměťového místa v RWM a skok, když nejsou shodné	3/2
	2. data			24
	3. rel			C

Tab. 28 (pokračování)

DJNZ Rn, rel			$Rn \leftarrow Rn - 1 \quad \{\text{pre } Rn < > 0 \quad PC \leftarrow PC + 2 + \text{rel}\}$	
Kód:	1. 1101 1 r r r	D8-F	Dekrementace obsahu Rn a skok, jestli obsah Rn není nulový	2/2
	2. rel			24
	3.			-
DJNZ direct, rel			$\text{direct} \leftarrow \text{direct} - 1 \quad \{\text{pre } \text{direct} < > 0 \quad PC \leftarrow PC + 2 + \text{rel}\}$	
Kód:	1. 1101 0101	D5	Dekrementace obsahu přímo adresovaného paměťového místa v RWM a skok, když obsah není nulový	3/2
	2. direct			24
	3. rel			-
NOP			$PC \leftarrow PC + 1$	
Kód:	1. 0000 0000	00	Prázdná instrukce	1/1
	2.			12
	3.			-

Tab. 28 (pokračování)

Hex	Byty	Mnem	Operandy	Hex	Byty	Mnem	Operandy
00	1	NOP		2B	1	ADD	A,R3
01	2	AJMP	kod adr	2C	1	ADD	A,R4
02	3	LJMP	kod adr	2D	1	ADD	A,R5
03	1	RR	A	2E	1	ADD	A,R6
04	1	INC	A	2F	1	ADD	A,R7
05	2	INC	data adr	30	3	JNB	bit, kod adr
06	1	INC	@R0	31	2	ACALL	kod adr
07	1	INC	@R1	32	1	RETI	
08	1	INC	R0	33	1	RLC	A
09	1	INC	R1	34	2	ADDC	A, #data
0A	1	INC	R2	35	2	ADDC	A, data adr
0B	1	INC	R3	36	1	ADDC	A,@R0
0C	1	INC	R4	37	1	ADDC	A,@R1
0D	1	INC	R5	38	1	ADDC	A,R0
0E	1	INC	R6	39	1	ADDC	A,R1
0F	1	INC	R7	3A	1	ADDC	A,R2
10	3	JBC	bit, kod adr	3B	1	ADDC	A,R3
11	2	ACALL	kod adr	3C	1	ADDC	A,R4
12	3	LCALL	kod adr	3D	1	ADDC	A,R5
13	1	RRC	A	3E	1	ADDC	A,R6
14	1	DEC	A	3F	1	ADDC	A,R7
15	2	DEC	data adr	40	2	JC	kod adr
16	1	DEC	@R0	41	2	AJMP	kod adr
17	1	DEC	@R1	42	2	ORL	data adr,A
18	1	DEC	R0	43	3	ORL	data adr,#data
19	1	DEC	R1	44	2	ORL	A, #data
1A	1	DEC	R2	45	2	ORL	A, data adr
1B	1	DEC	R3	46	1	ORL	A,@R0
1C	1	DEC	R4	47	1	ORL	A,@R1
1D	1	DEC	R5	48	1	ORL	A,R0
1E	1	DEC	R6	49	1	ORL	A,R1
1F	1	DEC	R7	4A	1	ORL	A,R2
20	3	JB	bit, kod adr	4B	1	ORL	A,R3
21	2	AJMP	kod adr	4C	1	ORL	A,R4
22	1	RET		4D	1	ORL	A,R5
23	1	RL	A	4E	1	ORL	A,R6
24	2	ADD	A, #data	4F	1	ORL	A,R7
25	2	ADD	A, data adr	50	2	JNC	kod adr
26	1	ADD	A,@R0	51	2	ACALL	kod adr
27	1	ADD	A,@R1	52	2	ANL	data adr,A
28	1	ADD	A,R0	53	3	ANL	data adr,#data
29	1	ADD	A,R1	54	2	ANL	A, #data
2A	1	ADD	A,R2	55	2	ANL	A, data adr

Tab. 29 Disassembler instrukcí

Hex	Byty	Mnem	Operandy	Hex	Byty	Mnem	Operandy
56	1	ANL	A,@R0	81	2	AJMP	kod adr
57	1	ANL	A,@R1	82	2	ANL	C, bit adr
58	1	ANL	A,R0	83	1	MOVC	A,@A+PC
59	1	ANL	A,R1	84	1	DIV	AB
5A	1	ANL	A,R2	85	3	MOV	dataadr,data adr
5B	1	ANL	A,R3	86	2	MOV	data adr, @R0
5C	1	ANL	A,R4	87	2	MOV	data adr, @R1
5D	1	ANL	A,R5	88	2	MOV	data adr, R0
5E	1	ANL	A,R6	89	2	MOV	data adr, R1
5F	1	ANL	A,R7	8A	2	MOV	data adr, R2
60	2	JZ	kod adr	8B	2	MOV	data adr, R3
61	2	AJMP	kod adr	8C	2	MOV	data adr, R4
62	2	XRL	data adr,A	8D	2	MOV	data adr, R5
63	3	XRL	data adr,#data	8E	2	MOV	data adr, R6
64	2	XRL	A, #data	8F	2	MOV	data adr, R7
65	2	XRL	A, data adr	90	3	MOV	DPTR, #data16
66	1	XRL	A,@R0	91	2	ACALL	kod adr
67	1	XRL	A,@R1	92	2	MOV	bit adr, C
68	1	XRL	A,R0	93	1	MOVC	A,@A+DPTR
69	1	XRL	A,R1	94	2	SUBB	A, #data
6A	1	XRL	A,R2	95	2	SUBB	A, data adr
6B	1	XRL	A,R3	96	1	SUBB	A,@R0
6C	1	XRL	A,R4	97	1	SUBB	A,@R1
6D	1	XRL	A,R5	98	1	SUBB	A,R0
6E	1	XRL	A,R6	99	1	SUBB	A,R1
6F	1	XRL	A,R7	9A	1	SUBB	A,R2
70	2	JNZ	kod adr	9B	1	SUBB	A,R3
71	2	ACALL	kod adr	9C	1	SUBB	A,R4
72	2	ORL	C, bit adr	9D	1	SUBB	A,R5
73	1	JMP	@A+DPTR	9E	1	SUBB	A,R6
74	2	MOV	A, #data	9F	1	SUBB	A,R7
75	3	MOV	data adr, #data	A0	2	ORL	C,/bit adr
76	2	MOV	@R0, #data	A1	2	AJMP	kod adr
77	2	MOV	@R1, #data	A2	2	MOV	C,bit adr
78	2	MOV	R0, #data	A3	1	INC	DPTR
79	2	MOV	R1, #data	A4	1	MUL	AB
7A	2	MOV	R2, #data	A5		rezerva	
7B	2	MOV	R3, #data	A6	2	MOV	@R0, data adr
7C	2	MOV	R4, #data	A7	2	MOV	@R1, data adr
7D	2	MOV	R5, #data	A8	2	MOV	R0, data adr
7E	2	MOV	R6, #data	A9	2	MOV	R1, data adr
7F	2	MOV	R7, #data	AA	2	MOV	R2, data adr
80	2	SJMP	kod adr	AB	2	MOV	R3, data adr

Tab. 29 (pokračování)

Hex	Byty	Mnem	Operandy	Hex	Byty	Mnem	Operandy
AC	2	MOV	R4, data adr	D7	1	XCHD	A,@R1
AD	2	MOV	R5, data adr	D8	2	DJNZ	R0,A
AE	2	MOV	R6, data adr	D9	2	DJNZ	R1,A
AF	2	MOV	R7, data adr	DA	2	DJNZ	R2,A
B0	2	ANL	C, bit adr	DB	2	DJNZ	R3,A
B1	2	ACALL	kod adr	DC	2	DJNZ	R4,A
B2	2	CPL	bit adr	DD	2	DJNZ	R5,A
B3	1	CPL	C	DE	2	DJNZ	R6,A
B4	3	CJNE	A, #data, k. adr	DF	2	DJNZ	R7,A
B5	3	CJNE	A,data adr, k. adr	E0	1	MOVX	A,@DPTR
B6	3	CJNE	@R0,#dat, k. adr	E1	2	AJMP	kod adr
B7	3	CJNE	@R1,#dat, k. adr	E2	1	MOVX	A,@R0
B8	3	CJNE	R0,#data,kod adr	E3	1	MOVX	A,@R1
B9	3	CJNE	R1,#data,kod adr	E4	1	CLR	A
BA	3	CJNE	R2,#data,kod adr	E5	2	MOV	A,data adr
BB	3	CJNE	R3,#data,kod adr	E6	1	MOV	A,@R0
BC	3	CJNE	R4,#data,kod adr	E7	1	MOV	A,@R1
BD	3	CJNE	R5,#data,kod adr	E8	1	MOV	A,R0
BE	3	CJNE	R6,#data,kod adr	E9	1	MOV	A,R1
BF	3	CJNE	R7,#data,kod adr	EA	1	MOV	A,R2
C0	2	PUSH	data adr	EB	1	MOV	A,R3
C1	2	AJMP	kod adr	EC	1	MOV	A,R4
C2	2	CRL	bit adr	ED	1	MOV	A,R5
C3	1	CRL	C	EE	1	MOV	A,R6
C4	1	SWAP	A	EF	1	MOV	A,R7
C5	2	XCH	A, data adr	F0	1	MOVX	@DPTR,A
C6	1	XCH	A,@R0	F1	2	ACALL	kod adr
C7	1	XCH	A,@R1	F2	1	MOVX	@R0,A
C8	1	XCH	A,R0	F3	1	MOVX	@R1,A
C9	1	XCH	A,R1	F4	1	CPL	A
CA	1	XCH	A,R2	F5	2	MOV	data adr,A
CB	1	XCH	A,R3	F6	1	MOV	@R0,A
CC	1	XCH	A,R4	F7	1	MOV	@R1,A
CD	1	XCH	A,R5	F8	1	MOV	R0,A
CE	1	XCH	A,R6	F9	1	MOV	R1,A
CF	1	XCH	A,R7	FA	1	MOV	R2,A
D0	2	POP	data adr	FB	1	MOV	R3,A
D1	2	ACALL	kod adr	FC	1	MOV	R4,A
D2	2	SETB	bit adr	FD	1	MOV	R5,A
D3	1	SETB	C	FE	1	MOV	R6,A
D4	1	DA	A	FF	1	MOV	R7,A
D5	3	DJNZ	data adr, kod adr				
D6	1	XCHD	A,@R0				

Tab. 29 (pokračování)

OBECNÉ POJMY

Před deseti či patnácti lety vývojáři v oblasti mikroprocesorové techniky měli značné obtíže při nasazování mikroprocesorů v praxi. Spousty z nich pracně shánělo základní softwarové vybavení tj. překladač zdrojového kódu, či relokátor (INTEL, Avocet). V oblasti hardwarového prostředí byla situace ještě komplikovanější, protože jen málo pracovišť bylo vybaveno komerčními vývojovými prostředky. Nedostatek financí se obcházel pracovním budováním vlastních pomůcek, ku příkladu simulátorů paměti a programátorů EPROM.

HARDWAROVÉ PROSTŘEDÍ

V současné době je situace poněkud odlišná v několika směrech. Pořízení editorů, překladačů, relokátorů, softwarových simulátorů je často bezplatné. Jednak výrobci mikroprocesorů a mikrokontrolérů nabízejí levné pořízení „starter kitů“, v aplikační oblasti je mnoho subjektů nabízejících různá řešení vývojových a konstrukčních pomůcek, až se někdy zdá, že je jich více, nežli uživatelů. Stačí si „zasurfovat“ po Internetu a nevíte, co si vybrat.

Ať používáte jakékoliv hardwarové prostředky, při přechodu na mikrokontroléry ATMEL nevzniknou žádné podstatné překážky. V případě použití mikrokontrolérů AT. 53, 55, 252 může být nepříjemné, že některé SFR váš překladač nezná a musíte proto použít přímou adresaci, nebo je deklarovat přiřazením EQU.

5 SOFTWAREVÉ PROSTŘEDKY A LADĚNÍ PROGRAMU

PROGRAMOVACÍ TECHNIKY

Dnešní mikrokontroléry jsou tak komplexně hardwarově vybaveny, že nezřídka vznikají aplikace s minimálním dodatečným obvodovým doplňkem, a proto se veškerá inteligence a výkonnost přenesla do software.

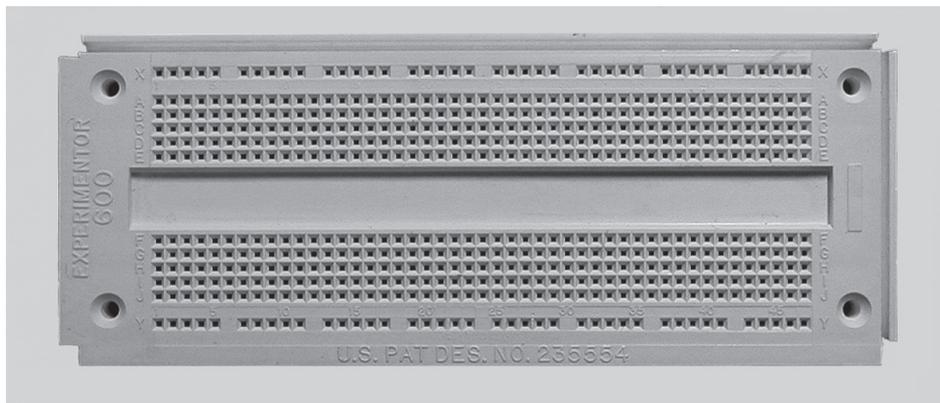
Při realizaci programu je nutné znát architekturu mikrokontroléru, možnosti jeho instrukční sady, ale i hardwarově-sofwarového vybavení, na kterém budeme aplikaci vyvíjet. Nutným předpokladem je však důkladná znalost požadavků kladených na řešení. Tak se popis v zásadě dá popsat v několika bodech:

1. Důsledná analýza požadované aplikace, návrh řídicích a signálových rozhraní, návrh algoritmů, průzkum knižních modulů, pokud to jde i studium konkurenčních řešení nebo vlastností.
2. Realizace spolehlivého zkušebního zapojení aplikace, v případě rozsáhlejších aplikací, po částech. Pomoc zkušeného obvodáře je k nezaplacení.
3. Zapsání logické kostry programu, rozdělení do menších celků, realizace podprogramu a programových modulů.
4. Vybrat vhodný programovací jazyk, pokud to jde:
 - Assembler – jazyk symbolických adres – vhodný především pro vývoj aplikací vyžadujících rychlost při menších nárocích na paměť programu i dat.
 - Vyšší programovací jazyk (C++, Basic), jež umožňují rychlejší vývoj aplikace a přehledný vývoj náročnějších aplikací.
5. Ladění pozůstává v zásadě z opakujících se překladů zdrojových textů do výkonného modulu ve strojovém tvaru, zavedení do emulační paměti, spuštění v aplikaci a sledování nabývajících vlastností aplikace.
6. Je-li aplikační program odladen a aplikace vykonává činnost podle zadání, je přeložen hotový program ve strojovém tvaru (formát .HEX, .IH, .BIN) zaveden do paměti programu, v našem případě interní FLASH EPROM, a mikrokontrolér je definitivně umístěn do aplikace.
7. Posledním krokem, ne však významem, je finalizace konstrukčního řešení celé aplikace.

ZKUŠEBNÍ OVĚŘOVÁNÍ ZAPOJENÍ

Podobně jako při práci s analogovými a číslicovými obvody, je velice důležité vyzkoušet si také zapojení navržená s mikrokontroléry v reálné činnosti. Mnozí zřejmě mají tendenci vyzkoušet svá řešení na univerzálních zkušebních deskách plošných spojů pájením součástek. Je to tradiční spolehlivá metoda, i když je možné použít nespájivá kontaktní pole,

kteřá jsou v současnosti běžně dostupná (viz obr. 61). Metodu preferují hlavně vzhledem k tomu, že obvody s mikrokontroléry jsou počtem spojů mnohdy dosti složité a je tu nedůvěra ve spolehlivost přítlačných kontaktů. To je pochopitelné, neboť značná část problémů v elektronice je způsobena hlavně špatnými kontakty.



Obr. 61 Bezkontaktní zkušební deska EXPERIMENTOR 600

Zde je na místě nutno poznamenat, že na rozdíl od některých autorů, kteří nejsou nakloněni těmto pomůckám, s ohledem na spolehlivost a cenu, máme velice dobré cca 15leté zkušenosti s pravidelným používáním těchto kontaktních nepájivých polí. Odzkoušeny byly na nich veškeré naše navržené elektronická zapojení a to různé složitosti, počínaje analogovými až po obvody s mikrokontroléry.

Je nutno konstatovat, že práce s těmito pomůckami je velice praktická vzhledem k tomu, že jakákoliv změna v obvodovém zapojení se realizuje velice jednoduše a rychle. Stačí mít k dispozici izolované tenké vodiče na koncích bez izolace, které pouhým zasunutím do dutinky vytvoří přepojení. Je to veliký rozdíl oproti práci s pájkou, kde není jednoduché v hustější spleti vodičů pájet a nezpůsobit nechtěný spoj. Také je důležité, že při pouhém zasouvání se šetří součástky, které zůstávají stále jako nové. Takzvané studené spoje nejsou problémem a zkoušené zapojení jsou funkční i po velice dlouhé době. K ceně pouze tolik, že ani slušná univerzální deska plošných spojů není levnou záležitostí, pokud si ji neuděláme sami (a i to něco stojí). Jako u všeho i zde platí – důležitá je kvalita výrobku.

Seznam obrázků

	str
1. Blokové schéma mikrokontrolérů ATMEL	7
2. Architektura paměti	8
3. Interní datová paměť RAM	9
4. Instrukční cyklus	15
5. Struktura jednotlivých portů mikrokontroléru	17
6. Operace na V/V bráně	18
7. Struktura zatěžovacích rezistorů jednotlivých vývodů portů P1 a P3	19
8. Ochrana vstupů mikrokontrolérů	20
9. Mikrokontrolér s externí pamětí programu	20
10. Časování při výkonu programu z externí paměti	21
11. Časování při čtení z externí paměti programu	22
12. Mikrokontrolér s externí pamětí dat	22
13. Časování při zápisu do externí paměti dat	23
14. Časování při čtení z externí paměti dat	23
15. Připojení sériové EEPROM v kaskádě	25
16. Připojení sériové EEPROM 3vodičové, 4vodičové	25
17. Připojení paralelní EEPROM	27
18. Režim 0 pro čítače 0 a 1	29
19. Režim 1 pro čítače 0 a 1	31
20. Režim 2 pro čítače 0 a 1	31
21. Režim 3 pro čítače 0 a 1	32
22. Záchytný režim časovače 2	33
23. Režim s automatickým přednastavením – čítá jen vpřed (DCEN = 0)	34
24. Režim s automatickým přednastavením – čítá vpřed i vzad (DCEN = 1)	34
25. Režim generátoru přenosové rychlosti	34
26. Sériový kanál režim 0	36
27. Sériový kanál režim 1	38
28. Sériový kanál režim 2	40
29. Sériový kanál režim 3	41
30. Zdroje přerušení	46
31. Zdroje přerušení dle priority	46

32.	Průběh odezvy na přerušení	48
33.	Reset – časování	50
34.	Resetovací obvod	52
35.	Způsoby zálohování napájení	53
36.	Hlídací obvod podpětí	53
37.	Režimy IDLE a POWER DOWN	54
38.	Zapojení oscilátoru	56
39.	Externí oscilátor	57
40.	Oscilátorové obvody	57
41.	Pouzďení mikrokontroléru AT89C51	60
42.	Blokové schéma mikrokontroléru AT89C51	61
43.	Schéma programování AT89C51	65
44.	Schéma verifikace AT89C51	65
45.	Průběhy signálů při programování AT89C51 (12V)	66
46.	Průběhy signálů při programování AT89C51 (5V)	67
47.	Pouzďení AT89C1051, AT89C2051, AT89C4051	72
48.	Blokové schéma AT89C1051, AT89C2051, AT89C4051	73
49.	Mapa SFR AT89C1051, AT89C2051, AT89C4051	74
50.	Schéma programování AT89C1051, AT89C2051, AT89C4051	76
51.	Schéma verifikace AT89C1051, AT89C2051, AT89C4051	76
52.	Průběhy programování a verifikace AT89C1051, AT89C2051, AT89C4051	76
53.	Pouzďení AT89C52, AT89S53, AT89C55, AT89S8252	80
54.	Schéma paralelního programování FLASH EPROM	83
55.	Schéma verifikace FLASH EPROM	83
56.	Schéma sériového programování AT89S8252	83
57.	SPI – blokové schéma	88
58.	SPI – Master-Slave propojení	88
59.	SPI – formát přenosu pro CPHA=0	89
60.	SPI – formát přenosu pro CPHA=1	89
61.	Bezkontaktní zkušební deska EXPERIMENTOR 600	112

Seznam tabulek

	str
1. Nižších 128 bytů interní RAM	10
2. Mapa SFR	11
3. Banky pracovních registrů	14
4. Port P3	17
5. Typy sériových a paralelních EEPROM pamětí	26
6. Režimy čítačů/časovačů	29
7. Režimy sériového kanálu	42
8. Přenosové rychlosti	44
9. Priority přerušení	47
10. Vektory přerušení	49
11. Registry SFR po inicializaci	51
12. Značení pouzder	63
13. Režimy ochrany mikrokontrolérů	64
14. Režimy programování AT89C51	64
15. Parametry programování AT89C51	67
16. Mezní parametry AT89C51	67
17. Statické vlastnosti AT89C51	68
18. Dynamické vlastnosti AT89C51	69
19. Značení pouzder AT89C51	70
20. Značení pouzder AT89C1051, AT89C2051, AT89C4051	71
21. Režimy programování AT89C1051, AT89C2051, AT89C4051	77
22. Parametry programování AT89C1051, AT89C2051, AT89C4051	77
23. Značení pouzder AT89C52, AT89S53, AT89C55, AT89S8252	79
24. Módy paralelního programování pro FLASH a EEPROM AT89S8252	82
25. Instrukční sada sériového programování	84
26. Nastavení hlídacího časovače	85
27. Nastavení dávkového přenosu SPI	87
28. Instrukce mikrokontrolérů ATMEL	93
29. Disassembler instrukcí	106

REJSTŘÍK

- A**
- adresa 12
 - adresace 13
 - bitová 9
 - indexová 13
 - nepřímá 13
 - přímá 13
 - registrová 13
 - specifikovaného registru 13
 - adresový prostor 8, 9, 10
 - algoritmus 64, 74
 - architektura 7
 - mikrokontrolérů 7
 - paměti 8
- B**
- banka registrů 9, 14
 - bit 9
 - budič 16
 - byt 13, 15
- C**
- CPU 8, 14, 55
 - cyklus 15
 - instrukční 15
 - strojový 15
- Č**
- čítač instrukcí (PC) 12, 14
 - čítač/časovač 13, 28
- D**
- data 16, 21
 - Data Polling 65, 75
 - D/A převodník 5
- E**
- EA 11, 21, 63
 - EEPROM 24, 26, 53
- F**
- fáze 15
 - FLASH 8, 21, 59, 63, 74
 - formát 84, 89
- G**
- GATE 29
- H**
- hodinový generátor 56
 - hradlo 16
- I**
- index 13
 - inicializace 50
 - instrukce 9, 12, 15, 91
 - aritmetické 91
 - logické 91
 - pro přenos dat 91
 - pro větvení 91
 - instrukční sada 91
- J**
- jazyk 111
 - programovací 111
 - symbolických adres 111
 - jednotka 15
 - aritmeticko-logická (ALU) 14
 - řídící 15
- K**
- kanál 12, 35, 42
 - sériový 12, 35, 42
 - kmitočet 52, 67
 - krokování 49
- M**
- maskování 45, 47
 - mezní parametry 67

mikroprocesor 5, 52, 78, 109
mód 13, 64, 71
multiplex 20, 62

N

napájení 52, 54

O

obsluha přerušení 47, 48
obslužný podprogram 45, 49
oscilátor hodinový 56

P

paměť
 dat 8
 externí 20
 interní 9, 13
 programu 8
parametry programování 67, 77
port 16
pouzďení 60, 72, 80
programování paměti FLASH 63, 74, 83
přenosová rychlost 18, 28, 31, 34, 43
přerušení 29, 45, 48
příznak 14, 30, 42, 55
PSEN 8, 16, 63

R

RAM 6, 21, 59, 73, 78
registr 9, 13, 14, 51
RESET 19, 50, 52
režim časovače 29, 30, 32

Ř

řídící signály 16, 64, 74

S

sběrnice 7, 62
 adresová 62
Serial Data Buffer (SBUF) 12, 35, 51

snížený příkon 54
 IDLE 54
 POWER DOWN 54
Special Function Registers (SFR) 9, 11, 51
SPI 6, 7, 17, 86
Stack Pointer (SP) 12, 15, 48, 51
stavové slovo (PSW) 12, 14
střadač 12, 13, 92

T

TIMER 7, 13, 85
typy kontrolérů 59
typy mikrokontrolérů 6

U

ukazatel dat (DPTR) 12, 15, 51
uzamykací bit 63, 75

V

vektory přerušení 49
verifikace 62, 65, 76, 83
vstupy 16, 19

W

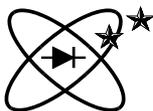
Watch Dog (WDT) 7, 54, 85

Z

zálohování 52
zdroje přerušení 47

LITERATURA

- [1] Levický D. a kol.: Jednočipové mikropočítače radu 8051, ČSVTS pri EF VŠT v Košiciach 1988
- [2] ATMEL: Technické listy k mikroprocesorom Atmel, 2000
- [3] Sobotka Z.: Otázky a odpovede z mikropočítačov APLIKÁCIE, ALFA Bratislava 1988
- [4] Sobotka Z.: Otázky a odpovede z mikropočítačov ARCHITEKTÚRA A PROGRAMOVANIE, ALFA Bratislava 1983
- [5] Starý J.: Mikropočítač a jeho programování, SNTL Praha 1984
- [6] Valášek P.: Monolitické mikroprocesory a mikropočítače, SNTL Praha 1989
- [7] Kočiš I., Šulko I.: Mikroprocesory a mikropočítače, ALFA Bratislava 1986
- [8] Hrbáček J.: Mikrořadiče PIC16CXX a vývojový kit PICSTART, BEN Praha 1998
- [9] Hrbáček J.: Programování mikrokontrolérů PIC16CXX, BEN Praha 1998
- [10] Skalický P.: Mikroprocesory Řady 8051, BEN Praha 1998
- [11] Burger I.: Stykové obvody mikropočítačov, ALFA Bratislava 1990
- [12] Dědina B., Valášek P.: Mikroprocesory a mikropočítače, SNTL Praha 1983



Elektronik - zásilková služba, 262 21 Obecnice 318
tel. i fax. (0306) 635 650, 21 963
zs@elektro-obecnice.cz

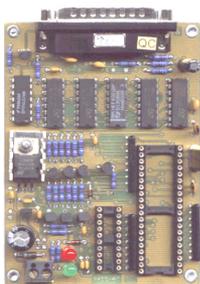
Programátory pamětí a mikroprocesorů firmy Atmel

Programátory jsou koncipovány jako levné kity bez krytování a vlastního napájecího zdroje, připojitelné k počítači standardu PC-AT přes rozhraní CENTRONICS. Na deskách plošných spojů programátorů jsou umístěny precizní patice pro vkládání programovaných obvodů. Dále je na desce umístěna svorkovnice pro připojení externího nestabilizovaného napájecího zdroje a konektor CANNON pro připojení prodlužovacího kabelu tiskárny.

P-89C5X

P-89C5X je programátor jednočipových mikroprocesorů standardu stavebnice MCS-51 typu AT89C1051, AT89C2051, AT89C4051, AT89C51, AT89C52 a AT89C55. Součástí programátoru je obslužný program na disketě, který umožňuje programování obvodů ze souboru v binárním tvaru, čtení neutajeného obsahu obvodu do souboru v binárním tvaru, mazání obvodu, verifikaci, čtení signatury, několikastupňové uzamčení obvodu a editaci.

cena:



980,- Kč bez DPH

P-24CEX

Programátor obvodů EEPROM 24C02, 24C04, 24C08, 24C16, 24C32 a 24C64. Na desce plošného spoje programátoru je umístěna precizní patice DIP8 pro vkládání programovaných obvodů. Připojuje se k externímu nestabilizovanému napájecímu zdroji 7 až 15V DC a konektoru CANNON pro připojení prodlužovacího kabelu tiskárny. Programátor má současně vyvedenou programovací sběrnici I²C na pájecí plošky tak, aby bylo možno programovat i obvody umístěné v jiném přístroji.



286,- Kč bez DPH

STARTÉR - KIT

Stavebnice startér kit je určen pro všechny ty, pokročilé zájemce v mikroprocesorové technice, kteří chtějí začít s aplikacemi jednočipových mikroprocesorů ATMELE AT89C2051. Umožňuje ověřování vlastních programů a zapojení. P-START.KIT 1 obsahuje všechny nezbytné obvody pro samostatnou činnost. Součástí plošného spoje je univerzální pájecí pole, na jehož okraj jsou vyvedeny vývody procesoru.

obj.název	popis	cena vč.DPH
P-START.KIT 1	Základní část	605.90
P-START.KIT 2	Blok výstupních relé	179.30
P-START.KIT 3	Blok vstupů	90.60
P-START.KIT 4	Blok RS232	87.20
P-START.KIT 5	Blok EEPROM	30.00
P-START.KIT 6	Blok tlačítek	20.00
(WEB-B8)	Krabička	186.70

P-START.KIT 2 až P-START.KIT 6 obsahuje jen polovodičové součástky, které se letují přímo do základní části P-START.KIT 1.

P-89C5XP

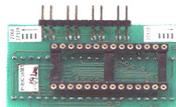


Programátor P-89C5XP je plně totožný s programátorem P-89C5X a navíc místo precizní patice s nulovou silou, kterou ocení ti, kdo programátor budou častěji používat. Programování je jak 5-ti voltové verzi tak také ve 12-ti voltové. Napájecí zdroj a propojovací kabel není součástí programátorů.

1346,- Kč bez DPH

P-89C5M1

Adaptér je určen pro programátor P-89C5X nebo P-89C5XP. Pomocí tohoto adaptéru lze programovat paměti typu 27C64, 27C128, 27C256 a 27C512. Adaptér se vkládá do objímky DIL 40.



Ke všem programátorům si lze doobjednat patřičné obvody a katalogové listy (jen v angličtině). Ceny sdělíme telefonicky.

<http://www.elektro-obecnice.cz>

Připravujeme programátor I2C Bus EEPROM - 24C01 - 24C512 - Siemens EEPROM - SDE2516, SDE2526, SDA2546, SDA2586, SDA3546, SDA3586, SDE 2506 - Atmel I2C EEPROM - AT17C65, AT17C128, AT17C256, AT17C512, AT17C010 - Atmel AVR (AT90S1200 ? AT89C8535) včetně Flash micro - Atmel AVR ATmega - ATmega603, ATmega103, ATmega161, ATmega163, ATTiny12, ATTiny15 - Atmel x51 programovatelné přes ISP - AT89S8252 a AT89S53 - Microwire EEPROM - 93C06, 93C46, 93C56, 93C66, 93C76, 93C86 - Microchip - PIC 16C84/16F84 , 16F873/874/876/877, 12C508/509 - SPI EEPROM - 25010, 25020, 25040, 25080, 25160, 25320, 25640, 25128, 25256. Předbežná cena osazené desky bez zdroje: 942,-Kč bez DPH.

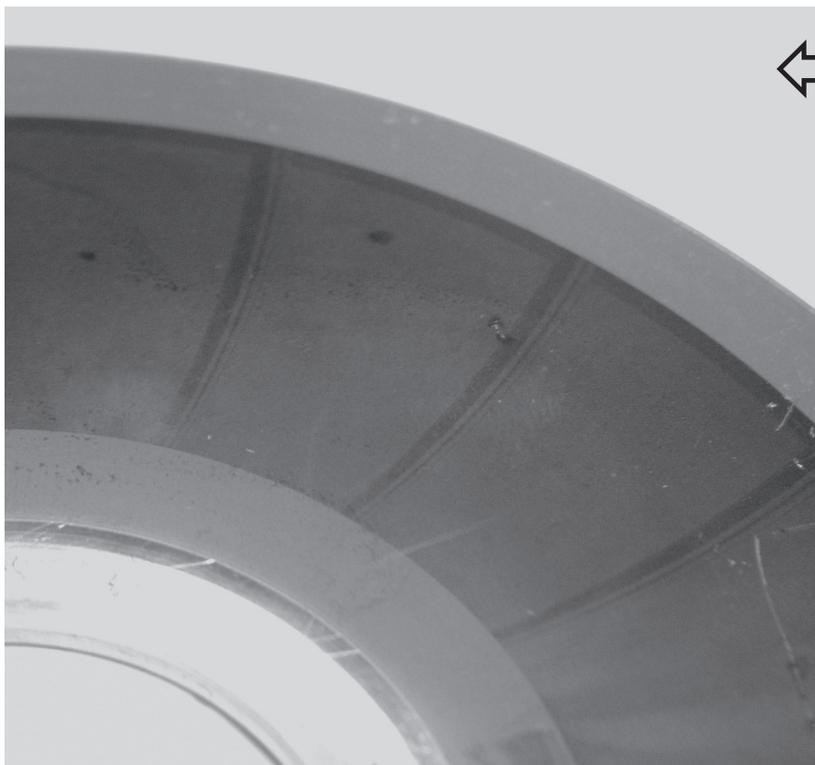
Naší hlavní náplní je prodej aktivních a pasivních elektrosoučástek, programátorů Atmel, odborné literatury, elektronických stavebnic a modulů, měřící techniky a nářadí pro elektrotechniku. V současné době zajišťujeme distribuci čítající cca 140 000 položek.

Rozeberte si PC

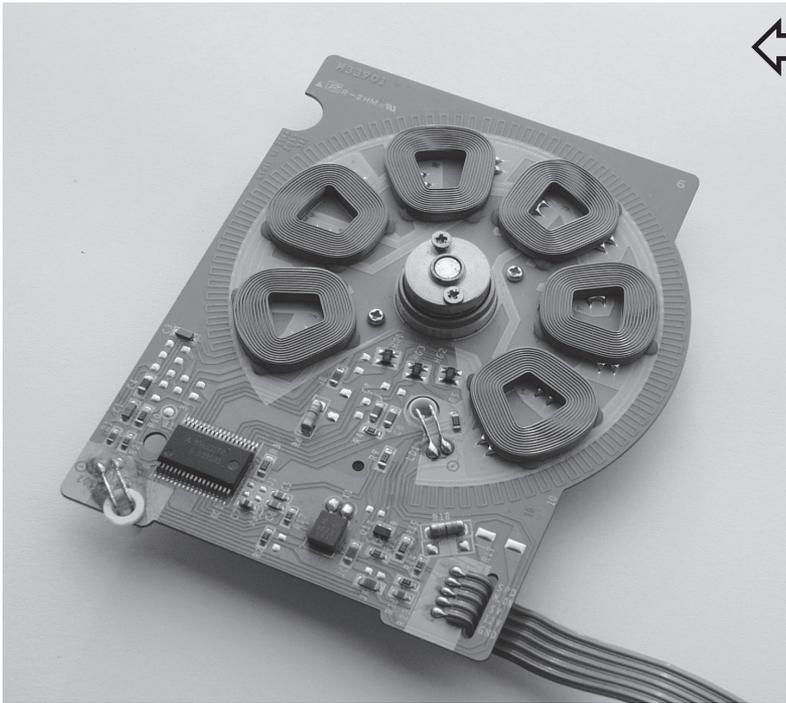
šílená kniha nakladatelství BEN – technická literatura

Příručka má podnadpis „**Kniha pro kutily nového tisíciletí**“, mohlo by se též dodat „... **a pro totální bastlíře**“, neboť to, co v knize najdete, **nemá ve světě obdoby**. Je tedy určena všem, kteří se chtějí snadno, rychle a bez horentních finančních nákladů dozvědět něco o elektronice, avšak **bez zbytečných a složitých teorií**. A aby pak to, co se dozví, mohli nějak prakticky využít. Kniha je psána „populárně technicky“ s humorně komentovanými příklady z praxe, **které se skutečně udály**. Je určena především pro začínající elektroniky, **badatele** a kutily.

Nevěříte? Kniha se již počátkem března 2001 objevila v naší prodejní síti. Rozsah 208 stran formátu B5, objednáč číslo 121051, MC 199 Kč.

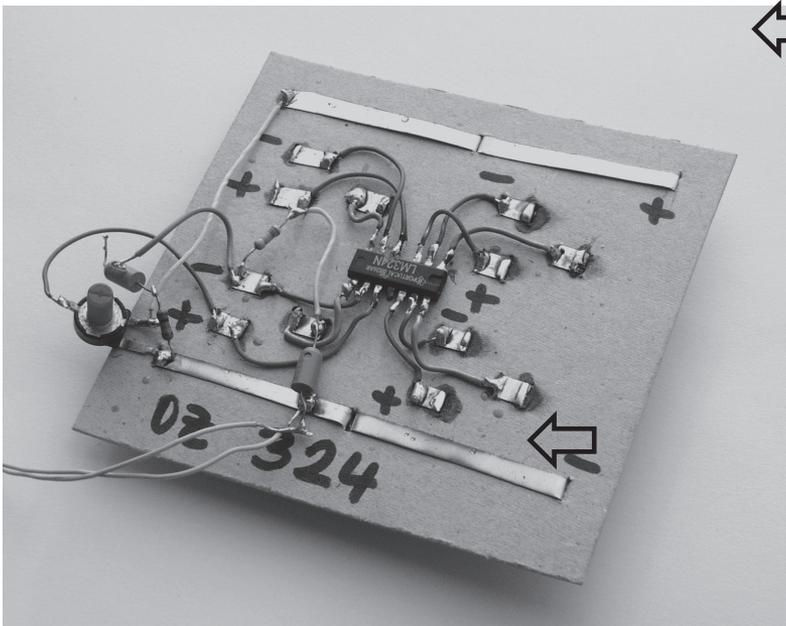


Již jste někdy viděli sektory vašeho hard disku „živě“? Jen račte! Stačí si rozmíchat náš „**magnetickej lógr**“ a vidíte data **na vlastní oči!** Jen si prosím nesrkněte jako jedna osoba zmíněná v knize ...



Schválně jestli poznáte, co je to na fotografii. Není to jen taková ledajaká deska. Její předností je, že je „za pár šupů“, že ji dostanete téměř všude a hlavně – že na ní jsou tři Hallovy generátory. Nevíte co s nimi? Dal by se s nimi postavit třeba **digitální kompas** ... Nevíte jak obvody odpájíte? Stačí se jen začíst ...

Diskuse a polemiky o tomto knižním titulu můžete sledovat po vydání (možná i před vydáním) na naší stránce <http://www.ben.cz/rpc>.



Co říkáte naší nové pokusné desce? Chce to pouze tvrdý papír, nůžky, fixu a nějaké děrovátko. Myslíte, že by mohlo časem navlhnout? Tak to zalejte parafínem nebo nějakou jinou „sračkou“. Konec s nepájivými poli a dalšími jinými pokusnými deskami. Dokud si to sami nevyzkoušíte, neuvěříte, že je něco takového vůbec možné! S chutí do toho ...



Mikroprocesory řady 8051

Jednočipové mikro počítače dnes najdeme v mnoha zařízeních a přístrojích, se kterými se denně běžně setkáváme. Mikro počítače řídí kancelářská zařízení jako jsou digitální telefony, faxy, telefonní ústředny, kopírky a tiskárny. V domácí a spotřební elektronice jako jsou rádia, televize, videa, CD přehrávače a zesilovače, ale i váhy, mikrovlnné trouby a regulátory topení si ani jejich existenci neuvědomujeme. Každé moderní zařízení z měřicí, automatizační a regulační techniky si lze dnes bez mikro počítače těžko představit. Proto je znalost návrhu a vývoje jednoúčelových, mikroprocesorem řízených, aplikací v současnosti velice důležitá.

Publikaci nelze považovat za vyčerpávající v oblasti procesorů s jádrem typu 8051, přináší však ucelený pohled na jádro tohoto procesoru. Obecně jsou popsány typy periférií i jejich použití, které výrobci integrují do moderních procesorů nejen z této řady. Popis některých zajímavých procesorů z dnes velmi rozmanité škály procesorů, byl nahrazen jednoduchým přehledem, který dobře poslouží k vyhledání vhodného procesoru pro danou aplikaci. V získaném prostoru jsou popsány první z nastupujících řad procesorů od firmy Intel řada MCS251 a Philips řada 80C51XA. Jejich architektura sice vychází z vlastností klasické 8051, ale jedná se v podstatě o zcela nové procesory s 16bitovou vnitřní architekturou, velkým adresovatelným prostorem a hlavně výrazně vyšším výkonem.

Podrobný popis instrukčního souboru s řadou praktických příkladů přináší čtenáři možnost zvládnout základy programování v jazyce symbolických adres. Příručka je určena především studentům a začátečníkům, kteří se rozhodli proniknout do problematiky stále oblíbené 8051, pochopit základy práce s těmito procesory a udělat tak nezbytný krok ke své první aplikaci.

Z obsahu: Mikroprocesory řady 8051, Instrukční soubor CPU8051, Řešené příklady a tvorba programu, Mikroprocesor MCS251 a 80C51XA, Co se obvykle nepublikuje.

Autor Petr Skalický, 160 stran A5, obj. číslo 180035, MC 199 Kč.

Dále vyšlo:

Komunikace mikrokontrolérů s okolím 1 a 2

Učebnice programování PIC

Mikrořadiče PIC16CXX

Programování mikrokontrolérů PIC16CXX

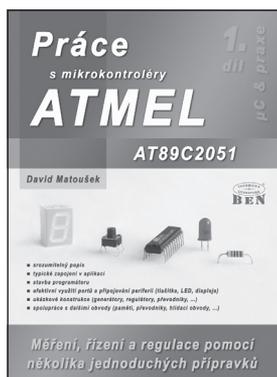
Mikrokontroléry PIC16C7X

Mikrokontroléry Motorola HC11

Mikrořadiče ZILOG Z8

Monolitické mikro počítače

Polovodičové paměti



Práce s mikrokontroléry Atmel AT89C2051

Měření, řízení a regulace pomocí několika jednoduchých přípravků

Kniha podrobně vysvětluje jednotlivé rysy mikrořadičů typu AT89C2051 a ukazuje jejich použití jak v klasických příkladech, tak i v dosud nepublikovaných konstrukcích. Velký důraz je kladen na srozumitelnost a postupné vysvětlování jednotlivých pojmů.

V úvodu je čtenář seznámen se základními pojmy mikroprocesorové techniky, následuje velmi stručný popis základních schopností řadičů ATMEL, který je zakončen popisem konstrukce programátoru a testovací desky (vývojového kitu).

Následuje popis programátorského modelu, instrukcí a assembleru mikrořadičů ATMEL. Tento výklad je doplněn praktickými příklady použití.

Po nezbytném úvodu je věnována pozornost popisu portů P1 a P3 včetně základních aplikací, které jsou určeny především začátečníkům. Poté se seznámíme s méně tradičním použitím portů P1 a P3, kromě jiného zde nalezneme realizaci nízkofrekvenčního generátoru, popis zmnožování vstupů a výstupů, konstrukci a ovládání vícesegmentových displejů, použití zabudovaného komparátoru pro měření kapacity, odporu a napětí (tedy jako A/D převodník).

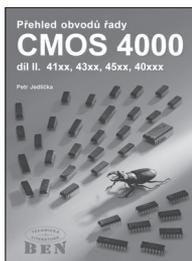
Přerušovací systém je vysvětlen na příkladech realizace nízkofrekvenčního generátoru s volitelným průběhem (tedy generátoru funkcí). Podobně sériová komunikace ukazuje velmi jednoduchou realizaci programovatelného generátoru nízkofrekvenčního signálu do 10 kHz. Čítače a časovače jsou předvedeny v aplikacích vytvářejících zvukové efekty, PWM regulátory, D/A převodníky a měřiče kmitočtu a přenesené i odporu či kapacity.

Poslední dvě kapitoly jsou věnovány použití perspektivních periferních obvodů vybavených sběrnici MicroWire (93C66, M5451B7, TLC549) nebo I2C (PCF8591), převodníků napětí nebo teploty na kmitočty realizovaných obvodem LM331 a řídicích obvodů TL77xxA a MAX690A.

Na doprovodném CD-ROM naleznete zdrojové texty všech 23 publikovaných příkladů a klíče plošných spojů všech 16 realizovaných přípravků.

Autor David Matoušek, 240 stran B5 + CD ROM, obj. č. 121093, MC 349 Kč.

ČÍSLICOVÁ TECHNIKA



Přehled obvodů řady CMOS 4000

Obsahuje základní vlastnosti a souhrnné přehledy obvodů řady CMOS 4000. U každého je uvedeno rozmístění a popis vývodů, dále pak stručný popis funkce, funkční schéma a podle potřeby funkční tabulka, logické schéma, časové průběhy. U některých obvodů i příklady aplikací. Autor Petr Jedlička,

I. díl - obvody 40xx; 180 stran B5, obj. č. 180005, MC 199 Kč,

II. díl - obvody 45xx, 40xxx; 256 str. B5, obj. č. 180022, MC 199 Kč.

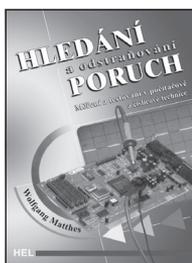


Přehled obvodů řady TTL 7400

Podobně jako kniha „Přehled obvodů řady CMOS 4000“ obsahuje základní vlastnosti a souhrnné přehledy obvodů řady TTL 7400. Přehled navíc obsahuje i zapojení vývodů pouzder PLCC a více aplikačních zapojení. Autor Petr Jedlička,

I. díl - obvody 74xx; 288 str. B5, obj. číslo 180034, MC 249 Kč.

II. díl - obvody 741xx; 304 str. B5, obj. číslo 180039, MC 299 Kč.



Hledání a odstraňování poruch –

– Měření a testování v počítačové a číslicové technice

Tato kniha se zabývá profesionálním hledáním chyb a poruch, měření a testování (zkoušením) v moderních elektronických systémech a přístrojích, které pracují převážně digitálně (číslícově). Jako příklad zde mohou posloužit typické osobní počítače „kompatibilní s IBM“ (PC). Ty jsou rozšířeny v milionech exemplářů a jsou mnohem náchylnější k chybným funkcím než mnohé jiné technické výtvořky. Chceme-li systém PC s přijatelnými náklady nakonfigurovat, uchovávat při životě a modernizovat, je téměř nezbytné provádět to svépomocí. Je samozřejmé, že základní dovednosti při hledání chyb (poruch), měření a testování je možno použít prakticky na všech elektronických systémech, přístrojích a funkčních jednotkách.

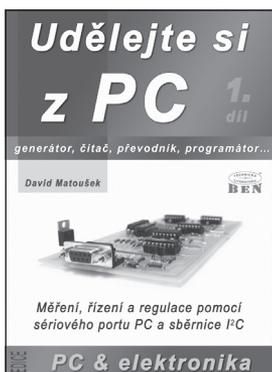
Autor Wolfgang Matthes, 608 stran B5, vázané, obj. č. 121078, MC 798 Kč. Vydalo nakladatelství HEL.

PŘIPRAVUJEME

Mikrokontroléry ATMEL AVR

Práce s mikrokontroléry ATMEL AT89S8252

Udělejte si z PC – 2. díl



Udělejte si z PC – generátor, čítač, převodník, programátor...

Měření, řízení a regulace pomocí sériového portu PC a sběrnice I²C

Kniha je určena čtenářům, kteří mají alespoň základní znalosti číslicové techniky a rámcové představy o programování. Jádrem knihy je totiž popis konstrukce a ovládání šesti elektronických přístrojů, které lze využít v amatérské praxi. Všechna tato zařízení se připojují k sériovému portu počítače a jsou ovládána programy, které běží na operačních systémech: Windows 95, Windows 98, Windows NT, Windows 2000 nebo Windows Me. Ovládací programy byly vytvořeny ve vývojovém prostředí C++ Builder verze 1.0.

První kapitola probírá teorii spojenou s ovládáním sériového portu jak pod Windows, tak i na úrovni operačního systému MS-DOS. Krátce je věnována pozornost i práci s paralelním portem. Kapitoly 2, 3, 4 a 5 popisují stavbu několika jednodušších zařízení. Jedná se impulzní generátor do 1 MHz, čítač do 16 MHz, programátor obvodů GAL a programovatelný generátor do 100 kHz. Šestá kapitola nejdříve popisuje chování sběrnice I²C a poté se věnuje výkladu tří obvodů pracujících s touto sběrnici. Jedná se o obvody: TDA8444 (8násobný 6bitový D/A převodník), PCF8574 (8bitový vstupně/výstupní port) a PCF8591 (jednoduchý 8bitový D/A převodník a 4kanálový 8bitový A/D převodník). Sedmá kapitola používá obvody popsané v kapitole 6 pro konstrukci měřicí desky vybavené převodníky A/D a D/A a číslicovými vstupy a výstupy. Jsou ukázány dvě aplikace této desky. Osmá kapitola popisuje stavbu programovatelného generátoru do 100 kHz vycházejícího z konstrukce popsané v kapitole 5. Dochází k rozšíření možných nastavení parametrů signálu a s tím je spojená i úprava ovládacího programu. V příloze jsou výkresy konstrukce desek plošných spojů pro výrobu všech uvedených zařízení v amatérských podmínkách.

Autor David Matoušek, 176 stran B5 + CD, obj. č. 121069, MC 249 Kč.



Číslicová technika

Kniha velice podrobně popisuje druhy číslicových obvodů a jejich použití. Je určena nejen začátečníkům, protože i pokročilí „bastlíři“ zde naleznou řadu dosud nepublikovaných konstrukcí.

Zvláště poslední kapitola ukazuje různé užitečné konstrukce vytvořené na bázi číslicových obvodů. Jedná se o zajímavá použití ovládacích tlačítek (například zapnutí a vypnutí jedním tlačítkem), časovací obvody, řízení výkonových obvodů (stejnoseměrné i střídavé obvody) včetně použití PWM.

Autor David Matoušek, 208 stran B5, obj. č. 121060, MC 249 Kč.

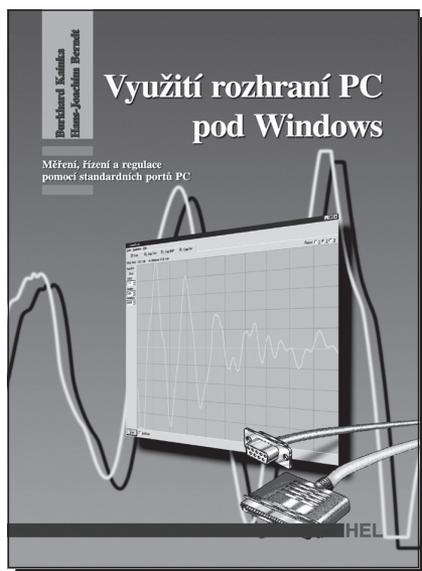
Za 200 Kč

osciloskop z Vašeho PC

Nebo byste rádi z vašeho počítače udělali „za pár babek“ síťový časový spínač, ovládali krokový motor, měřili čtyřkanalově napětí nebo kmitočty, navrhli si čítač či zapisovač, měřili teplotu a vlhkost vzduchu, automaticky testovali integrované obvody, postavili generátor funkcí, programátor paměti EPROM, paměťový osciloskop, ...

Řešení je poměrně jednoduché. V prodejní síti BEN – technická literatura je právě k mání kniha

Využití rozhraní PC pod Windows



ve které najdete kromě výše zmíněných návodů i to, jak ze zvukové karty „udělat“ ve Windows kompletní osciloskop pomocí programu **SS-CANP.EXE**. Ten je umístěn na doprovodném CD ROM. Na něm jsou všechny programové příklady ve VB5, Delphi 3 nebo Delphi 4 se všemi zdrojovými texty ve spustitelné formě, včetně **PORT.DLL**. CD obsahuje ještě jednu perličku – uživatelský program k univerzálnímu rozhraní **COMPUNI.EXE**.

Celý komplet stojí neuvěřitelných 198 Kč (tedy o 2 Kč méně než hlásá titulky), obj. číslo je 121040. **DOPORUČUJEME!**

Dále je v knize popsána stavba a provoz:

- ovládacích obvodů
- měřicích přístrojů
- analogově-číslicových převodníků
- měření se zvukovou kartou
- měření se záchytnou (capture) videokartou
- asynchronní sériový přenos
- a další aplikace PC.

Vedle přesného, srozumitelného a praktického popisu klasických rozhraní PC (s čím a jakým způsobem můžeme prostřednictvím software komunikovat) nás autoři seznamují se svépomocnou stavbou a programováním několika velmi zajímavých obvodů, které mohou být připojeny k portu počítače, aniž bychom z něj museli sundat kryt.

Základem aplikací popisovaných v knize jsou 32bitové systémy, tedy Windows 95/98 a NT. Přednost byla dána programovacím jazykům Visual Basic a Delphi.



*Věškerá technická
a počítačová literatura
pod jednou střechou*

Adresy prodejen technické literatury:

PRAHA 10, Věšínova 5, tel. (02) 74820211, 74818412

PRAHA 1, Jindřišská 29, tel. (02) 24398387

PLZEŇ, sady Pětatřicátníků 33, tel. (019) 7323574

BRNO, Cejl 51, tel. (05) 45242353

OSTRAVA, Českobratrská 17, tel. (069) 6117184

centrála:	BEN, Věšínova 5, 100 00 PRAHA 10
zásilková služba:	tel. (02) 74820411, 74816162 fax (02) 74822775
distribuce:	tel. (02) 74820211, 74818412, fax (02) 74822775
Internet:	http://www.ben.cz
e-mail:	knihy@ben.cz (objednávky zboží) redakce@ben.cz (připomínky ke knize)

CENTRÁLA



Věšínova 5,
100 00 PRAHA 10

V naší centrále
jsou
soustředěna
všechna
oddělení:

**prodejna
sklad
zásilková
služba
distribuce
nakladatelství**

Po - Pá 9.00 - 18.00 So 9.00 - 12.00

Pouhých 200 metrů od stanice metra „Strašnická“ !!

Pár slov o nakladatelství



<http://www.ben.cz>

*Nakladatelství **BEN** – technická literatura se věnuje vydávání převážně počítačové a elektrotechnické literatury. Nakladatelství je součástí stejnojmenné firmy, která se zabývá prodejem a distribucí veškeré technické a počítačové literatury, jež v poslední době vyšla. Dále pak prodejem a distribucí zejména českých titulů na CD ROM a DVD. Přehledy dostupné literatury – ediční plány, vydávané několikrát ročně, obdržíte na našich prodejnách, na vyžádání je zasíláme poštou. Celková nabídka je soustředěna do několika specializovaných prodejen po celé České republice. Jejich adresy najdete na předcházející straně.*

Adresa této knihy na Internetu: <http://shop.ben.cz/180046>

Ján Klůčik, Vojtech Fronc

Mikrokontroléry ATMEL s jádrem 8051

Vydalo nakladatelství BEN – technická literatura, Praha 2001
dotisk 1. vydání

Vedoucí nakladatelství Libor Kubica

Vedoucí redakce a DTP studia Martin Havlák

Odpovědný a technický redaktor Iveta Kubicová

Recenze Jaroslav Novák
Václav Vacek

Obálka Libor Kubica, Radka Kubicová (roháč)

Sazba Iveta Kubicová

Tisk Marten s. r. o.

Počet stran 128

objednací číslo 180046
EAN 9788073000080
ISBN 978-80-7300-008-3 (tištěná kniha)
978-80-7300-397-5 (elektronická kniha v PDF)

