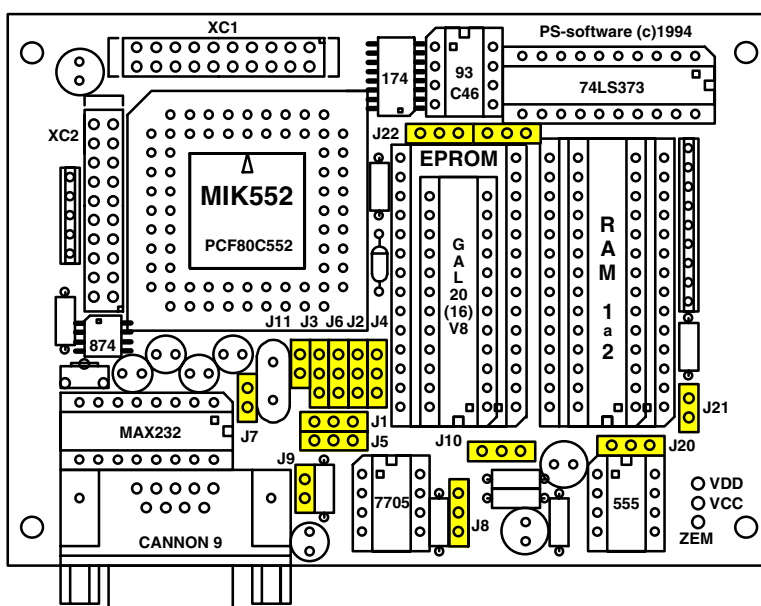


## Mikroprocesorový systém MIK552VR4

Univerzální mikroprocesorový systém MIK552 je osazen jednočipovým mikroprocesorem firmy Philips PCF80C552-5-16WP, který patří do vývojové řady procesoru 8051. Systém umožňuje připojit vnější paměť programu od kapacity 2kB do 64kB, vnější paměť dat od 2kB do 48kB a je vybaven sériovou pamětí EEPROM s kapacitou 128x8 bitů. Mezi hlavní výhody procesoru 80C552 patří 10-bitový převodník s 8 vstupními kanály, dva výstupy 8-bitové pulzně šířkové modulace, tři interní čítače, 4 vstupy záchytného systému, které mohou být využity jako další zdroje vnějších přerušení, 8 výstupů systému se 3 komparačními registry, rozšířená vnitřní datová paměť na 256 bytů, přístrojová sběrnice I<sup>2</sup>C a watchdog. Všechny důležité vstupy a výstupy procesoru jsou vyvedeny na dva konektory XC1 a XC2 typu PFL20 (2x10 vývodů). Pro snadnou komunikaci s nadřazeným počítačem, například typu PC, je systém vybaven obvodem MAX232, který zajišťuje převod duplexního sériového kanálu z úrovně TTL na úroveň V24. Pro realizaci multiprocesorových systémů je možné systém osadit rozhraním RS485. Takto zpracovaný sériový kanál je vyveden na standardní konektor CANNON 9 (vidlice). Řídicí systém je vybaven vstupy U<sub>CC</sub> a U<sub>DD</sub> pro připojení napájení systému. Nejsou-li využívány u procesoru režimy se sníženou spotřebou Idle a Power down mód, budou oba napájecí vstupy spojeny a připojeny na 5V. V ostatních případech se připojuje 5V na vstup U<sub>CC</sub> a zdroj záložního napětí na vstup U<sub>DD</sub>. Pomocí spínače J9 je možné z obvodu kontroly napájecího napětí U<sub>CC</sub> přivést signál na vstup vnějšího přerušení procesoru  $\overline{INT0}$ .

Univerzální mikropočítač je určen pro komunikační, měřicí, řídicí a regulační aplikace včetně jednoduššího zpracování signálů jako jsou:

*KONVERZE KOMUNIKAČNÍCH PROTOKOLŮ  
SBĚR DAT S PŘÍRAZENÍM ČASOVÝCH A KALENDÁŘNÍCH ÚDAJŮ  
ŘÍZENÍ BEZPEČNOSTNÍCH SYSTÉMŮ - KONTROLA A REGISTRACE PŘÍSTUPŮ  
ŘÍZENÍ PRŮMYSLŮVÝCH PROCESŮ  
SLEDOVÁNÍ A ŘÍZENÍ SPOTŘEBY ENERGIE  
SLEDOVÁNÍ A ŘÍZENÍ EXPERIMENTŮ*



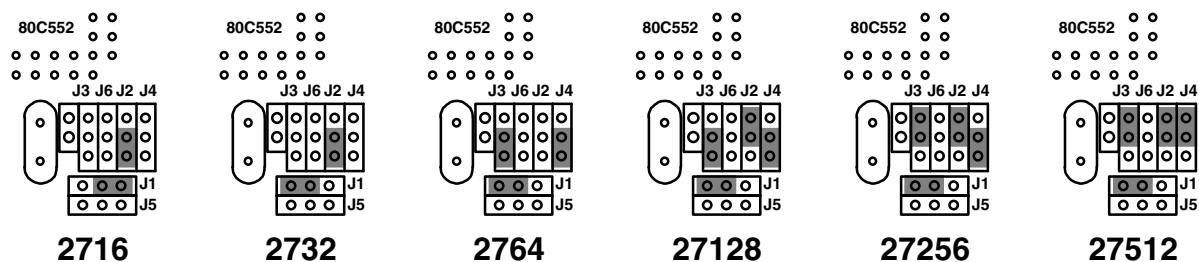
Obr.1 Systém MIK552VR4

## **Mikroprocesorový systém MIK552VR4 má tyto vlastnosti a parametry:**

Vnější paměť programu:	2kB (2716), 4kB (2732), 8kB (2764), 16kB (27128), 32kB (27256) a 64kB (27512)
Vnější datová paměť 1:	Paměť RAM v širokém pouzdře DIL. 2kB (6116) Adresy 0000H ÷ 07FFH opakující se na dalších adresách: 0800H ÷ 0FFFFH až 7800 ÷ 7FFFH nebo 8kB (6164) Adresy 0000H ÷ 1FFFFH opakující se na dalších adresách: 2000H ÷ 3FFFFH až 6000H ÷ 7FFFH, nebo 32kB (61256) Adresy 0000H ÷ 7FFFH
Vnější datová paměť 2:	Paměť RAM v úzkém pouzdře DIL. 8kB (6164) Adresy 8000H ÷ 9FFFH opakující se na adresách: A000H ÷ BFFFH nebo 32kB (61256) Adresy 8000H ÷ BFFFH použitelných pouze 16kB
Vnější sériová paměť EEPROM	128x8bitů (93C46) přístupná přes dodávané podprogramy v jazyce symbolických adres (Assembleru) a jazyce C.
Hodinový kmitočet:	1,2 až 16MHz standardně 11.052 MHz.
Vstupně/výstupní vodiče:	8 vstupů A/D převodníku nebo logických signálů, 2 výstupy PŠM nebo log.signálů, 2 vývody I <sup>2</sup> C, 18 vstupně/výstupních vodičů včetně vstupů přerušení, časovačů, záchytného systému a výstupů komparačního systému.
Maximální přenosová rychlost sériového kanálu:	100 kbitů/s
Napájecí napětí U <sub>CC</sub> a U <sub>DD</sub> :	5V ± 10%, U <sub>DD</sub> - je současně referenčním napětím převodníku A/D, není-li použit obvod MAX874 (zdroj referenčního napětí 4,096V)
Rozsah provozních teplot:	0 až 70°C, na vyžádání rozsah -20 až 85°C
Hodiny reálného času:	Zvláštní vybavení na přání zákazníka s indikací sekund, minut, hodin, dnů, dne v týdnu, měsíce a roku (na dvě platné cifry) v BDC kódu. Systém neumožňuje od těchto hodin generovat jakékoliv přerušení.
Odběr ze zdroje:	50 až 100mA - podle typu paměti Idle mód = I <sub>norm</sub> - (15÷30 mA) v závislosti na konfiguraci Power down mód = (1,5÷6 mA) pouze pro CMOS paměti a program v EPROM
Verze:	Řídící - aplikace v EPROM, Vývojová - aplikaci lze nahrát z PC do jedné z paměti RAM a spustit
Rozměry:	10,1 x 8,0 cm včetně konektoru, výška 2,0 cm 10,1 x 7,3 cm - plošný spoj 9,45 x 6,3 cm - rozteč děr o průměru 3 mm.

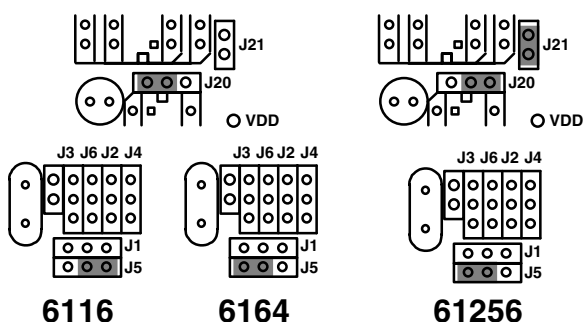
## Vlastnosti a konfigurace MIK552

Mikroprocesorový systém MIK552 je navržen jako uzavřený univerzální řídicí a vývojový systém (MIK552V3) systém, který obsahuje řadu funkcí potřebných k řízení jednoduchých i složitějších aplikací včetně možností zpracování signálů. Systém je vybaven vnější pamětí programu o velikosti 2, 4, 8, 16, 32, 64kB. Je-li systém použit v aplikaci, u které nevystačíme s kapacitou rozšířené vnitřní datové paměti procesoru (256 bytů), může uživatel využít vnější datovou paměť o kapacitě 2 nebo 8kB v širokém pouzdře DIL a 8 nebo 32kB v úzkém pouzdře DIL. Celková kapacita vnější datové paměti RAM tak může dosáhnout kapacity 2, 8, 10, 16, 32, 34, 40 a 48kB. Konfigurace příslušné vnější programové a datové paměti se provádí pomocí zkratujících propojek na šesti nožových přepínačích J1 až J6, jejichž umístění v mikroprocesorovém systému je zobrazeno na obr.1 (obrázek je v měřítku 1:1). Na obr.2 jsou zobrazeny všechny konfigurace přepínačů J1 a J4 pro vnější paměti programu realizované pamětmi EPROM

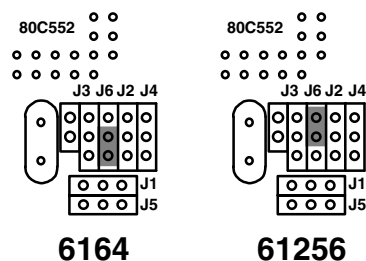


Obr.2. Konfigurace paměti EPROM

2716 (2kB) až 27512 (64kB). V případech pamětí 2716, 2732 a 2764 jsou přepínače, kde na **umístění zkratovací propojky nezáleží** a proto nemusí být vůbec použita. Na obr.3 je zobrazeno nastavení přepínačů J5, J20 a J21 pro vnější paměť dat realizovanou pamětí 6116 (2kB), 6164 (8kB) a 61256 (32kB) v širokém pouzdře DIL a na obr.4 je zobrazeno nastavení přepínače J6 pro vnější paměť dat tvořenou pamětí 6164 (8kB) nebo 61256 (32kB) v úzkém pouzdře DIL. Tmavě vybarvené plošky určují polohy zkratovacích propojek. Systém je dále vybaven sériovou pamětí EEPROM o kapacitě 128x8bitů, která je přístupná pomocí podprogramů dodávaných výrobcem v jazyce



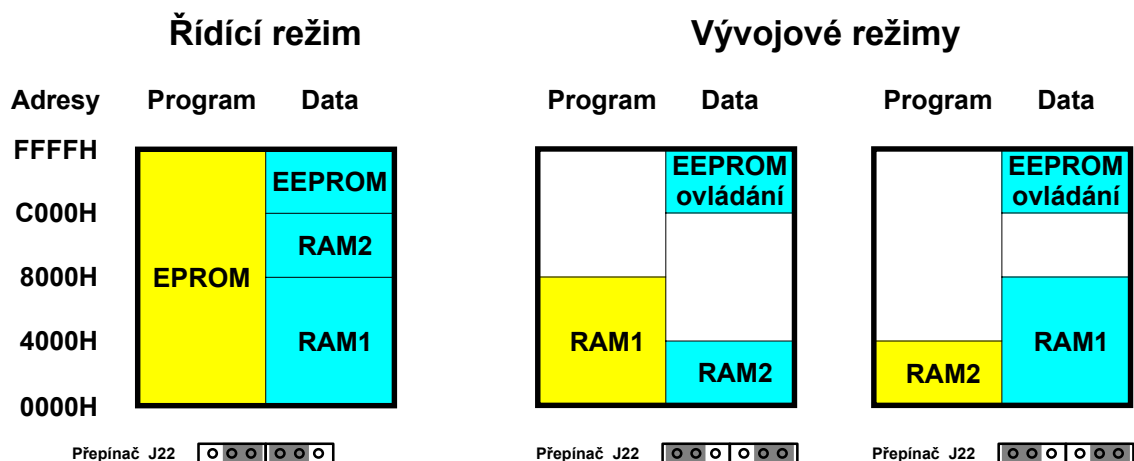
Obr.3. Konfigurace paměti RAM široké pouzdro DIL



Obr.4. Konfigurace paměti RAM úzké pouzdro DIL

symbolických adres a jazyce C. Paměť lze využít jako paměť konstant, korekčních hodnot nebo přístupových hesel, které v závislosti na aplikaci je třeba občas měnit. Na obr.5 je zobrazeno využití paměťových prostorů v popisovaném systému v řídicí verzi (aplikační program je uložen v EPROM) a v obou vývojových režimech ve verzi vývojové. Jestliže vývojovou verzi využijeme jako verzi řídicí, potom rozdělení bude stejné jako u verze řídicí s tím, že funkce pro přepínání paměťových prostorů leží ve stejném adresovém prostoru jako paměť EEPROM.

Vyjma konfiguračních přepínačů pro programovou a datovou paměť J1 až J6 je



Obr.5 Struktura paměťového prostoru MIK552 a nastavení přepínačů J22

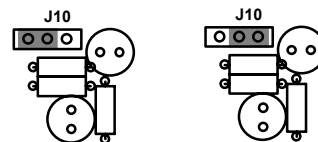
systém vybaven dalšími čtyřmi nebo pěti (MIK552VR4) spínači. Spínač J7 ovládá signál procesoru EW (Enable Watchdog). Watchdog ("hlídací pes") je čítač, který slouží ke kontrole správné činnosti programu v mikroprocesorovém systému. Protože správně probíhající program musí zajistit vynulování tohoto čítače dříve, než dojde k jeho přetečení a následnému vynulování procesoru, je činnost čítače watchdog špatně slučitelná s režimy procesoru se sníženou spotřebou. V módu Idle způsobí povolená činnost watchdog po uplynutí nastavené doby vynulování procesoru a jeho návrat k normální činnosti od adresy 0000H. V power down módu je vyřazen z činnosti i oscilátor procesoru a proto není činnost čítače watchdog možná. Proto výrobce zajistil, aby při povolené činnosti čítače watchdog (EW=0) byly instrukce pro uvedení systému do power down módu ignorovány (nelze nastavit bit PCON.0) a program pokračoval následujícími instrukcemi. Proto je systém vybaven spínačem J7, který ovládá signál povolení (spínač J7 je zkratován) nebo zakázání (spínač J7 rozpojen) činnosti obvodu watchdog. Spínač J11 (verze V3) slouží k nastavení signálu EA pro konfiguraci programové paměti procesoru do log.0 (spínač spojen) nebo log.1 (spínač rozpojen). Je-li program umístěn ve vnější paměti programu EPROM, potom musí být EA=0, je-li využívána vnitřní programová paměť procesoru (87C552 nebo obdobné procesory), potom by mělo být EA=1.

Standardně je mikroprocesorový systém MIK552 nulován třemi způsoby. Základní nulování je odvozeno z napěťového komparátoru napájecího napětí  $U_{CC}$  a může být konfigurováno přepínačem J10 na dvě varianty:

- 1) Systém je nulován v případě poklesu napájecího napětí  $U_{CC}$  pod minimální napětí  $U_{CCmin} = 4,5V$ .
- 2) Systém je nulován při náběhu napětí  $U_{CC}$  a není nulován při poklesu napětí.

Na obr.6 je zobrazena poloha zkratovací propojky na přepínači J10 pro obě varianty. Ve vývojovém režimu je možné s ohledem na počáteční nastavení tohoto obvodu využívat **pouze** druhou variantu nulování.

Stejná situace je i v případě využití vývojového systému jako řídicího systému. Dalším zdrojem přerušení je vstupně/výstupní vývod procesoru RESET vyvedený



příp.1

příp.2

Obr.6 Konfigurace nulování

na uživatelský konektor XC2. Zdrojem nulování může být čítač watchdog, který při přetečení generuje na tomto vývodu po dobu 3 strojových cyklů log.1. Stejným způsobem, tj. přivedením log.1 na tento vývod, může uživatel generovat nulování systému MIK552 a zároveň může z tohoto vývodu snímat pomocí jednoho vstupu obvodu LS, ALS, CMOS nebo HCT informaci o nulování systému a využívat ji k nulování uživatelského zařízení. Pro správnou činnost systému musí být na vývodu RESET procesoru napětí menší než 0,7V. Nulování systému MIK552 umožňuje i signál sériové linky DTR (Data Transmit Ready - připravenost PC), který je ovládán nejnižším bitem registru pro řízení modemu u počítače PC (adresa 3FCH (pro COM1), adresa 2FCH (pro COM2)). Nulování z nadřazeného počítače je umožněno po zkratování spínače J8 nastavením příslušného bitu v registru modemu do úrovně log.1. Na konektoru CANNON 9 vývodu DTR to odpovídá napětí +7V až +12V. Není-li zkratovací propojka použita, je tato možnost nulování vyřazena z činnosti. Ve verzi MIK552VR4 je spínač J8 nahrazen přepínačem. Je-li propojka v poloze blíže kraji desky, potom nulování z počítače PC zajistí vynulování celého systému. Tato možnost musí být používána ve vývojovém režimu. Přepnutí přepínače do opačné polohy zajišťuje nulování pouze procesoru a tím i testované aplikace běžící v paměti RAM bez návratu do vývojového prostředí uloženého v paměti EPROM.

Posledním konfiguračním spínačem je spínač J9, který slouží k připojení nulovacího výstupu napěťového komparátoru 7705 s aktivní úrovní log.0 ke vstupu vnějšího přerušení  $\overline{INT0}$ . Zkratováním spínače J9 je výstup komparátoru propojen se vstupem  $\overline{INT0}$  a při poklesu napájecího napětí může být generováno přerušení zajišťující uchování nezbytných hodnot v zálohované vnitřní paměti procesoru ( $U_{DD}$ ) nebo externí datové paměti tvořené obvodem zero power RAM. Využívání tohoto režimu, při kterém je vhodné zajistit přechod procesoru do režimu se sníženou spotřebou nebo napájením

tzv. power down módu, je možný pouze u řídicích verzí. V takovém případě je však nezbytné generovat nulování systému pouze po náběhu napájecího napětí  $U_{CC}$  (viz. přepínač J10), které obnoví normální činnost procesorového systému. Není-li power down mód odvozen od poklesu napájecího napětí  $U_{CC}$ , musí být normální činnost procesoru aktivována uživatelským nulováním. **Je-li vývod  $\overline{INT0}$  využit jako výstup, spínač J9 musí být rozpojen.** U verze MIK552VR4 jsou ještě přepínače J22, které umožňují zablokovat vývojové režimy a deska se stává pouze řídicí verzí obr.5.

Z mikroprocesoru 80C552 jsou vyvedeny všechny důležité vstupy a výstupy na dva konektory XC1 a XC2 s 2x10 vývody (PFL20) včetně sériového kanálu konvertovaného obvodem MAX232 do standardního rozhraní V24 nebo RS485 a vyvedeného na standardní konektor CANNON o 9 vývodech. V následujících tabulkách je popsáno umístění jednotlivých vývodů procesoru 80C552 na konektorech XC1(tab.1), XC2(tab.2) a sériového konektoru CANNON (tab.3 a tab.4).

**Konektor XC1**

Vývod	Název	Funkce
1	ADC7	Osmý vstup A/D převodníku nebo vstup brány 5 bit 7
2	ADC6	Sedmý vstup A/D převodníku nebo vstup brány 5 bit 6
atd.		
8	ADC0	První vstup A/D převodníku nebo vstup brány 5 bit 0
9	$U_{DD}$	Záložní napájecí napětí 5V - zálohování vnitřní RAM CPU, zdroj referenčního napětí pro A/D převodník není-li MAX874.
10	AVSS	Zem A/D převodníku
11	PWM0	Výstup pulzně šířkové modulace 0
12	PWM1	Výstup pulzně šířkové modulace 1
13	P41	I/O brána 4 bit P4.1 nebo výstup kompar.systému CMSR1
14	P40	I/O brána 4 bit P4.0 nebo výstup kompar.systému CMSR0
15	P43	I/O brána 4 bit P4.3 nebo výstup kompar.systému CMSR3
16	P42	I/O brána 4 bit P4.2 nebo výstup kompar.systému CMSR2
17	P45	I/O brána 4 bit P4.5 nebo výstup kompar.systému CMSR5
18	P44	I/O brána 4 bit P4.4 nebo výstup kompar.systému CMSR4
19 a 20	GND	Uzemnění

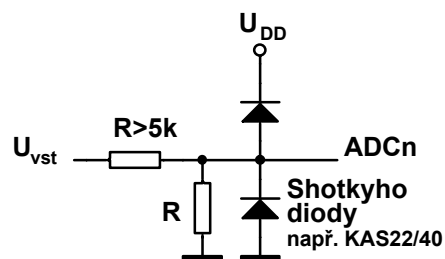
*Tabulka 1. Obsazení vývodů na konektoru XC1*

Jak vyplývá z tab.1 je na konektoru vstupů A/D převodníku vyvedeno kromě uzemnění GND také záložní napětí  $U_{DD}$ . Důvodem tohoto opatření je nutnost splnit

přísné kritérium pro vstupní napětí kteréhokoliv vstupu A/D převodníku. Vstupní napětí ADC0 až ADC7 **musí splnit tuto podmínku**

$$AU_{SS} - 0,2V < ADCn < AU_{DD} + 0,2V$$

Pokud si uživatel není jist, zda tato podmínka bude splněna, doporučujeme provést ošetření signálu podle obr.7.



Obr.7 Ochranný obvod A/D převodníku

### Konektor XC2

Vývod	Název	Funkce
1	$\overline{INT1}$	I/O brána 3 bit P3.3 nebo vstup vnějšího přerušení $\overline{INT1}$
2	T1	I/O brána 3 bit P3.5 nebo vstup čítače/časovače 1
3	$\overline{INT0}$	I/O brána 3 bit P3.2 nebo vstup vnějšího přerušení $\overline{INT0}$
4	T0	I/O brána 3 bit P3.4 nebo vstup čítače/časovače 0
5	SDA	I/O brána 1 bit P1.7 nebo datový vodič sběrnice I <sup>2</sup> C
6	SCL	I/O brána 1 bit P1.6 nebo hodinový vodič sběrnice I <sup>2</sup> C
7	RT2	I/O brána 1 bit P1.5 nebo vstup čítače/časovače 2
8	T2	I/O brána 1 bit P1.4 nebo vstup čítače/časovače 2
9	CT3I	I/O brána 1 bit P1.3 nebo vstup záchytného systému 3
10	CT2I	I/O brána 1 bit P1.2 nebo vstup záchytného systému 2
11	CT1I	I/O brána 1 bit P1.1 nebo vstup záchytného systému 1
12	CT0I	I/O brána 1 bit P1.0 nebo vstup záchytného systému 0
13	OUT1	bit D3 výstupu 74174 s adresou C000H ÷ FFFFH ext. RAM. Použití výstupu je možné v řídicím verzi MIK552 <b>za předpokladu</b> , že bude log.0 na bitech D0 až D2 (tj. X X X OUT1 0 0 0), ve vývojové verzi přes dodané programy Eepromv.
14	P47	I/O brána 4 bit P4.7 nebo výstup kompar.systému CMT1
15	P46	I/O brána 4 bit P4.6 nebo výstup kompar.systému CMT0
16	RESET	Vstup/výstup umožňující vynulovat systém MIK552 úrovní log.1 nebo přenášet nulovací signál do řízeného systému.
17	U <sub>CC</sub>	Napájecí napětí 5V
18	U <sub>DD</sub>	Záložní napájecí napětí 5V - zálohování vnitřní RAM CPU, zdroj referenčního napětí pro A/D převodník, není-li použit MAX874.
19 a 20	GND	Uzemnění

Tabulka 2. Obsazení vývodů na konektoru XC2

**Konektor CANNON 9 pro rozhraní RS232**

Vývod	Název vůči PC	Funkce na desce systému MIK552
1	DCD	Data Carrier Detect - nezapojen
2	RxD	Výstup přenášených sériových dat do PC
3	TxD	Vstup přijímaných sériových dat z PC
4	DTR	Data Terminal Ready - Signál indukující připravenost PC. Může být využit k nulování MIK552.
5	GND	Uzemnění
6	DSR	Data Set Ready - Signál indikující připravenost periferie. Nastaven na trvalou připravenost MIK552.
7	RTS	Request To Send - Připravenost PC k vysílání. Propojen s vývodem 8.
8	CTS	Clear To Send - Připravenost periferie k příjmu. Propojen s vývodem 7.
9	RI	Ring Indicator - Hlášení o příjmu - nezapojen.

*Tabulka 3. Obsazení vývodů na konektoru CANNON***Konektor CANNON 9 pro rozhraní RS485**

Vývod	Název	Funkce na desce systému MIK552
1	GND	Uzemnění
2		Nezapojen
3		Nezapojen
4	Tx / Rx+	Kladný výstup nebo vstup přenášených sériových dat
5	Tx / Rx-	Záporný výstup nebo vstup přenášených sériových dat
6		Nezapojen
7		Nezapojen
8	Tx / Rx+	Kladný výstup nebo vstup přenášených sériových dat
9	Tx / Rx-	Záporný výstup nebo vstup přenášených sériových dat

*Tabulka 4. Obsazení vývodů na konektoru CANNON*

Řídicí i vývojový systém MIK552 je realizován na jednom plošném spoji s označením MIK552V nebo (V3) a jednotlivé verze se od sebe liší pouze v osazení programovatelného obvodu U7. Řídicí verze je osazena obvodem GAL16V8 a jeho jednotlivé výstupní vývody jsou naprogramovány na kombinační výstupy, které realizují logické kombinační rovnice z tab.5. Aktivační signál pro paměť EPROM je trvale aktivován a



proto je nutné mikroprocesorový systém MIK552 osadit pamětí s přístupovou dobou alespoň 250ns. Pro snížení spotřeby jsou aktivací signály vnějších datových pamětí funkcemi jak adresových, tak i řídicích signálů ( $\overline{RD}$ ,  $\overline{WR}$ ) a proto je třeba systém osadit pamětmi RAM s přístupovou dobou alespoň 200ns pro piezokeramický rezonátor 12MHz.

Řídící verze MIK552	
$\overline{CSROM} = VCC$	$\overline{CSR1} = \overline{A15} * (\overline{RD} + \overline{WR})$
$\overline{CSR2} = \overline{A15} * \overline{A14} * (\overline{RD} + \overline{WR})$	$\overline{OER1} = \overline{RD}$
$\overline{ZAPIS} = \overline{A15} * \overline{A14} * \overline{WR}$	$RESET = RST + \overline{RSTPC}$
$\overline{D0} = \overline{OUT} \quad \text{a} \quad \overline{D0.TRST} = \overline{A15} * \overline{A14} * \overline{RD}$	

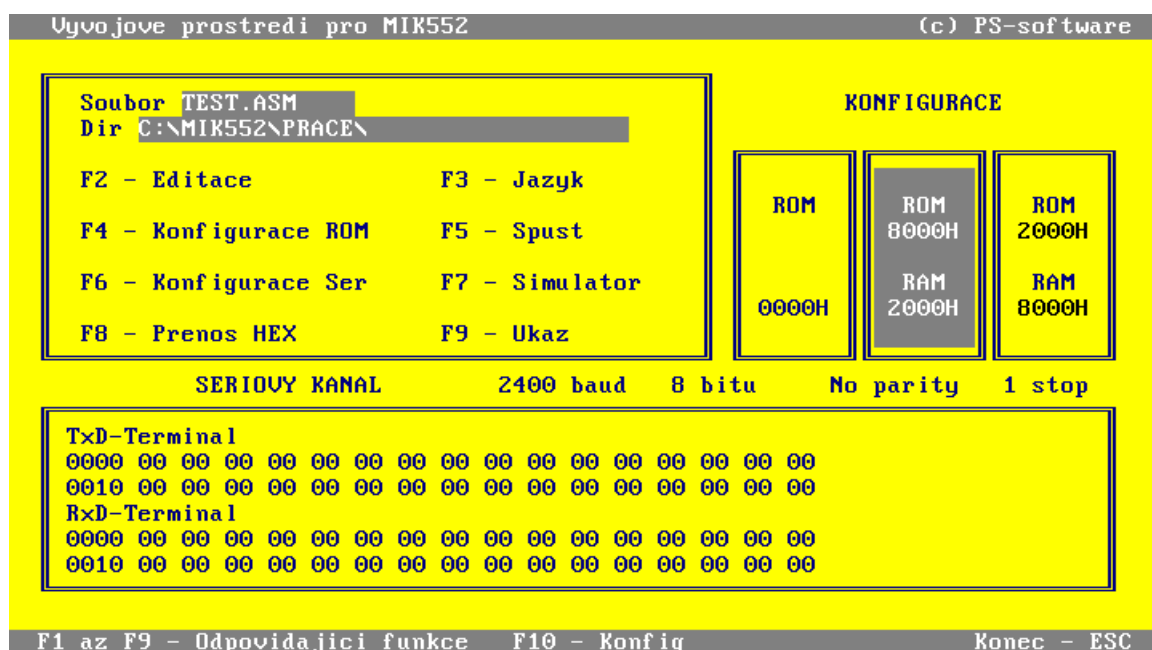
Tabulka 5. Logické funkce realizované obvodem GAL16V8

Vývojová verze je osazena obvodem ATF20V8BQL nebo PALCE22V10 a z jeho výstupních vývodů jsou dva naprogramovány na sekvenční výstup a zbývající výstupy jsou kombinační. Logické rovnice, které realizují jednotlivé výstupy nejsou veřejně přístupné. Aktivací signál pro paměť EPROM je trvale aktivován po první náběžné hraně signálu  $\overline{PSEN}$ , při kterém jsou oba signály OUT1 a OUT2 nulové. Proto je nutné mikroprocesorový systém MIK552 osadit pamětí EPROM s přístupovou dobou alespoň 250ns. Pro snížení spotřeby jsou aktivací signály vnějších datových pamětí funkcemi jak adresových, tak i řídicích signálů ( $\overline{RD}$ ,  $\overline{WR}$ ,  $\overline{PSEN}$ ) a proto je třeba systém osadit pamětmi RAM s přístupovou dobou alespoň 200ns pro datovou paměť ( $\overline{RD}$ ,  $\overline{WR}$ ) a 120ns pro programovou paměť ( $\overline{PSEN}$ ) pro piezokeramický rezonátor 12MHz.

## Vývojové prostředí MIK552

Vývojové prostředí pro mikroprocesorový systém MIK552 umožňuje vývoj programového vybavení v jazyce symbolických adres nebo jazyce C bez dalšího hardwarového vybavení jako je například simulátor EPROM. Prostředí umožňuje snadné spouštění uživatelem definovaného editoru, assembleru, překladače jazyka C a softwarového simulátoru, prohlížení vytvořených programů a jejich překladů, včetně přijatých znaků ze sériové linky a přenesení přeložených souborů v INTEL\_HEX formátu do jedné z pamětí RAM systému MIK552. Protože prostředí prozatím neumožňuje definovat body zastavení (break point), je vybaveno výraznější podporou pro sériovou linku, která uživateli umožňuje přijímat vyslané hodnoty po sériové lince na nastavené přenosové rychlosti, zobrazovat na obrazovce PC a nebo ukládat do souboru s příponou .SER o velikosti maximálně 4kB .

Vlastní prostředí se instaluje z distribuční diskety MIK552 V1.2 příkazem INSTALL A: nebo INSTALL B:, podle toho, do které mechaniky byla vložena. Na disku C: se vytvoří direktorář MIK552 a dva poddirektáře C:\MIK552\PRACE a C:\MIK552\DODAT. Direktorář C:\MIK552 je určen pro vlastní vývojové prostředí (MIK552.EXE) a jeho konfigurační soubor (MIK552.CFG). Pro vyvíjené programy je určen direktorář C:\MIK552\PRACE. Pro uživatelem definovaný editor, assembler, překladač a simulátor je určen direktorář C:\MIK552\DODAT. Uvedené cesty si může uživatel změnit v konfiguračních okénkách. Po vlastním spuštění programu MIK552 se na obrazovce objeví obr.1, který se skládá ze tří oken a řádku nápovědy. Největší okno zobrazuje možnosti, které posky-



Obr.1. Vývojové prostředí MIK552

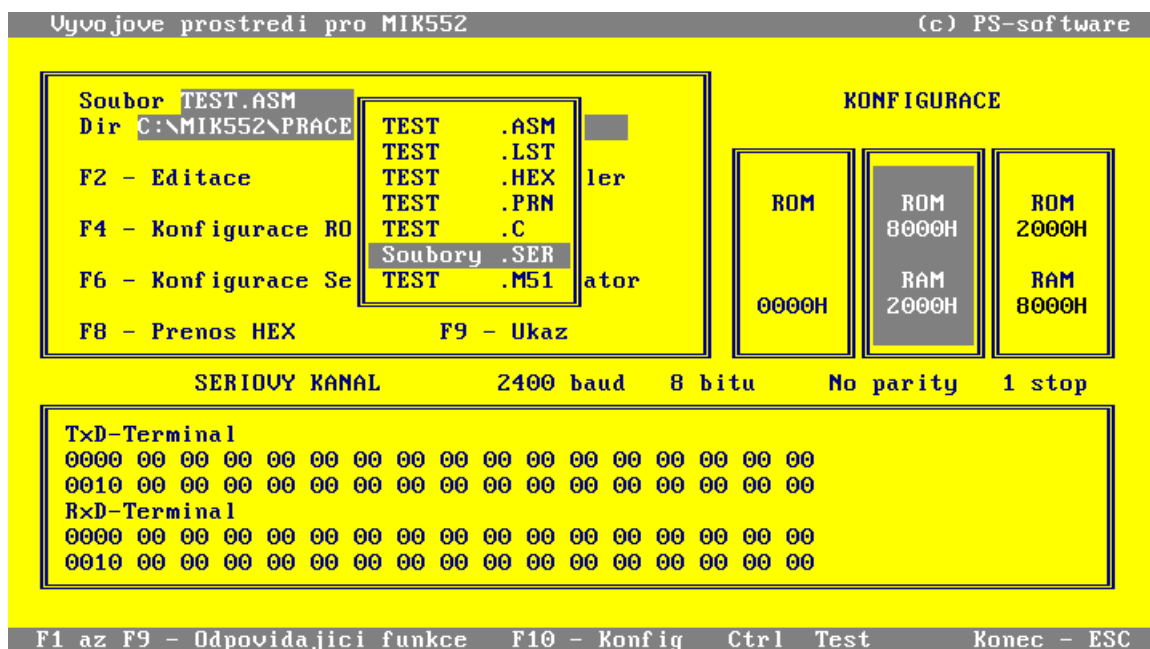
tuje program MIK552. Tři okénka pod názvem KONFIGURACE určují aktuální konfiguraci systému a poslední okno slouží uživateli k obsluze duplexního sériového kanálu po spuštění uživatelské aplikace v systému. Asi dvě vteřiny po spuštění aplikace se zvýrazní jedno ze tří konfiguračních okének určujících aktuální konfiguraci. Je-li systém osazen jednou pamětí RAM, potom ve vývojovém režimu může představovat pouze paměť programu, a proto bude zvýrazněno prvé konfigurační okénko s nápisem ROM s hodnotou odpovídající velikosti paměti. Je-li systém osazen dvěma paměťmi RAM (RAM1 a RAM2), potom bude zvýrazněno druhé okénko, v kterém je paměť RAM1 vnější paměť programu (ROM) a paměť RAM2 vnější paměť RAM. Oba paměťové prostory začínají na adrese 0000H a končí na adrese odpovídající kapacitě dané paměti mínus jedna. Ve zbývajícím paměťovém prostoru dochází k jejich zrcadlení. Pomocí **klávesy F4** lze konfigurační políčko zvýraznit a pomocí šipek (←,→) vybrat druhou variantu (tj. RAM1 = vnější paměť RAM, RAM2 = vnější paměť programu ROM). Je-li systém vybaven pouze jednou pamětí, potom nelze provést ve vývojovém režimu změnu konfigurace.

Největší okno informuje o jménu vyvíjeného uživatelského programu a direktoráři, v kterém je umístěn. Změna jména nebo direktoráře je umožněna pomocí kláves (↑,↓), které zvýrazní vybraný řádek, který je potom možné konfigurovat. Povolenými řídicími klávesami jsou (→,←, Del, Back, Insert, Home, End). Jestliže neznáme přesně jméno programu s kterým chceme pracovat, zmáčkneme **klávesu F1**, která způsobí vytvoření nabídky programů s právě uvedenou příponou. Je-li zapsán v okénku soubor řetězec FILE.C, potom jsou vyhledávány soubory s příponou C. Přípona .\* je též povolena, ale je třeba pamatovat na to, že nabídka souborů je omezena počtem 20. Potvrzením vybraného souboru (**klávesa Enter**) je jméno souboru přeneseno do příslušné řádky.

**Klávesy F2** - editor, **F3** - assembler, **Ctrl~F3** - jazyk a **F7** - simulátor umožňují vyvolat uživatelem definované sekvence programů z konfiguračního menu (F10 - viz. dále). Každý příkaz se skládá z vlastního jména programu umístěného obvykle v direktoráři C:\MIK552\DODAT, typu přípony zpracovávaného souboru a parametrů, s kterými má být vlastní program spuštěn. **Klávesa F8** umožňuje natažení přeloženého programu ve formátu INTEL HEX do konfigurací určené paměti RAM a **klávesa F5** umožňuje vlastní spuštění vyvíjeného programu. Je-li v přenášeném programu byte jehož adresa přesahuje kapacitu paměti osazenou na modulu, je uvedená skutečnost indikována chybovým hlášením, po kterém může být vlastní aplikace spuštěna se všemi důsledky chybného chování nebo zbloudění programu. Počáteční adresa přenášeného programu v INTEL\_HEX formátu je uživateli nabídnuta jako adresa spuštění programu a může být uživatelem přepsána. Programem VYVOJ1 (nebo VYVOJ2) (EEPROM) je potom vygenerována sekvence, která zajistí obvodové odpojení paměti EPROM a připojení zvolené paměti RAM jako programové paměti. Spuštěnou aplikaci lze zatím zastavit pouze individuálním programem (např. čekajícím na příchod signálu ze sériové linky) nebo vy-

nulováním systému MIK552 při současném zmáčknutí kláves Ctrl a R. Na obrazovce se objeví nápis *Nuluji systém MIK552* a činnost uživatelské aplikace je zastavena. Vývoj některých uživatelských aplikací usnadní možnost konfigurace příjmu znaků ze sériové linky. Jsou-li při spuštěné aplikaci zmáčknuty klávesy **Ctrl~A**, přejde vývojové prostředí do režimu znakového příjmu bajtů přicházejících po sériové lince. Zmáčknutím kláves **Ctrl~H** se opět vrátíme do standardního režimu. Jestliže ve spuštěné aplikaci zmáčkne- me klávesy **Ctrl~F** a zadáme jméno souboru, potom do mikropočítače MIK552 je po sériové lince vyslán textový nebo binární soubor zadaného jména. Uvedená skutečnost je indikována nápisem *Vysílám* v posledním řádku obrazovky. V průběhu vysílání jsou přijímány počítačem PC znaky vysílané mikropočítačem, ale zobrazeny budou až po odvyslání celého souboru. Počet současně přijatých znaků by neměl překročit počet 4096. V opačném případě nebudou přijaté znaky správně zobrazeny.

**Klávesa F9** umožňuje prohlížení souborů souvisejících s vyvíjeným programem obr.2. V nabídce se objeví jména souborů, se kterým uživatel pracuje. Je-li jedna z variant vybrána, potom obsah souboru je možné si prohlížet za použití kláves ↑, ↓, ←, →, PgDn, PgUp, Home, Back a End. Pro snadné vyhledávání chyb v assembleru byla zavedena funkce hledající nápisy ERROR (znak E), pro jazyk C byla navíc zavedena funkce hledající nápisy WARNINGS (znak W) a pro obecné použití funkce hledající maximálně 8-mi znakový řetězec (znak F) bez rozlišení velkých a malých písmen. Poslední nabídku tvoří soubory s příponou .SER, které vzniknou zápisem přijímaných znaků při spuštěné aplikaci, je-li tato varianta vybrána. Konfiguraci typu zápisu do sou-



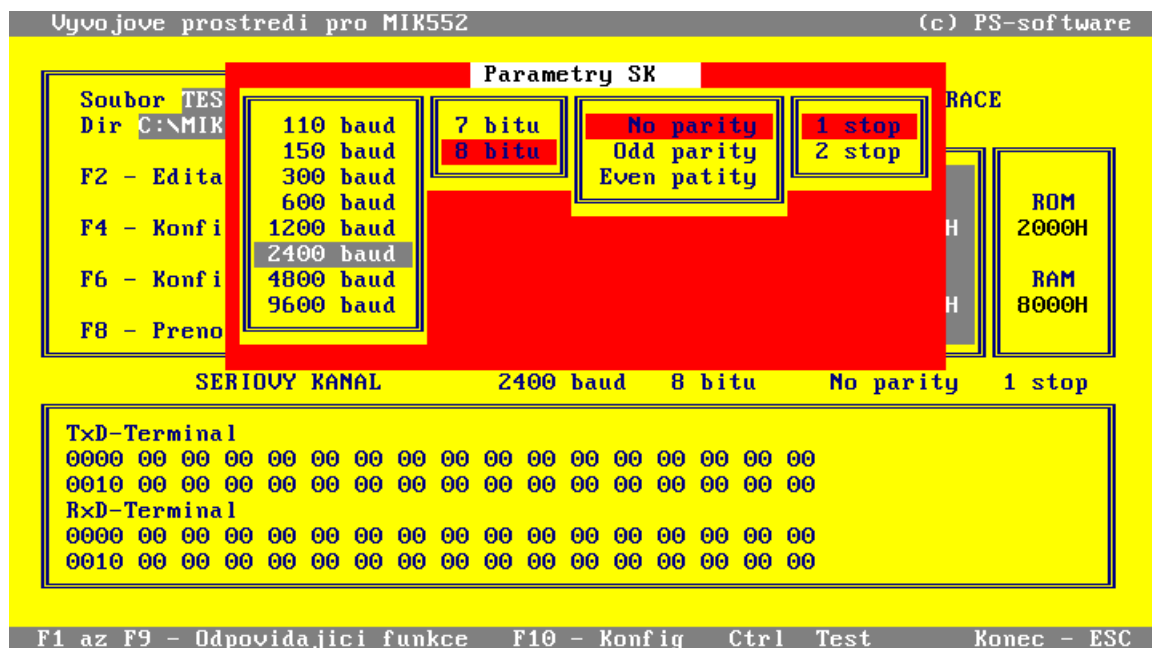
Obr.2 Nabídka souborů k prohlížení

boru nalezneme pod klávesou **F6**, kde můžeme vybrat zápis ASCII, kdy soubor je tvo-

řen přijatými binárními hodnotami nebo zápis HEX, který je podobný paměťovému výpisu přijímaných znaků.

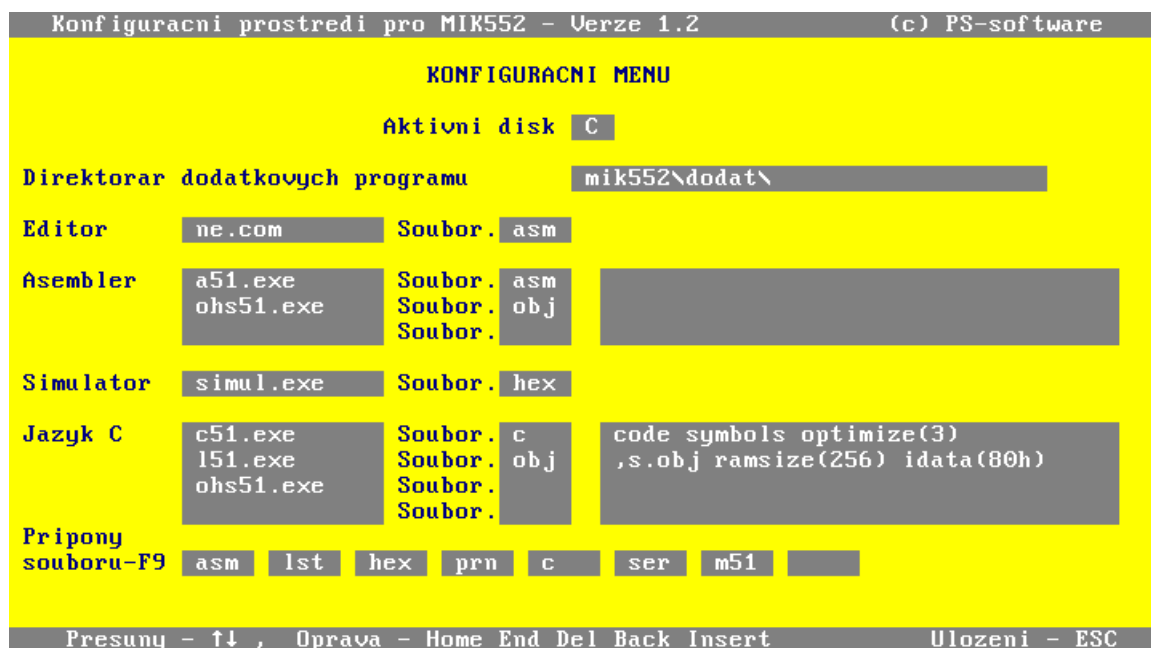
Hlavní důraz u vývojového prostředí MIK552 je kladen na využití možností vyplývajících z existence sériové linky. Ta je využívána v systémovém režimu k přenosu vyvíjených programů do mikroprocesorového systému a v uživatelském režimu k přenosu hodnot do nebo z vytvářeného programu. Jinými slovy po spuštění aplikace může uživatel využívat počítač PC jako terminál vysílající do systému MIK552 zvolené znaky a zároveň jej využívat k příjmu hodnot vysílaných vytvořenou aplikací. Příjem je realizován vždy na obrazovku počítače PC, případně mohou být přijaté znaky uloženy do zvoleného souboru (Jméno.SER), jsou-li hodnoty vysílány velmi rychle. Přenos v uživatelském režimu po sériové lince lze konfigurovat po zmáčknutí **klávesy F6** a vybráním položky **Parametry SK**. Na obrazovce se objeví situace z obr.3. Zvýrazněnou položku lze pomocí šipek přemístit na odpovídající parametr a potvrdit klávesou **Enter**. Poslední možností, která dosud nebyla popsána je možnost opakovaného vysílání zadané sekvence zatím o maximálně 32 znacích. Sekvence může být zadávána hexadecimálně nebo přímo znakově na konci příslušného řádku. Po jeho zadání je možno zvolit periodu opakování, jejíž přesnost v závislosti na typu počítače a jeho rychlosti by se měla pohybovat okolo 10 až 15%.

Po zmáčknutí **klávesy F10** se dostaneme do uživatelského menu obr.4, v kterém si uživatel definuje svůj vlastní ASCII editor pro vytváření programů v assembleru nebo jazyce C, překladač jazyka symbolických adres nebo jazyka C a případně i programový simulátor pro ověření určitých částí vytvářeného programu. Každý příkaz v konfigurač-



Obr.3 Volba parametrů sériového kanálu

ním menu se skládá z vlastního jména spouštěného programu obvykle umístěného v direktorii C:\MIK552\ DODAT, typu přípony zpracovávaného souboru a parametrů, s kterými má být vlastní program spuštěn. V případě Jazyka C (KEIL V3.21), na kterém byla funkce ověřena, bylo nezbytné zařadit do souboru AUTOEXEC.BAT nastavení, která jsou uložena v souboru stejného jména. V posledním řádku menu je možné konfigurovat pořadí přípon pro nabídku F9 s tím, že pátou položku *Soubory.SER* nelze přesouvat.



Obr.4 Konfigurační menu prostředí MIK552

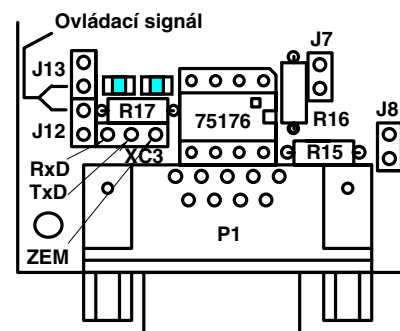
Vývojové prostředí V1.2 bylo rozšířeno o testovací rozhraní, které umožňuje ověření funkce mikroprocesorového systému MIK552. Díky rozhraní je snadný přístup k obsahu paměti EEPROM i ke zjištění hodnoty na zvoleném vstupním vodiči nebo kanále A/D převodníku. Přejít do testovacího prostředí se uskuteční po zmáčknutí klávesy ( **T** ), která je uvedena jako poslední příkaz v hlavní nabídce. Testovací rozhraní zajišťuje po zmáčknutí příslušné klávesy následující funkce:

- ( **R** ) - **testování velikostí pamětí RAM** a jejich umístění v systému. Uvedené adresové prostory jsou platné pouze při využití systému MIK552 v řídicím režimu (aplikační program v EPROM). Ve vývojovém režimu začínají adresové prostory pamětí RAM s danou kapacitou vždy od adresy 0000H nezávisle na tom, zda jsou nakonfigurovány jako paměť programu nebo dat obr.5 předcházející části.
- ( **I** ) - **testování logických úrovní** na vstupech brány P1,P3 a P4 mikroprocesoru. Zkratováním příslušného vstupu můžeme určit, zda vstup je funkční.

- ( **M** ) - **testování výstupů** pulzně šířkové modulace. Po zmáčknutí klávesy musí být na výstupech impulzní napětí v TTL úrovních s periodou cca 11 $\mu$ s. Po ukončení příkazu přejdou výstupy do úrovně log.1.
- ( **P** ) - **testování logických úrovní** na výstupech brány P1,P3 a P4 mikroprocesoru. Tato část umožňuje testovat buď všechny výstupy najednou, nebo nastavit na jednotlivých výstupech příslušné úrovně log.0 - **0**, log.1 - **1** nebo střídající se log.0 a log.1 - **X**. Po zmáčknutí klávesy a příslušné volby, musí být na měnících se výstupech impulzní napětí v TTL úrovních s periodou cca 20 $\mu$ s. Po ukončení příkazu přejdou výstupy do úrovně log.1.
- ( **S** ) - **testování sériového kanálu**. Při nesprávně přijatém znaku je příslušný byte barevně zvýrazněn.
- ( **E** ) - **testování sériové EEPROM**. Po zmáčknutí klávesy je na obrazovce zobrazen obsah paměti EEPROM. Zadáním hexadecimální adresy (00 až 7F) a příslušné hodnoty (00 až FF) bude modifikován obsah příslušného paměťového místa a opět zobrazen obsah paměti. Zadání adresy v rozsahu (80 až FF) způsobí naplnění celé paměti zadanou hodnotou.
- ( **A** ) - **testování A/D převodníku** a jeho analogového multiplexeru. Přivedením napětí (0 až  $U_{dd}$ ) na příslušný kanál A/D převodníku může být otestována jeho správná činnost. Je-li systém vybaven napěťovou referencí, potom rozsah převodníku (000 až 3FF) odpovídá napětí na vstupu (0 až  $U_{ref}$ ).

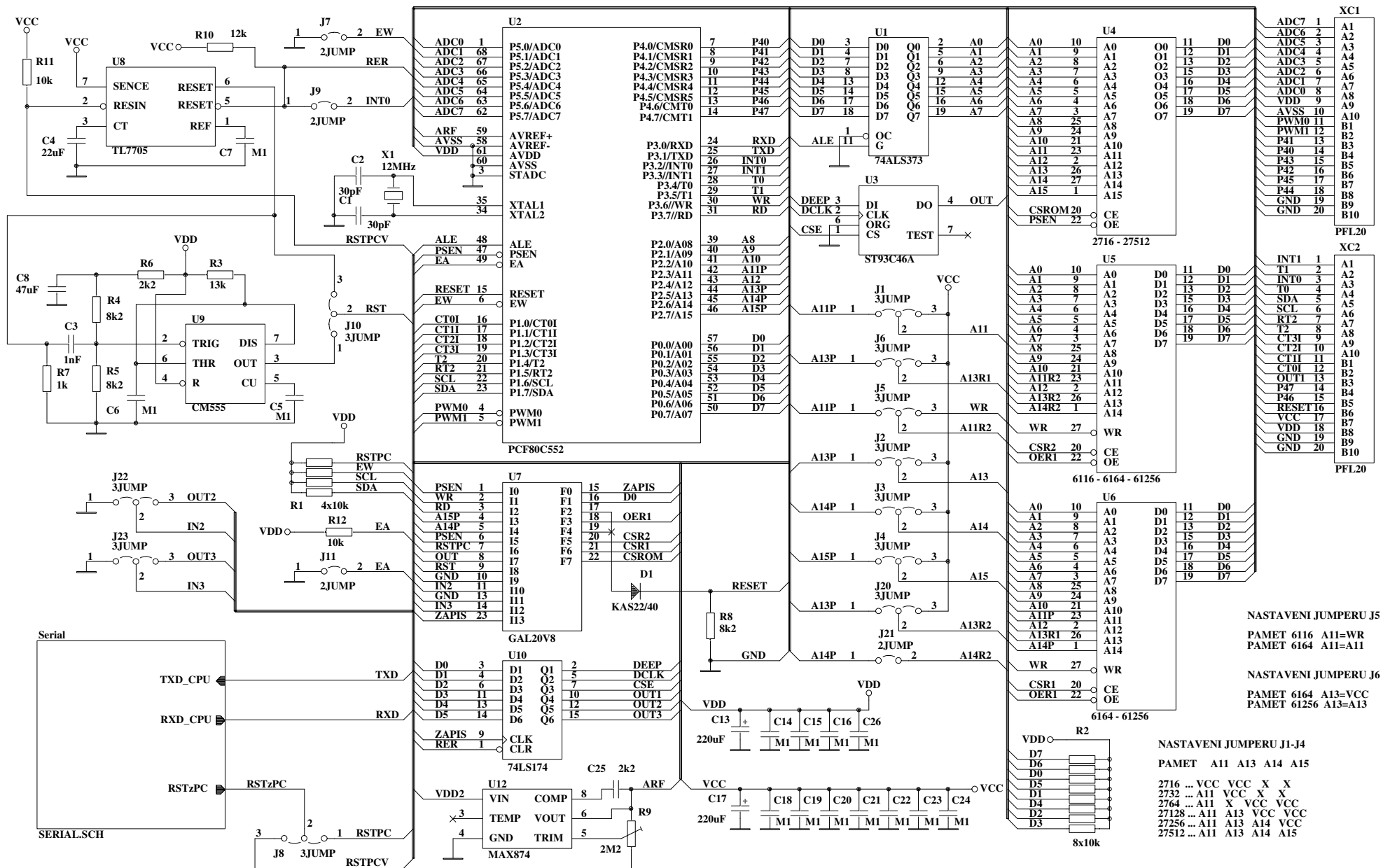
## Konfigurace MIK552 s rozhraním RS485

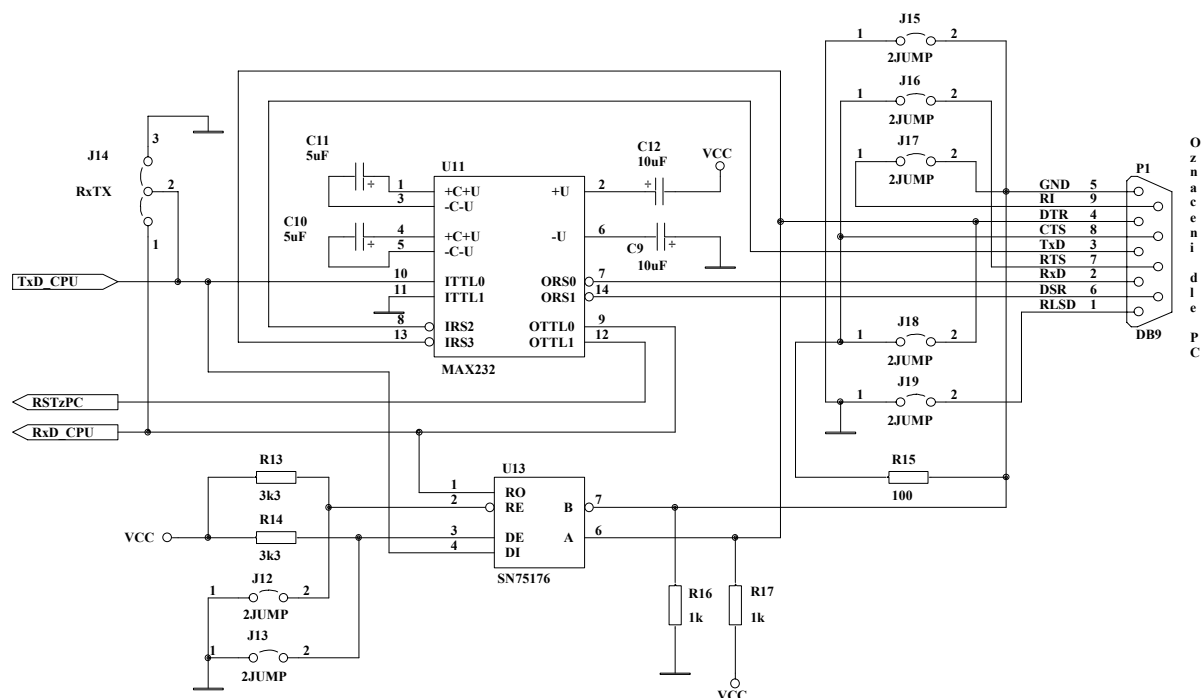
U mikroprocesorového systému MIK552V3 může být sériové rozhraní RS232 pro komunikaci s nadřazeným počítačem typu PC nahrazeno rozhraním RS485. Tím je umožněna realizace multiprocesorových systémů s procesory z řady 8051 komunikujícími mezi sebou v módu 2 nebo 3. Sběrnice RS485 je tvořena symetrickým dvou vodičovým vedením zakončeným charakteristickou impedancí umožňující nesoučasný obousměrný sériový přenos. Jeden z procesorů může vysílat, ostatní mohou přijímat. Teprve na vyzvání řídicího procesoru může být některý z podřazených procesorů začít vysílat. Převod signálů TxD a RxD procesoru na rozhraní 485 zajišťuje obvod SN75176, který je vybaven dvěma aktivačními signály DE (spínač J13) a RE (spínač J12). Signál DE=1 aktivuje přenos signálu TxD procesoru na sběrnici RS485, signál RE=0 aktivuje přenos ze sběrnice na vstupní signál procesoru RxD. Spínače J12 a J13 z obr.1 může být rozhraní konfigurováno pouze pro vysílání nebo příjem. Je-li třeba přepínat příjem s vysíláním, potom je třeba spojit prostřední kontakty J12 a J13 a připojit je na ovládající signál. Protože může nastat situace, při které není aktivní ani jeden vysílač, je třeba na dvou vodičovém vedení nastavit napěťové úrovně tak, aby odpovídaly log.1 na vstupu procesoru RxD. Proto je vedení ošetřeno třemi odpory. Mezi vodiči je zapojen odpor R15=100Ω, který představuje charakteristickou impedanci dvou vodičového vedení. Vodič A je potom připojen přes odpor R16=1kΩ na napájení a vodič B na přes stejný odpor R17 na zemní vodič. Tak je zajištěno, aby v klidovém stavu bylo na sběrnici napětí mezi vodiči A a B větší než 200mV, což odpovídá logické jedničce. Jestliže deaktivujeme obvod SN75176 nastavením signálů DE=0 (spínač J13 zkratován) a signál RE=1 (spínač J12 rozpojen), potom můžeme ze systému MIK552V3 vyvést sériové rozhraní v úrovních TTL přes nožový konektor XC3. Signály procesoru 80C552 přivedené na konektor XC3 jsou zobrazeny na obr.1.



Obr.1 Rozhraní RS485







## Seznam komunikačních příkazů pro systémy MIK552

Ovládací program systémů MIK552 a MIK537 dodržuje následující komunikační protokol mezi programem běžícím v počítači PC a monitorem běžícím v systému MIK. Příkazy lze rozdělit na příkazy ovládací, které jsou v každém systému MIK, a dodatečně přidané testovací příkazy (okolo roku 1996-7). Ovládací příkazy slouží ke zjištění konfigurace systému, nahrání programu do zvolené paměti RAM, spuštění programu uloženého v dané paměti RAM a ke zjištění obsahu paměti RAM (není programem PC implementována).

### Ovládací příkazy

**R** - zjistí obsazení systému pamětmi RAM a vrátí získanou informaci ve zprávě: *RAMAdrlow1Adrhight1Adrlow2Adrhight2* nebo *RAMAdrlow1Adrhight1C000* není-li v systému přítomna druhá paměť RAM

**H** - přenos Intel-hex souboru do výše adresované paměti RAM. Přenos je ukončen přijetím sekvence *":00000001FF", 0Dh, 00h*. Vlastní přijetí příkazu k přenosu je potvrzeno zprávou *"LOADH"*, jeho správný přenos zprávou *"END"* a nesprávné přijetí zprávou *"EEND"*.

**L** - přenos Intel-hex souboru do výše adresované paměti RAM. Přenos je ukončen přijetím sekvence *":00000001FF", 0Dh, 00h*. Vlastní přijetí příkazu k přenosu je

*potvrzeno zprávou "LOADL", jeho správný přenos zprávou "END" a nesprávné přijetí zprávou "EEND".*

*G - vlastní spuštění programu ve výše adresované paměti paměti RAM.*

*V okamžiku spuštění je paměť přeadresována do prostoru 0000 až kapacita paměti. Přijetí příkazu je potvrzeno zprávou "GOH", po kterém je očekáváno přijetí čtyřech HEXznaků představující spouštěcí adresu. Přijetí adresy a vlastní spuštění je potvrzeno zprávou "Y".*

*S - vlastní spuštění programu ve výše adresované paměti paměti RAM.*

*V okamžiku spuštění je paměť přeadresována do prostoru 0000 až kapacita paměti. Přijetí příkazu je potvrzeno zprávou "GOL", po kterém je očekáváno přijetí čtyřech HEXznaků představující spouštěcí adresu. Přijetí adresy a vlastní spuštění je potvrzeno zprávou "Y".*

*D - modifikace a výpis obsahu paměti v nepřeadresovaném tvaru (není ve vývojovém systému využívána). Přijetí příkazu je potvrzeno zprávou "ZADEJ ADRESU", po kterém se přijímají čtyři HEXznaky představující adresu modifikovaného paměťového místa. Po přijetí adresy je vyslána zpráva "ZADEJ HODNOTU", po které se přijímají dva HEXznaky představující hodnotu modifikovaného paměťového místa. Po něm následuje výpis 256 paměťových míst od zadané adresy logicky vynásobené hodnotou FFF0h.*

*T - přechod do testovacího režimu. ). Přijetí příkazu je potvrzeno zprávou "TEST". Po této odpovědi je možno zadávat tyto příkazy, které se ruší vysláním znaku 08h.*

*S - test sériového kanálu vrací znak, který byl do sériového kanálu vyslán.*

*M - generuje impulzy na výstupech PWM.*

*I - vypisuje stav vstupů P1, P3, P4 a P5 zakončených řetězcem "UKON".*

*A - vypisuje stav na analogových vstupech brány P5.*

*E - vypíše stav EEPROM. Další obsluha je podobná modifikaci a výpisu paměti RAM*