

13. SBĚRNICE A ROZHRANÍ MIKROPOČÍTAČOVÝCH SYSTÉMU

Jednotlivé bloky mikropočítačového systému spolu většinou komunikují prostřednictvím sběrnic. Sběrnice též slouží ke styku mikropočítače s okolím. Mezi sběrnicí a funkčním blokem a mezi jednotlivými funkčními bloky mikropočítače lze definovat rozhraní. Tato definice zahrnuje prostředky a pravidla pro spojení jednotlivých funkčních bloků systému a pro jejich spojení se sběrnicí.

13.1 Rozhraní mezi bloky systému na zpracování dat

Rozhraní určuje mechanické, elektrické, funkční a operační vlastnosti styku mezi dvěma konkrétními částmi systému na zpracování dat.

Po mechanické stránce charakterizuje rozhraní především typy kabelů, konektorů a rozměry a tvar mechanických dílů, jejichž kompatibilita je podmínkou správného propojení.

Po elektrické stránce jsou rozhodující přípustné úrovne napětí a proudu a přiřazení logických stavů elektrickým veličinám.

Funkční vlastnosti definují význam jednotlivých signálů a jejich časové průběhy včetně tolerancí. Definice funkčních vlastností tvoří obvykle nejrozsáhlejší část definice rozhraní. Implementace funkčních vlastností je nejnákladnější záležitost a pokud nejsou u dvou částí systému shodné, představuje to obvykle nejjzávažnější omezení z hlediska možností jejich vzájemného propojení.

Mechanické, elektrické a funkční vlastnosti rozhraní se většinou definují tak, že jsou nezávislé na vlastnostech částí systému, které rozhraní spojuje.

Naproti tomu operační vlastnosti rozhraní definující informační kódy, abecedy, formáty informačních bloků nebo zpráv, apli-

kační, diagnostické a testovací programy a pod., jsou závislé na vlastnostech části systému, které rozhraní spojuje. Proto se operační vlastnosti rozhraní mnohdy do jeho obecné definice nezahrnují.

13.2 Sběrnice

Sběrnicí nazýváme informační cestu, umožňující přenos informací mezi jednotlivými bloky systému pro zpracování dat a mezi systémem a jeho okolím, kterou časově sdílí několik bloků systému. Sdílení sběrnice se při přenosu informace zajišťuje technickými prostředky, které vyhradí přenosu informace z určeného informačního zdroje určenému příjemci informace jistý časový úsek. Další časové úseky jsou vyhrazeny přenosům z dalších informačních zdrojů pro jiné příjemce informace. Podstata sběrnicového uspořádání je znázorněna na obr. 13.1. Sběrnice je tvořena soustavou n vodičů, na kterou je připojen zdroj informace i kombinované bloky, které mohou sloužit jako zdroje i jako příjemci informace. Jejich počet je omezen jen vlastnostmi použitých technických prostředků.

Sběrnice dělíme podle následujících kritérií:

- a) podle výstupního obvodu připojujícího informační zdroj ke sběrnici - na sběrnice s logickými členy jejichž výstupní obvod má otevřený kolektor a na sběrnice s logickými členy, jejichž výstupní obvod je třístavový,
- b) podle úrovní signálů - na sběrnice s úrovněmi TTL, ECL a pod.,
- c) podle metody přenosu - na sběrnice symetrické a asymetrické - na sběrnice s asynchronním a synchronním přenosem,
- d) podle metody sdílení - na sběrnice s cyklickým sdílením a sběrnice s libovolným přístupem,
- e) podle směru přenosu - na sběrnice jednosměrné a sběrnice obousměrné,
- f) podle způsobu přenosu dat - na sběrnice se sériovým a paralelním přenosem.

Volba nejvhodnějšího typu sběrnice závisí především na prostorovém uspořádání systému, vzdálenosti mezi jeho jednotlivými bloky, náročích na rychlosť přenosu informace a přístupu k jednotlivým informačním zdrojům a v neposlední řadě i na ceně systému.

13.2.1 Technické prostředky sběrnic

K nejjednodušším budičům sběrnice patří logické členy s výstupním obvodem s otevřeným kolektorem. Sběrnice s těmito budiči je znázorněna na obr. 13.2. Na datové vstupy A1 a A2 logických členů se přivádí datová informace, která má být předána prostřednictvím sběrnice. Sdílení sběrnice informačními zdroji bloku 1 a bloku 2 se řídí signály na vstupech B1 a B2 logických členů tak, aby byl v každém časovém okamžiku ke sběrnici připojen jen jediný výstup logického členu. Na tomto výstupu se objeví signál, který je inverzí signálu na jeho datovém vstupu. Každý blok, který informaci se sběrnice přijímá je vybaven příslušnými technickými prostředky. V obr. 13.2 slouží jako příjemci informací invertory. Příklad časového průběhu signálu na vstupních a výstupních obvodech sběrnice jsou uvedeny na obr. 13.3. Není-li ke sběrnici připojen ani jediný informační zdroj, je její stav určen odporem R, jehož hodnota je volena na základě kompromisu mezi dvěma požadavky: při velké hodnotě tohoto odporu vzniká náběžná doba přechodu sběrnice do stavu 1, při malé hodnotě vznikají proudové špičky při změnách stavu sběrnice, které při paralelním uspořádání sběrnice mohou způsobit přeslechy mezi jednotlivými vodiči sběrnice. Hodnota odporu R musí být v předepsaných mezích, odpovídajících počtu připojených logických členů.

K dalším budičům sběrnice patří logické členy s třístavovým výstupním obvodem. Tyto logické členy mají kromě základních dvou stavů 0 a 1 ještě třetí stav s vysokou výstupní impedancí. Do tohoto třetího stavu se výstupní obvod uvádí signálem na zvláštním řídicím vstupu. Vzhledem k tomu, že k sběrnici nepřipojené výstupy představují vysokou impedanci, je počet informačních zdrojů a příjemců připojitelných ke sběrnici větší než u sběrnice vybavené logickými členy s výstupním obvodem s otevřeným kolektorem. Zapojení sběrnice i její funkci můžeme i v tomto případě popsat v obr. 13.2 a 13.3. V obr. 13.2 lze vynechat odpor R, neboť třístavové výstupní obvody mají ve stavu 1 podstatně nižší impedanci než výstupní obvody s otevřeným kolektorem. Je však třeba pamatovat na to, aby nebyly ke sběrnici připojeny současně dva nebo více informačních zdrojů, neboť by došlo k přetížení popřípadě ke zničení jejich výstupních obvodů. U sběrnice s logickými členy s otevřeným kolektorem současné připojení několika informačních

zdrojů může vést pouze ke ztrátě informace, ale nezpůsobuje přetížení výstupních obvodů.

Chování sběrnicových systémů z hlediska jejich buzení je do značné míry ovlivňováno vstupními charakteristikami připojených příjemců informace. Nejvýhodnějším řešením jsou kombinované výstupní obvody přijímač-vysílač, jejichž vstupní a výstupní impedance jsou přizpůsobeny tak, aby přechod ze stavu příjem do stavu vysílání ovlivnil sběrnicový systém co nejméně. Zapojení přijímače-vysílače je znázorněno na obr. 13.4.

Uvedené budiče jsou určeny pro sběrnice s asymetrickým přenosem. Při této metodě přenosu jsou signálová napětí všech vodičů vztažena ke společnému zemnímu vodiči, jak znázorňuje obr. 13.5. Každý vodič sběrnice můžeme pokládat za vedení, které má charakteristickou vlnovou impedanci Z_0 . Aby nedocházelo při přenosu k odrazům na vedení, je nutné, aby vedení bylo zakončeno impedancí $Z = Z_0$. Je výhodné, aby jak informační zdroj, tak příjemce informace měl impedanci, která je rovna charakteristické vlnové impedanci. Pro malé vzdálenosti se nemusí vedení a impedance zdroje a příjemce přizpůsobovat, neboť odrazy jsou zanedbatelné. Za malou lze pokládat vzdálenost od 25 cm do 500 cm podle požadavků na doby nástupních a sestupních hran signálů. Předností asymetrické metody přenosu jsou malé požadavky na technické prostředky, neboť pro přenos jednoho signálu postačuje jeden vodič. Nevýhodou je značná citlivost na rušivé signály (obr. 13.6).

Při symetrické metodě přenosu je hodnota signálu určena nikoliv jeho napětím vůči zemi, ale je dána rozdílovou hodnotou mezi dvěma vodiči, které jsou pro přenos signálu vyhraženy. Pro zvětšení odolnosti proti poruchám může být symetrie uspořádání zajištěna fyzickým uspořádáním dvojice vodičů. Rušení pak sice vyvolává na obou vodičích dvojice rušivá napětí, avšak vyhodnocovaná rozdílová hodnota se ovlivňuje jen velmi málo. Princip symetrické metody přenosu znázorňuje obr. 13.7 a příklad uspořádání používajícího tuto metodu je na obr. 13.8.

13.2.2 Metody sdílení sběrnice

Nejjednodušší metodou sdílení sběrnice je cyklické sdílení. Sběrnicový systém je v tomto případě řízen generátorem hodino-

vých impulsů, jehož výstupní hodinové impulsy jsou rozvedeny ke všem blokům systému a určují časové intervaly, které jsou vyhrazeny jednotlivým informačním zdrojům a příjemcům informace.

V takto řízeném systému (obr. 13.9) se zřizuje určitá informační cesta jen ve vyhrazeném časovém intervalu.

Složitější metodou sdílení používají sběrnicové systémy s libovolným přístupem, ve kterých se sběrnice přiděluje informačním zdrojům na základě žádosti o přidělení sběrnice. Pokud nelze žádosti o přidělení sběrnice okamžitě vyhovět, řadí se žádosti do fronty na přidělení sběrnice, která se buduje buď podle zvoleného prioritního schématu, anebo pouze podle pořadí, v jakém požadavky do fronty přicházejí. Zatímco řídící sběrnice při cyklickém sdílení rozvádí pouze hodinové impulsy má ve sběrnicovém systému s libovolným přístupem podstatně složitější charakter, neboť její řídící signály musí zabezpečovat styk mezi jednotlivými bloky na úrovni dialogu, t.j. předávání žádostí o jejich potvrzení.

Příkladem sběrnicového systému s libovolným přístupem je standardní sběrnice IMS - 2 (IEC 488), kterou podrobněji popiseme v jedné z následujících kapitol. Bloky, které jsou k této sběrnici připojeny, mohou mít různý charakter (obr. 13.10). Jeden z nich řídí sdílení sběrnice a nazývá se řidič. Žádosti o přidělení sběrnice sdělují jednotlivé bloky prostřednictvím signálu na řídící sběrnici řidiče, který ji podle prioritního schématu přidělí jednomu ze žádajících bloků, kterému bude předávat zprávu. Pak proběhne přenos dat, ve kterém přejímají bloky střídavě úlohu mluvčího nebo posluchače. Jednotlivé byty tvořící zprávu se předávají asynchronně pomocí signálů sběrnice řízení přenosu. Po ukončení přenosu se sběrnice uvolní a přidělí se dalšímu žádajícímu bloku.

13.2.3 Metody přenosu po sběrnici

Podle směru přenosu lze sběrnice rozdělit na sběrnice jedno-směrné a obousměrné. V uspořádání s jednosměrnými sběrnicemi, znázorněném na obr. 13.11, slouží jedna sběrnice k zápisu dat, druhá sběrnice ke čtení dat. Obě sběrnice jsou propojeny prostřednictvím vazebního členu, takže jejich obsah je shodný. K přenosu

slova délky n je třeba $2n$ budičů. Každý blok připojený ke sběrnici má oddělený vstup a výstup. Uspořádání sběrnicového systému s jednosměrnými sběrnicemi dovoluje snadno zavést galvanické oddělení jednotlivých částí systému pomocí optoelektronických členů, přechody z asymetrické na symetrickou sběrnici a pod.

Na obousměrné sběrnici, vyznačené na obr. 13.1 a 13.2 se přenáší informace oběma směry. Toto uspořádání je výhodné především pro přenos informací uvnitř číslicového systému, neboť přináší úspory v počtu použitých vodičů, kterých je třeba pro přenos slova délky n pouze n , avšak nevýhodou je menší zatížitelnost sběrnice, obtížné galvanické oddělení částí systému a poměrně nákladná metoda pro zvýšení logického zisku, neboť vřazení obou-směrného zesilovače není možné, a je nutné vřadit do sběrnice dva opačně zapojené jednosměrné zesilovače a podle směru přenosu na sběrnici provozovat jen jeden z nich.

Podle způsobu přenosu dat rozdělujeme sběrnice na sběrnice se sériovým přenosem dat a s paralelním přenosem dat. Při sériovém přenosu jsou jednotlivé byty slova přenášeny po jediném vodiči v časové posloupnosti bit po bitu. Přitom je samozřejmě nutné zabezpečit, aby příjemce informace rozeznal začátek a konec slova. Podle metody, kterou se tento požadavek zabezpečuje, rozeznáváme přenos asynchronní a přenos synchronní. Při asynchronním přenosu se do vysílaných dat vkládají synchronizační byty nebo značky, které slouží příjemci informace k udržení synchronizmu bez ohledu na přijímaná data. Zvláštním případem asynchronního přenosu je přenos arytický, ve kterém se synchronizační byty rozběhový a závěrný, doplňují ke každému přenášenému znaku. Při synchronním přenosu se znaky vysílají tak, že poslední bit vysílaného znaku bezprostředně navazuje na první bit následujícího znaku. Příjemce musí udržovat synchronizaci s vysílaným signálem po celou dobu přenosu informačního bloku. Synchronizace se zajišťuje např. předáváním řídicích synchronizačních pulsů nebo jinými metodami, kterých si blíže povídáme v následujících odstavcích.

Sběrnice s paralelním přenosem dat musí mít nejméně tolik vodičů, kolik bitů mají paralelně přenášená slova. Bitovou synchronizaci není třeba zajišťovat a tím se zjednoduší obvody řídi-

cí přenos informace. Na druhé straně však rostou náklady na budící a čtecí obvody. Pro každý bit slova musí být zvláštní budící a čtecí obvod. I při paralelním přenosu se data mohou předávat synchronně a asynchronně. Vzhledem k tomu, že paralelní přenos se uplatňuje především pro přenos informací na malé vzdálenosti, používá se častěji asynchronní přenos, který sice vyžaduje složitější řídící obvody a větší počet řídicích signálů, ale dovoluje přizpůsobit rychlosť přenosu rychlostem zdrojů a příjemců informace. Řídící signály při této metodě styku oznamují připravenost příjemce informace k převzetí znaku a potvrzují jeho převzetí. Potvrzuje též platnost dat předávaných informačním zdrojem.

13.2.4 Sběrnice v mikropočítáčových systémech

V mikropočítáčových systémech, zvláště pak v systémech multiprocesorových se setkáváme s několika typy sběrnic, které plní různorodé požadavky, které jsou na ně v systému kladený. Příklad připojení mikropočítáčových stavebnicových prvků a systémů ke sběrnicím je uveden na obr. 13.12. První a druhý mikropočítáčový systém na uvedeném obrázku obsahuje sběrnice, které slouží ke styku mikroprocesoru s paměti a s dalšími stavebnicovými prvky. Tyto sběrnice musí především splňovat požadavky na rychlou výměnu informací mezi mikroprocesorem a dalšími stavebnicovými prvky. Dalšími požadavky je vhodná metoda sdílení sběrnice, nejlépe podle zvoleného prioritního schématu. Vzhledem k malým vzdálenostem mezi stavebnicovými prvky a procesorem lze připustit systém s poměrně velkou šíří toku, řádově do sta vodičů. Tento sběrnicový systém v zásadě odpovídá mezimodulové sběrnici středních a velkých počítáčů, popřípadě známé standardní sběrnici UNIBUS minipočítáčů PDP firmy Digital Equipment. Na rozdíl od těchto systémů, ve kterých mnohdy pracují sběrnice synchronně, v mikropočítáčových systémech se většinou používá i v tomto případě asynchronní přenos.

Dalším typem sběrnice v obr. 13.12 je vnější sběrnice, sloužící pro připojení přídavných zařízení, např. měřících přístrojů na vzdálenost do několika metrů. Požadavky na rychlosť jsou nižší než u vnitřních sběrnic; je možné je splnit paralelní sběrnicí s menší šíří toku, obsahující např. osm až dvacet vodičů.

K vzájemnému styku mikropočítacových systémů, popřípadě ke styku mikropočítáče se vzdálenými terminály slouží sériová sběrnice. Při menších požadavcích na rychlosť přenosu se používá asynchronní metoda přenosu, pro větší rychlosti metoda synchronní.

Sběrnice, používané v mikropočítacových systémech jsou včetně jejich nejdůležitějších charakteristik přehledně uspořádány v tab. 13.1.

Vraťme se ještě k obr. 13.12, na kterém je v prvním mikropočítacovém systému čárkovaně vyznačen další. Toto znázornění má ilustrovat dva rozdílné systémy, a sice multipočítacový systém a multiprocesorový systém. Multipočítacový systém je charakterizován tím, že na řešení zadáne úlohy se podílí několik mikropočítacových systémů, z nichž každý řeší část úlohy. K vzájemné koordinaci při řešení úlohy postačí, když si předávají jen relativně malou část z celkového objemu zpracovávaných informací. Tyto systémy většinou vzájemně komunikují prostřednictvím sériových sběrnic, jejichž přenosová rychlosť je pro menší objemy přenášených informací dostatečná.

Pro multiprocesorové systémy je charakteristické, že procesory sdílí společné systémové prostředky, jako např. paměti, komunikační adaptory, periferie. Kromě toho však každý mikroprocesor komunikuje se svými vlastními systémovými prostředky. V těchto systémech je objem vzájemně předávaných informací podstatně větší. Společná systémová sběrnice pro komunikaci mikroprocesorů s vlastními i společnými systémovými prostředky se brzy ukázala jako nevyhovující řešení a východisko bylo nalezeno v hierarchické struktuře sběrnic. Představitelkou sběrnice pro komunikaci multiprocesorových systémů se společnými systémovými zdroji je sběrnice Multibus Intel.

Použití hierarchické struktury sběrnic v jednoprocesorových systémech řeší časové disproporce vznikající různou rychlosťí funkčních bloků. Jestliže všechny funkční bloky mikropočítáče komunikují prostřednictvím jediné sběrnice, probíhá tato komunikace s nemenným strojovým cyklem, nezávislým na rozdílné rychlosti samotných funkčních bloků. I když některé mikroprocesory mohou kompenzovat další vybavovací dobu paměti nebo periferií přechodem do

stavu čekání, lze dosáhnout rychlosti přenosu po jediné systémové sběrniči v mezích asi 4 až 6 MHz.

V systémech s hierarchickou strukturou sběrnic se používá rychlá hlavní sběrnice pro komunikaci mezi mikroprocesorem a rychlými paměti. Druhá vstupně-výstupní sběrnice je oddělena od hlavní sběrnice pomocným vstupně-výstupním procesorem. Toto uspořádání znázorňuje obr. 13.13. Rychlosť přenosu po hlavní sběrnici je pak omezena jen na dobu cyklu mikroprocesoru, operační paměti a vstupně-výstupního procesoru. Rychlosť přenosu po vstupně-výstupní sběrnici je pak možné volit v rozmezí 4 až 5 MHz, což zjednodušuje potřebné technické prostředky. Vstupně-výstupní procesor řídí všechny vstupně-výstupní operace a provádí inicializaci, dohled a diagnostiku všech periferních zařízení. Komunikuje s hlavním procesorem jen tehdy, když má připravena data pro programové zpracování. Je proto vybaven vyrovnávací pamětí, takže může připravit k zpracování celé informační bloky, nebo zprostředkovat připojeným periferním zařízením přímý styk s operační pamětí. Systémy s hierarchickou strukturou sběrnic zjednoduší a zrychlují vstupně-výstupní operace a usnadňují modulární řešení programových a technických prostředků.

14. SYSTÉMOVÉ SBĚRNICE MIKROPOČÍTAČŮ

14.1 Sběrnice Multibus

Sběrnice Multibus byla definována jako standardní sběrnice pro propojení jednodeskových mikropočítačů řady SBC 80, jejíž jednotlivé členy dala postupně na trh firma Intel, a které zaznamenaly značný komerční úspěch. Na rozdíl od jiných výrobců, např. Digital Equipment, která definici svojí sběrnice UNIBUS nepublikovala především z obchodních důvodů, aby zadržala výrobě konkurenčních periferních zařízení připojitelných ke sběrnici, postupovala firma Intel zcela opečně. Publikovala v lit. /3/ veškeré informace o sběrnici Multibus a tím vytvořila jeden ze základních předpokladů k tomu, aby význam sběrnice přerostl rámcem jediné firmy a aby se stala nejvzácnějším kandidátem na standardní mikropočítačovou sběrnici, neboť během jediného roku vytvořilo dvacet jiných výrobců různá periferní zařízení, která jsou s touto sběrnicí kompatibilní. K dalším předpokladům patří promyšlený přístup při řešení této sběrnice, který jí dává univerzální charakter. Tím se liší od konkurenčních sběrnic, např. sběrnice S-100, která též našla široké uplatnění především na trhu stavebnic mikropočítačů pro amatéry. Jednodeskové mikropočítače a další stavebnicové bloky jsou koncipovány tak, že po připojení ke sběrnici Multibus vytvářejí systém s hierarchickou strukturou sběrnic, vyznačený na obr. 14.1. Každý stavebnicový blok má vlastní vstupy/výstupy a vlastní paměť. Při vnitřních operacích jednotlivé prvky bloku komunikují prostřednictvím vlastní sběrnice a nepožadují přístup k systémové sběrnici Multibus. Tím se značně redukuje četnost žádostí o přidělení sběrnice, neboť žádosti uplatňují jen ty funkční bloky, které chtějí získat přístup k společným systémovým prostředkům, jako např. paměťovým místům společné paměti nebo společným vstupně-výstupním zařízením. Požadavky na přístup k vlast-

ním systémovým prostředkům a k společným systémovým prostředkům rozlišují technické prostředky funkčního bloku obvody styku se sběrnicí Multibus na základě adresy požadovaného prostředku. Je-li tato adresa v adresovacím rozsahu vlastních systémových prostředků, funkční blok přidělení sběrnice nepožaduje, v opačném případě předává žádost a na její základě je mu přidělena. Sběrnice ovšem nesmí být obsazena jiným funkčním blokem; jinak musí čekat žádající blok až se uvolní. Řídící signály sběrnice Multibus musí tedy indikovat zda je sběrnice volná, nebo zda je obsazena. Současně uplatněné žádostí o přidělení systémové sběrnice, se řeší postupně podle priorit jednotlivých funkčních bloků, pomocí technických prostředků.

14.1.1 Mechanické a elektrické vlastnosti

Sběrnice tvoří propojení mezi jednotlivými funkčními bloky systému, realizované na standardních deskách s rozměry 305 x 65 mm, které jsou vybaveny na obou delších stranách konektory. Tyto konektory usnadňují vzájemné propojení desek a ostatních částí systému. Konektor Pl s.ouží k připojení k standardní sběrnici Multibus. Všechny signály jsou po elektrické stránce kompatibilní s logickými obvody TTL.

14.1.2 Funkční vlastnosti

Přehled signálů sběrnice Multibus a jejich přiřazení vývodům konektoru Pl je uveden v tab. 14.1. Z celkového počtu 86 vývodů je 24 věnováno napájení, 16 vývodů obousměrné datové sběrnici a dalších 16 vývodů adresové sběrnici. Řídící sběrnice obsahuje signály řízení paměti (MRDC/, MWTC), řízení vstupů/výstupů (IORC/IOWC), řízení přístupu k sběrnici (BCLK/, CCLK/, EPRN, EPRO/, FREQ/, BUSY/) a potvrzující signály (XACK/, AACK).

Funkční bloky připojené ke sběrnici Multibus dělíme na řídící a podřízené. Řídící funkční bloky mohou předávat žádosti o přidělení sběrnice a po jejím přijetí řídit přenos po sběrnici. V okamžiku, kdy je přijata žádost funkčního bloku o přidělení sběrnice, stává se hlavním řídícím funkčním blokem a rá přístup k adresové sběrnici. Podřízené funkční bloky mají přístup k datové sběrnici pouze v době, kdy jsou adresovány. Totéž se týká i ří-

dicích bloků, které právě nerají funkci hlavního řídícího bloku. Jako podřízené bloky jsou v systému klasifikovány všechny funkční bloky, které nemají právo řídit přenos na stěrniči Multibus, přitom ovšem mohou být vybaveny vlastním mikroprocesorem a vykonávat i složitější funkce.

Řízení asynchronního přenosu

Sběrnice Multibus používá asynchronní metodu přenosu, která urožňuje, aby funkční bloky měly na sobě rezavíslé časování. Jestliže hlavní řídící blok, např. jednodeskový mikropočítač, rebo řadič přímého styku s parětí vyšle povelový signál k čteri nebo zápisu obsahu paměťového místa, čeká s vysláním dalšího povelu až do té doby, dokud adresovaný funkční blok, provádějící povel, nepotvrdí jeho ukončení. Asynchronní přenos je řízen časovou posloupností povelových a potvrzujících signálů (obr. 14.2). Při provádění jakéhokoliv povelu musí hlavní řídící funkční blok předepsat obsah adresové sběrnice. Jestliže chce předávat povel zápisu, musí předepsat rovněž obsah datové sběrnice. Pak teprve s minimálním zpožděním 50 nsec může předat povelový signál zápisu, buď MWTC/ nebo IOWC/. Tento povelový signál zůstává v aktivním stavu až do té doby, dokud adresovaný funkční blok nepředá potvrzující signál XACK/. Aby se předešlo blokování systému v důsledku poruchy funkčního bloku, je doba čekání na potvrzení omezena na 10 ms. Po jejím uplynutí počítá řídící funkční blok další činností, např. přechodem k diagnostické proceduře. Adresovaný funkční blok předává po povelu zápisu potvrzující signál XACK/ tehdy, když ukončil zápis buď do vyrovnavacího registru nebo do paměťového místa. Hlavní řídící funkční blok pak růžne okamžitě zrušit povelový signál a nejdříve za 50 nsec adresové a datové signály.

Při provádění povelu čtení je časový sled signálů obdobný.

Hlavní řídící funkční blok začánuje provádění povelu adresovými signály a pak nejdříve za 50 ns předává povelový signál čtení MRDC/ nebo IORC/. Adresovaný funkční blok potvrzuje signálem XACK/ povel čtení až tehdy, když předal obsah adresovaného paměťového místa na datovou sběrnici. Signál XACK/ v tomto případě potvrzuje, že data na datové sběrnici jsou platná. Hlavní řídící funkční blok potvrdí ukončením povelového signálu MRDC/

resp. IORC převzetí přečtených dat z datové sběrnice. Adresovaný funkční blok může pak okamžitě zrušit datové signály a hlavní řídicí blok se zpožděním 50 ns adresové signály.

Některé mikroprocesory, např. mikroprocesor 8080A, reagují na přijetí potvrzujícího signálu o převzetí dat s jistým definovaným zpožděním. V těchto případech lze vyložit časovou ztrátu způsobenou tímto zpožděním použitím signálu předběžného potvrzení AAC_Y/, který generuje funkční blok ještě předtím, než připravil data na datovou sběrnici. Vnitřní časování řídicího a adresovaného funkčního bloku však musí být v takovém vztahu, aby data ne datové sběrnici byla v okamžiku převzetí platná. Pokud nelze tento požadavek zaručit, je lépe od předběžného potvrze i upustit a využívat potvrzující signál XACK/ i za cenu uvedených časových ztrát.

Rízení přístupu k sběrnici

Metoda sdílení sběrnice musí být řešena tak, aby sběrnice byla přidělováním na základě priorit. Jinak by se značně snižoval význam metody přímého styku s pamětí, jejíž potřebné technické prostředky patří do vytavení téměř všech mikropočítačových systémů. Rádič přímého styku s pamětí mívá vyšší prioritu než mikroprocesor, neboť čte nebo zapisuje data do paměti předvaná synchronně žádanou přenosovou rychlostí. Opožděné přijetí žádosti o přidělení sběrnice tohoto rádiče by mohla znamenat ztrátu jednoho nebo dokonce několika předávaných znaků. Opožděné přijetí žádosti mikroprocesoru o přidělení sběrnice však znamená většinou pouze zpomalení jeho činnosti, nikoliv však ztrátu zpracovávané informace.

Přístup k sběrnici Multibus je řešen technickými prostředky pomocí řídicích signálů, jejichž význam je uveden v tab. 14.2. Podobně jako v systémech přerušení se používá bud metody sériového nebo paralelního zpracování žádostí o přidělení sběrnice.

Sériové zpracování žádostí o přidělení sběrnice

Uspořádání funkčních bloků při sériovém zpracování žádostí o přerušení a propojení jejich řídicích signálů je znázorněno na obr. 14.3. Priority řídicích funkčních bloků jsou dány jejich umístěním. Sousední funkční bloky mají propojen vždy vstup prioritního signálu BPRN/ s výstupem prioritního signálu BPRO/, čímž

se vytvoří v systému řetěz sériově zapojených obvodů pro zpracování žádostí o přidělení sběrnice. Kromě tohoto propojení není třeba dalších technických prostředků. Potřebné obvody jsou součástí řídicích funkčních bloků jak je zřejmě např. z obr. 14.1, v němž jsou zahrnuty do obvodů styku se sběrnicí Multibus.

Jestliže činnost kteréhokoli řídicího funkčního bloku např. vstupně/výstupní operace vyžaduje přidělení sběrnice, a nevyžaduje-li je řídicí funkční blok s vyšší prioritou, a není-li obsazena, generuje žádající blok jednak signál EPRO/, kterým blokuje žádosti s nižší prioritou, jednak signál EUSY/, kterým po převzetí sběrnice blokuje i žádost s vyšší prioritou. Je-li však sběrnice obsazena nebo na zpracování čeká nezpracovaný požadavek s vyšší prioritou, pak žádající blok musí čekat do doby, až se sběrnice uvolní nebo až se zpracuje požadavek s vyšší prioritou. Čekající blok vzorkuje stav vstupu EPRN/ a EUSY/ vždy v době sestupné hrany signálu ECLK/. Řízení sběrnice převeze až v okamžiku, kdy to připustí tyto dva vstupní stavы. Synchronní zpracování vstupních signálů EPRN/ a EUSY/ dovoluje generovat výstupní signály EUSY/ a EPRO/ jen v době sestupné hrany hodinového signálu ECLK/. Tak se řeší problém současně vzniklých žádostí o přidělení sběrnice v několika řídicích funkčních blocích.

Je třeba si uvědomit, že funkční bloky pracují vzhledem k činnosti sběrnice asynchronně. I samotný přenos signálů priorit ze vstupu EPRN/ na výstup EPRO/ se předává asynchronně, jen se zpožděním logických členů obvodů styku se sběrnicí Multibus, které tyto signály zpracovávají. Jestliže vznikne žádost s nejvyšší prioritou, pak se prioritní signál blokující všechny žádosti s nižší prioritou šíří sériově zapojenými obvody styku se sběrnicí Multibus jednotlivých funkčních bloků. Riziko současného přidělení sběrnice několika funkčním blokům při synchronním zpracování řídicích signálů vzniká jen tehdy, jestliže by se signál EPRO/, vzniklý v době sestupné hrany hodinového signálu ECLK/ neuplatnil během periody hodinového signálu na vstupu bloku s nejnižší prioritou. Tím je omezen počet řídicích funkčních bloků, které lze zapojit do série na tři až čtyři podle typu funkčního bloku. V rozsáhlějších systémech je nutné zvětšit periodu hodinového signálu ECLK/. Perioda signálu ECLK/ generovaného standardními

technickými prostředky jednodeškových mikropočítačů SBC80-05 nebo SBC80/20 je 100 ns. Oba tyto mikropočítače jsou vybaveny přepínači, kterými lze zdroj hodinového signálu odpojit od sběrnice a sběrnici vytvárit vnějším zdrojem hodinového signálu BCLK/ s delší periodou.

Jednodeškové mikropočítče SRC 80-05 a SRC 80/20 jsou též vybaveny další funkční možnosti, tzv. zámkkem sběrnice, umožňujícím obsazení sběrnice až do doby, kdy ji speciální instrukce uvolní pro příjem dalších žádostí o přidělení. Zatímco přidělení sběrnice v normálním provozu je platné jen pro přenos jediného slova, dovoluje zásek sběrnice, aby řídící funkční blok vybavený touto možností tlokoval všechny další žádosti a uskutečnil tak časově kritické činnosti, např. styk s velmi rychlou pamětí, spočívající v přenosu víceslovné informace.

Paralelní zpracování žádostí o přidělení sběrnice

Uspořádání systému s paralelním zpracováním žádostí o přidělení sběrnice Multitus je uvedeno na obr. 14.4. Prioritu žádostí řeší v tomto případě zvláštní technické prostředky uvedené v čárkovaném ohrazeném bloku. Jestliže vznikně při činnosti některého funkčního bloku žádost o přidělení sběrnice, generuje se signál BREQ/. Signály BREQ/ ze všech řídicích funkčních bloků se přiveďou na vstupy dekodéru priorit, který na svém vstupu generuje kód žádosti s nejvyšší prioritou. Ten zpracuje další dekodér, který na svém výstupu aktivuje jediný ze vstupních signálů BPRN/, toho bloku, který uplatnil žádost o přidělení sběrnice s nejvyšší prioritou. V naznačeném uspořádání a při periodě hodinových signálů BCLK/ 100 ns je zpoždění obvodů pro paralelní zpracování žádostí o přidělení sběrnice dostatečně nízké, aby bylo možno řešit prioritu nejen osmi řídicích funkčních bloků, ale při rozšíření prioritních obvodů i většího počtu bloků.

14.2 Sběrnice Microbus

Sběrnici Microbus, tvořící rozhraní mezi mikroprocesorem a ostatními prvky systému, jako např. paměti, přídavnými zařízeními, komunikačními adaptéry atd., určenou pro poměrně malé systémy s méně než deseti stavebnicovými prvky, navrhla firma National Semiconductor. Rozhraní, které reprezentuje sběrnice Microbus,

vyhovuje nejen styku mezi mikroprocesory a stavebnicovými prvky mikropočítačových systémů vyráběnými touto firmou, ale je voleno tak, aby vyhovělo s minimálním přizpůsobením i ke styku mezi mikroprocesory a stavebnicovými prvky mikropočítačových systémů dalších předních výrobců, např. Intel, Motorola, Zilog, atd.

Od sběrnice Multibus se liší především tím, že datová sběrnice je pouze osmibitová. Vzhledem k tomu, že sběrnice není určena pro složitější systémy s hierarchickou strukturou, odpadají řídicí signály sdílení sběrnice, takže řídicí sběrnice zahrnuje jen 14 signálů, z nichž tři jsou určeny jako vzorkovací signály pro čtení, tři pro zápis a další tři pro řízení přímého styku s pamětí. K dalším řídicím signálům patří signál inicializace systému, signál žádosti o přerušení a signály řízení asynchronního přenosu. Adresová část sběrnice Microbus obsahuje šestnáct adresových signálů.

I odrotnější popis sběrnice vzhledem k jejímu dosud malému rozšíření neuvádíme. Čtenář je nalezne např. v /4/.

14.3 Sběrnice S-100

Sběrnice S-100 byla definována při vývoji mikropočítačové stavebnice Altair, kterou dala na trh v r. 1975 firma MITS. I když nebyla zpočátku miněna jako standardní sběrnice, velký komerční úspěch stavebnice Altair způsobil, že ji přijaly desítky výrobců pamětí, vstupně-výstupních zařízení a dalších prvků a dílů mikropočítačových stavebnic, kteří se chtěli na tomto komerčním úspěchu podílet. Koncepte mikropočítačové stavebnice Altair vychází stejně jako koncepte jednodeskových mikropočítačů Intel z představy funkčních bloků realizovaných na standardních deskách propojených standardním způsobem.

14.3.1 Mechanické a elektrické vlastnosti

Sběrnice S-100 tvoří propojení mezi jednotlivými funkčními bloky systému, realizovanými na standardních deskách s rozměry 254 x 127 mm. Desky jsou vybaveny na jedné delší straně přímým konektorem se 100 vývody, jak je patrné z obr. 14.5. Tento počet vývodů dal této sběrnici též jméno (S-100 - standardních 100 signálů). Všechny signály jsou po elektrické stránce kompatibilní

s logickými obvody TTL. Budíče signálů mají logický zisk 10. Vstup signálu smí představovat maximální zátěž jednoho standardního vstupu TTL. Při použití logických členů s nízkou spotřebou a za předpokladu, že signály nejsou přivedeny na všechny desky, lze vybudovat systém s 20 až 25 deskami, aniž by se překročila povolená logická zátěž budíčů signálů.

Úplná mikropočítáčová stavebnice sestává z kazety, která obsahuje matriční desku s plošným spojem, propojujícím konektory pro zasunutí standardních výmenných desek. Na deskách je umístěn procesor, paměti, vstupně-výstupní moduly, atd. Plošný spoj pro propojení konektorů tvoří 100 paralelně vedených vodičů. K mikropočítáčové stavebnici patří též napájecí zdroje.

14.3.2 Funkční vlastnosti

Mikropočítáčová stavebnice Altair používala stavebnicové prvky mikropočítáče MCS-80 firmy Intel, což se projevilo i při návrhu sběrnice S-100; její řídicí signály jsou orientovány na tento mikropočítáč. Tato orientace je pokládána za jednu z nevýhod této sběrnice, neboť použití mikropočítáčových prvků, především mikroprocesorů jiných výrobců, není sice vyloučeno, ale přináší komplikace při návrhu procesorových desek, neboť technické prostředky se musí doplnit o obvody přizpůsobení k sběrnici S-100. Signály a napájecí vodiče obsazují 82 ze 100 vývodů konektoru standardní desky; zbývajících 18 vývodů zůstává volných pro pozdější doplňování, např. k rozšíření systému S-100 pro šestnáctibitové mikropočítáče. Zvláštností jsou výlučně jednosměrné signálové cesty. Příklad uspořádání je uveden na obr. 14.6.

Adresová a datová sběrnice systému S-100

Adresová sběrnice odpovídá přímo adresové sběrnici mikroprocesoru 8080A; obsahuje signály A0 až A15, které jsou přímo na procesorové desce vedeny přes budíci třístavové obvody řízené signálem ADDR DSB. To umožňuje přístup ke sběrnici S-100 i jiným funkčním blokům viz. tab. 14.8. Obousměrná datová sběrnice mikroprocesoru 8080 tvořená signály D0 až D7 je na procesorové standardní desce upravena na dvě jednosměrné datové sběrnice, vstupní datovou sběrnici, obsahující signály D10 až DI7, a výstupní datovou sběrnici, obsahující signály D00 až D07. Výstupní dato-

vá sběrnice je vybavena třístavovými budiči řízenými signálem DODSB. Rozdělení obousměrné datové sběrnice na dvě jednosměrné přináší výhody, spočívající ve zjednodušeném časování přesunů dat po těchto sběrnicích a v snadnějším řešení sdílení těchto sběrnic, např. při přímém styku s pamětí. Kromě toho se odstraní riziko vyplývající z přechodových dějů, k nimž dochází na obousměrné sběrnici při výměně směru přenosu. Nevýhodou je ovšem potřebný větší počet signálových vodičů.

Řídící sběrnice systému S-100

Řídící sběrnice S-100 používá poměrně velký počet celkem 44 řídicích signálů, které zprostředkují vzájemný styk funkčních bloků umístěných na standardních deskách. Obvody pro zpracování žádostí o přidělení sběrnice jsou na rozdíl od sběrnice Multibus řešeny pomocí obvodů na standardní desce procesoru. Pro zpracování žádostí o přidělení sběrnice podle priorit nejsou vyčleněny zvláště signály.

Řídící signály se podle funkcí rozdělují do osmi skupin: K časovacím signálům patří čtyři signály, uvedené v tab. 14.3. První dva se používají k synchronizaci procesoru s ostatními funkčními bloky. Jsou to hodinové signály Ø1 až Ø2 mikroprocesoru 8080A, které jsou stejně jako ostatní signály sběrnice S-100 přivedeny přes budiče, takže mají dostatečný logický zisk. K dalším signálům patří signál základního krystalového oscilátoru, který slouží k tvorbě vnitřních časovacích signálů funkčních bloků systému, a signál reálného času.

K další skupině vstupních řídicích signálů procesoru tab. 14.4 patří jednak signály připravenosti funkčních bloků systému k přenosu dat a sice signál PRDY ze samotného systému a signály XRDY z vnějších jednotek, které jsou na standardní desce procesoru logicky sečteny a přivedeny na vstup READY mikroprocesoru 8080. Dále jsou to signály žádosti o přerušení PINT a žádosti o přidělení sběrnice PHOLD, které jsou po invertování bezprostředně připojeny k vstupům INT a HOLD mikroprocesoru 8080.

Výstupní řídící signály procesoru tab. 15.5 jsou přímo odvozeny z řídicích výstupních signálů mikroprocesoru 8080. Na desce procesoru jsou tyto signály vedeny přes budiči obvody. Jejich iden-

tifikátory začínají písmenem P; následující symboly jsou shodné s identifikátory odpovídajících signálů mikroprocesoru.

Stavové signály tab. 14.6 se generují na základě obsahu stavového slova, které na začátku každého strojového cyklu předává mikroprocesor 8080 na datovou sběrnici. Toto stavové slovo se ukládá do stavového registru na desce procesoru a řídí vykonání předepsaného strojového cyklu. Zpracování stavového slova na desce procesoru zjednodušuje logické obvody ostatních funkčních bloků, které nemusí být vybaveny technickými prostředky pro zpracování stavového slova a jsou řízeny stavovými signály sběrnice S-100.

Ze skupiny signálů řízení paměti z tab. 14.7 se pro paměti, které nejsou vybaveny obvody pro ochranu informace využívá pouze signál MWRT. Ostatní tři signály jsou určeny k řízení obvodů ochrany paměti. Adresový prostor paměti vybavených těmito obvody je rozdělen do zón s rozsahem od 256 bytů do 4 bytů. Zónu lze pak deklarovat jako chráněnou nebo nechráněnou tak, že se na adresovou sběrnici zapiše počáteční adresa zóny a současně se generuje signál PROT, má-li být obsah zóny chráněn, nebo signál UNPROT v případě opačném. Adresuje-li se při zápisu paměťové místo chráněné zóny, signál PS vyvolá přerušení, nebo se využívá k signalizaci na pultu operátora.

Signály řízení přístupu k sběrnici S-100 uvedené v tab. 14.8 ovládají výstupní třístavové budiče adresové, datové a řídicí sběrnice na desce procesoru. Převedením těchto budičů do nevodivého stavu získají přístup ke sběrnicím jiné funkční bloky. Signálem SSW DSB se odpojuje vstup datového budiče na desce procesoru od sběrnice, takže lze zavést do procesoru data z jiného informačního zdroje, např. z klávesnice na pultu operátora.

Význam inicializačních signálů a signálů řízení režimu je zřejmý z tab. 14.9.

Koncepce napájení sběrnicového systému S-100 vychází ze samostatných napěťových stabilizačních obvodů, kterými jsou vybaveny jednotlivé desky. Výhody této koncepce spočívají v snížení nebezpečí nežádoucích vazeb přes napájecí zdroje. Další výhodou jsou jednoduché společné neregulované napájecí zdroje +8 V, +16 V a -16 V, ze kterých se na jednotlivých deskách generuje napětí + 5 V pro obvody TTL a další napětí pro obvody MOS.

Nevýhodou je však poměrně velká tepelná ztráta na jednotlivých deskách, což je nutné respektovat pečlivým návrhem chlazení stavebnicové jednotky. Mnohdy je třeba již u malých systémů volit nucený oběh chladicího vzduchu.

Další nevýhodou tohoto sběrnicového systému je bezprostřední souseďství signálů datové a adresové sběrnice, což vede k přeslechům a snížení šumové imunity. Tuto nevýhodu odstraňují výrobci základního plošného spoje kostry vkládáním stínících vodičů.

Přes tyto nevýhody však se sběrnicový systém S-100 prosadil takovou měrou, že výrobci stavebnicových dílů mikropočítáčových systémů vyráží dvě verze svých výrobců; jedna z nich je kompatibilní se sběrnici Multibus, druhá, určená především pro amatéry je kompatibilní se sběrnici S-100.

15. STANDARDNÍ PARALELNÍ ASYNCHRONNÍ ROZHRANÍ

V předchozích kapitolách jsme poznali rozhraní mezi systémovou sběrnicí mikropočítače a jeho funkčními bloky. Toto rozhraní však samozřejmě nemá univerzální charakter. V číslicových systémech se definují pro různé účely další typy rozhraní, jak ilustruje obr. 15.1, ve kterém jsou uvedena rozhraní definovaná v jednotém systému malých počítačů SMEP. Rozhraní M1 mezi systémovou sběrnicí a funkčními bloky je ekvivalentní tomu, kterým jsme se již zabývali. Rozhraní M3 je určeno pro sériový styk na bázi proudové smyčky s proudem 10, resp. 20 mA a používá se k připojení vnějších zařízení do vzdálenosti 200 až 500 m. Rozhraní mezi zařízením ukončujícím datový okruh a koncovým zařízením se označuje M4 a používá se při přenosu dat mezi vnějším zařízením a číslicovým systémem nebo mezi dvěma systémy na velké vzdálenosti. Těmto typům rozhraní se budeme věnovat v následujících kapitolách. Konečně je v obr. 15.1 uvedeno rozhraní M2, které se používá pro paralelní připojení přídavných zařízení do vzdálenosti asi 10 až 15 m, a rozhraní určená pro připojení mechanismů vnějších pamětí k odpovídajícím řídicím jednotkám, označovaná M5 až M9.

V jednotném systému elektronických počítačů je definováno rozhraní S3, odpovídající rozhraní M2 systému SMEP. Základem pro jejich definici se stalo rozhraní podle / 5 /, normalizované ve Vel. Británii, které se pro svoje výhodné vlastnosti používá v dosti širokém měřítku v mnoha zemích a o jehož přijetí jako mezinárodní normy se jedná v rámci mezinárodní elektrotechnické komise IEC. V rámci jednotných systémů počítačů se nazývá paralelním radiálním rozhraním / 6 / . I když uvedené definice nejsou zcela shodné, mají společný jeden podstatný znak, a to metodu řízení asynchronního přenosu. S uvedeným rozhraním se setkáme i v mikropočítačových systémech, neboť jím jsou vybavena přídavná zařízení,

která se k mikropočítači připojují, např. snímač děrné pásky a další zařízení, vyznačená v obr. 15.1. Proto se jím budeme zabývat v této kapitole a nazveme jej standardním paralelním asynchronním rozhraním.

15.1 Vlastnosti rozhraní

Mechanické a elektrické vlastnosti rozhraní jsou uvedeny v odpovídajících normách a doporučeních, viz lit. /5/ a /6/.

Funkční vlastnosti rozhraní jsou popsány v tab. 15.1, která obsahuje přehled signálů a jejich významy pomocí časových diagramů obr. 15.2 a 15.3. Přesná definice funkčních vlastností je uvedena v lit. /5/, popř. /6/.

Asynchronní přenos znaku se řídí obdobným způsobem, jako na sběrnici Multibus (kap. 14). Popíšeme si jej podle obr. 15.2. Signál SO generovaný zdrojem informace a signál AO generovaný příjemcem informace mají v podstatě stejný význam; jejich stav 1 znamená, že zařízení je schopno bezchybně pracovat. Jejich stav tedy nezáleží na stavu přijímaného signálu AO resp. SO, ale výlučně na vnitřním stavu zařízení. Je-li např. příjemce informace rádková tiskárna, pak stav signálu AO závisí na tom, zda jsou v činnosti její napájecí zdroje, zda snímače mechanické části neindikují závadu, zda je založen papír, atd. Není-li signál SO, resp. AO v stavu 1, nelze jednoznačně interpretovat ostatní stykové signály.

Signály SO a AO během přenosu znaku musí být ve stavu 1 a jejich stav se nesmí měnit. Přenos znaku zahajuje příjemce informace přechodem signálu AC do stavu 1, čímž informuje zdroj informace o své připravenosti k převzetí znaku. Zdroj informace na základě stavu 1 signálu AC připraví platná data. Platnost datových signálů potvrdí přechodem signálu SC do stavu 1. Příjemce informace pak převezme připravená data a potvrdí jejich převzetí přechodem signálu AC do stavu 0. Přenos znaku pak ukončí zdroj informace přechodem signálu AC do stavu 0. Teprve pak může příjemce informace zahájit přenos dalšího znaku přechodem signálu AC do stavu 1.

V obr. 15.2 je vyznačen další signál AE. Tímto signálem může informovat příjemce informace informační zdroj o tom, že rozeznal

chybu v přijatých datech. Přesná interpretace tohoto signálu není předepsána, ale je věcí dohody uživatelů rozhraní.

K řízení přenosu bloku dat se používá další řídicí signál ST vyznačený na obr. 15.3. Je-li tento signál při předávání znaku ve stavu 1, pak příjemce informace jej pokládá za poslední znak v bloku dat. Využití a přesná interpretace tohoto signálu je opět záležitostí uživatelů rozhraní, neboť data mohou být předávána buď po znacích nebo po blocích pevné i proměnné délky. Využívá-li se signál ST k řízení přenosu bloku dat, pak po detekci signálu ST příjemce informace kontroluje správnost přijatého bloku např. vyhodnocením jeho cyklického zabezpečovacího kódu. Iředtím, než příjemce informace potvrdí stavem 1 připravenost k převzetí prvního znaku dalšího bloku dat, nastaví podle výsledku kontroly správnosti přijatého bloku dat signál AF do odpovídajícího stavu. Uživatelsky definované operační vlastnosti rozhraní rohou např. předepisovat při signálizované chybě v přijatém bloku tento blok odkovat. Nehlásí-li stav 1 signálu AF chybu, pokračuje se přenosem dalšího bloku dat. Operační vlastnosti rozhraní mohou využívat odlišnou interpretaci bloku s délkou jednoho znaku buď jako řídicího znaku neto znaku se stavovou informací.

Základní funkční vlastnosti rozhraní, definované v /5/ tvoří základ definice rozhraní systému JSEP a SMEP, ve kterém je rozšířen počet datových a řídicích signálů, viz /6/ .

15.2 Stavebnicové prvky pro realizaci rozhraní mezi mikroročítáčem a vnějším zařízením

Stavebnicový prvek pro realizaci tohoto rozhraní patří k základním prvkům, kterým výrobci mikropočítáčových systémů začali vytavovat mikropočítáčové stavebnice. Popíšeme si podrobněji jeden z nich, vazební obvod 8255 firmy Intel.

Programovatelný vazební otvod 8255

Vazební obvod 8255 lze programovat tak, aby umožnil realizovat požadované funkční a operační vlastnosti rozhraní s minimálními nároky na potřebné programové vybavení mikropočítáčového systému.

Vnitřní uspořádání vazebního obvodu 8255 je uvedeno v obr.15.4.

K systémové sběrnici mikropočítáčového systému se vazební obvod připojuje prostřednictvím budiče datové sběrnice a řídicích obvodů čtení zápisu. Význam signálů rozhraní těchto bloků a systémové sběrnice mikropočítáče je uveden v tab. 15.2. Vztah těchto signálů k signálům systémové sběrnice mikropočítáče, orientované na systém 8080, např. sběrnice Multibus nebo S-100 je zřejmý z obr. 15.5. Operace zápisu datových a řídicích slov a čtení datových slov řídí adresové a řídicí signály tohoto rozhraní v soulaze s tab. 15.3.

Další skupinu signálů tvoří signály rozhraní s vnějším zařízením zpracovávané vazebními členy A, B a C. Jejich význam je dán pracovním režimem, do něhož je vazební obvod přiveden při programové inicializaci. Vazební členy A, B a C nejsou totožné. Vazební obvod A obsahuje osmibitový výstupní budič střadačového typu a osmibitový vstupní budič. Vazební obvod B obsahuje osmibitový vstupně-výstupní budič střadačového typu a osmibitový vstupní budič. Vazební člen C obsahuje osmibitový výstupní budič střadačového typu a osmibitový vstupní budič. Tento vezetní člen se v některých pracovních režimech rozděluje na dva čtyřbitové vazební členy, jejichž využití si podrobněji popíšeme v dalším textu.

Stavem 1 signálu RESET se vazební obvod 8255 převádí do základního vstupního režimu, ve kterém se všech 24 signálů rozhraní s vnějším zařízením chová jako vstupní signály. V tomto stavu zůstává vazební obvod 8255 i po přechodu signálu RESET do stavu 0.

Řídicím slovem předaným výstupní instrukcí OUT do řídicího registru (tab. 15.3), se převede vazební obvod do jednoho z následujících režimů: základního vstupně-výstupního režimu C, vzorkovacího vstupně-výstupního režimu 1 a režimu obousměnné sběrnice 2. Iředstava o rozdělení signálů rozhraní vazebního obvodu 8255 s vnějším zařízením v jednotlivých pracovních režimech je uvedena v obr. 15.5. Iří změně pracovního režimu se automaticky nuluje všechny výstupní registry i stavové příznaky.

Řídicí slovo má dva různé formáty; v prvním definuje pracovní režim vazebního obvodu, v druhém umožnuje řízení jednotlivých bitů vazebního členu C. Formát řídicího slova určuje bit D7, jak je zřejmé z tab. 15.4 a 15.5. Formát řídicího slova definujícího pra-

covní režimy umožňuje tyto režimy kombinovat pro různé skupiny výstupů, což dovoluje pružnější přizpůsobení požadavkům rozhraní s vnějším zařízením. V pracovních režimech 1 a 2 se mohou některé výstupní signály skupiny C využít jako signály žádosti o přerušení. Těmto signálům jsou přiděleny stavové bity maskování přerušení, k jejichž řízení lze využít mechanismu řízení jednotlivých bitů vazebního člena skupiny C. Pomocí těchto bitů lze maskovat žádosti o přerušení vnějších zařízení. Nastavení maskovacího bitu do stavu 1 povoluje přijetí žádosti o přerušení, stav 0 má význam opačný.

V pracovním režimu C jsou podle obr. 15.5 rozdeleny signály rozhraní s vnějším zařízením do čtyř skupin, z nichž každá může být podle tab. 15.4 definována jako skupina vstupních nebo výstupních signálů. Řídícím slovem lze tedy předepsat šestnáct různých konfigurací, jejichž přehled je uveden v tab. 15.6. Definuje-li se skupina jako výstupní, je obsah datového slova zapsaného do příslušného vazebního člena uložen v registru a je k disposici na jeho výstupu až do dalšího zápisu. Definuje-li se skupina jako vstupní, čte se takový stav jejich vstupních signálů, jaký je v okamžiku operace čtení.

Pracovní režim 1 je určen pro asynchronní přenos vstupně-výstupních datových slov, jehož řídící signály se generují technickými prostředky vazebního obvodu 8255. K přenosu datových informací slouží vazební členy A a B, ke generaci řídicích signálů slouží vazební člen C, a sice jeho bity PC3 + PC5 naleží vazebnímu čленu skupiny A, jeho bity PC0 + PC2 vazebnímu členu skupiny B ve vstupním režimu; ve výstupním režimu vazebnímu členu A naleží bity PC3, PC6 a PC7 a u člena B je přiřazení stejné jako v předchozím případě. Zbývající bity vazebního člena C mají univerzální použití. Jak výstupní, tak vstupní data se vzorkují a ukládají do registrů vazebních členů.

V pracovním režimu 1 lze předepsat čtyři různé konfigurace. Jsou-li předepsány vazební členy A nebo B jako vstupní, pak interpretaci jejich signálů a požadovaná řídící slova ilustruje obr. 15.6 a tab. 15.7.

Přenos slova zahajuje zdroj informace přechodem signálu STB do stavu 0, po němž následuje zápis vstupních dat do vstupního

registru, indikovaný přechodem signálu IBF do stavu 1, který potvrzuje informačnímu zdroji převzetí datového slova. Zdroj informace může pak zrušit datovou informaci, nesmí však předat další slovo do té doby, pokud signál IBF nepřejde do stavu 0. Pokud není přerušení maskováno stavem bitů FC4 resp. FC2 tab. 15.7, vyvolává stav signálu IBF žádost o přerušení stavem 1 signálu INTR.

Fřijme-li mikroprocesor tuto žádost, přečte datovou informaci ze vstupního registru vazebního obvodu 8255 na systémovou sběrnici na základě signálu RD a dále ji zpracuje. Nástupní hranou signálu RD se ukončuje cyklus převzetí datového slova a může se zahájit další cyklus.

Činnost vazebních členů A a B ve funkci výstupních členů ilustrují obr. 15.7 a druhá část tab. 15.7. Vzhledem k tomu, že řízení přenosu je obdobné jako v předcházejícím případě, nebude je již podrobněji popisovat. Formáty řídicích slov dovolují předepsat v pracovním režimu 1 nezávisle směr přenosu vazebního členu A a B, takže obvod 8255 může vytvořit čtyři různé konfigurace.

Pracovní režim 2 lze definovat jen pro vazební člen A, který pak uskutečňuje obousměrný asynchronní přenos datových slov. Přenos jednoho slova je řízen technickými prostředky vazebního obvodu 8255 pomocí řídicích signálů, jejichž význam odpovídá pracovnímu režimu 1. Z hlediska dalšího výkladu je podrobnější popis tohoto pracovního režimu nepodstatný a proto odkazujeme čtenáře na /9/.

Zbývá podotknout, že v pracovních režimech 1 a 2, ve kterých vazební člen C generuje nebo přijímá signály řídicí asynchronní přenos, představuje jeho obsah stavovou informaci, kterou lze převzít běžnou vstupní instrukcí. Formáty stavových slov jsou uvedeny v tab. 15.8. Jejich obsah reprezentuje stav periferních zařízení a průběh přenosu, takže jich lze využít při tvorbě obslužných programů.

15.3 Vazební obvod 8255 jako standardní paralelní asynchronní rozhraní

Uvažme možnost realizace standardního paralelního rozhraní s asynchronním přenosem pomocí vazebního obvodu 8255. Omezíme se na výklad asynchronního přenosu znaku, jak jej uvádí obr. 15.2,

neboť rozšíření programového a technického vybavení pro asynchronní přenos bloku dat podle obr. 15.3 lze již snadno odvodit. V obr. 15.8 je uvedeno zapojení obvodu 8255 pro případ, kdy vazební obvod 8255 slouží jako zdroj dat. Pro tento případ se nabízí použití vazebního obvodu 8255 v pracovním režimu 1. Jediným doplňkem technických prostředků je invertor, který je nutno vřadit mezi výstup signálu OBFA a rozhraní, aby signál SC měl polaritu předepsanou definicí funkčních vlastností rozhraní. Programové vybavení mikropočítáče pro řízení přenosu ilustruje tab. 15.9. V podprogramu inicializace se posloupností dvou instrukcí MVI a OUT převádí vazební obvod 8255 do požadovaného pracovního režimu. Vazební člen A je převeden do výstupního pracovního režimu 1, vazební člen B, který se v příkladu nevyužívá, do výstupního pracovního režimu 0. Bity PC4 a PC5, které se nevyužívají pro řízení asynchronního přenosu dat se převádějí do výstupního režimu, bity PC0 až PC2 do vstupního režimu. V podprogramu zahájení přenosu se využívá možnost přímého řízení jednotlivých bitů vazebního členu C k nastavení signálu S0 do stavu 1. Přechod signálu AC do stavu 1 vyvolává žádost o přerušení signálem INT1A. Přerušením vyvolaný podprogram předání znaku převezme znak z datového zásobníku, předá jej do vazebního členu A a aktualizuje ukazatel datového zásobníku. Asynchronní přenos znaku pak zajistí technické vybavení vazebního obvodu 8255 prostřednictvím řídících signálů, jejichž časový diagram je uveden v obr. 15.8. Program uvedený v tab. 15.9 není úplný, neboť neřeší inicializaci ukazatele datového zásobníku, testování stavu signálu A0, kterým příjemce informace potvrzuje stav připravenosti, atd.; má pouze ilustrovat využití pracovního režimu 1 ke zjednodušení programu asynchronního přenosu.

Technické a programové prostředky pro případ, v němž vazební obvod 8255 má sloužit jako příjemce dat se na první pohled zdá jí zcela analogická a odpovídající řešení je uvedeno na obr. 15.9 a v tab. 15.10. K podprogramu zahájení přenosu je třeba pouze podotknout, že pro zjednodušení v něm není uvedeno testování podmínek, za kterých může dojít k převedení signálu AO do stavu 1. Tyto podmínky charakterizující stav připravenosti příjemce informace jsou samozřejmě závislé na jeho konkrétním uspořádání. Vzhledem k tomu se v tab. 15.10 rovněž neřeší problém testování připravenosti příjemce informací během přenosů dat. Pokud bychom nevyžadovali chybové hlášení prostřednictvím signálu AE, pak by zapoje-

ní podle obr. 15.9 vyhovělo požadavkům na asynchronní přenos znaku s jistým omezením, které vyplýne z dalšího popisu činnosti. Povšimněme si však, že na rozdíl od předchozího obrázku zapojení obsahuje dva invertory. Představíme-li si, že rozhraní vazebních obvodů 8255 z obr. 15.8 a z obr. 15.9 je společné, tj. že zprostředkuje přenos dat mezi dvěma mikropočítáčovými systémy, pak za předpokladu, že připustíme odchylku od standardního paralelního rozhraní s asynchronním přenosem, a že invertory není třeba vrádit z jiných důvodů např. na základě požadované vstupní a výstupní impedance pro přenos vedením, je možné dva invertory vypustit. Jedná se o invertory vložené mezi výstup signálu OBFA a vstup signálu STBA. Invertor přizpůsobující signál IBFA signálu ACK tvoří nezbytnou součást zapojení, což představuje jistou nevýhodu při aplikaci vazebního obvodu 8255 v pracovním režimu 1.

Další nevýhodu, týkající se vstupního pracovního režimu 1 budeme demonstrovat na příkladu signálu AE, požadovaného definicí standardního paralelního rozhraní. Podle obr. 15.2 se požaduje, aby signál AE byl ve stavu 1 při AC+SC=0, a aby přišel do stavu 0 při AC.SC=1. Podprogram převzetí znaku uvedený v tab. 15.10 je sice vyvolán přerušením signálem INTRA, generovaným při stavu signálu AC+SC=0, ale jakmile se uskuteční čtení znaku z vazebního člena A do střadače, po kterém je teprve možné provést kontrolu parity převzatého znaku, přechází automaticky signál AC do stavu 1; nelze tedy signál AE generovat podle požadavků standardního paralelního rozhraní, ale tak, jak znázorňuje obr. 15.9, tj. přechází do stavu 1 při SC.AC=1 a do stavu 0 při SC+AC=0.

V podprogramu převzetí znaku v tab. 15.10 se po přerušení signálem INTRA nejdříve nastaví signál AE do stavu 0, pak se přečte znak z vazebního člena A do střadače, nastaví se příznak parity pomocí instrukce ANI a v závislosti na jeho stavu podprogram buď pokračuje zápisem převzatého znaku do datového zásobníku a aktualizací jeho ukazatele, nebo v případě zjištění chybnej parity zápisem znaku X namísto přijatého znaku do datového zásobníku, nastavením signálu AE do stavu 1 a přechodem na návěstí LAB, po kterém se aktualizuje ukazatel datového zásobníku a podprogram ukončuje svou činnost. Podprogram ukončení přenosu, uvedený v tab. 15.10, je vyplán sestupnou hranou signálu SO a zakazuje přerušení vyvolávané signálem INTRA.

Jak z předchozího výkladu vyplývá, spočívá nevýhoda pracovního režimu 1 v tom, že se signál AC generuje okamžitě po čtení obsahu vazebního členu. Tato nevýhoda se uplatňuje i v jiných situacích, např. při blokovém přenosu dat. Podprogram převzetí znaku např. v tomto případě testuje, zda převzatý znak neoznačuje konec bloku dat. I po rozzeznání takového znaku se generuje signál AC, který zdroj informace např. snímač děrné pásky vyhodnotí tak, že připraví další znak patřící již k dalšímu bloku dat. Uvedené chování známená komplikace při návrhu programového zabezpečení asynchronního přenosu bloků dat.

Proto je mnohdy výhodnější řešit vstupní paralelní rozhraní s asynchronním přenosem pomocí pracovního režimu 0 i za cenu programového řízení signálu AC. V obr. 15.10 a tab. 15.11 je naznačeno jedno z možných řešení, ve kterém se změny stavu řídicích signálů SO a SC zpracovávají stejně jako v předchozích příkladech, systémem přerušení. Na rozdíl od předchozích řešení je třeba v tomto případě doplnit technické prostředky o obvody pro zpracování těchto signálů. V příkladech podprogramů v tab. 15.11 se předpokládá, že systém přerušení reaguje na změny stavů signálů SC a SO. Pokud by reagoval pouze na úrovně signálů, bylo by třeba podprogramy doplnit o maskování těch stavů signálů, které v daném okamžiku není třeba zpracovávat.

Zápis programu je na základě předchozích příkladů již snadno pochopitelný. Liší se pouze ve zpracování chyby parity. Má-li se signál AE generovat podle definice standardního paralelního rozhraní s asynchronním přenosem podle obr. 15.2, pak je chybu parity třeba zaznamenat do příznaku chyby v podprogramu převzetí znaku a stav tohoto příznaku testovat v podprogramu ukončení převzetí znaku. V tab. 15.11 není uveden podprogram ukončení přenosu, vyvolávaný sestupnou hranou signálu SO, neboť jeho činnost je závislá na konkrétním řešení systému přerušení.

16. SBĚRNICE A SYSTÉMY PRO MÍSTNÍ PŘENOS DAT

16.1 Sběrnice IMS-2

Sběrnicový systém IMS-2 je na rozdíl od sběrnicových systémů, popsaných v předcházejících odstavcích určen pro propojení zařízení a přístrojů na větší vzdálenost, asi do 20 metrů. Původně byl navržen firmou Hewlett-Packard jako rozhraní pro propojení zařízení používaných v přístrojové technice. Operace na sběrnici lze řídit řídicí stanicí, např. počítačem, stolním počítačem nebo kalkulátorem anebo kterýmkoliv ze zařízení připojeným ke sběrnici a vytaveným potřebnými technickými nebo programovými prostředky. Propojení sběrnicových systémů IMS-2 je podstatně jednodušší než propojení systémových stěrnic minipočítačů nebo mikropočítačů; přenos dat je však podstatně pomalejší a realizace požadovaných funkčních vlastností je též mnohem složitější. V rámci RVFP je definice sběrnicového systému IMS-2 ve stadiu projednávání. Byla však již přijata Mezinárodní komise pro elektrotechniku pod názvem IEC 488. Odpovídající americká norma je IEEE 488.

16.1.2 Funkční vlastnosti

K sběrnici IEC 488 může být připojeno maximálně 15 zařízení. Sběrnici tvoří 16 vodičů, které se po funkční stránce dělí do třech skupin. Přehled signálů sběrnice IEC 488 je uveden v tab. 16.1. Po obousměrné datové sběrnici se přenášejí data paralelně byte po byte. Přenášená data mohou mít buď význam adres, dat programů (např. výsledků měření), příkazů a povelů určených jednotce nebo všem zařízením připojeným ke sběrnici, anebo mohou obsahovat stavové informace. Data vyskytující se na datové stěrnici se interpretují v závislosti na stavu řídicího signálu ATN. Jestliže je signál ATN ve stavu 1, pak data na sběrnici mají význam adres

nebo povelù, které jsou určeny všem zařízením. V opačném případě jsou data na sběrnici určena těm zařízením, které získaly oprávnění prostřednictvím sběrnice komunikovat.

Při funkčním popisu sběrnice se zavádí následující pojmy: mluvčí, posluchač a řidič. Zařízení, které předává data na sběrnici, se nazývá mluvčí. Zařízení, která tato data přijímají, se nazývají posluchači. Provoz na sběrnici řídí zařízení, nazývané řidič. Ten je oprávněn předávat povely určené všem zařízením připojeným ke sběrnici, a přijímat stavové informace předávané těmito zařízeními.

Asynchronní přenos po sběrnici se řídí stykovými signály řízení přenosu, uvedené v tab. 16.1 a v obr. 16.1. Sigrál \overline{DAV} předává mluvčí a jeho stavem 0 potvrzuje platnost dat, která předává na datovou sběrnici. Posluchač potvrzuje přijetí těchto dat signálem NRFD ve stavu 0 a signálem NDAC ve stavu 1. Jestliže se v daném okamžiku jen jedno zařízení připojené ke sběrnici ve funkci aktivního posluchače, pak tento stav signálu NDAC potvrzuje převzetí dat. Na základě těchto stavů signálů přiřazuje mluvčí signálu \overline{DAV} stav 1. Je-li posluchač schopen převzít další byte, přiřazuje signálům NRFD stav 1 a signálu NDAC stav 0, čímž se ukončuje posloupnost řídicích signálů předání jednoho bytu. Jestliže je v daném okamžiku několik zařízení připojených ke sběrnici v roli aktivních posluchačů, pak každé z nich předává stejnou posloupnost řídicích signálů. Jediný rozdíl spočívá v tom, že se jednotlivé fáze cyklu přenosu dat neukončí dříve, dokud všechna zařízení nepředají potvrzující signál. Díky tomu, že signály NRFD a DAC jsou předávány na sběrnici prostřednictvím budíčů s otevřeným kolktorem a díky interpretaci jejich významu není připravenost pro příjem dat rozeznána mluvčím dříve než jsou všichni posluchači k příjmu připraveni. Podobně se nepotvrdí převzetí dat do okamžiku, dokud data nepřevezmeme nejmalejší zařízení, patřící mezi posluchače. Rychlosť přenosu se přizpůsobuje nejmalejšímu posluchači, který se podílí na konverzaci. Tato vlastnost je jedním z charakteristických rysů každého asynchronního přenosu. Je třeba si však uvědomit, že pomalejší zařízení připojené ke sběrnici neovlivňuje nepříznivě rychlosť přenosu, pokud není v daném okamžiku oprávněno se na konverzaci podílet.

Inicializační signál IFC generuje řidič tehdy, chce-li uvést sběrnicový systém do klidového stavu. Požaduje se, aby zařízení připojené ke sběrnici IMS-2 ukončila přechod do klidového stavu nejpozději v době 100 μ s po zahájení inicializace, tj. po přechodu signálu IFC do stavu 0. Přechod do klidového stavu se ovšem týká jen těch vnitřních stavů zařízení, které souvisí s jeho rozhraním se sběrnicí. Lze jej řešit programovým vybavením mikropočítáčového systému, ale z požadavku na dobu odezvy na signál IFC vyplývá, že stav tohoto signálu musí zpracovat systém přerušení tohoto mikropočítáče.

Řídicí signál ATN umožnuje rozlišení mezi povelovým režimem ATN ve stavu 0 a textovým režimem ATN ve stavu 1. Řidič může kdykoliv generovat signál ATN (přiřadit mu stav 0), aby mohl předepsat operace, které mají na sběrnici protíkat. Ke generaci signálu ATN může dojít např. uprostřed cyklu přenosu dat mezi mluvčím a posluchači. Požaduje se, aby odezva zařízení připojitelných ke sběrnici byla krátká než 200 ns. V této době musí všechna zařízení ohlásit připravenost ke komunikaci s řidičem, a to stavem 1 signálů NRFD a DAV. Mluvčí musí současně uvolnit datovou sběrnicu. Rychlá odezva zařízení umožňuje, aby řidič převzal rychle stavové informace ze zařízení, která jsou vybavena prostředky pro výkonání povelu paralelní výzvy. Při výkonávání tohoto povelu se stavové slovo předává bez potvrzujících řídicích signálů. Zařízení, která nejsou vybavena prostředky pro výkonání tohoto povelu, nesmí blokovat sběrnicu.

Všechna zařízení musí být vybavena technickými prostředky pro zpracování signálu ATN. I když je rozhraní implementováno mikropočítáčovým systémem, zpracovávají tyto technické prostředky signál ATN nezávisle, bez intervence mikropočítáčového systému, jehož rychlosť je v tomto případě nedostatečná.

Na výkonávání ostatních povelů uskutečňovaných asynchronním přenosem nejsou kladený omezující časové požadavky, t-kže je může výkonávat programové vybavení mikropočítáčového systému. Řidič nepokračuje dalším povelom do té doby, než všechna zařízení nepotvrdí převzetí povelu prostřednictvím řídicích signálů (obr. 16.1).

Signál EOI má dva významy. V povelovém režimu slouží ve spojení se signálem ATN k předání paralelní výzvy těm zařízením, kte-

rá jsou vytavena technickými prostředky pro její vykonání. Paralelní výzvu může sdílet až osm zařízení připojených ke sběrnici, z nichž každé má přiřazen jeden z datových signálů. Jestliže tedy řidič přiřadí signálům ATN a EOI stav 0, každé z těchto zařízení předává jednobitovou stavovou informaci prostřednictvím přiřazeného datového signálu. Řidič tak může např. identifikovat, které ze zařízení vyvolalo prostřednictvím signálu SRQ přerušení. Paralelní výzva je jedinou metodou, která umožňuje řadiči rychle převzít stavovou informaci ze zařízení. Obr. 16.2 znázorňuje řešení obvodů pro zpracování paralelní výzvy. Mikropočítač, který řídí činnost zařízení, předává prostřednictvím výstupního člena adresu paralelní výzvy, která je závislá na konfiguraci systému a která může být i předepsána dálkově. Současně se předává stavový bit. Jestliže se na vstupech ATN a EOI objeví současně stav 0, pak se tyto signály zpracované Schmittovými klopnými obvody přivedou na dva vstupy trojvstupového logického součinového člena a na jeho třetí vstup se přivede stavový bit. Výstup logického člena je spojen s výběrovým vstupem dekodéru dvojkového kódu na kód 1 z 8. Jestliže je stavový bit ve stavu 1 a přijde paralelní výzva, předává se na adresovaný datový výstup (jeden ze vstupů-výstupů DIO1 až DIO8) stav 0.

V datovém režimu slouží signál EOI mluvčímu k označení posledního předávaného bytu dat. Posluchači, kteří stav tohoto signálu monitorují, mohou na základě jeho stavu 0 ukončit přenos.

Signál REN generuje řidič. Jestliže je tento signál aktivní a systém je v povelovém režimu, tj. stav signálu ATN je 0, pak přejdou vybraná zařízení připojená ke sběrnici z režimu ručního ovládání ovědacími prvky na panelech do režimu dálkového ovládání. Signál REN musí zůstat ve stavu 0 po celou dobu, kdy se požaduje dálkové ovládání. Po přechodu do režimu dálkového ovládání pracují všechny dálkově ovládané přístroje jako posluchači, alespoň z počátku. Tento postup dává předpoklady k automatickému testování systému. Po přechodu signálu REN do stavu 1 se musí všechny přístroje vrátit do režimu ručního ovládání během 100 µs, což může zajistovat podprogram obsluhy přerušení mikropočítače, vyvolaný přechodem signálu REN do stavu 1.

Signál SRG je žádostí o obsluhu a vyvolává přerušení činnosti řidiče. Řidič rozezná zařízení, které přerušení vyvolalo buď para-

lelní výzvou, popsanou v předcházejícím textu, nebo sériovou výzvou. Sériová výzva se uskutečňuje posloupností povelů, které vyzvedávají stavovou informaci z jednoho zařízení po druhém. Tento postup je samozřejmě pomalejší než paralelní výzva, umožňuje však převzít podrobnou stavovou informaci a zvolit obsluhu přerušení žádajícího zařízení, neboť při sériové výzvě je k dispozici celá datová sběrnice, kdežto při paralelní výzvě jen jeden její bit. Navíc lze algoritmus sériové výzvy implementovat programem mikropočítače.

Ve sběrnicovém systému IMS-2 se předává celkem 47 dálkových zpráv, jejichž posloužitelnosti jsou předepsány operačními vlastnostmi. Všechna zařízení ke sběrnici připojitelná musí být schopna tyto zprávy generovat, přijímat a zpracovávat. Některé z těchto dálkových zpráv, tj. zpráv předávaných po sběrnici, jsou uvedeny v tab. 16.2. V povelech se nevyužívá bitu DIO8 datové sběrnice a lze jej využít jako paritní bit povelu; zařízení musí být pak odpovídajícím způsobem vybavena. Všeobecné povely jsou určeny všem zařízením, která jsou připojena ke sběrnici. Tak např. povel nulování zařízení, který má formát X 0 0 1 0 1 0 0 se využívá k přivedení všech zařízení připojených ke sběrnici do klidového stavu. Na rozdíl od nulování, vyvolaného stavem '0 signálu IFC se však v tomto případě inicializuje celé zařízení, nikoliv jen jeho rozhraní se sběrnicí IMS-2. Adresované povely jsou určeny pouze zařízením, která jsou v daném okamžiku nastavena jako posluchači. Tak např. povel GET, skupinový povel ke spuštění neuvedený v tab. 16.2 s formátem X 0 0 0 1 0 0 0, inicializuje současně předem programovanou činnost všech posluchačů. Povel UNL ruší nastavení zařízení jako posluchačů, ke kterému došlo po předchozím povelu MIA.

Tak např. posloupnost dálkových zpráv UNL, MIA, MTA, DAB₁, DAB₂, DAB_n předepisuje měření napětí adresovaným voltmetrem ve zvoleném rozsahu. V tomto případě ruší povel UNL všechny předchozí posluchače. Povel MIA obsahuje adresu voltmetu a určuje jej jako posluchače. Povel MTA obsahuje adresu řidiče, kterým se sám určí jako mluvčí. Posloupnost datových bytů DAB₁ až DAB_n určuje měřicí rozsah voltmetu a další údaje, které jsou závislé na samotném voltmetu. Obdobnou posloupností UNL, MTA, MIA, DAB₁, DAB_n lze předepsat jako příjemce naměřených výsledků jiné připoje-

né zařízení, jehož adresa je obsažena v povelu MTA.

V každém případě musí být posloupnost povelů zpracována technickými nebo programovými prostředky rozhraní se sběrnicí IMS-2. Zpracování spočívá v nastavení požadovaných vnitřních stavů a vytvoření požadovaných vnitřních signálových cest, kterým jsou pak datové informace obsažené v datových bytech DAB předávány přímo z informačního zdroje např. vyrovnávací paměti voltmetru příjemci informace např. vyrovnávací paměti tiskárny. Nezáleží ovšem na tom, jakými prostředky jsou požadované vlastnosti rozhraní implementovány, zda pomocí diskrétních logických obvodů, obvodů s rozsáhlou integrací anebo mikropočítáčem.

Sériová výzva, o které jsme se zmínili v předchozím textu, se uskutečňuje posloupností povelů UNL, SFE, MTAl, MTA2 ... MTAn. Zařízení, jehož adresa je obsažena v povelu MTAl až MTAn odpovídá buď pouze stavovým slovem RQS nebo posloupnosti RQS, STB. Po převzetí stavového slova identifikujícího příčinu přerušení musí řidič zahájit odpovídající podprogram obsluhy. Použití mikroprocesoru jak na straně řidiče, tak i v samotných zařízeních zjednoduší stěr dat, generaci požadovaných posloupností a testování specifických podmínek, tj. všech součástí procesu dialogu mezi řidičem a zařízeními připojenými ke sběrnici.

Sběrnicový systém zprostředkovává komunikaci mezi třemi částmi zařízení, které uskutečňují základní činnosti. Z tohoto hlediska je vhodné rozdělit funkce zařízení do dvou kategorií: přístrojové funkce a funkce rozhraní. K přístrojovým funkcím, které nejsou v standardu sběrnicového systému IMS-2 definovány, patří např. řízené měření napětí apod. Funkce rozhraní generují a zpracovávají zprávy; jejich úkolem je zřídit a zrušit tok informací mezi dvěma nebo více zařízeními. Sběrnicový systém musí zprostředkovat přenos přístrojových zpráv např. DAB i zpráv pro řízení rozhraní, což ilustruje (obr. 16.3.) Vztahy mezi funkcemi rozhraní a přístrojovými funkcemi i jejich vzájemné vztahy musí být přesně definovány. Metodou používanou ve standardu IMS-2 je reprezentace těchto funkcí pomocí vývojového diagramu. Každý uzel vývojového diagramu reprezentuje vnitřní stav systému. K orientovaným hranám vývojového diagramu jsou připsány podmínky, za kterých může dojít k přechodu do jiného vnitřního stavu.

Příklad vývojových diagramů přístrojových funkcí a funkcí rozhraní je uveden v obr. 16.4, ve kterém jsou stavy přístrojových funkcí označeny dfsn, stavy funkcí rozhraní IFSN. RMN jsou dálkové zprávy a Imn jsou místní zprávy. Vývojové diagramy jednoznačně určují, za jakých podmínek může dojít ke změnám přístrojových funkcí a funkcí rozhraní. Výhodou reprezentace pomocí vývojových diagramů je jednoznačnost a nezávislost na implementaci funkcí.

Standard IMS-2 definuje deset funkcí rozhraní. Ně každý přístroj připojený ke sběrnici musí být všemi funkcemi vybaven. K základním komunikačním funkcím patří první čtyři uvedené v tab. 16.3: řízení asynchronního přenosu zdroje SH, příjemce AH, mluvčí T a posluchač L. K inicializačním funkcím patří nulování zařízení DC a inicializace základních funkcí DT. Funkce vyžádané obsluhy SR a paralelní výzvy umožňuje předání stavové informace buď společnou datovou sběrnici nebo vyhrazeným bitem této sběrnice. Funkce RL umožňuje volit mezi místním ovládáním z panelu přístroje nebo dálkovým ovládáním. Konečně funkce řidiče umožňuje předávat všeobecné a adresné povely, přijímat žádosti o obsluhu a odpovědi na paralelní výzvu a generovat řídící signály IFC a REN.

Jako příklad definice funkce posluchače si uvedeme její zjednodušený vývojový diagram (obr. 16.5). K inicializaci této funkce tj. k přechodu do klidového stavu LIDS dochází jednak při zapnutí zařízení, jednak při přijetí signálu IFS, a to z jakéhokoliv aktivního stavu. Jestliže je signál IFS ve stavu 0 buď je zapnut ovládací prvek lon (jen poslouchej) na panelu, anebo zařízení přijme vlastní adresu posluchače a současně je jeho funkce AH řízení asynchronního přenosu příjemce ve stavu převzetí dat ACDS, pak přejde funkce posluchače do stavu LADS. V tomto stavu však není posluchač oprávněn přijímat zprávy ze sběrnice. Teprve až řidič ukončí posloupnost povelů MTA a MLA a určí tak mluvčího a všechny posluchače, přejde signál ATN do stavu 0 a funkce posluchače do aktivního stavu LACS. Zařízení, jehož rozhraní se sběrnici IMS-2 přejde do tohoto stavu, je oprávněno přijímat data ze sběrnice. Přístrojové funkce zařízení tak mohou být řízeny prostřednictvím přístrojových zpráv předávaných po sběrnici. Z aktivního stavu LACS přechází zařízení do klidového stavu LIDS až tehdy, přijme-li povel UNL, který generuje řidič.

Standard sběrnicového systému IMS-2 obsahuje vývojové diagramy všech deseti funkcí rozhraní.

16.1.3 IMS-2 a mikropočítáčové systémy

Mikropočítáčové systémy jsou předurčeny k realizaci funkcí rozhraní a přístrojových funkcí sběrnicového systému IMS-2 z několika důvodů. Jednak je šíře datových cest v obou systémech stejná. Dále se v otou případech používá asynchronní metoda přenosu, která dovoluje přizpůsobení přenosové rychlosti na sběrnici technickým možnostem mikropočítáčových systémů. Technické a programové vybavení mikropočítáčových systémů umožňuje zpracování celých informačních bloků. Velmi výhodně lze rovněž využít možnosti vybavení mikropočítáčového systému vícenásobnými vstupně-výstupními členy.

Při zpracování informačních bloků s posloupnostmi dálkových zpráv lze tyto posloupnosti při vysílání předem připravit v operační paměti mikropočítáčového systému a pak je postupně vysílat pomocí programových indexových smyček. Při příjemu lze zprávu uložit v paměti pro pozdější zpracování nebo přijímané byty ihned dekódovat, popřípadě provádět jejich konverzi do jiného kódu. Podle potřeby je možné upravit formáty vysílaných i přijímaných zpráv.

K tomuto zpracování se výhodně využívá bytově orientovaných aritmetických a logických operací, kterými je vybavena většina mikropočítáčových systémů a instrukcí pro práci s tabulkami obsahujícími konverze kódů.

Vícenásobné vstupně-výstupní členy usnadňují třídění a zpracování odlišně kódovaných zpráv určených různým přístrojovým funkcím, neboť není nutné vybavit přístroj pro každý druh přijímané zprávy úplnými technickými prostředky, ale pouze speciálními vstupy. Mikropočítáčový systém rozezná na základě přístrojové adresy jak má zprávu zpracovat.

Při realizaci funkcí sběrnice IMS-2 mikropočítáčovým systémem je třeba vzít v úvahu především dva činitele: potřebnou přenosovou rychlosť a cenu. Při požadavku velké přenosové rychlosti je mnohdy nezbytné implementovat přístrojové funkce i funkce roz-

hraní technickými prostředky. Mnohdy lze však využít metody přímého styku s pamětí mikropočítače, jejíž realizaci většina výrobců mikropočítačů usnadňuje tím, že dodává zvláštní stavebnicový prvek, řadič přímého styku s pamětí.

Ekonomická výhodnost využití mikropočítačového systému oproti pevně řízenému systému je závislá především na přístrojových funkcích a funkcích rozhraní, které je třeba v uvažovaném zařízení realizovat. Jen některé funkční vlastnosti vyžadující časovou odezvu kratší než 200 ns je nutné realizovat technickými prostředky. Ve všech ostatních případech má návrhář systému možnost si řešení zvolit. Ve většině případů představuje cenově výhodnější alternativu implementace funkcí rozhraní i přístrojových funkcí programovým vybavením mikropočítačového systému, což ilustruje tab.

16.3. Funkce řízení asynchronního přenosu zdroje i příjemce lze sice řešit výlučně programovými prostředky, ale této možnosti se obvykle nevyužívá, neboť by se tak výrazně snížila přenosová rychlosť. Všechny ostatní funkce lze řešit programovými prostředky jak uvádí levý sloupec této tabulkы. Toto řešení však není z funkčního hlediska optimální. Při vyšších požadavcích na přenosové rychlosti je výhodnější řešit většinu funkcí rozhraní technickými prostředky a jen některé z nich např. DC, DT a přístrojové funkce programovým vybavením mikropočítačového systému, jak naznačuje pravý sloupec v tab. 16.3. Programové vybavení řeší v tomto případě především následující úkoly: dekódování přístrojových funkcí, jejich vykonávání a vazbu mezi funkcemi rozhraní DC a DT a přístrojovými funkcemi.

Vzhledem k tomu, že sběrnicový systém IMS-2 již přijalo více než stovku výrobců, dá se v nejbližší budoucnosti očekávat, že mikropočítačové stavebnice budou doplněny o další stavebnicový prvek, který bude realizovat funkce rozhraní s touto sběrnicí. Např. firma Intel uvádí předběžné údaje o stavebnicovém prvku 8291, který je určen k realizaci funkce mluvčího - posluchače. Tento prvek má umožnit komunikaci sběrnice IEC 488 s mikropočítačem řízenou programem nebo řadičem přímého styku s pamětí 8259. Jeho technické prostředky mají umožnit zpracování paralelní výzvy a další funkce rozhraní. K implementaci funkce řadiče je určen další stavebnicový prvek 8292. Návrhář systému se pak může věnovat návrhu programového vybavení implementujícího přístrojové funkce

a jejich vazbu s funkcemi rozhraní.

16.2 Systém CAMAC

Systém CAMAC (Computer Aided Measurement and Control) definuje standardní rozhraní mezi přístrojovými moduly. Standardní moduly s jednoduchou nebo vícenásobnou šířkou lze zasunout do standardního rámu, který je vybaven konektory, propojenými standardní sběrnicí. Mechanické uspořádání systému CAMAC je znázorněno na obr. 16.6. Prostřednictvím sběrnice "Dataway" komunikují moduly vzájemně mezi sebou a mezi řadičem rámu, jehož prostřednictvím mohou jednotlivé moduly komunikovat s počítačem. Organizace sběrnice je uvedena v tab. 16.4. Adresy jednotlivých modulů jsou dekódovány logickými obvody řadiče, který vybírá jeden z 24 modulů rámu pomocí výběrového signálu N. Požadavek na obsluhu může předat kterýkoliv z modulů svým stavovým signálem LAM (Look At Me).

Řadič rámu určený pro zprostředkování styku modulů s počítačem obsahuje obvody pro výběr modulů, dekódování řídicích signálů a povelů, řídicí a stavové registry a obvody systému přerušení. Jeho algoritmy činnosti při řízení různých typů přístrojových modulů jsou poměrně složité, takže je výhodné řadič realizovat mikropočítáčovým systémem. Přenosová rychlosť sběrnice CAMAC je 106 slov/s; délka slova je 24 bitů. Při vyšších požadavcích na rychlosť přenosu je nutné použít vyrovnávací paměti.

V rozsáhlejších systémech CAMAC se používá hierarchická struktura sběrnic, vyznačená na obr. 16.7. Sběrnice "Branch Highway" může být řešena podle požadavků na rychlosť přenosu a na vzdálenost mezi počítačem a rámy systému CAMAC buď se sériovým nebo paralelním přenosem. Oba tyto typy sběrnic jsou v systému CAMAC definovány.

Mnoho komerčně vyráběných přístrojů je vybaveno rozhraním podle definice systému CAMAC. Výrobci minipočítáčů a mikropočítáčů však většinou nevybavují své výrobky potřebnými technickými prostředky. Důvodem je podstatně složitější rozhraní systému CAMAC ve srovnání s běžnými osmi nebo šestnáctibitovým rozhraním mikropočítáčů a minipočítáčů. Očekává se, že realizace řadiče sběrnice pomocí mikropočítáče přinese zlepšení této situace. Zatím zůstává využití systému CAMAC omezeno na náročné přístrojové systémy.

17. DÁLKOVÝ PŘENOS DAT

Komunikace mezi mikropočítačovými systémy se na větší vzdálenosti většinou uskutečňuje prostřednictvím datových spojů telekomunikačních sítí. Mikropočítačové systémy jsou vybaveny technickými prostředky, např. komunikačními adaptéry, které umožňují realizovat doporučené nebo normalizované rozhraní pro připojení k datovým spojům. Jejich programové vybavení dovoluje realizovat řídicí postupy, tj. soustavu pravidel pro navázání a zrušení spojení a výměnu zpráv. Předtím, než se budeme věnovat technickým a programovým prostředkům mikropočítačových systémů pro dálkový přenos dat, seznámíme se se základními telekomunikačními prostředky přenosu dat, se strukturou sítí datových spojů a s jejich řízením.

17.1 Telekomunikační prostředky a metody přenosu dat

K základním typům telekomunikačních spojů, které zprostředkují přenos informací mezi dvěma nebo více místy, patří telegrafní, telefonní nebo datové spoje. Spoj sestává z několika okruhů řazencích za sebou. Okruh je souhrn přenosových prostředků k vytvoření přímého obousměrného spojení, zahrnující pouze kanály pro směr tam a zpět. Kanál je souhrn prostředků, které zajistují přenos jedním směrem. Jestliže jsou kanály pro oba směry realizovány odděleně, mluvíme o čtyřdrátovém okruhu, v opačném případě mluvíme o dvoudrátovém okruhu.

Spoje se vždy sestavují z okruhů stejného typu. Mohou být buď komutované nebo pevné. Pevný spoj se sestavuje na předem dohodnutou dobu, komutovaný spoj pouze na dobu potřebnou k přenosu. Komutovaný spoj se sestavuje na základě volby vyslané volajícím účastníkem. Po ukončení komunikace se na základě signálu vybave-

ní spoj ruší. Komutované sítě mají obvykle hierarchickou strukturu, ve které se vytváří komutovaný spoj mezi dvěma účastníky z určité oblasti prostřednictvím ústředny nižší úrovně. Při spojení účastníků z různých oblastí se na vytváření komutovaného spoje podílí ústředny vyšší úrovně.

Podle směru přenosu se spoje dělí na spoje se simplexním, poloduplexním a duplexním provozem. Simplexní spoje umožňují přenos dat jen jedním směrem. Poloduplexní spoj umožňuje přenos dat oběma směry, ale nikoliv současně. Jen jeden z účastníků může v daném okamžiku vysílat. Pokud by oba účastníci vysílali současně, došlo by k znehodnocení přenášené informace. Příkladem poloduplexního spoje je dvoudrátový dálnopisný spoj. Duplexní spoj umožňuje současný přenos dat oběma směry. Příkladem duplexního spoje je čtyřdráťový telegrafní nebo telefonní spoj, popřípadě i dvoudrátový telefonní spoj, ovšem za předpokladu, že se např. rozdělí jeho kmitočtové pásmo na dvě části, které slouží pro přenos jedním a druhým směrem.

Podle typu přenášeného signálu se spoje rozdělují na spoje analogové a číslicové. Analogové spoje mohou přenášet spojitý signál, číslicové spoje pouze dvoustavový signál.

Vlastnosti telekomunikačních spojů a dalších telekomunikačních prostředků a metody přenosu musí vyhovovat mezinárodním doporučením mezinárodního poradního sboru pro telegrafii a telefonii CCITT, které mají zajistit možnost přenosu dat při mezinárodním telekomunikačním spojení. V případech, kdy je nutné doporučení a normy respektovat na úrovni návrhu mikropočítačového systému, uvedeme odkaz na příslušné doporučení CCITT, popřípadě s ním související normy, vydávané mezinárodní normalizační organizací ISO, popřípadě mezinárodní komisi pro elektrotechniku IEC.

17.1.2 Metody přenosu dat po telekomunikačních spojích

Metody přenosu jsou určeny vlastnostmi telekomunikačních spojů, vlastnostmi měničů signálů, které přizpůsobují datový signál vlastnostem použitého telekomunikačního spoje a vlastnostmi koncových zařízení, které spolu komunikují.

Podle směru přenosu se dělí přenos na simplexní, poloduplexní a duplexní. Přitom však nezáleží na vlastnostech samotného te-

lekomunikačního spoje. Simplexním přenosem mínime přenos, při kterém jsou přenášena data jen jedním směrem; přitom však mohou být informace o průběhu přenosu předávána směrem opačným, takže k uskutečnění přenosu používáme duplexní telekomunikační spoj.

Při duplexním přenosu jsou data přenášena oběma směry. K přenosu informací o průběhu přenosu slouží buď datový kanál pro opačný směr v tomto případě se informace o průběhu přenosu v jednom směru vkládá do toku dat, vysílaných opačným směrem anebo jsou vyhrazené zvláštní kontrolní kanály pro přenos těchto informací.

Při poloduplexním přenosu se data přenášejí v daném okamžiku jen jedním směrem.

Jestliže se data přenášejí po blocích zabezpečených proti chybám, provádí přijímací stanice kontrolu správnosti přijatého bloku. Při duplexním provozu, kdy se jednotlivé bloky dat vysílájí bezprostředně za sebou, musí informace o správnosti přijatého bloku obsahovat číslo bloku, ke kterému se vztahuje, neboť se předává v době, kdy se již vysílá další blok. Při poloduplexním provozu se bloky předávají postupně a před vysíláním dalšího bloku se čeká na potvrzení správnosti předcházejícího bloku, takže jako kontrolní informace postačí buď kladné nebo záporné potvrzení. Tak se zjednoduší i řídicí postupy pro provoz při poloduplexním přenosu oproti postupům pro provoz při duplexním přenosu.

Podle počtu současně přenášených informací se dělí přenos na paralelní a sériový. Při paralelním přenosu se obvykle nepoužívá paralelních vedení, ale přenosové pásmo telekomunikačního spoje se rozdělí na potřebný počet podkanálů paralelním modelem.

Podle způsobu znakové synchronizace se dělí přenos dat na přenos asynchronní a synchronní. Při asynchronním přenosu se do vysílaných dat vkládají synchronizační bity nebo znaky, které slouží k udržení synchronismu přijímací stanice a vysílání.

Při arytmickém přenosu, který je zvláštním případem asynchronního přenosu, se synchronizační bity doplňují ke každému přenášenému znaku. Při synchronním přenosu se synchronizační znaky nebo synchronizační sledy vkládají obvykle jen na začátku každého bloku a přijímací stanice musí udržet synchronizaci s vysílací stanicí po dobu vysílání celého bloku dat. Data se v tomto případě vysílají tak, že znaky za sebou bezprostředně následují. Délka bloku

se pohybuje od několika znaků do několika set znaků a jeho délka obvykle odpovídá délce informačního bloku paměťového media, ze kterého je čten nebo do kterého je zapisován. Výhodou synchronního přenosu je dosažení větší přenosové rychlosti než při asynchronním nebo arytmickém přenosu, neboť mezi přenášené znaky není nutné vkládat synchronizační znaky ani mezery, které snižují přenosovou rychlosť, tj. počet znaků, přenášených za jednotku času. Formát synchronního přenášeného bloku dat je buď pevný nebo proměnný. Výhodnější je proměnný formát, neboť dovoluje přizpůsobení délky bloku délce předávané zprávy. V tomto případě je začátek a konec zprávy označován zvláštními znaky. Typický formát bloku s proměnnou délkou je uveden v tab. 17.1.

Při přenosu dat je třeba zajišťovat nejen synchronizaci znakovou, ale i synchronizaci bitovou, která zajišťuje vzorkování bitů přenášené informace v optimálním časovém okamžiku. Bitovou synchronizaci při synchronním přenosu zajišťují obvykle synchronní modemy, které vyhodnocují změny stavu signálu a předávají koncovému zařízení informaci o tom, kdy má signál vzorkovat.

Při arytmickém přenosu se bitová i znaková synchronizace odvozuje od prvního přechodu rozběhového prvku, jak znázorňuje obr. 17.1. Každý znak při arytmickém přenosu se skládá ze tří částí: z rozběhového prvku, pěti nebo osmi datových prvků, z nichž jeden může mít význam paritního bitu a z jednoho až dvou závěrečných prvků. V obr. 17.1 jsou uvedeny formáty znaku při arytmickém přenosu s pětiprvkovou abecedou CCITT2 a s osmiprvkovou abecedou CCITT5. Jakmile přijímač zjistí rozběhový přechod, začne odměřovat časové intervaly umožňující vzorkovat stav signálu v optimální době, tj. uprostřed doby vymezené pro přenos jednoho prvku. K tomu se např. využívá vnějšího hodinového signálu, jehož perioda je kratší než doba, vynaložená pro přenos jednoho prvku, což znázorňuje obr. 17.2, ve kterém je doba $T_d = 16 T_h$. Přijímač vzorkuje v klidovém stavu stav přijímaného signálu při každé nástupní hraně hodinového signálu T_h . Přechod signálu do stavu 0 se pokládá za rozběhový prvek až tehdy, jestliže byl detekován při devíti za sebou následujících periodách hodinového signálu. Tímto způsobem se zvýší odolnost přenosu proti poruchám, neboť se sníží pravděpodobnost, že porucha bude interpretována jako rozběhový prvek.

Se vzrůstající rychlostí přenosu však tato odolnost proti rušení klesá, neboť vzrůstá pravděpodobnost, že porucha bude trvat dosta- tečně dlouho. Vyhodnotí-li přijimač přechod přijímaného signálu jako rozběhový prvek, vzorkuje stav signálu po celistvých násob- cích dob $T_d = 16 \text{ Th}$, a vyhodnocuje jednotlivé bity datového signá- lu, jak znázorňuje obr. 17.3. Jako další zabezpečení vyhodnocuje i závěrné prvky a jejich nesprávný příjem se vyhodnotí jako chyba přenosu, označovaná jako chyba rámce.

Mezinárodní abecedy č. 2 a č. 5

Při dálkovém přenosu dat se používá mnoho různých kódů a abe- ced. Kódem rozumíme soubor dvojkových posloupností nebo soubor pravidel pro jejich tvorbu. Každému prvkůmu kódu můžeme přiřadit znak. Přiřazení znaků určitému kódu označujeme jako abecedu. Při dálkovém přenosu dat má největší význam mezinárodní abeceda č. 2 a mezinárodní abeceda č. 5, která se používá pro přenos dat po telefonní a telexové síti. Nevýhodou mezinárodní abecedy č. 2 je je- jí poměrně malý počet znaků, takže není pro aplikaci v systémech zpracování dat nevhodnější, neboť počítače pracují s rozsáhlejší- mi abecedami, jejichž znaky jsou přiřazeny osmibitovému kódu. Přesto však tuto abecedu uvádíme v tab. 17.2 a tab. 17.3, neboť se s ní setkáme při připojení zařízení určených pro dálkopisné sí- tě k mikropočítačovým systémům.

V systémech dálkového zpracování dat se používá mezinárodní abeceda č. 5 podle doporučení CCITT V3, normalizované jako abeceda ISO7. Její znaky jsou přiřazeny sedmibitovému kódu, zabezpečenému osmým paritním bitem. Mezinárodní referenční verze této abecedy je uvedena v tab. 17.4. Kromě této verze se používají národní a ap- likačně orientované verze, které mohou volitelným pozicím této ta- bulky přiřadit jiné symboly. První dva sloupce tab. 17.4 jsou vy- hrazeny řídicím znakům. V dalších sloupcích je uveden základní soubor grafických znaků písmena, číslice, symboly. Pomocí řídi- cích znaků SI a SO lze předepsat přechod do jiného souboru grafic- kých znaků, např. znaků ruské abecedy.

Řídicí znaky se dělí do čtyř kategorií. K první z nich patří řídicí znaky přenosu TC: ACK, DLE, ENQ, EOT, ETB, ETX, NAK, SOH, STX a SYN; jejich význam je uveden v tab. 17.5. K další patří zna- ky úpravy záznamu FE; které jsou určeny především k uspořádání

a umístění informací zaznamenávaných zobrazovacím zařízením a tiskárnou: BS, CR, FF, HT, LF a VT. Jejich význam (tab. 17.6) vychází z představy, že stránka je složena z určitého počtu řádek skládajících se ze znaků. Znaky tvořící řádek zabírají určitý počet míst, zvaných znaková místa. Aktivním místem se rozumí znakové místo, ve kterém by se objevil vyhodnocovaný znak, kdyby měl být otisknut. Řídicí znaky ovládání přídavného zařízení DC jsou určeny k ovládání lokálních nebo vzdálených zařízení, připojených na systém zpracování dat nebo telekomunikační systém. V tab. 17.7 je uvedeno jejich preferované užití; jejich význam je nutné definovat individuálně pro každou aplikaci. Znak DCL se např. užije k zapnutí zařízení, DC2 k jeho uvedení do zvláštního provozního režimu. Opětne použití znaku DCL má pak význam návratu do základního provozního režimu.

Řídicí znaky pro oddělování informace jsou čtyři: FS, GS, RS, US. Těchto oddělovačů informace lze používat jak v hierarchickém pořadí, tak nehierarchicky. V hierarchickém použití je vzestupně pořadí US, RS, GS, FS. Oddělovačem nejvyšší úrovně je FS. V hierarchickém použití nemohou být data rozdelená oddělovačem určité úrovně dále rozdělena oddělovačem vyšší úrovně, ale pouze oddělovačem nižší úrovně. Význam řídicích znaků pro oddělování je nutné definovat individuálně při každé aplikaci; v tab. 17.8 se uvádí pouze datové celky, které řídicí znaky vymezují při hierarchickém použití.

Význam ostatních řídicích znaků, které nelze zařadit do těchto kategorií, je uveden v tab. 17.9. Přesnou definici mezinárodní abecedy č. 5 včetně jejího vyjádření na paměťových mediích uvedenou v doporučeních CCITT a návazných normách ISO nalezně čtenář v /7/, /8/

17.1.3 Měniče signálů

Měniče signálů přizpůsobují datové signály předávané koncovými zařízeními použitému telekomunikačnímu spoji (obr. 17.4). Podle druhu telekomunikačního spoje rozdělujeme měniče na:

- měniče pro přenos v základním pásmu
- telegrafní měniče signálu
- měniče pro přenos po telefonních spojích - modemy

měniče pro přenos po širokopásmových spojích - širokopásmové modemy.

Měniče pro přenos v základním pásmu jsou určeny pro použití do vzdálenosti několika kilometrů. Při kratších vzdálenostech lze docílit přenosových rychlostí až 9 600 bitů/s. K těmto měničům patří např. měniče pro přenos dat nízkou úrovní po metalických spojích, tzv. měniče GDN, které se využívají pro připojení terminálů ke koncentrátoru a v mnoha dalších aplikacích. Jejich význam se vzhledem k nízké ceně a poměrně velkým dosažitelným přenosovým rychlostem pro přenos dat na menší vzdálenosti stále zvětšuje.

Telegrafní měniče jsou určeny k přeměně signálů z koncového zařízení na signály vhodné pro přenos po telegrafních spojích, tj. s jednoduchým proudem 40 mA a dvojím proudem 2×20 mA nebo 2×10 mA. Podobně jako měniče pro přenos v základním pásmu nemění telegrafní měniče polohu signálů v kmitočtovém pásmu a proto jsou principiálně velmi jednoduché. Bývají však obvykle vybaveny technickými prostředky, umožňujícími provoz na dálnopisné komutované síti, čímž jejich složitost a cena vzrůstá; přesto však patří tyto měniče mezi nejméně nákladné.

Měniče pro přenos dat po telefonních spojích upravují datový signál tak, aby jeho kmitočtové spektrum souhlasilo s kmitočtovým pásmem telefonního spoje, tj. 300 až 3 400 Hz. Úprava spočívá v modulaci datového signálu na nosný signál na začátku telefonního spoje a v jeho demodulaci na konci spoje, čímž se získá původní datový signál. Spojením zkrácených slov modulátor-demodulátor vznikl název pro tento druh měničů, které se nazývají modemy. Modemy lze rozdělit podle několika hledisek. Podle způsobu přenosu je dělíme na sériové a páralelní; podle typu provozu na asynchronní a synchronní, podle přenosové rychlosti na pomalé (do 1 200 bit/s), středně rychlé (od 1 200 bit/s do 9 600 bit/s) a rychlé (nad 9 600 bit/s). Modemy mohou pracovat s různými typy modulace, např. amplitudovou, kmitočtovou, fázovou i kombinovanou.

Z hlediska požadavků na vlastnosti telefonního spoje je jedním ze základních parametrů modemů modulační rychlosť, což je převrácená hodnota délky trvání nejkratšího jmenovitého intervalu, po který nedochází ke změně stavu signálu. Vyjadřuje se v baudech (Bd). Pro uži-

vatele modemu je však podstatným parametrem přenosová rychlosť, která udává počet bitů, přenášených modemem za jednotku času. Tyto dva parametry jsou si rovny pouze tehdy, používá-li modem dvoustavovou modulaci. V každém jednotkovém intervalu se v tomto případě přenáší jednobitová informace. Vícestavovou modulací se přenáší v jednotkovém intervalu vícebitová informace. Tak např. čtyřstavová modulace přenáší v jednotkovém intervalu dvoubitovou informaci, přenos n-bitové informace v jednotkovém intervalu vyžaduje 2^n -stavovou modulaci. Příklad čtyřstavové amplitudové modulace je uveden v obr. 17.5. Zvětší-li se však přenosová rychlosť vícestavovou modulací, vznikají nároky na parametry telekomunikačního spoje i modemu, neboť při detekci vícestavově modulovaného signálu je třeba rozlišit menší modulační změny než u dvoustavově modulovaného signálu. Rovněž poruchy lze hůře rozlišit od informačního obsahu. Typ telekomunikačního spoje tedy předurčuje přenosovou rychlosť modemu, které lze pro něj použít.

17.1.4 Rozhraní v systémech dálkového přenosu dat

V systémech přenosu dat se obvykle definují tři rozhraní, která jsou vyznačena na obr. 17.6 a pro která se v jednotném systému elektronických počítačů JSEP zavedlo označení rozhraní S1, S2 a S3. V jednotném systému malých počítačů SMEP podle obr. 15.1 je rozhraní S2 ekvivalentní rozhraní M4, S3 M2. Rozhraní S1 mezi zařízením ukončujícím datový okruh tj. měničem signálu a datovým spojem definuje především přenosové charakteristiky signálů, např. vysílací úrovňy apod. Definice rozhraní S1 zaručuje správnou činnost mnoha kanálových nosných systémů z hlediska telekomunikační sítě, z hlediska zatížení a hluků, z hlediska vyhovujícího přenosu a spolupráce s telekomunikačními zařízeními. Definice tohoto rozhraní je obsažena v doporučeních CCITT / 7 / a / 8 /. Rozhraní S1 se především týká návrhu zařízení ukončujícího datový okruh, takže se jím nebudeme podrobněji zabývat a odkazujeme čtenáře na uvedenou literaturu.

Rozhraní S2 je rozhraním mezi zařízením ukončujícím datový okruh a koncovým zařízením. Pod pojmem koncové zařízení se rozumí jakýkoliv zdroj nebo příjemce dat. Rozhraní S2 má základní význam, neboť jím jsou propojeny technické prostředky, které jsou vlast-

nictvím správy spojů a technické prostředky uživatele; budeme se tomuto rozhraní věnovat poněkud podrobněji. Výrobci mikropočítačových systémů vybavují mikropočítačové stavebnice stavebnicovým prvkem, nazývaným komunikační adaptorem, který usnadňuje realizaci tohoto rozhraní.

Dalším rozhraním, uvedeným na obr. 17.6 je rozhraní S3. Toto rozhraní je vnitřním rozhraním koncového zařízení a obvykle se tak označuje rozhraní mezi periferním zařízením a řídící jednotkou. Obecně však struktura koncového zařízení může obsahovat jakékoliv rozhraní, kterým jsme se zabývali v předechozích odstavcích.

Vraťme se k rozhraní S2. Mechanické a elektrické vlastnosti tohoto rozhraní jsou definovány v doporučeních CCITT V24 a V28, jeho funkční vlastnosti v doporučeních V24 a X21 viz lit. [7] a [8]. Definice mechanických vlastností určuje zejména konektory, kterým musí být vybaveno koncové zařízení i zařízení ukončující datový okruh, což zajišťuje propojitelnost dvou jednotek různých výrobců. Elektrické vlastnosti rozhraní jsou definovány pomocí parametrů ekvivalentního vazebního obvodu. Funkční vlastnosti rozhraní S2 jsou určeny vazebními obvody mezi koncovým zařízením a zařízením ukončujícím datový okruh. Vazební obvody se dělí do čtyř kategorií: zemnící obvody, datové obvody, řídící obvody a časové obvody. Nejdůležitější vazební obvody a jejich význam je uveden v tab. 17.10. Časový průběh signálů postupujících těmito vazebními obvody při vysílání a příjmu dat je znázorněn na obr. 17.7. Z uvedených obvodů se vždy využívají jen ty obvody, které jsou nezbytné pro styk s konkrétním zařízením ukončujícím datový okruh. Jestliže se např. jedná o modem s přenosovou rychlosí 4 800 bit/s, který nebývá vybaven zpětným kanálem, nevyužívají se datové ani řídící obvody, které jsou určeny pro zpětný kanál. Využití uvedených vazebních obvodů pro normalizované modemy je definováno v dalších doporučeních CCITT. Kromě obvodů pro rozhraní S2 tab. 17.10 je definována další skupina signálů pro automatické volání, které se používají při dálkovém přenosu dat po komutovaných spojích a které slouží jako rozhraní pro připojení zařízení pro automatické navázání spojení. Toto rozhraní se někdy označuje jako rozhraní S4. Jeho obvody se však nebudeme zabývat.

Operační vlastnosti rozhraní S2 se staly rovněž předmětem nor-

malizačního úsilí. Popíšeme je podrobněji v následujícím odstavci.

17.2 Organizace přenosu dat

K tomu, aby mohlo dojít k výměně zpráv mezi koncovými zařízeními, musí být dohodnuta soustava pravidel, která určuje postup při navazování spojení a jeho ukončování a při navazování přenosu a jeho ukončování. Dále musí být dohodnut formát zpráv, tj. posloupnost znaků nebo bitů ve zprávě a používaná abeceda. Tato soustava syntaktických a sémantických pravidel, která předepisuje strukturu zpráv, povелů a význam přenášených prvků zpráv se nazývá řídicí postup. Řídicí postup lze navrhnut tak, aby vyhovoval všem velmi různorodým požadavkům kladeným různými typy telekomunikačních spojů; druhy přenosu, strukturou telekomunikační sítě, po které se přenos dat uskutečňuje, zvoleným typem provozu, atd. Při přenosu dat po pevných spojích však např. odpadá problém navázání a ukončení přenosu, ten se týká pouze přenosu dat po komutovaných spojích. Univerzální řídicí postupy jsou příliš složité a proto se kromě nich zavedly a normalizovaly i řídicí postupy, které jsou vhodné jen pro určité typy telekomunikačních spojů a druhy přenosu. Způsob propojení technických prostředků přenosu dat značně ovlivňuje řídicí postupy, které je nutné pro přenos dat použít.

17.2.1 Struktura sítí přenosu dat

Nejjednodušším typem spoje je tzv. dvoubodový spoj, který zajišťuje přenos mezi dvěma koncovými zařízeními, tedy spojení dvou bodů. Dvoubodový spoj může být buď simplexní, poloduplexní nebo plně duplexní, podle toho, zda zajišťuje přenos dat jen jedním směrem nebo oběma směry. Pro jeho znázornění i pro znázornění složitějších struktur se používá grafického zobrazení, tzv. orientovaného grafu. Každý uzel orientovaného grafu odpovídá koncovému zařízení, každá orientovaná hrana spoje s jedním směrem přenosu. V obr. 17.8 je uvedeno grafické znázornění dvoubodového spoje. Někdy se používá neorientovaného grafu. V tomto případě každá neorientovaná hrana grafu odpovídá spoji pro oba směry přenosu.

Obecně však spoj nemusí obsahovat jen dva uzly; pak mluvíme o mnohabodovém spoji. Příklady mnohabodových spojů jsou uvedeny

v obr. 17.9. V prvním případě jsou uzly spojeny tak, že každý uzel je dostupný přímo z ostatních uzlů, pak mluvíme o jejich paralelním spojení. V druhém případě je znázorněn mnohabodový spoj se sériovým spojením uzlů, při kterém jsou uzly obecně dostupné jen přes jiné uzly. Konečně poslední příklad v tomto obrázku znázorňuje kruhový spoj, ve kterém se přenos uskutečňuje jen v jednom směru. Výměna zpráv je však možná oběma směry, neboť každý uzel je dostupný po obvodu spoje z kteréhokoliv jiného uzlu.

Obecně lze z těchto základních typů spojů vytvářet složitější sítě, např. hvězdicové, polygonální a síť s hierarchickou strukturou. V dalším výkladu se však omezíme na základní typy spojů.

Podle způsobu řízení rozdělujeme systémy pro dálkový přenos dat na systémy centralizované a decentralizované. V centralizovaných systémech je určen jeden z uzlů sítě jako řídicí a činnost všech ostatních je tomuto uzlu podřízena. V řídicím uzlu sítě je obvykle umístěn počítač, který řídí činnost všech koncových zařízení, umístěných v ostatních uzlech. V decentralizovaných systémech je několik řídicích uzlů, které sice vzájemně komunikují, ale nejsou si podřízeny. Mnohdy je však určen jeden z nich dočasné jako nadřízený, místo řízení se však podle okamžité potřeby měří. Decentralizovaný systém řízení se většinou používá v složitějších systémech s hierarchickou strukturou. V těchto systémech se inteligence systému rozptyluje po síti, takže se hlavní řídicí uzel např. hlavní počítač odlehčuje od přenosových funkcí; vztájí však samozřejmě nárky na vybavení ostatních uzlů i na používané řídicí postupy. Obecně však se snižují nároky na telekomunikační prostředky.

Způsob řízení cvlivňuje i topologie sítě, takže v grafech, znázorňujících sítě se vyznačuje řídicí uzel zvláštním symbolem, např. dvojitým kroužkem.

17.2.2 Řídicí postupy

Při přenosu dat mezi koncovými zařízeními musí řídicí postup obsahovat pravidla pro řízení všech fází, kterými tento přenos prochází: navázání spojení, navázání přenosu, výměny dat, ukončení přenosu a zrušení spojení. Kromě toho musí definovat strukturu

zpráv a povelů.

Nejjednodušším případem je přenos dat na dvoubodovém pevném spoji, proto vysvětlíme základní pojmy na tomto případu.

Namísto pojmu uzel budeme v dalším textu používat pojmu stanice. Budeme předpokládat, že jedna ze stanic je řídící. Jen tato řídící stanice má právo zahájit přenos dat. Stanici, která má v daném okamžiku právo vysílat, budeme nazývat hlavní stanicí. I na mnohabodovém spoji bude vždy jen jedna hlavní stanice. Fáze navázání přenosu, zahajuje řídící stanice dotazem, zda je podřízená stanice schopna data přijmout. Podřízená stanice budou neodpoví, protože je např. mimo provoz, nebo odpoví záporně, protože je např. obsazena jinou činností, nebo odpoví kladně. Kladná odpověď ukončuje fázi navázání přenosu.

Fázi přenosu dat zahajuje řídící stanice zasláním prvního bloku zprávy. Jestliže podřízená stanice přijala blok zprávy bez chyby přenosu a nezařezla chybu v zabezpečovacích znacích, odpovídá kladně. V opačném případě odpoví záporně. Na kladnou odpověď předává řídící stanice další blok zprávy. Zápornou odpověď interpretuje jako žádost o opakování bloku. Fáze přenosu dat pokračuje předáváním dalších bloků zprávy do té doby, až jsou všechna data předána. Pak následuje fáze ukončení přenosu, kterou vyvolá řídící stanice vysláním příslušného povelu. Řídící postupy obvykle též dávají možnost podřízené stanici, aby si v průběhu přenosu dat mohla vyžádat ukončení přenosu.

Uvedený postup zajišťuje možnost zaslání dat z řídící stanice do stanice podřízené. Mluvíme v tomto případě o tzv. režimu výběru, neboť řídící stanice si vybírá podřízenou stanicí, která chce zaslát data. Režim výběru je charakterizován tím, že řídící stanice je současně hlavní stanicí. Termín režim výběru nalézá své plné oprávnění při přenosu na mnohabodových spojích.

Kromě režimu výběru se používají ještě další režimy, a sice režim výzvy a režim ohlášení. Oba dva jsou určeny k tomu, aby řídící stanice mohla převzít datu od podřízené stanice. V prvním případě vyzývá řídící stanice podřízenou k předání zprávy. Přitom může nastat několik případů. Podřízená stanice neodpovídá, je-li mimo provoz, nebo odpoví záporně, neboť nemá k odeslání žádnou

zprávu. Má-li však připravenou zprávu, pak ji odesílá namísto odpovědi. Při vysílání zprávy se stává hlavní stanicí, která řídí fázi přenosu dat i fázi ukončení přenosu. Režim výzvy je tedy charakterizován tím, že podřízená stanice je hlavní stanicí. Na mnohobodových spojích uskutečňuje řídicí stanice výzvy všem podřízeným stanicím postupně buď ve zvoleném pořadí nebo podle zvoleného prioritního schématu.

Režim ohlášení se liší od režimu výzvy tím, že se podřízená stanice ohlašuje řídicí stanici, a pak převezme řízení, neboť vyvolává přenos. O přijetí ohlášení však rozhoduje původní řídicí stanice, takže i v režimu ohlášení zůstávají původní řídicí stanici některé řídicí funkce. Režim ohlášení je tedy charakterizován tím, že řídicí stanice je současně hlavní stanicí, avšak funkce řídicí stanice má pouze přechodný charakter, neboť po ohlášení tato funkce přechází na řídicí stanici sítě, kterým je, jak jsme uvedli v předchozím odstavci, většinou počítač nebo řídicí jednotka.

Při přenosu dat na komutovaných spojích musí řídicí postupy zajistit navázání a zrušení spojení. Tyto fáze jsou závislé na typu telekomunikačního spoje. Nebudeme se proto těmito fázemi zabývat. Postup při navázání a zrušení spoje v dálnopisné síti telex i v komutované telefonní síti je předmětem doporučení CCITT.

17.2.3 Formát zpráv

K podmírkám, které vytváří předpoklady pro úspěšnou výměnu zpráv mezi koncovými zařízeními, patří jednotné uspořádání jednotlivých prvků zprávy, popřípadě povelů a příkazů.

Formát zpráv se dělí na dvě skupiny: znakově orientovaný formát a bitově orientovaný formát. Ve znakově orientovaném formátu je délka přenášené zprávy a jejich bloků celistvým násobkem délky přenášených značek, které je nutné interpretovat jako znaky dohodnuté abecedy. V bitově orientovaných formátech je formát nezávislý na použité abecedě a zpráva představuje z hlediska přenosu jen posloupnost bitů. Její interpretace je záležitostí koncových zařízení.

Výhody i nevýhody obou formátů vyplynou z dalšího textu.

Vlastní obsah zprávy je z hlediska přenosu dat nepodstatný a označuje se jako text. Formát zprávy se skládá ze tří hlavních částí: záhlaví, textu a zakončení. Záhlaví a zakončení nazýváme obálkou. Její základní funkcí je ohrazení a zabezpečení textu. K dalším funkcím patří řízení a směrování přenosu. V znakově orientovaných formátech používá obálka stejnou abecedu jako text zprávy. Některé abecedy mají zvláštní znaky pro vytvoření formátu. V mezinárodní abecedě č. 5 jsou to např. řídící znaky přenosu, které jsme uvedli v tab. 17.5. Je-li třeba v těchto formátech přenést zprávu v jiné abecedě, používá se tzv. transparentního textu, tj. textu, který se nedá interpretovat v abecedě předepsané řídicím postupem. Musí se však zabezpečit, aby nemohlo dojít k záměně znaků tohoto textu s řídícími znaky přenosu. Záhlaví se obvykle skládá ze dvou částí; části uživatelské, která se vytváří před přenosem a je součástí zprávy, a z části přenosové, která se vytváří a modifikuje jednotlivými telekomunikačními prostředky při přenosu dat a před zpracováním se odstraňuje.

Úkolem zakončení je oddělit od sebe zprávy nebo jejich jednotlivé bloky a předat zabezpečovací část přenášené zprávy nebo bloku. Zabezpečovací část obsahuje v znakově orientovaných formátech jeden nebo několik bytů paritního, iteračního, cyklického nebo jiného detekčního nebo opravného kódu. V bitově orientovaných formátech má stejný význam pole zadané délky, obsahující zabezpečovací bity. Zabezpečovací část se vytváří v závislosti na obsahu zprávy podle pravidel pro tvorbu zabezpečovacího kódu vysílací stanice. Podle stejných pravidel se vytváří i na přijímací straně a při zakončení se srovná zabezpečovací část vytvořená v přijímací stanici, se zabezpečovací částí přijatou v rámci zakončení z vysílací stanice. Podmínkou potvrzení správnosti přijaté zprávy je shoda přijaté a vytvořené zabezpečovací části.

Příklad znakově orientovaného formátu zprávy, používající mezinárodní abecedu č. 5 jsou uvedeny v tab. 17.11. Význam řídících znaků přenosu je uveden v tab. 17.5. V posledním řádku tabulky je uveden příklad formátu zprávy s transparentním textem. K přechodu do transparentního režimu se používá autoregistr DLE, následovaný znakem STX. V tomto režimu nerozeznává přijímací stanice řídící znaky, kterým nepředchází znak DLE, ale interpretuje je jako text.

Znak ETX, který se náhodně vyskytne v transparentním textu, se neinterpretuje jako konec zprávy, ale jako ukončení se interpretuje pouze dvojice DLE ETX, DLE ITB nebo DLE ETB. Aby ani náhodný výskyt znaku DLE nebyl chybně interpretován, kontroluje se při vysílání a vysílací stanice jej automaticky zdvojuje, jakmile se vyskytne. Přijímací stanice pak vždy jeden z dvojice znaků DLE vyřazuje z textu. Následuje-li však za znakem DLE řídicí znak, provádí odpovídající přenosovou funkci.

V bitově orientovaných formátech je základním prvkem formátu rámec, což je blok s minimální délkou 32 bitů, který je oboustranně ohrazen křídelními značkami, tvořenými posloupností bitů 01111110. Aby nebyla náhodná posloupnost bitů interpretována jako křídelní značka, vkládá vysílací stanice automaticky po každé posloupnosti pěti bitů ve stavu 1 bit ve stavu nula, kterou přijímací stanice automaticky vyřazuje. K dalším polím rámce náleží adresové pole obsahující adresu stanice, které je zpráva určena, řídicí pole obsahující příkazy, popř. pořadová čísla rámce, informační pole, obsahující text a konečně zabezpečovací pole rámce. Příklad formátů, používaných v řídicím postupu HDLC je uveden v obr. 17.10. Formát je určen obsahem řídicího pole. Formát samostatného rámce není zabezpečen pro přenos posloupnosti rámců, takže neobsahuje pořadové číslo rámce. Řídicí slovo obsahuje kód řídicích povelů a odpovědí, které dovolují primární tj. nadřízené starici převést sekundární tj. podřízenou stanici do požadovaného režimu nebo naopak umožňují sekundární stanici předat odpověď primární stanici. Význam některých povelů je uveden v tab. 17.12. Dohlížecí rámec zahajuje a řídí přenos informace v informačních rámcích. Kód povelu nebo odpovědi v dohlížecím rámcu obsahuje povel nebo informaci o připravenosti stanice k příjmu, jak jsou uvedeny v tabulce. Pořadové číslo Nr obsažené v řídicím poli označuje rámec, jehož příjem je očekáván. Potvrzuje se tak příjem rámců s nižšími pořadovými čísly. Informační rámec obsahuje v řídicím poli kromě pořadového čísla Nr i pořadové číslo Ns rámce, který je vysílán. Slouží přijímací stanici k porovnání s pořadovým číslem rámce, jehož příjem je očekáván, což znázorňuje obr. 17.11. Zabezpečení posloupnosti rámců se zajišťuje srovnáním obou pořadových čísel Nr a Ns, která musí souhlasit.

17.2.4 Příklady řídicích postupů

Pro ilustraci si uvedeme příklad znakově orientovaného řídicího postupu, vycházejícího ze základního postupu ISO - řídicího postupu BSC a příklad bitově orientovaného řídicího postupu HDLC. V prvním případě budeme předpokládat poloduplexní provoz na pevném mnohobodovém spoji, v druhém případě duplexní provoz.

Algoritmus činnosti při navázání spojení v režimu výběru je uveden v obr. 17.12. Budeme předpokládat, že řídicí stanice je počítač. Řídicí postup je součástí jeho telekomunikačního programového vybavení, které pak na základě zdrojového programu uživatele řídí prostřednictvím programu kanálu přenos dat. Výběr se zahajuje povelom zápis, kterým se vysílá posloupnost SYN SYN EOT, který má za úkol přivést mnohobodový spoj do stavu klidu. Pak následuje posloupnost ADRB ADRB ENQ, obsahující adresu podřízené stanice B, se kterou řídicí stanice chce navázat přenos. Možné reakce stanice B na posloupnost výběru jsou vyznačeny v pravém sloupci obr. 17.12. Odpovědi se čtou na základě povelu čtení a programové vybavení řídí odpovídající činnosti řídicí stanice. Jestliže stanice B neodpovídá, protože např. nerozeznala svou adresu nebo je vypnuta, opakuje se několikrát výběrová posloupnost; jsou-li i následující pokusy neúspěšné, vyvolává se odpovídající procedura obnovy (tyto činnosti nejsou na obrázku uvedeny). Jestliže stanice B má nezpracovanou stavovou informaci, indikující např. chybu parity paměti dat, odpovídá posloupností RVI, která má význam odmítání výběru a je tvořena posloupností znaků DLE a grafického znaku **≤** (znak DLE - autoregistr, viz tab. 17.5 mění význam znaku za ním následujícího tak, že je interpretován jako další řídicí znak přenosu RVI). Řídicí stanice ukončuje přenos dat znakem EOT, za kterým obvykle následuje výzva k předání nezpracované stavové informace a vyvolání odpovídající procedury obnovy. Podřízená stanice může být zaměstnána autonomní činností. V tom případě odpovídá řídicím znakem WACK, tvořeným posloupností znaků DLE; (středník). Řídicí stanice reaguje ukončením přenosu znakem EOT. Konečně poslední možností je kladné potvrzení výběru podřízenou stanici znakem ACKO, tvořeným posloupností DLE 0 (nula). Znakem ACKO se potvrzuje každý správně přijatý sudý blok, znakem ACK1 každý správně přijatý lichý blok. Potvrzením výběru je přenos na-

vázán a řídicí stanice reaguje přechodem do fáze výměny dat, tj. předáním prvního bloku textu. Tato fáze je znázorněna na obr.

17.3. Blok textu má formát STX text ETX BCC (předává-li se text v jednom bloku) nebo STX text ETB BCC (předává-li se v několika blocích). Jestliže podřízená stanice neodpovídá, řeší tuto situaci řídicí stanice, obdobně jako při výběru. Odpovědí NAK informuje podřízená stanice řídicí stanici o tom, že došlo k chybě přenosu, např. k nesouhlasu ve znaku zabezpečení přenosu BCC. Řídicí stanice reaguje opakováním vysílaného bloku. Odpověď EOT ukončující přenos vysílá podřízená stanice tehdy, obsahuje-li nepřípustné znaky nebo povely, nebo došlo-li při zápisu textu k chybě parity vyrovnávací paměti. Řídicí stanice pak obvykle pokračuje výzvou k převzetí stavové informace upřesňující příčinu ukončení provozu. Odpověď WACK je kladným potvrzením; znamená však, že blok textu obsahoval povel, který vyvolal činnost, kterou je podřízená stanice zaměstnána, takže nemůže přijmout další blok textu. Konečně odpověď ACK1 potvrzuje bezchybně přijatý blok. Řídicí stanice buď pak ukončuje přenos anebo předává další blok textu.

Z uvedeného příkladu je zřejmé, že řídicí postup BSC je v podstatě poloduplexní, neboť před vysíláním dalšího bloku musí vysílací stanice čekat na potvrzení předchozího bloku. Použitím duplexního spoje se provoz zrychlí jen tím, že při změnách směru přenosu není nutné vřazovat časové prodlevy eliminující přechodné jevy, ke kterým dochází na poloduplexním spoji.

Řídicí postup HDLC je určen pro řízení duplexního provozu. Lze jej aplikovat i na poloduplexních spojích; nevyužije se však všech jeho možností. Příklad algoritmů činnosti při navázání přenosu a výměny dat je uveden v obr. 17.14. Symbolický formát předávaných rámců zjednodušeně znázorňuje formáty z obr. 17.10.

V první části obrázku je naznačena činnost při navázání přenosu stanice A se stanicí B, v druhé části obrázku prokládané vysílání stanice A stanicím B a C. Činnost je zřejmá z popisu, uvedeném v obrázku. Je zřejmé, že použitý princip zabezpečení posloupnosti rámců a způsob jejich potvrzování, vyložený v obr. 17.11 dává řídicímu postupu HDLC duplexní charakter, neboť potvrzení bezchybného příjmu rámce nemusí následovat bezprostředně po jeho přijetí. Vysílací stanice může pokračovat vysíláním dalších

rámců; k jejich potvrzení dojde až na vyžádání, a to i v průběhu vysílání dalších bloků.

17.3 Dálkový přenos dat a mikropočítáčové systémy

17.3.1 Stavebnicové prvky pro realizaci rozhraní S2

Výrobci mikropočítáčových systémů rozpoznali poměrně brzy významnou úlohu, kterou má dálkový přenos dat při nasazení mikropočítáčových systémů v mnoha aplikačních oblastech a doplnili soubor stavebnicových prvků o specializované stavebnicové prvky, určené pro realizaci některých funkčních a operačních vlastností rozhraní S2. Popíšeme si dva představitele těchto stavebnicových prvků, které dala na trh firma Intel. První z nich, komunikační adaptér 8251 má univerzálnější charakter a lze jej užít jak k realizaci dálkového přenosu dat, ve kterém se využívá např. základního řídicího postupu ISO, tak i při řešení jednoduchého rozhraní při asynchronním přenosu dat, např. při připojení dálnopisu. Druhý z nich, řadič HDLC/SDLC 8273 je určen k realizaci rozhraní při dálkovém přenosu dat s použitím řídicího postupu HDLC, popř. obdobného postupu SDLC firmy IBM.

Programovatelný komunikační adaptér 8251

Komunikační adaptér 8251 lze programovat tak, aby vyhověl různorodým požadavkům rozhraní podle typu telekomunikačního spojování signálu i zvoleného řídicího postupu. Úkolem komunikačního adaptoru a programového telekomunikačního vybavení mikropočítáče je převést všechny úlohy spojené s dálkovým přenosem dat, které jsme alespoň rámcově poznali v předchozích odstavcích, na bytově orientované vstupně-výstupní operace, které jsou součástí architektury mikropočítáčového systému.

Vnitřní uspořádání komunikačního adaptoru je uvedeno v obr. 17.15. K systémové sběrnici mikropočítáčového systému se komunikační adaptér připojuje prostřednictvím budiče datové sběrnice a řídicích obvodů čtení zápisu. Význam signálů rozhraní těchto bloků a systémové sběrnice mikropočítáče je uveden v tab. 17.13. Vztah těchto signálů k signálům systémové sběrnice mikropočítáče orientované na systém 8080 např. sběrnice Multibus nebo S-100 je zřejmý z obr. 17.16. Operace zápis datových, řídicích a povelových

slov a čtení datových a stavových slov řídí adresové a řídicí signály tohoto rozhraní v souhlase s tab. 17.14. Zapojení na obr. 17.16 představuje jednu z možných alternativ, odpovídající konkrétní konfiguraci systému. K signálům uvedeným na obrázku mohou přistoupit ještě signály TxRDY, TxE a RxRDY, s nimiž se seznámíme v dalším textu a jejichž stav může být testovat program nebo může vyvolávat přerušení.

Další skupinu signálů tvoří signály rozhraní se zařízením ukončujícím datový okruh. Tyto signály odpovídají doporučení RS-232-C. Toto doporučení je obdobné doporučení V24, s kterým jsme se seznámili v předchozích kapitolách; proto je v tab. 17.15 uveden pouze vztah mezi signály komunikačního adaptoru 8251, určenými pro toto rozhraní a odpovídajícími vazebními obvody podle doporučení V24. Význam signálů v těchto vazebních obvodech byl vysvětlen v tab. 17.10. Signál časové základny při vysílání TxC určuje při synchronním režimu přímo přenosovou rychlosť; při asynchronním režimu je přenosová rychlosť zlomkem kmitočtu signálu TxC; tento zlomek může mít hodnotu $n = 1$, $1/16$ nebo $1/64$; což se určuje povelem, který předepisuje režim komunikačního adaptoru. Tak např. pro přenosovou rychlosť 50 bit/s a $n = 1$ je $TxC = 50$ Hz, pro $n = 1/16$ je $TxC = 800$ Hz a pro $n = 1/64$ je $n = 3,2$ kHz. Zdrojem signálu TxC je v každém případě zvláštní kmitočtový zdroj. Synchronizaci tohoto signálu a signálu TxD (103) zajišťuje v synchronním režimu komunikační adaptér. Dalším signálem časové základny je signál RxC (115), který zajišťuje synchronizaci při příjmu.

Zdrojem časové základny v synchronním režimu je synchronní modem; kmitočet časové základny je v tomto případě roven přenosové rychlosti. Při asynchronním přenosu je zdrojem časové základny zvláštní kmitočtový zdroj; požadavky kladené na jeho kmitočet jsou obdobné, jako na kmitočet signálu TxC. Synchronizace mezi signálem RxD a RxC pro $n = 1$ se musí zajistit zvláštními technickými prostředky, pro $n = 1/16$ a $n = 1/64$ není třeba synchronizaci zajišťovat – komunikační adaptér interně spouští i zastavuje časovou základnu a vzkrujuje datové a závěrné prvky tak, jak znázorňuje obr. 17.2 a 17.3.

Poslední skupinu signálů tvoří signály řízení obvodů vysílače a přijímače. Význam těchto signálů je uveden v tab. 17.16. Tyto signály mohou být zpracovány systémem přerušení. V tomto případě

se stávají součástí rozhraní se systémovou sběrnicí mikropočítáče. Signál TxRDY je maskován bitem TxEN stavového slova. Všechny signály mají odpovídající bity ve stavovém slovu, jejichž význam je stejný s výjimkou bitu TxRDY, který není maskován stavem TxEN. Všechny nebo některé z těchto signálů mohou být zpracovány programem, tj. převzetím stavového slova a testováním odpovídajících bitů. Způsob zpracování se určí na základě systémové analýzy.

Programovatelnost komunikačního adaptoru 8251 spočívá v možnosti přivést jej do požadovaného stavu prostřednictvím předepsané posloupnosti řídicích slov. Tato posloupnost musí následovat po každé inicializaci vyvolané buď vnějším signálem RESET nebo zápisem bitu INTERNAL RESET v povelovém slovu. Příklad předepsané posloupnosti řídicího slova a povelových a datových slov je znázorněn v obr. 17.17. První slovo po inicializaci je interpretováno jako řídicí. Toto slovo má dva různé formáty; jeden pro synchronní a druhý pro asynchronní režim. Formát řídicího slova je určen bity D1 a DO. Jsou-li oba bity nulové, jde o formát synchronního režimu (tab. 17.17). Po tomto řídicím slovu musí následovat jedna nebo dvě kódové kombinace, které předepisují kódy znaku SYNC1 a SYNC2. Počet znaků SYNC určuje bit SCS řídicího slova. Jeden nebo dva znaky SYNC s takto předepsanými kódy vkládá automaticky komunikační adaptér do toku vysílaných dat, jakmile při vysílání neobdrží včas data. Mikropočítákový systém je o tom informován signálem TxE. Při příjmu se bitem ESD předepisuje buď interní nebo externí synchronní režim. V interním synchronním režimu musí být předepsán v prvním povelovém slovu bit EH (tab. 17.19), který předepisuje hledání znakové synchronizace. Obsah vyrovnavací paměti přijímače se v každé periodě signálu časové základny \overline{RxC} (115) porovnává s kódem znaku SYNC 1. Jestliže je komunikační adaptér programován pro přenos s dvěma znaky SYNC, zjišťuje se, zda kód následujícího přijatého znaku odpovídá kódu znaku SYNC 2. Po rozumném předepsané posloupnosti znaků SYNC přejde komunikační adaptér do stavu znakové synchronizace, což se indikuje signálem SYNDAT a nastavením odpovídajícího bitu ve stavovém slovu, jehož formát je uveden v tab. 17.20. Bit SNYDET se automaticky nuluje po čtení stavového slova. Tím se však neruší režim znakové synchronizace, který trvá až do povelu, obsahujícího bit EH ve stavu 1. V externím synchronním režimu se znaková synchronizace

ustavuje v okamžiku, kdy signál SYNDAT přejde do stavu 1. Po příjmu znaku se jeho kód zapisuje do vyrovnávací paměti přijímače a signál RxRDY přechází do stavu 1, čímž signalizuje procesoru, že je připraven znak k převzetí. Jestliže nebyl do té doby převzat předchozí znak, ustavuje se bit OE stavového slova, indikující přetečení vyrovnávací paměti přijímače. Rovněž se kontroluje parita, pokud byla tato kontrola bitem PEN řídicího slova předepsána, a v případě zjištění chyby parity se nastavuje bit PE stavového slova.

Předepíše-li řídicí slovo asynchronní režim, pak při vysílání vkládá komunikační adaptér 8251 rozběhový prvek, za nímž následuje datové slovo, připravené ve srovnávací paměti vysílače. Počet bitů datového slova je předepsán obsahem řídicího slova asynchronního režimu, jehož formát je uveden v tab. 17.18; za datovým slovem pak následují závěrné prvky, jejichž počet je rovněž předepsán obsahem řídicího slova. Jestliže není do vyrovnávací paměti vysílače zapsáno datové slovo, zůstává výstup TxD (103) ve stavu 1, dokud povelové slovo nepředepíše vyslání znaku s kódovým přiřazením samé nuly (bit SBRK). Při příjmu se při činiteli bitové rychlosti 1/16 nebo 1/64 vyhodnocuje rozběhový prvek způsobem, uvedeným v obr. 17.2. Jestliže je předepsán činitel bitové rychlosti rovný 1, zajišťuje se vzorkování jinými technickými prostředky. Další bity se vzorkují metodou, uvedenou v obr. 17.3. Jestliže se nevyhodnotí předepsaný počet závěrných prvků, indikuje se tato chyba rámce stavem 1 bitu FE stavového slova. Přijímač i však postačuje k jeho funkci jediný závěrný prvek. Po příjmu znaku se jeho kód zapisuje do vyrovnávací paměti přijímače.

Kontrola znaku, signalizace jeho připravenosti a hlášení chyb probíhá stejně jako v synchronním režimu. O průběhu operací předepsaných komunikačnímu adaptoru se může program informovat čtením obsahu stavového slova. Čtení se předepisuje stavem signálů podle tab. 17.14.-Během operace čtení stavového slova se blokuje jeho aktualizace. Pokud je aktualizace povolena, dochází k ní nejdéle do 28 ns po vnější události, která ji má vyvolat, např. po vyprázdnění vyrovnávací paměti vysílače nebo po naplnění vyrovnávací paměti přijímače.

Programovatelný řadič řídicích postupů HDLC/SDLC 8273

Programovatelný řadič 8273 je určen k realizaci rozhraní S2 mezi mikropočítačem a zařízením ukončujícím datový okruh v případech, používá-li se řídicího postupu HDLC, normalizovaného organizací ISO, nebo řídicího postupu SDLC firmy IBM. Použití tohoto stavebnicového prvku umožňuje zjednodušit programové vybavení mikropočítače realizujícího řídicí postup neboť mnohé potřebné procedury se zjednoduší na předání povelu řadiči 8253. I když jsou uvedené řídicí postupy určeny pro synchronní přenos, lze řadič řídicích postupů HDLC/SDLC programovat pro použití asynchronního přenosu. V tomto případě se používá inverzní kódování bez návratu k nule NRZI. Je charakterizováno tím, že k přechodu stavu signálu dochází tehdy, jestliže se vysílá binární nula. Výhodou použití kódování NRZI při použití řídicích postupů HDLC, resp. SDLC je to, že k přechodům dochází nejdéle po pěti datových prvcích, kdy se vkládá do toku dat nula. Udržení bitové synchronizace je pak snadnější než při jiném kódování. Příklad kódování NRZI je v obr. 17.18. Řadič 8253 je vybaven číslicovým fázovým závěsem, který na základě přechodu signálu s kódem NRZI generuje signál časové základny, zajišťující bitovou synchronizaci. V synchronním a asynchronním režimu automaticky vkládá binární 0 do toku vysílání křídelních znáček a vypouští je při příjmu, zajišťuje vysílání křídelních znáček a jejich detekci při příjmu, stejně tak, jako znaků ABORT, IDLE, EDP. Rovněž obstarává zabezpečení rámce, tj. generaci cyklického zabezpečovacího kódu při vysílání a jeho verifikaci při příjmu. Aplikace řadiče 8253 je možná jak primární řídicí stanici, tak i v sekundární podřízené stanici a v kruhových sítích SDLC. Tato poslední aplikace se však již vymyká z rámce této publikace, takže odkazujeme čtenáře na lit. 8

Vnitřní uspořádání řadiče řídicích postupů HDLC/SDLC je uvedeno v obr. 17.19. Kromě obvodů analogických těm, které jsme poznali v obr. 17.15, obsahuje sedm speciálních registrů, jejichž význam bude vysvětlen dále, a dva vyrovnávací datové registry - registr vysílače a registr přijímače. K systémové sběrnici mikropočítače se řadič připojuje prostřednictvím budiče datové sběrnice a řídicích obvodů čtení a zápisu, což ilustruje obr. 17.20. Význam signálů rozhraní řadiče 8253 se systémovou sběrnici mikro-

počítače je obdobný, jak byl vysvětlen v tab. 17.13. Jediným rozdílem je to, že odpadá signál C/D a namísto něho se používají signály A0 a A1, které se obvykle propojují se stejnojmennými signály adresové sběrnice mikropočítače. Odlišná je též interpretace operací řízených řídicími signály rozhraní, uvedená v tab. 17.21. Kromě toho jsou součástí rozhraní signály žádosti o přerušení pro vysílání TxTNT a pro příjem RxINT, které jsou zpracovány systémem přerušení mikropočítače.

Dalším rozhraním je rozhraní s řadičem přímého styku s pamětí. Význam signálů tohoto rozhraní je uveden v tab. 17.22. Rozhraní umožňuje řídit nezávisle přenos dat pro vysílání a příjem, což si vyžadují řídící postupy HDLC, resp. SDLC, jejichž charakter je duplexní. Vyšší přenosová rychlosť na rozhraní S1 vyžaduje, aby se přenos mezi mikropočítačem a řadičem 8273 uskutečňoval metodou přímého styku s pamětí. Řadič lze však též zapojit tak, aby přenos na této úrovni byl řízen programem. V tomto případě se požadavky na zpracování datového slova předávají obsahem stavového slova, když se řadič 8273 převede do režimu programem řízeného přenosu (viz soubor povelů v tab. 17.24).

Rozhraní se zařazením ukončujícím datový okruh je tvořeno dvěma vazebními obvody: vstupním vazebním obvodem A a výstupním vazebním obvodem B. Formát vstupních a výstupních slov a význam jejich bitů je uveden v tab. 17.23. Převod z paralelního formátu datových slov v registru vysílače řídí signál časové základny TxC (114), opačný převod do registru přijímače signál RxC (115). Povelem lze převést řadič 8273 do diagnostického režimu, ve kterém se interně přepisují data z výstupu TxD (103) na vstup RxD (104), což umožňuje, aby systém vyslal zprávu sám sobě a zkонтroloval tak svoji činnost až na úrovni tohoto rozhraní. Pokud se týká asynchronního režimu činnosti, řízeného dalšími signály DPLL a 32CLK, odkazujeme čtenáře na lit. (9).

Činnost řadiče 8273 lze rozdělit do tří fází: povelové, prováděcí a ukončovací.

Během povelové fáze předává mikropočítač povel do povelového registru řadiče 8273, popřípadě parametry povelu do registru parametrů. V prováděcí fázi vykonává řadič 8273 povel předaný v předchozí fázi. V ukončovací fázi signalizuje řadič 8273 mikropočítači

ukončení povelu a formuluje informace o úspěšném či chybném vykonávání povelu ve výsledkových registrech. V této fázi se čtou obsahy některých registrů, aby podle výsledku povelu mohl program v souladu s pravidly řídicího postupu předat další povel.

Povelová fáze je tedy řízena programovým vybavením mikropočítače. Některé povely vyžadují bližší specifikaci pro provádění, ta se předává v dalších slovech postupně zapisovaných do registru parametrů. Povel může obsahovat až čtyři parametry. Vývojový diagram povelové fáze je uveden v obr. 17.21. Z něho vyplývá, že povel nelze zapsat tehdy, je-li zařízení obsazeno, což indikuje bit CBSY stavového slova. Formát stavového slova je uveden v tab. 17.24. Vzhledem k tomu, že řadič 8273 má jen jeden soubor jednoúčelových registrů, lze provádět současně jen jeden povel. Duplexní povaha zařízení tím však není narušena, neboť lze postupně reagovat na žádosti o přerušení vyvolávané jak přijímačem, tak vysílačem.

V prováděcí fázi se povel provádí buď v režimu přímého styku s pamětí, tj. autonomně - bez účasti mikropočítače - anebo v režimu programem řízeného přenosu pomocí přerušení vyvolávaného při přenosu každého bytu. Ukončovací fáze se zahajuje buď úspěšným ukončením povelu, nebo zjištěním chyby při provádění povelu. Výsledky operací se dělí do dvou kategorií podle typu operace a způsobu zpracování a to na přímé a nepřímé. Přímé výsledky jsou obsaženy vždy ve výsledkovém registru RES. Nepřímé výsledky jsou obsaženy ve výsledkovém registru vysílače TxI/R nebo přijímače RxI/R. Výsledek je obsažen buď v jednom nebo několika bytech. První byte nese vždy informaci o příčině přerušení. Kódové zobrazení uvádí tab. 17.25 a tab. 17.26.

Povely, předávané řadiči 8273 mikropočítačem během povelové fáze lze rozdělit do čtyř kategorií: inicializační povely, povely určené přijímači, povely určené vysílači a povely určené zařízení, ukončujícímu datový okruh. Jejich přehled je uveden v tab. 17.27.

Do kategorie inicializačních povelů patří prvních osm povelů v tabulce. Tyto povely nevyvolávají přerušení po ukončení předepsané operace, ani se nepřipravuje informace o tom, jak operace proběhla. První dva povely umožňují nastavit režim jednobitového zpoždění, používaný v kruhových sítích, a režim programem řízeného

přenosu, používaný v případě, nelze-li přenos dat uskutečnit přímým stykem s pamětí. Další dva povely umožňují tyto režimy zrušit. Povel specifikující režim činnosti umožňuje předepsat způsob zpracování některých znaků a posloupnosti s ohledem na použitý řídicí postup, konfiguraci komunikační sítě a typ zařízení ukončujícího datový okruh. Parametrem tohoto povelu lze např. předepsat režim HDLC, ve kterém se jako předčasné ukončení interpretuje posloupnost sedmi binárních jedniček, na rozdíl od normálního režimu, v němž se takto interpretuje posloupnost osmi binárních jedniček. Dále lze předepsat režim zápisu do vyrovnavací paměti 8273, ve kterém se první dva byty rámce, obvykle adresové a řídicí pole, ukládají do vyrovnavací paměti 8273. Není-li tento režim předepsán, předávají se tyto dva byty do operační paměti stejně jako ostatní data. Nezajišťuje-li zařízení ukončující datový okruh počáteční synchronizaci před vysláním křídelní značky, lze předepsat parametrem povelu režim předsynchronizace, ve kterém se před křídelní značkou vysílají dva byty synchronizační posloupnosti s kódovou kombinací 00 (hex) pro kódování NRZI a 55 (hex) pro jiné kódování, aby byl zajištěn dodatečný počet modulačních přechodů pro bitovou synchronizaci protější stanice. Další povел specifikující režim sériového vstupu-výstupu umožňuje předepsat režim kódování NRZI a diagnostický režim, ve kterém se interně přepisují data z výstupu vysílače na vstup přijímače, jak jsme vysvětlili při výkladu rozhraní řadiče 8273 s modelem.

Kategorie povelů určených přijímači obsahuje čtyři povely předepisující buď všeobecný příjem (ve kterém se přijímají rámce bez ohledu na obsah jejich adresových polí) nebo výběrový příjem (ve kterém se přijímají jen ty rámce, jejichž obsah adresových polí souhlasí s obsahem adresových polí zadaných v rámci parametrů). Pokud jsou v řídicím postupu HDLC použita rozšířená adresovací pole, musí o přijetí rámce rozhodnout programové vybavení. Povel výběrového příjmu v režimu krunové sítě respektuje zvláštnosti této konfigurace. Poslední z povelů určených přijímači ukončuje příjem bezprostředně po jeho převzetí. Povely této kategorie vyvolávají po ukončení předepsané příjmové operace přerušení. Výsledek operace je pro převzetí připraven ve výsledkovém registru RxI/R.

K třetí kategorii povelů patří šest povelů určených vysílači. Patří k nim především povел k vyslání rámce, k jehož parametry

patří údaje o jeho délce, které se různě interpretují, podle toho, zda řadič 8273 je v režimu zápisu do vyrovnávací paměti nebo nikoliv. Je-li v režimu tohoto zápisu, předávají se adresová a řídicí pole jako parametry, a do délky rámce se nezapočítávají. V opačném případě se jako parametr předává pouze délka rámce; uvedená pole se přejímají z operační paměti jako data a do délky rámce se započítávají. Řadič 8273 doplňuje automaticky rámcem o křídelní značky a pole zabezpečení a provádí vkládání binárních nul. Povel vysílání v transparentním režimu však předepisuje vyslání dat tak, jak jsou uloženy v operační paměti, aniž by řadič tato data podle pravidel řídicího postupu. Povely předčasného ukončení zařazují znak předčasného ukončení, po kterém následuje přechod do klidového stavu. Obsah znaku i klidový stav je závislý na typu povelu a na režimu, do kterého byl řadič uveden inicializačním povelom. Povely této kategorie vyvolávají po ukončení předepsané operace přerušení s tím, že její výsledek se uloží do registru TxI/R.

K poslední kategorii poveli patří povely pro zařízení ukončující datový okruh. Tyto povely umožňují číst vstupní slova a měnit jednotlivé bity výstupního slova rozhraní řadiče 8273 se zařízením ukončujícím datový okruh a řídit tak toto zařízení. Formát vstupních a výstupních slov rozhraní byl uveden v tab. 17.23.

Podrobnější popis poveli s uvedením formátů parametrů nalezne čtenář v lit. (9).

17.3.2 Využití stavebnicových prvků při realizaci řídicích postupů

Stavebnicové prvky pro realizaci rozhraní S2 a některých řídicích postupů jsou z technologického hlediska poměrně náročné obvody LSI, které co do složitosti i hustoty součástek několikanásobně překonávají samotný mikroprocesor 8080; přebírájí však jen základní úkoly předepsané algoritmy řídicích postupů. Větší část těchto úloh musí řešit programové vybavení mikropočítačových systémů. Přesto však aplikace těchto stavebnicových prvků zvětšuje podstatným způsobem propustnost systému, neboť pokud by byly základní úkoly např. serializace a deserializace informace, vzorkování datových prvků, vkládání znaku SYNC v základním řídicím postupu ISO, nebo binárních nul ve vyšším řídicím postupu HDLC

řešeny programovým vybavením mikropočítače, bylo by toto řešení časově příliš náročné. Pokud by byly řešeny technickými prostředky s nižším stupněm integrace, bylo by takové řešení neúnosně nákladné.

Je třeba si však uvědomit, že tyto obvody řeší jen některé ze základních úloh předepsaných řídicími postupy. Základní postup ISO např. předepisuje vkládání znaku SYNC po předepsaných časových prodlevách. K odměřování těchto časových prodlev je třeba doplnit technické prostředky o generátor časových prodlev, např. obvod 8253, a vložení znaku SYNC do toku dat řešit programovým vybavením. Procedura vkládání znaku SYNC do toku dat se vyvolává přerušením generátorem časových prodlev po uplynutí časové prodlevy. Rovněž časové prodlevy, předepsané např. časovým diagramem v obr. 17.7 se musí řešit obdobným způsobem. Z přehledu obvodů styku S2, uvedených v tab. 17.10, a ze srovnání, uvedeného v tab. 17.15 je zřejmé, že pro většinu zařízení ukončujících datový okruh neumožňují stavebnicové prvky zpracovat všechny potřebné signály rozhraní, a je třeba doplnit technické prostředky o další obvody, např. o vazební obvod 8255, popřípadě o řadič přerušení 8214 nebo 8259 a zajistit jejich programové ovládání.

Implementace řídicího postupu se zadánou konfigurací komunikační sítě a typem zařízení ukončujícím datový okruh není jednoduchou záležitostí a vyžaduje od návrháře dokonalou znalost algoritmu řídicího postupu, technických a programových prostředků mikropočítačových systémů a znalost telekomunikačních prostředků.

Poznatky uvedené v tomto odstavci budou však užitečné i těm, kteří se nebudou přímo touto problematikou zabývat. Kromě získání obecného přehledu byly probrány mnohé pojmy s obecnou platností: formáty slov, zpráv, řídicí a stavová slova, atp. Komunikace mezi mikroprocesorem a dalšími speciálními stavebnicovými prvky, např. řadičem pružného disku apod. probíhá obdobným způsobem, tj. výměnou povelových, datových, stavových a výsledkových slov. I vnitřní struktura těchto stavebnicových prvků je v mnohých ohledech shodná nebo podobná. Kromě toho lze tyto poznatky uplatnit i tam, kde nejde o dálkový přenos dat, ale ve všech případech, kdy se data mezi mikropočítačově orientovanými systémy přenášejí po sériové sběrnici.

18. MULTIMIKROPROCESOROVÉ A MULTIPOČÍTAČOVÉ SYSTÉMY

Mikroprocesory a mikropočítačové systémy nalezly úspěšné uplatnění v široké aplikační oblasti. Jednotlivé aplikace lze charakterizovat tím, že nároky na zpracování dat nejsou příliš vysoké. Patří k nim např. aplikace v přístrojové technice, v inteligentních terminálech, v řadičích periferních zařízení počítačů, v řadičích procesorů průmyslových, dopravních a jiných a v speciálních procesorech velkých počítačů.

Aplikace mikroprocesorů v rozsáhlejších systémech s většími nároky na rozsah a rychlosť zpracování dat však zůstává otevřeným problémem, jehož řešení se zvláště v posledních letech věnuje mnoho pozornosti. Zájem o řešení tohoto problému je vyvoláván především vzrůstajícími nároky na systémy zpracování dat. Aplikace mikroprocesorů v rozsáhlejších systémech by tyto systémy ekonomicky zvýhodnila a zajistila by potřebné zvýšení spolehlivosti, neboť by umožnila realizovat větší redundaci, než je dosud z ekonomického hlediska únosné. Možnost vytvářet redundantní strukturu vyplývá přímo ze samotné povahy systému, který by byl tvořen velkým počtem identických procesorů. Před nástupem mikropočítačů nebylo možné o takovéto struktuře systému uvažovat, neboť cena jednotlivých procesorů byla tak vysoká, že jejich počet v systému byl značně omezený.

Multiprocesorové systémy přinášejí další výhody, vyplývající z jejich modulové struktury. Systém tvořený identickými procesory a doplněný propojovacími prvky s dobře definovanými rozhraními lze snadno vyrobit, testovat a využívat. Výhody modulární struktury zatím poznali výrobci i uživatelé výpočetních systémů pouze v paměťových jednotkách.

Aplikace mikroprocesorů v rozsáhlejších systémech je však

podmíněna vyřešením následujících problémů, jejichž přehled si uvedeme:

K nejtíživějším problémům patří rozklad úlohy na úlohy dílčí, aby každá z nich mohla být řešena buď jedním procesorem, nebo skupinou procesorů. Snahou je vyřešit rozklad úlohy automatickými programovacími prostředky, např. vhodně řešenými komplátory tak, aby se tímto rozkladem nezabýval programátor, který by jinak musel mít podrobné znalosti systému. S rozkladem úlohy je spojena otázka sdílení systémových zdrojů např. pamětí a vstupně-výstupních zařízení jednotlivými procesory nebo jejich skupinami při samotném výpočtu úlohy. Řešení problému sdílení musí být takové, aby bylo vyloučeno zablokování systému, ke kterému může dojít tím, že každý z procesorů čeká na systémové zdroje, přidělené jinému procesoru nebo skupině procesorů a žádný z procesorů nemůže pokračovat ve výpočtu, dokud jeho žádost o systémové zdroje není přijata. S problémem rozkladu úlohy je dále spojena otázka synchronizace mezi současně a nezávisle řešenými dílčími úlohami, aby mohly být dílčí výsledky bez snížení propustnosti systému dále zpracovány.

Rovněž není dosud vyřešena otázka nejhodnější struktury propojení pro vzájemnou komunikaci mezi procesory i mezi procesory a paměti a otázka řídicích postupů, jejichž pravidly by se měla tato komunikace řídit. Tato otázka je spojena s vyřešením vhodného adresovacího mechanismu s virtuálními adresami, který by umožňoval přístup jak k individuálnímu tak ke společnému fyzickému adresovacímu prostoru s přiměřenou úrovní ochrany dat. Patří sem i problém připojení vnějších zařízení a především vnějších pamětí, které musí být voleno tak, aby tato zařízení se stala integrální součástí multiprocesorového systému.

Uvedené požadavky mohou být splněny vhodnou volbou struktury systému. Předtím, než popíšeme jednu z možných alternativ, uvedeme v přehledu uspořádání, která přicházejí v úvahu a která byla z hlediska aplikace v multiprocesorových systémech analyzována.

18.1 Klasifikace číslicových systémů

K rozdělení číslicových systémů na jednotlivé kategorie můžeme použít různých hledisek. V tomto odstavci uvedeme dvě rozdílné klasifikace a několik příkladů.

Jedna z možných, velmi často používaných klasifikací, je uvedena na obr. 18.1. Číslicové systémy se podle této klasifikace rozdělují na dvě velké skupiny: systémy s jedním a s více procesory. Druhou skupinu, systémy s více procesory, lze dělit na základě rozdílů v programovém vybavení do dvou kategorií označovaných jako multipočítáčové a multiprocesorové systémy. Někdy se též označují jako systémy s volnou vazbou a s těsnou vazbou, popřípadě se pro multipočítáčové systémy používá pojmu systémy s rozdelenou inteligencí. V těchto systémech mají jednotlivé mikropočítáče vlastní specializované operační systémy, neboť každý mikropočítáč má přidělenou specializovanou funkci, která spoluuvytváří funkční vlastnosti multipočítáčového systému. Tato statická alokace funkcí umožňuje rozklad programového vybavení na programové segmenty, které mohou být přiřazeny jednotlivým mikropočítáčům. Vzájemná komunikace mezi procesory je na relativně velmi nízké úrovni, neboť objem informací, které si musí procesory předávat, je poměrně malý. Počítáče mohou být uspořádány buď místně, tj. např. v rámci jedné budovy nebo podniku, popřípadě v dálkovém uspořádání. Pak mluvíme o tzv. počítáčových sítích.

Multiprocesorové systémy se vyznačují tím, že obsahují jednotný operační systém, pod jehož řízením se úlohy dynamicky přiřazují jednotlivým procesorům systému. Je evidentní, že programové vybavení je podstatně složitější než v předcházejícím případě. Vzájemná komunikace mezi procesory i mezi jednotlivými programovými segmenty je rovněž na vyšší úrovni než v případě multipočítáčových systémů. K typickým znakům multiprocesorových systémů patří jednotný operační systém a sdílení společných vstupně-výstupních zařízení a sběrnic. Komunikace na všech poptávkách úrovních se zajišťuje různými konfiguracemi technického vybavení, které může sloužit jako kritérium pro další kategorizaci typických multiprocesorových systémů, a sice na systémy s časově sdílenou společnou sběrnicí, s vícenásobným vstupním výstupním členem paměti a s přepínací maticí. Systémy s časově sdílenou sběrnicí mohou mít jednoduchou nebo hierarchickou strukturu sběrnic, kterou jsme poznali v předcházejících kapitolách.

Kvazimultiprocesorové systémy nemají všechny znaky typických multiprocesorových systémů. Jejich název vyplývá z toho, že pod pojmem procesor obvykle chápeme jednotku, schopnou dekódovat a vy-

konávat instrukci. V tomto smyslu se tedy nejedná o skutečné multiprocesorové systémy, neboť vícenásobné jsou pouze výkonné nebo výpočetní jednotky. Patří k nim systémy s paralelními procesory, systémy s překrývanou činností a systémy odolné proti chybám. K systémům s paralelními procesory patří systémy s procesorovým polem, systémy s asociativními procesory a procesorové soubory. V systémech s procesorovým polem předepisují instrukce současné zpracování vektorů dat. Systémy s asociativními procesory zpracovávají data na základě jejich obsahu a nikoliv na základě jejich adresy. Systémy s překrývanou činností provádějí opakující se operace např. styk s operační pamětí, převzetí instrukce, vykonání instrukce s časovým překrytím, čímž se dosahuje větší propustnost systému. Systémy odolné proti chybám mají takovou organizaci a strukturu, která zajišťuje jejich maximální spolehlivost.

V obr. 18.2 je uvedena další klasifikace číslicových systémů, která dělí systémy z funkčního hlediska do čtyř kategorií. Pod čárkovou čarou je naznačen vztah mezi takto chápánými kategoriemi a kategoriemi, které jsme zavedli v předcházejícím textu.

K prvním kategoriím patří systémy s jediným tokem instrukcí a jediným tokem dat. Tyto systémy můžeme podle předcházejícího dělení označit jako jednoprocesorové systémy. Jejich blokové schéma je naznačeno v obr. 18.3. Druhou kategorii představují systémy s jediným tokem instrukcí a vícenásobným tokem dat. Tyto systémy obsahují jediný řadič, který přebírá a zpracovává instrukce. Instrukce však předepisuje činnost několika výpočetních jednotek, jak znázorňuje obr. 18.4. Tyto výpočetní jednotky řeší stejnou úlohu, pouze s rozdílnými datovými soubory. K této kategorii patří systémy s paralelními procesory. K třetí kategorii patří systémy s více-násobným tokem instrukcí a jediným tokem dat. Do této kategorie nelze jednoznačně přiřadit některou kategorii podle předešlé klasifikace. Můžeme tam řadit systémy s překrývanou činností; ty však mnohdy na základě své struktury patří mezi systémy třetí nebo čtvrté kategorie. Konečně poslední kategorie představuje systémy s více-násobným tokem instrukcí a vícenásobným tokem dat, jejichž strukturu znázorňuje obr. 18.5. Jejich velká propustnost vyplývá z možnosti současně nezávisle zpracovávat datové soubory a předávat si výsledky tohoto zpracování. Je však podmíněna synchronizací mezi jednotlivými nezávisle zpracovávanými dílčími úlohami a vhodným rozkla-

dem úlohy na tyto dílčí úlohy. Tento problém se u jiných kategorií nevyskytuje. Této kategorii lze přiřadit typické multiprocesorové systémy i multipočítacové systémy.

18.2 Vlastnosti systémů s několika procesory a jejich srovnání

V tomto odstavci uvážíme vlastnosti všech kategorií systémů s více procesory s ohledem na vhodnost jejich aplikace v rozsáhlých číslicových systémech s velkou propustností: multipočítacových systémů, typických multiprocesorových systémů a systémů s paralelními procesory. Tyto tři kategorie lze posuzovat z jednoho ohledu - podle stupně vazby mezi jednotlivými procesory v systému. Tento stupeň vazby můžeme chápát jako nejdelsí dobu přístupu libovolného procesoru systému k společným datovým strukturám.

Jednotlivé počítače multipočítacového systému jsou propojeny sériovými nebo paralelními informačními cestami a komunikace mezi nimi se řídí pravidly řídicích postupů, z nichž některé jsme poznali v předchozích kapitolách. Přenosová rychlosť informačních cest je maximálně 10 M bit/s. Tato přenosová rychlosť je sice dostatečná pro mnohé aplikace mikroprocesorů, jako např. v terminálových sítích, ale z ohledu požadavků na aplikace mikroprocesorů při výstavbě rozsáhlějších systémů s velkou propustností nemusí vyhovovat.

V typických multiprocesorových systémech má každý procesor přímý přístup k datům ve společné operační paměti prostřednictvím časově sdílené sběrnice, kterou jsme poznali v předchozích kapitolách, nebo prostřednictvím vícenásobného vstupně-výstupního člena operační paměti. Tato druhá alternativa je znázorněna na obr. 18.6. Synchronizaci mezi jednotlivými procesory v multipočítacových systémech zajišťují zprávy, které si vzájemně vyměňují jednotlivé mikropočítače. Na rozdíl od této metody, která přináší omezení dané přenosovou rychlosťí informačních cest, se synchronizace v multiprocesorových systémech většinou zajišťuje sdílením dat ve společné operační paměti. Rychlosť synchronizace je pak dána dobou přístupu jednotlivých mikroprocesorů k společné operační paměti. Z tohoto ohledu je určujícím faktorem struktura obvodů zajišťujících přístup k této paměti. Podle typu této struktury se v obr. 18.1 klasifikují typické multiprocesorové systémy na tři typy, což svěd-

čí o hledání nevhodnější struktury. Vznikly přitom i systémy s hierarchickou kombinovanou strukturou, např. s vícenásobnou sběrnicí a vícenásobným vstupním i výstupním členem.

V systémech s paralelními procesory, ve kterých všechny výpočetní jednotky zpracovávají data na základě jediného toku instrukcí, je představa komunikace mezi procesory totožná s představou řídicích signálů vyměňovaných mezi řadičem a výpočetními jednotkami. Tyto signály se obvykle přenáší přímo, bez přístupu k operační paměti, takže tato struktura má nejrychlejší komunikaci.

Kvantitativní představa o době přístupu ke sdíleným datovým strukturám je uvedena na obr. 18.7. Tento parametr je měřítko vhodnosti dané kategorie číslicového systému pro uvažovanou aplikaci. Jestliže aplikace vyžaduje vzájemnou interakci jednotlivých procesů v intervalech několika sekund, bude vyhovovat organizace multipočítáčového systému, např. mikropočítáčové sítě. Na druhé straně implementace algoritmu předepisujícího paralelní vyčíslení aritmatického výrazu vyžaduje interakci během každého instrukčního cyklu; tento požadavek splní jen systém s paralelními procesory. Střední interval požadované interakce mezi paralelně probíhajícími procesy se stává tedy základním parametrem, který popisuje aplikace a je východiskem pro volbu struktury systému s několika procesory, jenž ji bude schopen řešit.

18.3 Synchronizace procesů v multiprocesorovém systému

Z předchozího textu vyplývá, že pro většinu aplikací bude nejvhodnější strukturou některá z variant multiprocesorového systému. Synchronizace procesů v takovém systému se nejlépe zajistí prostřednictvím společné operační paměti, ve které dochází k výměně zpráv mezi procesy zpracovávanými jednotlivými procesory. Přitom není rozhodující, zda se styk se společnou operační pamětí uskutečňuje prostřednictvím vícenásobného vstupně-výstupního člena této paměti nebo časově sdílené sběrnice. Aby se předešlo konfliktům vznikajícím současnými žádostmi o styk se společnou operační pamětí, je výhodné, aby každý procesor měl vlastní operační paměť, ve které by mohl uložit program a data zpracovávaného procesu a společnou operační paměť by využíval jen k zabezpečení požadované interakce mezi jednotlivými procesy.

Společná operační paměť se obvykle dělí na zóny pro výměnu zpráv a jednu zónu stavových slov. Každému procesoru je přidělena jedna zóna pro výměnu zpráv, jež se dále dělí na subzóny, někdy nazývané schránky. Každá schránka je určena pro zprávu od jednoho procesoru. Každá zóna má tedy $N - 1$ schránek, kde N je počet procesorů systému. Mnohdy se vyčleňuje jedna zóna, která obsahuje zprávy pro všechny ostatní procesory. V zóně stavových informací se ukládá informace o stavu všech schránek zóny pro výměnu zpráv, např. o tom, zda schránka obsahuje zprávu nebo je prázdná, popřípadě identifikaci druhu zprávy, tj. jejího formátu a interpretace. Příklad organizace společné operační paměti je uveden v obr. 18.8.

Každý procesor systému může na základě obsahu zóny stavové informace žádat o styk s operační pamětí, aby mohl převzít pro něj určenou zprávu a zpracovat ji. K převzetí stavové informace dochází buď metodou obdobnou režimu výzvy (kap. 17), při níž si jednotlivé procesory systému postupně vyzvedávají stavovou informaci, nebo metodou přerušení, při níž procesor, který zapsal zprávu pro jiný procesor systému a současně předal odpovídající stavovou informaci, vyvolává přerušení všech procesorů systému. Procesory pak po přerušení testují jím přidělenou stavovou informaci. Jestliže některý z procesorů rozezná, že jeho zóna obsahuje zprávu, převeze ji a po přerušení testuje jemu přidělenou stavovou informaci. Metoda přerušení je z časového hlediska výhodnějším řešením, vyžaduje však odpovídající technické vybavení. Další možnosti, která je z časového hlediska nejvýhodnější, avšak z hlediska potřebného technického vybavení nejnáročnější, je metoda přerušení, při níž vyvolává procesor odesílající zprávu přerušení jen toho procesoru, který je příjemcem zprávy.

Složitost systému přerušení v multiprocesorových systémech nařístá. Je třeba rozeznávat celkem čtyří úrovně přerušení:

- přerušení, ke kterým dochází na úrovni jednotlivých procesorů, např. přerušení vyvolané přetečením registrů, nedovoleným přístupem k operační paměti apod.
- úroveň přerušení na vnitřní systémové úrovni; sem patří přerušení od systémových periferií; na této úrovni se většinou používá ještě statické prioritní schéma
- úroveň přerušení na úrovni operačního systému; toto přerušení vyvolávají programově jednotlivé úlohy a operační systém před

předáním řízení další úloze obvykle kontroluje stav celého systému; používá se většinou mnohem komplikovanější prioritní schéma s dynamickou alokací priorit řízenou programem operačního systému

- úroveň přerušení na vnější systémové úrovni, ke které patří přerušení vyvolávané žádostmi procesorů o komunikaci s jinými procesory systému, o styk s jinými systémovými prostředky nebo se společnou bází dat; i na této úrovni je nezbytným předpokladem flexibilního systému dynamická alokace priorit.

Požadavek synchronizace procesů v multiprocesorovém systému ovlivňuje podstatným způsobem jeho architekturu. V tomto odstavci jsme poukázali jen na dva aspekty, které jsou s tímto problémem spojeny. Další aspekty vyplynou z příkladu, který uvádíme v dalším textu.

18.4 Příklad multiprocesorového systému

Systém C_m^* je příkladem multiprocesorového systému, který byl vyvinut s cílem podrobit analýze a prakticky vyzkoušet možnosti, které nabízí systém tvořený několika sty procesorových elementů pro realizaci výpočetního systému s velkou propustností.

Při volbě architektury systému C_m^* byly vodítkem jednak praktické zkušenosti získané s podobnými systémy na různých pracovištích - jednak technické a ekonomické úvahy, které záhrnovaly např. vlastnosti stávajících obvodů LSI, požadavky na rozšiřitelnost systému jak po stránce technických požadavků tak technického vybavení a pod.

Předchozí experimenty např. ukázaly, že není nezbytnou podmínkou, aby všechny procesorové elementy měly stejný přístup ke všem částem operační paměti, neboť většina paměťových referencí směřuje obvykle k poměrně úzké oblasti adresového prostoru paměti. Lokální reference na programové instrukce a okamžitá data představují např. v realizovaném systému C_m^* více než 90 % všech paměťových referencí. Globální reference na společnou zónu paměti mohou mít tedy podstatně delší přístupovou dobu než lokální reference.

Uspořádání systému C_m^* je uvedeno v obr. 18.9. Základním prvkem je počítačový modul tvořený procesorovým elementem, lokální pamětí, vstupními zařízeními a lokálním přepínačem realizující

cím rozhraní mezi počítačovým modulem a ostatním systémem. Operační paměť systému tvoří výhradně lokální paměti jednotlivých počítačových modulů. Vyšším prvkem hierarchické struktury systému C_m^* je až čtrnáctiprvkový soubor počítačových modulů. Počítačové moduly spolu komunikují prostřednictvím sběrnic. Tuto komunikaci řídí speciální adresovací procesor. Paměťové reference jednotlivých procesorových elementů se mohou vztahovat ke kterémukoliv paměťovému místu operační paměti. Lokální přepínač na základě jednoduchých mapovacích tabulek rozhodne o tom, zda paměťová reference se vztahuje k lokálnímu adresovému prostoru nebo ke globálnímu adresovému prostoru. V prvním případě zajistí přístup k adresovanému paměťovému místu samotný lokální přepínač. V druhém případě předá lokální přepínač paměťovou referenci ke zpracování adresovacímu procesoru. Ten rozliší, zda se paměťová reference vztahuje k adresovému prostoru vlastního souboru paměťových modulů, anebo k adresovému prostoru jiného souboru a zprostředuje uskutečnění instrukce s globální paměťovou referencí buď prostřednictvím odpovídajícího lokálního přepínače nebo prostřednictvím adresovacího procesoru, k jehož adresovému prostoru se reference vztahuje. Adresovací procesory spolu komunikují prostřednictvím mezipočítačových sběrnic; ke každému z nich mohou být připojeny dvě sběrnice. Když nejsou dva adresovací procesory, které spolu potřebují komunikovat, propojeny přímo sběrnicí, zprostředuji komunikaci další adresovací procesory.

Jednotlivé procesorové elementy nemusí rozlišovat typy paměťových referencí; obracejí se k virtuálnímu systémovému adresovému prostoru s délkou adresového slova 28 bitů, tj. s rozsahem asi $2,7 \cdot 10^8$ slov. Překlad virtuální adresy na adresu fyzického adresového prostoru odpovídajícího procesorového modulu zajišťuje technické a programové vybavení systému C_m .

Charakteristickým znakem architektury systému C_m je předávání zpráv po paketech. Sběrnice nejsou vyhrazeny paměťové referenci po celou dobu jejího zpracování tak jak je to obvyklé, ale pouze po dobu, která je nutná k předání jednoho paketu obsahujícího adresu nebo adresu a data z jednoho uzlu sběrnice k jinému uzlu.

Předpokládá to ovšem vybavení těchto uzlů vyrovnávacími registry nebo paměti, které slouží pro uložení paketů. Tím se zvýší využití sběrnic, sníží počet konfliktních požadavků na jejich přidělení a nebezpečí samoblokování systému, neboť po dobu, kdy např. adresovací procesor zpracovává přijatou paměťovou referenci, uvolňuje se sběrnice pro komunikaci jiných procesorů.

Architektura systému C_m nabízí mnoho možností vzájemné komunikace mezi procesory a pamětí. Uživatel jej proto může interpretovat buď jako multiprocesorový systém nebo jako multipočítáčový systém a odpovídajícím způsobem programovat uživatelskou úlohu.

18.4.1 Mechanismus virtuálního adresování

Virtuální adresový prostor má rozsah 270 MB. Je rozdělen na segmenty, z nichž každý má rozsah maximálně 4 KB. Program se odvolává na jednotlivé segmenty nepřímo pomocí globálních odvolávek, které kromě globálního jména segmentu specifikují i právo přístupu k segmentu. Každý procesor má adresový prostor s rozsahem 64 KB, který je rozdělen do šestnácti stránek. Každá stránka může být současně reprezentována jedním ze segmentů virtuálního adresového prostoru. Nejvyšší stránka v adresovém rozsahu každého procesoru je rezervována pro programovou interakci s adresovacím procesorem. Obsahuje 15 pseudoregistrů, jejichž obsah definuje vztah mezi stránkami ve fyzickém adresovém prostoru procesoru a segmenty ve virtuálním adresovém prostoru.

Každý programový modul je reprezentován svými relokačními tabulkami. Tyto tabulky mu umožňují přístup ke všem segmentům virtuálního adresového prostoru, které má přiděleny. Změna zobrazení stránek fyzického adresového prostoru procesoru na segmenty virtuálního adresového prostoru se dosáhne změnou obsahu uvedených pseudoregistrů. Příslušný adresovací procesor na základě zápisu do této registrů nastavuje své relokační tabulky. Jestliže se změna dotýká segmentu, který je fyzicky umístěn v lokální paměti procesorového modulu, pak adresovací procesor aktualizuje relokační tabulku lokálního přepínače. Lokální paměť počítačového modulu má adresový rozsah 256 KB.

První úroveň mechanismu virtuálního adresování se uplatňuje při lokálních paměťových referencích (obr. 18.10). Adresa generova-

ná procesorem se překládá tak, že nejnižších 12 bitů zůstává beze změny a nejvyšší čtyři bity se použijí jako vstupy do relokační tabulky lokálního přepínače. Tato relokační tabulka generuje 6 bitů, které ukazují na začátek stránky ve fyzickém adresovém prostoru počítačového modulu. Nejnižších 12 bitů má význam posunutí v rámci stránky.

Druhou úroveň mechanismu virtuálního adresování znázorňuje obr. 18.11. Jestliže lokální přepínač rozezná, že virtuální adresa není obsažena ve fyzickém adresovém prostoru vlastního počítačového modulu, předá požadavek na zpracování adresovacímu procesoru. Tento procesor převeze adresu spolu s číslem počítačového modulu požadujícího paměťovou referenci s informací, zda vyžaduje čtení nebo zápis. Jestliže adresovací procesor rozezná, že požadovaná adresa leží uvnitř vlastního souboru počítačových modulů, přeloží ji pomocí relokačních tabulek a předá spolu s číslem modulu, který adresa obsahuje, a to ve formě fyzické adresy. Lokální přepínač adresovaného modulu ji přijme a vyvolá operaci čtení nebo zápis přímým stykem s lokální operační pamětí. Po ukončení operace předává lokální přepínač žádost o ukončení adresovacímu procesoru, ten po jejím přijetí potvrdí procesoru, který paměťovou referenci žádal, její úspěšné ukončení a při operaci čtení mu předá data.

Třetí úroveň mechanismu virtuálního adresování představuje obr. 18.2. Využívá se tehdy, jestliže adresovací procesor rozezná virtuální adresu, která není obsažena v jeho relokačních tabulkách, tj. není obsažena v operační paměti jeho souboru počítačových modulů. V tomto případě předává jméno segmentu 16 bitů, operační kód 3 bitů a posunutí 12 bitů, tj. úplnou virtuální adresu požadovaných dat s informací o typu paměťové reference. Tato informace je předána po mezipočítačové sběrnici ostatním adresovacím procesorům, a předává se mezi těmito procesory, dokud ji nepřeveze procesor, k jehož fyzickému adresovému prostoru operační paměti souboru se virtuální adresa vztahuje. Tento adresovací procesor pak uskutečňuje zobrazení virtuální adresy do fyzického adresového prostoru a předá fyzickou adresu příslušnému lokálnímu přepínači, který provede požadovanou operaci a potvrdí adresovacímu procesoru její ukončení, popřípadě mu též předá data. Potvrzující zpráva se pak předává adresovacímu procesoru toho souboru, který o paměťovou referenci žádal. Adresovací procesor pak ukončí zpracování paměťové referen-

ce předáním potvrzení, popřípadě dat příslušnému počítačovému modulu.

18.4.2 Uspořádání adresovacího procesoru

Z předchozího odstavce vyplývá, že operace, spojené s mechanismem adresování jsou poměrně náročné a mohly by výrazně snížit propustnost systému. Proto byla zvolena taková struktura, která dovoluje, aby jednotlivé fáze operace, a sice komunikace s počítačovými moduly, výpočet virtuální adresy, komunikace po mezipočítačové sběrniči a provedení předeepsané operace čtení nebo zápisu lokálním přepínačem byly na sobě nezávislá aby jeden adresovací procesor mohl současně zpracovat osm paměťových referencí.

Uvedená struktura je znázorněna na obr. 18.13. Adresovací procesor tvoří tři jednotky: řídící procesor sběrniče, mapovací procesor a vazební obvod. Řídící procesor sběrniče řídí všechny přenosy po mezipočítačové sběrniči. Mapovací procesor zobrazuje adresy z virtuálního do fyzického prostoru a naopak; kromě toho koordinuje všechny činnosti, které v adresovacím prostoru probíhají. Vazební obvod realizuje rozhraní s dvěma mezipočítačovými sběrnicemi a má dva vazební členy. Komunikace po mezipočítačových sběrnicích se uskutečňuje formou krátkých zpráv podle pravidel řídícího postupu, jehož algoritmy jsou implementovány mikroprogramem mapovacího procesoru.

Rozhraní těchto tří funkčních bloků adresovacího procesoru tvoří pět front, které umožňují nezávislé zpracování osmi paměťových referencí. Každý proces spojený se zpracováním jedné paměťové reference má k dispozici osm univerzálních registrů a čtyři spojovací registry podprogramů. Řídící procesor sběrniče udržuje stavovou informaci o stavu každého procesoru a přiděluje mu číslo. Přijaté žádosti o paměťové reference řadí do fronty na zpracování adres. Mapovací procesor je zpracuje a řadí buď do fronty požadavků na operační paměť, nebo do jedné z front na vysílání mezipočítačovou sběrniči. Přijaté informace z mezipočítačových sběrnic řadí vazební obvod do fronty požadavků na obsluhu anebo potvrzujících zpráv, předávaných jinými adresovacími procesory.

Řídící procesor sběrniče je horizontálně mikroprogramovaný procesor s kapacitou řídící paměti 256 x 40 bitů, s dobou mikro-

instrukčního cyklu 100 ns. Mapovací procesor je stejného typu a má řídicí paměť 1000×80 bitů a paměť s libovolným výběrem s kapacitou 5000×16 bitů pro uložení informací o odvolávkách na segmenty virtuální paměti.

18.4.3 Zhodnocení aplikačních možností

Architekturu systému je nutné posuzovat především z hlediska snadnosti jejího programování. Tomuto hledisku byl podřízen i návrh architektury systému C_m . Projevuje se to např. v návrhu mechanismu virtuálního adresování, který byl popsán v předcházejících odstavcích. Tento mechanismus umožňuje, aby se operační paměť systému jeyila programátoru jako zcela homogenní a spojitá, ačkoliv je fyzicky rozdělena do jednotlivých počítačových modulů.

Tuto skutečnost ilustruje obr. 18.14, ve kterém je uveden příklad, jak může programátor organizovat program a jeho datové struktury ve virtuální operační paměti. Jednotlivé komponenty programu mohou být zavedeny do libovolných fyzických operačních pamětí počítačových modulů, na základě jejich umístění jsou inicializovány relokační tabulky příslušných procesorů, které zajistí adresovatelnost jednotlivých komponent programu.

Z hlediska programátora je uživatelský program zaveden ve virtuální paměti tak, jak znázorňuje horní část obrázku. V dolní části obrázku je znázorněno umístění jednotlivých komponent programu ve fyzických operačních pamětech počítačových elementů 1 až 3. Při výpočtu programu A se převzetí jeho jednotlivých instrukcí uskutečňuje lokálními paměťovými referencemi, tedy bez časových ztrát. Rovněž při zpracování okamžitých hodnot i při práci se zásobníkem pracuje program A na základě lokálních paměťových referencí. Když však dojde k vyvolání procedury B, která vyžaduje přístup k proměnným pole C, pak dochází k tomu, že odpovídající paměťové reference jsou globální, ovšem uvnitř souboru počítačových modulů. Jestliže lokální přepínač rozezná tuto referenci, předá virtuální adresu adresovacímu procesoru, který ji převede na fyzickou adresu operační paměti počítačového modulu 1 nebo počítačového modulu 2, což závisí na řádku pole, ve kterém je požadovaná proměnná, a zahájí styk s touto pamětí prostřednictvím příslušného lokálního přepínače. Programátor ani samotný procesor počítačového modulu 1 nerozeznává

zda přístup k proměnné C byl zajištěn lokální nebo globální referenci. Jediným rozdílem je podstatně delší doba přístupu k této proměnné.

V zásadě je možné, aby program A prováděl procesor, který není ve stejném počítačovém modulu, v jehož operační paměti je umístěn program; je ovšem nutné správně inicializovat relokační tabulky. Výpočet programu se tím ovšem podstatně prodlouží.

Pokud rozklad úlohy umožňuje, aby procedura B mohla současně pracovat s proměnnými z pole C, je možné alokovat proceduru B současně ve dvou nebo několika operačních pamětech, jak znázorňuje obr. 18.15. Předáním vhodně volených parametrů této proceduře lze zajistit, aby pracovala s proměnnými umístěnými v různých oblastech pole C.

Zatímco lze zajistit, aby se alokace procesorů provádějících program i alokace segmentů virtuální paměti v operačních pamětech počítačových modulů uskutečňovala automaticky operačním systémem bez zásahu operátora, rozklad úlohy na paralelně probíhající dílčí úlohy musí řešit dosud programátori. Problém dekompozice úlohy na dílčí paralelní úlohy je předmětem výzkumu tak, jak jsme již uvedli v úvodu této kapitoly. Některé jazyky, např. Pascal a Modula, byly doplněny prostředky pro vyjádření algoritmů, které mají explicitní paralelismus (viz [10]).

Dalším problémem, který je spojen s možnostmi aplikace systému C_m , je realizace jeho nestandardních technických prostředků, především lokálního přepínače a adresovacího procesoru.

Problém realizace lokálního přepínače není tak tíživý. Dá se očekávat, že rozvoj technologie dovolí v nejbližší době realizovat počítačový modul, včetně všech jeho prvků, tj. procesoru, lokálního přepínače a lokální operační paměti v jediném pouzdro. Adresovací procesor však představuje závažnější problém, neboť ve stávajícím provedení představuje asi 500 integrovaných obvodů střední integrace,

Systém C_m tedy zatím nevyřešil všechny problémy, které jsme si uvedli v úvodu této kapitoly. Jeho popisem jsme chtěli ilustrovat, jak složitá a obtížná cesta vede k jejich řešení.

19 LITERATURA

1. Literatura, na kterou jsou odkazy v textu:

- /1/ Intel Corporation, Santa Clara: Intel 3000 Mikroprogramming Manual, 1976
 - /2/ Intel Corporation, Santa Clara: Intel 3000 Reference Manual, 1975
 - /3/ Intel Corporation, Santa Clara: Intel Multibus Interfacing, Pub. AP-28, 1977
 - /4/ FORCE, J.: Microprocessor bus could cure designers' woes. Electronics, July 20, 1978
 - /5/ British standard 4421: 1969, UDC 681.327. Specifications for a digital input/output interface for data collection systems
 - /6/ Systéma malých elektronických mašin. Interfejs dlja radialnovo podklučenija ustrojstv s paralelnoj pěredačej informacijí „IRRP“.
 - /7/ Zelená kniha, sv. VIII - Přenos dat. Překlady doporučení Zelené knihy CCITT, sv. VIII. Praha, NADAS, 1974
 - /8/ Oranžová kniha, sv. VIII - Přenos dat. Překlady doporučení Oranžové knihy CCITT, sv. VIII. Praha, NADAS, připravuje se k vydání.
 - /9/ Intel Corporation, Santa Clara: MCS-80 Users' Manual, 1978
 - /10/ BRINCK, P., HANSEN, J.: The architecture of concurrent programs. New Jersey, Prentice Hall, 1978
 - /11/ SWAN, J.R., FULLER, S.H., SIEWOREK, D.P.: Cm*- A modular multi-microprocessor. In: Proceedings of National Computer Conference, 1977
2. Další doporučená literatura:
- /12/ HILBURN, J.L., JULICH, P.N.: Microcomputers, Microproces-

- sors: Hardware, Software and Applications. Englewood Cliffs, N.J., Prentice Hall, 1976
- /13/ SOUČEK, B.: Minicomputers and Microcomputers. New York, John Wiley and Sons, 1976
- /14/ PEATMAN, J.B.: Microcomputer based design, New York, Mc Graw Hill Inc, 1977
- /15/ SIPPL, Ch. J.: Microcomputer Handbook. New York, Mason / Charter Publishers, Inc., 1977

Ing. Zdeněk SOBOTKA, CSc
Ing. Jaroslav STARÝ

MIKROPOČÍTAČE

Textová část

DT 681.32-181.4

Vydalo Ústředí technického průzkumu a služeb - ÚTEPS v TESLA - VÚST, Novodvorská 994, Praha 4 - Braník ve spolupráci s po-bočkou ČSVTS - pro interní potřebu hospodářských organizací TESLA a spolupracujících organizací jako svou neperiodickou interní publikaci č. 81-08-279/l - redakce Cyril Kavan - vy-robeno v ÚTEPS - 390 stran, vydání druhé, náklad 700 výtisků. V publikaci nebylo provedeno lektorování. Publikace je pro-dejná pouze socialistickým organizacím.