

# Sériová rozhraní SPI, Microwire, I<sup>2</sup>C a CAN

K.D. 2002

## 1 Úvod

Sériová komunikační rozhraní se v mikropočítačové technice používají ke dvěma základním účelům:

1. Ke komunikaci mezi jednotlivými mikropočítačovými moduly. Typická délka vedení je zde v řádu jednotek až stovek metrů, takže fyzická vrstva v různé míře řeší i problémy odolnosti proti rušení atd. Kromě známých a rozšířených rozhraní (RS232, RS485) se používají i specializovaná rozhraní resp. sběrnice (CAN, TTP). Při rostoucích nárocích na přenosovou kapacitu se pro toto propojení často používá i Ethernet.
2. Ke komunikaci mezi integrovanými obvody, případně ke komunikaci mezi mikropočítačovými moduly na krátkou vzdálenost. Typická délka vedení zde nepřesahuje jednotky metrů. Často používaná rozhraní pro komunikaci mezi jednotlivými IO jsou rozhraní Microwire, SPI a I<sup>2</sup>C. Tato rozhraní budou spolu s rozhraním CAN předmětem následujícího popisu.

Důvodem používání sériové komunikace mezi jednotlivými obvody je především zmenšení počtu vývodů jejich pouzder. Při použití sériových pamětí se zredukuje množství adresních, datových a řídicích vývodů obvyklé (nikoliv sériové) paměti na tři až čtyři vývody. To umožňuje zmenšit rozměry pouzdra i spojové desky, protože odpadá prostorově náročné propojování velkým počtem vodičů.

Další výhodou může v některých případech být možnost připojení obvodů se sériovým rozhraním i k mikrokontrolérům bez vyvedené vnitřní sběrnice, v krajním případě i bez příslušného řadiče sériového rozhraní. Funkce řadiče je potom realizována programově s využitím několika vývodů vhodného portu mikrokontroléru. Nelze tak většinou dosáhnout plné rychlosti daného rozhraní, ale tato skutečnost nemusí být v některých případech na závadu.

## 2 Rozhraní SPI

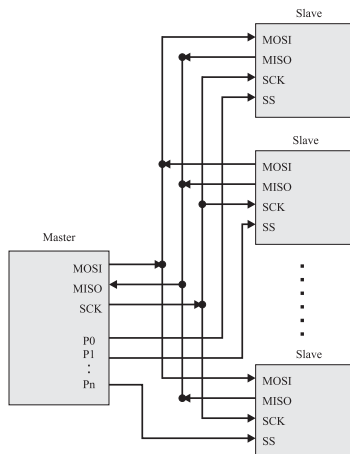
Rozhraní SPI je určeno především pro připojení vnějších pamětí, A/D převodníků a dalších obvodů k mikrokontroléru, případně pro vzájemnou komunikaci mezi mikrokontroléry. U některých mikrokontrolérů je SPI využíváno i pro programování jejich vnitřní paměti Flash.

Základní koncepce systému využívajícího sběrnici SPI je následující (viz Obr. 1):

V systému mohou být zapojeny dva nebo více obvodů. Jeden z obvodů, obvykle procesor, je typu Master, ostatní jsou typu Slave. Jednotlivé obvody jsou propojeny čtyřmi vodiči:

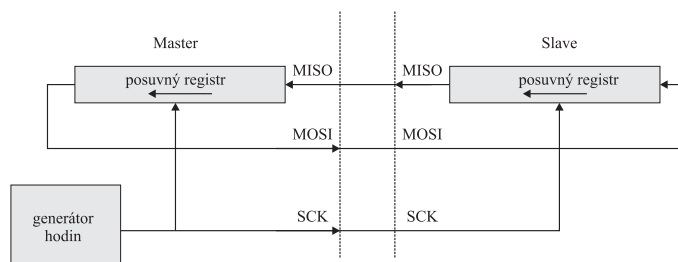
- Datový výstup MOSI (Master Out, Slave In) obvodu Master je připojen na vstupy MOSI všech obvodů Slave.

- Datový vstup MISO (Master In, Slave Out) obvodu Master je propojen s výstupy MISO všech obvodů Slave.
- Výstup hodinového signálu SCK je připojen na vstupy SCK všech obvodů Slave.
- Každý obvod Slave má vstup SS (Slave Select) pro výběr obvodu. Je-li SS v neaktivní úrovni, je rozhraní SPI daného obvodu neaktivní a jeho výstup MISO je ve vysokoimpedančním stavu. Vstupy SS jednotlivých obvodů jsou samostatnými vodiči propojeny s obvodem Master. Je-li obvodem Master mikrokontrolér, bývají tyto vodiče připojeny k některému z jeho portů. Tak lze snadno vybírat obvod, se kterým má být v daném okamžiku vedena komunikace.



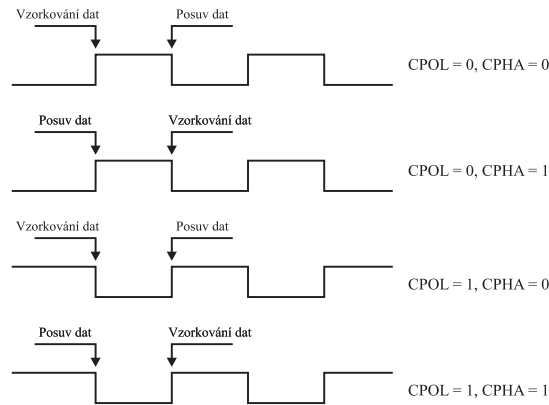
**Obr. 1** Celková koncepce systému se sběrnici SPI.

Přenosy na sběrnici SPI probíhají vždy mezi obvodem Master a některým z obvodů Slave. Oba obvody obsahují posuvné registry, které jsou v okamžiku komunikace propojeny tak, jak je schematicky naznačeno na Obr. 2.



**Obr. 2** Propojení stanic Master a Slave.

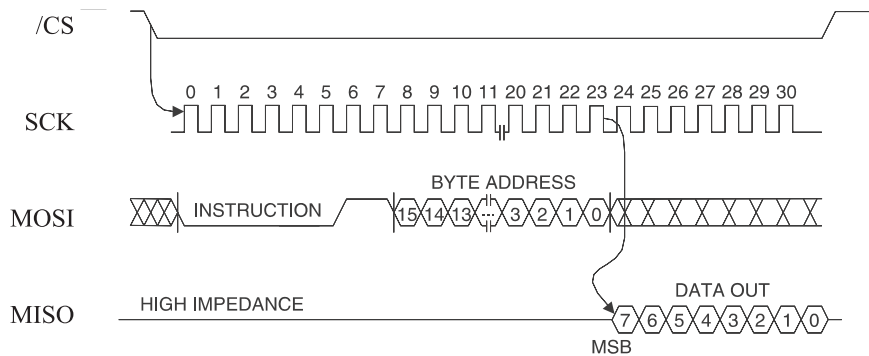
Obvod Master generuje hodinový signál, který řídí posouvání obou posuvných registrů. Kľidová úroveň signálu SCK a vztah mezi datovým a hodinovým signálem je dán parametry CPOL a CPHA (viz Obr. 3). Pokud je rozhraní SPI realizováno specializovaným řadičem, je obvykle možné tyto parametry v řadiči nastavit. Je-li rozhraní SPI realizováno programově, musí být okamžiky změny úrovně datových a hodinových signálů zvoleny tak, aby přijímající obvod vzorkoval ustálená data.



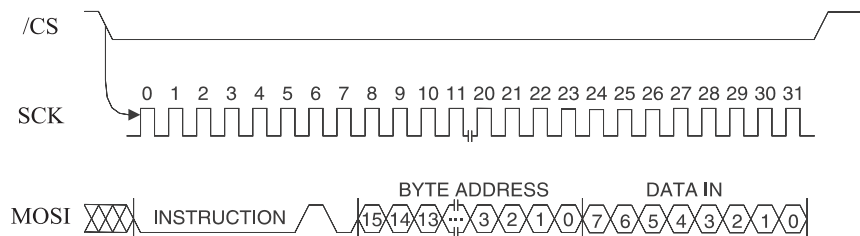
**Obr. 3** Význam parametrů CPOL a CPHA na SPI.

Napěťové úrovně jednotlivých signálů rozhraní SPI jsou dané použitou technologií. Maximální frekvence hodinového signálu je 2 MHz.

Na Obr. 4 a Obr. 5 je příklad komunikace se sériovou pamětí při čtení dat. Mikrokontrolér musí nejprve do paměti zapsat povel (čtení) a adresu dat. Potom jsou z paměti přečtena příslušná data.



**Obr. 4** Čtení dat z paměti s rozhraním SPI.

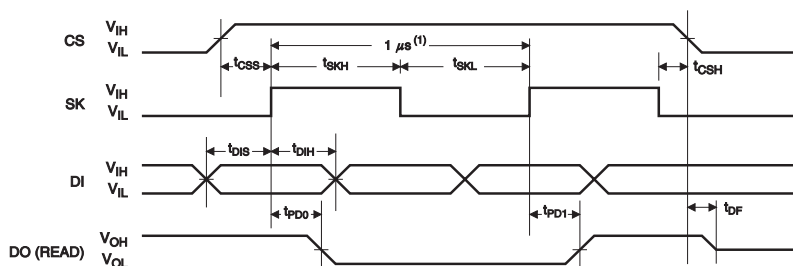


**Obr. 5** Zápis dat do paměti s rozhraním SPI.

### 3 Rozhraní Microwire

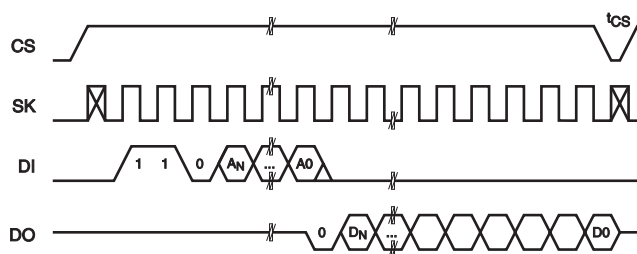
Microwire je jednoduché rozhraní v principu podobné rozhraní SPI. Mikrokontrolér je se všemi připojenými obvody propojen dvěma datovými vodiči DI a DO a jedním hodinovým vodičem SK. Pro výběr jednotlivých obvodů slouží samostatné výběrové vodiče CS. V obvyklé konfiguraci je rozhraním Microwire propojen mikrokontrolér s jedním nebo více paměťovými nebo jinými periferními obvody. Mikrokontrolér potom generuje hodinový signál SK a zapisovaná data DI. Čtená data jsou přenášena vodičem DO.

Základní časové relace mezi datovými a hodinovým signálem na straně periferních obvodů jsou znázorněny na Obr. 6. Obvody vzorkují vstupní data a mění hodnotu výstupních dat při vzestupné hraně signálu SK. Mikrokontrolér musí toto časování respektovat: Zapisovaná data musí být měněna v dostatečném předstihu ( $t_{DIS}$ ) před hranou SK, čtená data smí být vzorkována nejdříve po uplynutí větší z dob  $t_{PDO}$  a  $t_{PD1}$ . Maximální frekvence signálu SK je udávána 1 MHz.



Obr. 6 Vztah mezi signály SK, DI a DO na rozhraní Microwire.

Formát přenášených dat není rozhraním Microwire obecně řešen a závisí na konkrétním připojeném obvodu. Počet bitů v jednotlivých přenosech nemusí být v některých případech násobkem 8. Jako příklad je na Obr. 7 uveden formát dat při čtení z paměti EEPROM typu 93C66 ( $512 \times 8$  bitů). Po aktivaci paměti signálem CS je nejprve vysílán tzv. start bit, který je vždy 1. Potom je vysílán dvoubitový povel 10 (read) a potom postupně adresní bity A8 – A0. Celkem je tedy do paměti vysláno 12 bitů. Při vysílání posledního adresního bitu (A0) přejde výstup paměti z vysokoimpedančního stavu do úrovně 0. Při dalších 8 hodinových pulsech jsou na výstupu DO vysílány jednotlivé datové bity D7 – D0.



Obr. 7 Čtení dat z paměti 93C66 s rozhraním Microwire.

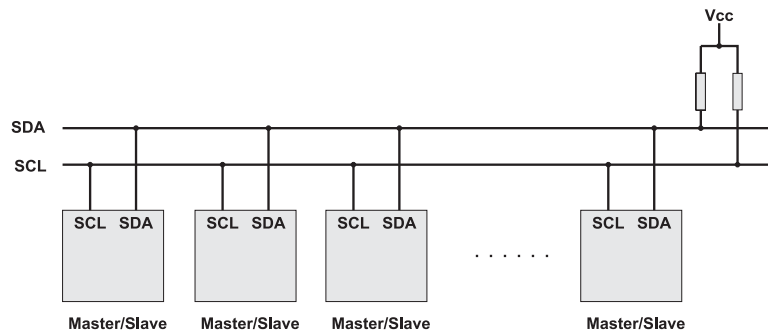
Pro řízení přenosů na rozhraní Microwire se často používá programové řešení, kdy jsou datové i hodinový signál připojeny na porty mikrokontroléru a programově nastavovány a čteny. Je-li mikrokontrolér vybaven řadičem SPI, lze jej v některých případech využít i pro rozhraní Microwire. Předpokladem je, že připojené obvody vyžadují nejprve vyslání start bitu

v úrovni 1. Nuly předcházející start bit jsou těmito obvody ignorovány a mohou proto být využity k prodloužení vysílaných dat a povelů na délku danou násobky 8.

## 4 Rozhraní I<sup>2</sup>C

### 4.1 Celková koncepce I<sup>2</sup>C

Na rozdíl od SPI a Microwire je I<sup>2</sup>C sběrnice typu multimaster. Řeší proto i arbitraci pro přístup na sběrnici a má zabudovaný mechanismus adresování jednotlivých připojených stanic nebo integrovaných obvodů. Základní koncepce sběrnice I<sup>2</sup>C je na Obr. 8. Každá stanice má určenou svou vlastní adresu o délce 7 nebo 10 bitů, která slouží k jejímu výběru i k arbitraci.

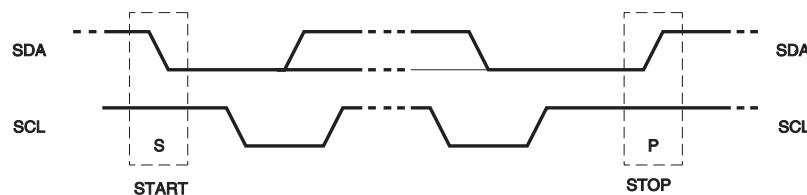


**Obr. 8** Celková koncepce sběrnice I<sup>2</sup>C.

### 4.2 Fyzická vrstva

Jednotlivé stanice rozhraní I<sup>2</sup>C jsou propojeny jedním datovým vodičem (SDA) a jedním hodinovým vodičem (SCL). Z elektrického hlediska jsou oba vodiče typu otevřený kolektor. Jejich maximální délka je dána jejich nejvyšší přípustnou kapacitou 400 pF.

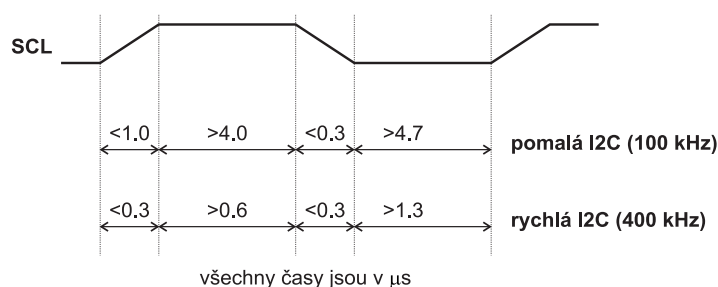
V klidovém stavu (volná sběrnice) jsou oba vodiče v úrovni H. Při probíhající přenosu jsou na SDA vysílány jednotlivé datové bity přičemž platí pravidlo, že logická úroveň na SDA se smí měnit pouze je-li SCL v úrovni L. Toto pravidlo je porušeno ve dvou speciálních případech – vysílání podmínek START a STOP, které se používají k zahájení arbitrace a k ukončení přenosu (viz Obr. 9).



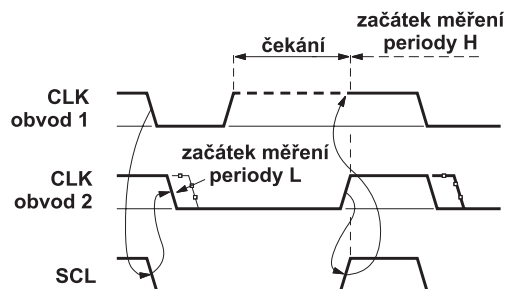
**Obr. 9** Podmínky START a STOP.

Maximální přípustná frekvence signálu SCL je podle verze I<sup>2</sup>C 100 kHz nebo 400 kHz. Pro obě frekvence je specifikována minimální povolená doba setrvání SCL v úrovni L i H (viz Obr. 10). Při arbitraci i při přenosu dat si jednotlivé stanice synchronizují generátory hodin tak, že trvání úrovně H na SCL je odměřováno vnitřním časovačem každé stanice až od okamžiku, kdy SCL skutečně úrovně H dosáhne (protože je SCL typu otevřený kolektor, může být v úrovni L držen i v situaci kdy se daná stanice snaží nastavit úroveň H). Podobně je doba trvání úrovně L na SCL odměřována od sestupné hrany (viz Obr. 11).

Tento mechanismus umožňuje některé ze stanic zpomalit přenos: pomalá stanice může podržet po určitou dobu signál SCL v úrovni L a tím zabránit vysílající stanici ve vyslání dalšího bitu.



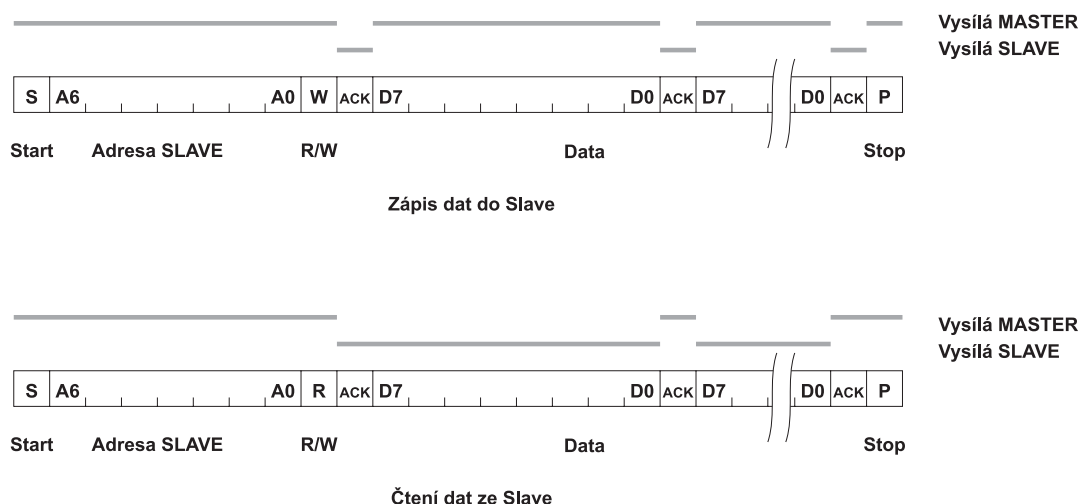
**Obr. 10** Časování signálu SCL na I<sup>2</sup>C.



**Obr. 11** Synchronizace hodin SCL.

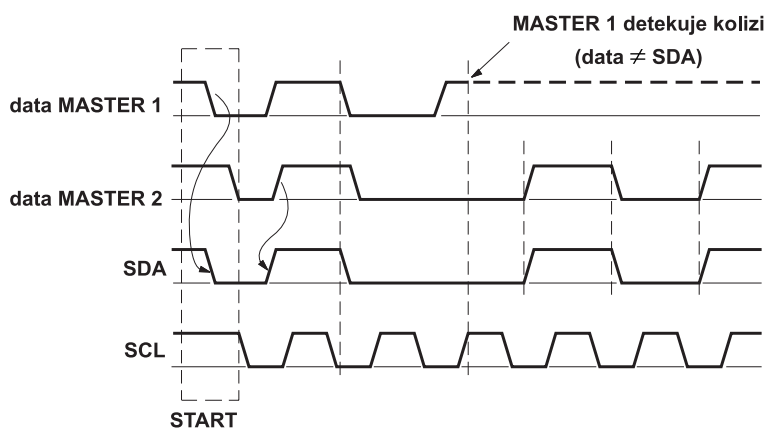
### 4.3 Linková vrstva

Formát rámce se sedmibitovou adresou při přenosu na I<sup>2</sup>C je na Obr. 12. Každému přenosu předchází vyslání podmínky START. Potom je vysílána 7bitová adresa příjemce a jeden bit R/W, který indikuje požadovanou operaci (čtení/zápis). Další bit ACK je vysílán s úrovní H a je určen k potvrzení přijímací stanicí. Dále jsou přenášena data ve směru určeném předchozím bitem R/W. Každý byte je následován jedním bitem ACK. Po ukončení přenosu je vysílána podmínka STOP.



**Obr. 12** Formát rámců na sběrnici I<sup>2</sup>C.

**Arbitrace** Pro arbitraci se na I<sup>2</sup>C používá metoda s detekcí kolize. Každá ze stanic může zahájit vysílání, je-li předtím sběrnice v klidovém stavu. Během vysílání musí neustále porovnávat vysílané bity se skutečným stavem SDA. Je-li zjištěn rozdíl mezi očekávaným a skutečným stavem linky SDA, je to indikace kolize mezi několika stanicemi. Vzhledem k charakteru sběrnice (otevřené kolektory) může k této situaci dojít, pokud určitá stanice vysílá úroveň H, zatímco jiná stanice vysílá úroveň L. Stanice, která na lince zjistí úroveň L, zatímco sama vysílá H, musí vysílání okamžitě ukončit (viz Obr. 13).



**Obr. 13** Arbitrace na I<sup>2</sup>C. Master1 ztrácí právo na sběrnici.

K arbitraci většinou dochází během vysílání několika prvních bitů, kdy je vysílána adresa přijímací stanice. Pokud by se např. dvě stanice současně pokusily o zápis do stejného obvodu, nastane kolize až při přenosu vlastních zapisovaných dat. V krajním případě, kdy několik stanic současně zapisuje stejná data na stejnou adresu, nemusí být kolize vůbec detekována.

**Adresování** Každá stanice (v praxi např. každý integrovaný obvod) připojená na I<sup>2</sup>C má přidělenou 7bitovou adresu. Po zachycení podmínky START porovnávají všechny obvody svou adresu s adresou, která je vysílána na sběrnici. Zjistí-li některý z obvodů shodu, je vysílání určeno právě jemu a musí přijetí adresy potvrdit bitem ACK. Potom přijímá resp. vysílá další data.

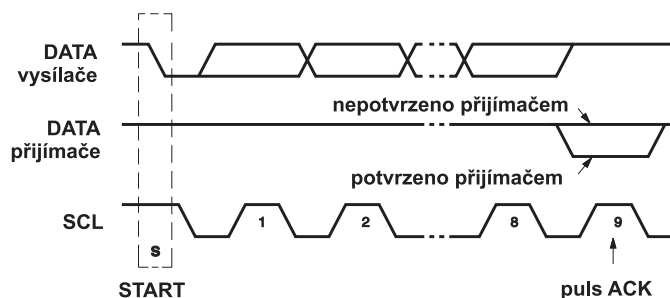
V praxi je většinou několik adresních bitů jednotlivých obvodů určeno už při výrobě (viz Tab. 1). Zbývající bity se volí pomocí příslušných vývodů daného obvodu, které se podle potřeby připojí na úroveň L nebo H.

Několik adres je na I<sup>2</sup>C vyhrazeno pro speciální účely. Např. adresa 0000000 je určena pro vysílání „broadcast“, adresy 0000011, 00001XX a 11111XX jsou rezervovány pro další účely.

Adresa 11110aa indikuje 10bitové adresování: aa zde označuje dva nejvyšší bity adresy stanice, zbývajících 8 bitů je vysíláno v následujícím bytu.

**Poznámka:** Popisované adresování se týká adresy jednotlivých *stanic* na I<sup>2</sup>C. Je-li stanicí např. paměťový obvod, jedná se o adresu celého obvodu. Adresa jednotlivých *slov* v paměti a požadovaná operace se do paměti přenáší v datové části rámce I<sup>2</sup>C.

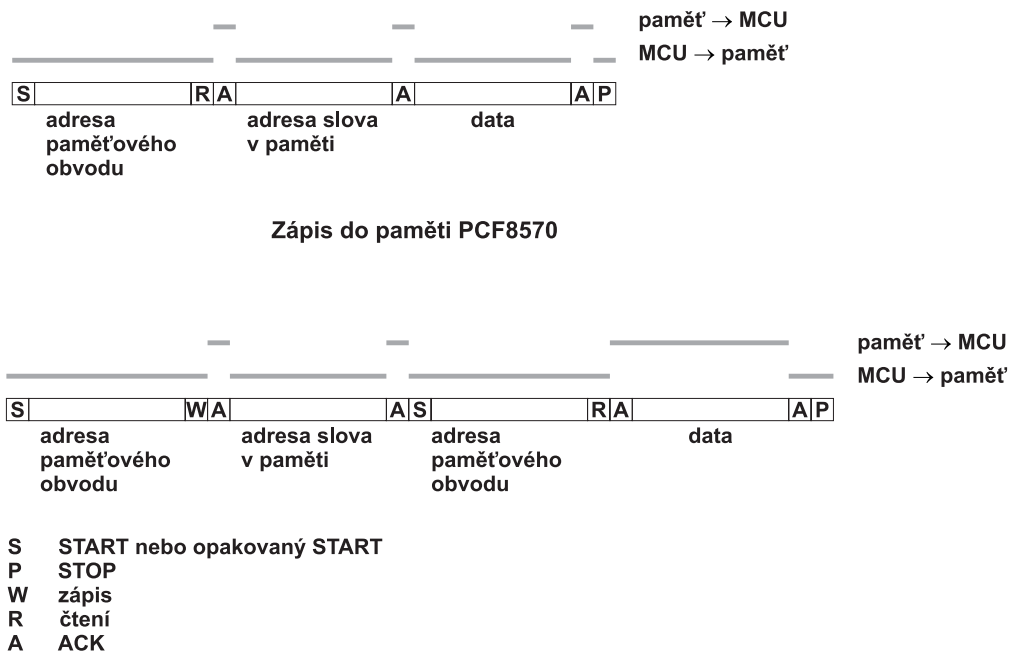
**Potvrzování** Každý vysílaný byte (včetně adresy) je následován vysláním jednoho bitu ACK. Vysílající stanice jej vysílá v úrovni H. Přijímající stanice potvrzuje přijetí tím, že v době vysílání ACK připojí SDA na úroveň L (viz Obr. 14). Pokud vysílající stanice nedostane potvrzení příjmu, ukončí vysílání podmínkou STOP.



**Obr. 14** Potvrzování příjmu bitem ACK.

**Příklad** Na Obr. 15 jsou uvedeny příklady komunikace na I<sup>2</sup>C při čtení a zápisu do paměti Philips PCF8570C. Při čtení s náhodným přístupem se musí do paměti nejprve zapsat adresa dat, která potom mohou být zapsána nebo přečtena. Před vlastním čtením dat z paměti opakuje master (tj. mikrokontrolér) podmínku START a vysílá rámec s povelům READ.





**Obr. 15** Zápis a čtení paměti Philips PCF8570.

PART NUMBER	FUNCTION	I <sup>2</sup> C ADDRESS						
		A6	A5	A4	A3	A2	A1	A0
—	General call address	0	0	0	0	0	0	0
—	Reserved addresses	0	0	0	0	0	X	X
PCD3311/12	Tone generator DTMF/modem/musical	0	1	0	0	1	0	A
PCF8200	Voice synthesizer (male or female)	0	0	1	0	0	0	0
PCF8566	96-segment LCD driver 1:1-1:4 Mux	0	1	1	1	1	1	A
PCF8568	LCD row driver for dot matrix displays	0	1	1	1	1	0	A
PCF8569	Column driver for dot matrix displays	0	1	1	1	1	0	A
PCF8570/71	256 × 8, 128 × 8 static RAM	1	0	1	0	A	A	A
PCF8570C	256 × 8 static RAM	1	0	1	1	A	A	A
PCF8573	Clock/calendar	1	1	0	1	0	A	A
PCF8574	I <sup>2</sup> C bus to 8-bit bus converter	0	1	0	0	A	A	A
PCF8574A	I <sup>2</sup> C bus to 8-bit bus converter	0	1	1	1	A	A	A
PCF8576	160-segment LCD driver 1:1-1:4 Mux	0	1	1	1	0	0	A
PCF8577	64-segment LCD driver 1:1-1:2 Mux	0	1	1	1	0	1	0
PCF8577A	64-segment LCD driver 1:1-1:2 Mux	0	1	1	1	0	1	1
PCF8578	Row/column LCD dot-matrix driver	0	1	1	1	1	0	A
PCF8579	Row/column LCD dot-matrix driver	0	1	1	1	1	0	A
PCF8581	128-byte EEPROM	1	0	1	0	A	A	A
PCF8582	256 × 8 EEPROM	1	0	1	0	A	A	A
PCF8583	256 × 8 RAM with clock/calendar	1	0	1	0	0	0	A
PCF8591	4-channel, 8-bit A/D plus 8-bit D/A	1	0	0	1	A	A	A
PCF8594	512-byte EEPROM	1	0	1	0	A	A	A
SAA1064	4-digit LED driver	0	1	1	1	0	A	A
SAA1136	PCM-Audio indent-word interface	0	0	1	1	1	1	0
SAA1300	5-bit high current driver	0	1	0	0	0	A	A
SAA5243/45	Enhanced teletext circuit	0	0	1	0	0	0	1
SAA7191	S-VHS digital multistandard decoder "square pixel"	1	0	0	0	1	A	1
SAA7192	Digital color space converter	1	1	1	0	0	0	A
SAA7199	Digital encoder	1	0	1	1	0	0	0
SAA9020	Field memory controller	0	0	1	0	1	A	A
SAA9051	Digital multi-standard TV decoder	1	0	0	0	1	0	1
SAA9068	(PIPCO) Picture-in-picture controller	0	0	1	0	0	1	A
SAB3035/36/37	(CITAC) CPU interface for tuning and control	1	1	0	0	0	A	A
SAF1135	Data line decoder	0	0	1	0	0	A	A
TDA4670	Picture signal improvement circuit	1	0	0	0	1	0	0
TDA4680	Video processor	1	0	0	0	1	0	0
TDA8421	Hi-fi stereo audio processor	1	0	0	0	0	0	A
TDA8425	Audio processor w/loudspeaker channel	1	0	0	0	0	0	1
TDA8440	Switch for CTV receivers	1	0	0	1	A	A	A
TDA8442	Interface for color decoders	1	0	0	0	1	0	0
TDA8443	YUV/RGB interface circuit	1	1	0	1	A	A	A
TDA8444	Octuple 6-bit DAC	0	1	0	0	A	A	A
TDA8461	PAL/NTSC color decoder	1	0	0	0	1	0	A
TEA6100	FM/IF and tuning interface	1	1	0	0	0	0	1
TEA6300/6310T	Sound fader control circuit	1	0	0	0	0	0	0
TSA5511/12/14	PLL frequency synthesizer for TV	1	1	0	0	0	A	A
TSA6057	Radio tuning PLL frequency synthesizer	1	1	0	0	0	1	A
UMF1009	Frequency synthesizer	1	1	0	0	0	A	A
—	Reserved addresses	1	1	1	1	X	X	X

X = Don't care.

A = Can be connected high or low by the user.

Tab. 1 I<sup>2</sup>C adresy některých obvodů Philips

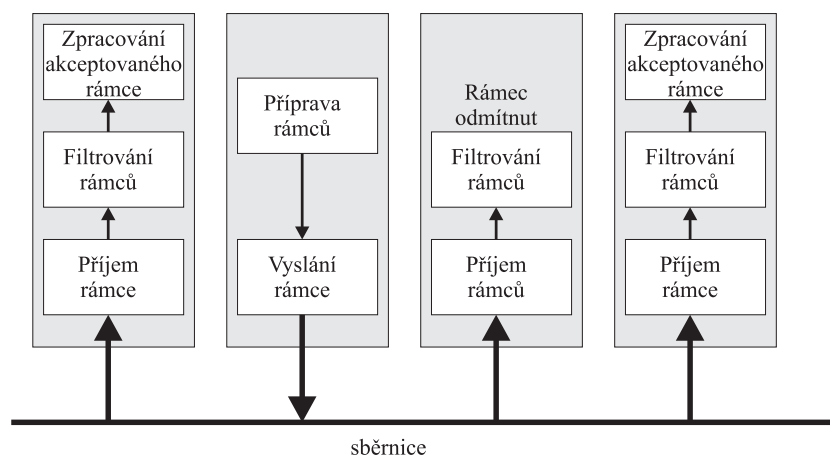
## 4.4 Praktická realizace

Rozhraní I<sup>2</sup>C uvedla firma Philips pro účely propojování různých obvodů ve výrobcích spotřební elektroniky. Jak je zřejmé z **Chyba! Nenalezen zdroj odkazů.**, vyrábí tato firma (ale i mnohé další) celou řadu obvodů s tímto rozhraním. Obvody se většinou připojují k mikrokontrolérům, které jsou vybavené příslušným řadičem. Čistě programová realizace I<sup>2</sup>C přichází v úvahu pouze ve zjednodušené situaci, kdy je zapotřebí k určitému mikrokontroléru připojit jeden nebo více obvodů s tímto rozhraním. Programové řešení *všech* vlastností I<sup>2</sup>C včetně arbitrace, kontroly bitů ACK, synchronizace SCL atd. by vyžadovalo rychlé vzorkování signálů SCL a SDA. K tomu by byl zapotřebí dostatečně výkonný mikrokontrolér, který by byl realizací I<sup>2</sup>C značně zatížen.

## 5 Sběrnice CAN

### 5.1 Koncepce sběrnice CAN

CAN (Controller Area Network) je sběrnice typu multimaster, která je určena k propojování mikropočítačů, různých inteligentních čidel a akčních členů nejen v průmyslu, ale i v automobilech. Její základní koncepce je následující: Jednotlivé stanice připojené na sběrnici vysílají svá data bez ohledu na to, je-li v systému nějaký „zájemce“ nebo nikoliv. Datové rámce proto neobsahují adresu příjemce, nýbrž jen identifikaci která určuje, jaká data rámec obsahuje. Rámec je přijat všemi přijímači na sběrnici. Každý z přijímačů nezávisle použije identifikaci rámce k rozhodnutí, má-li být právě přijatý rámec akceptován, tj. předán k dalšímu zpracování nebo má-li být vymazán. Každý z přijímačů tak akceptuje pouze datové rámce, které jsou pro danou stanici významné (viz Obr. 16).



**Obr. 16** Koncepce sběrnice CAN.

Pro sběrnici CAN existují dvě základní specifikace, zahrnuté v ISO 11898. CAN 2.0A definuje sběrnici s 11bitovým identifikátorem, CAN 2.0B potom definuje sběrnici s rozšířeným 29bitovým identifikátorem rámce. Kromě toho existují starší zpětně kompatibilní specifikace CAN 1.0, CAN 1.1 a CAN 1.2.

### 5.2 Fyzická vrstva

#### 5.2.1 Logické úrovně

Sběrnice CAN je realizována jedinou datovou linkou, na které se rozlišují dvě logické hodnoty: tzv. dominantní úroveň (dále ji budeme označovat jako **d**) a recesivní úroveň (dále **r**). Vztah mezi dominantní a recesivní úrovní je následující: je-li na sběrnici připojeno několik stanic a alespoň jedna z nich vysílá úroveň **d** zatímco ostatní vysílají úroveň **r**, je na sběrnici úroveň **d**. Vysílají-li všechny stanice úroveň **r**, je na sběrnici úroveň **r**.

Sběrnice s výše popsanými úrovněmi by mohla být realizována například vodičem, který je buzen budiči s otevřeným kolektorem a přes rezistor připojen na kladné napájecí napětí. Vy-

soká napěťová úroveň by potom odpovídala úrovni **r** a nízká napěťová úroveň by odpovídala úrovni **d**. V praxi se však pro realizaci sběrnice CAN systém s otevřenými kolektory nepoužívá (viz dále).

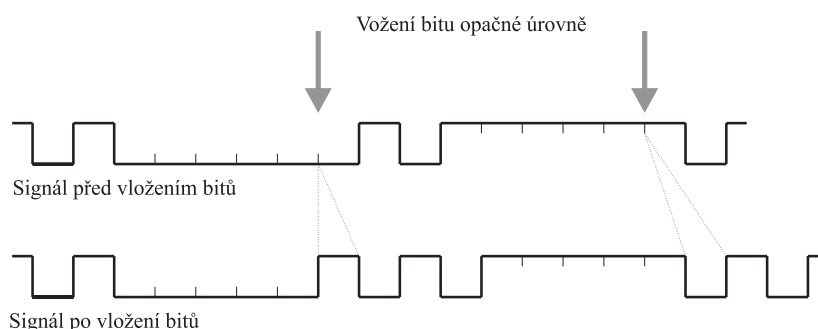
Popsané vlastnosti dominantní a recesivní úrovně jsou potřebné pro arbitrační mechanismus, kterým je řešen přístup na sběrnici CAN a dále tím, že se pro CAN mohou použít spoje s různými elektrickými vlastnostmi (viz odstavec 5.2.3).

## 5.2.2 Kódování dat a synchronizace hodin

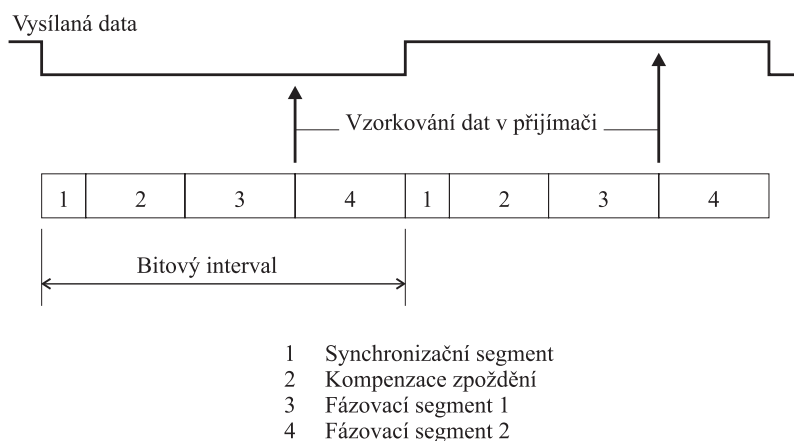
Jednotlivé stanice na sběrnici CAN jsou propojeny pouze datovými vodiči. K přenosu se používá kód NRZ.

Na rozdíl od sběrnic I<sup>2</sup>C, SPI atd. není použit samostatný vodič pro hodinový signál. Každá stanice si generuje interní hodinový signál, který je synchronizován s vysílající stanicí pomocí datového signálu. Protože modulace NRZ sama nezaručuje spolehlivou synchronizaci hodin přijímače, je použita technika vkládaných bitů: Obsahují-li vysílaná data 5 po sobě následujících bitů stejné logické hodnoty, je za ně vložen jeden bit opačné hodnoty. Ten je na straně přijímače z přijatých dat automaticky odstraněn. Tímto způsobem je zaručeno, že v datovém signálu bude alespoň jedna synchronizační hrana po 5 bitových intervalech (Obr. 17).

Každý bitový interval je rozdělen na 4 časové segmenty různé délky (viz Obr. 18). K vysílání dat dochází na začátku 1. segmentu, ke vzorkování na straně přijímače pak na konci 3. segmentu. Je-li přijímačem detekována změna úrovně vstupního datového signálu jinde než na konci 4. segmentu, je hodinový signál příslušným způsobem doladěn.



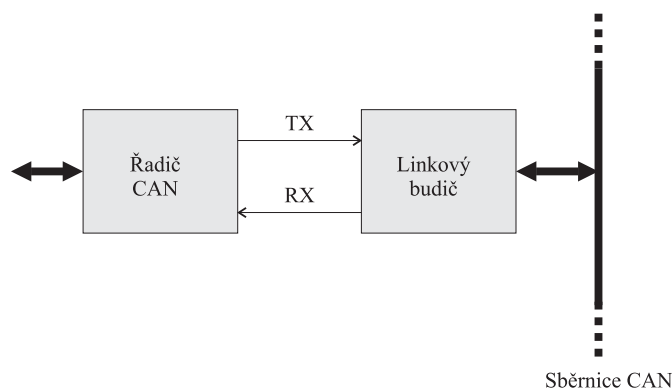
**Obr. 17** Vkládání synchronizačních bitů.



**Obr. 18** Rozdělení bitového intervalu na 4 segmenty.

### 5.2.3 Přenosové médium

Samotná norma CAN neobsahuje specifikaci přenosového média. V praxi se pro definování vlastností média používá několik různých standardů, např. ISO 11898, ISO 11519-1 nebo ISO 11992. Každý z těchto standardů definuje elektrické vlastnosti přenosového média a reprezentaci logických hodnot **r** a **d**. Linkové budiče bývají obvykle řešeny specializovaným obvodem, který je připojen na výstupy řadiče CAN (viz Obr. 19).



**Obr. 19** Řadič a linkový budič sběrnice CAN.

### Linkové budiče podle ISO 11898

Pro ilustraci si uvedeme charakteristiky média a budičů pro standard ISO 11898.

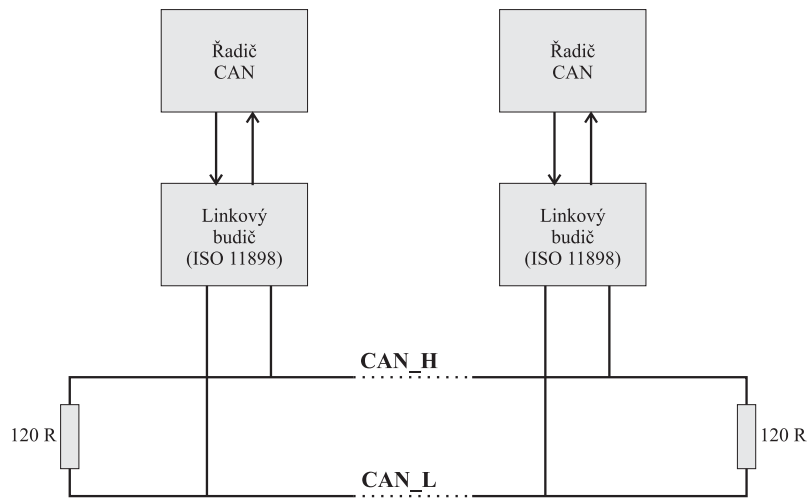
Pro přenos dat podle tohoto standardu se používá symetrické vedení s kabelem o charakteristické impedanci  $120\Omega$ . Oba konce jsou vedení jsou opatřeny zakončovacími odpory. Pro rozlišení logických úrovní **r** a **d** je podstatné rozdílové napětí na vodičích CAN<sub>H</sub> a CAN<sub>L</sub>. Je-li

$$U_{\text{CAN}_H} \leq U_{\text{CAN}_L} + 0.5V \text{ ,}$$

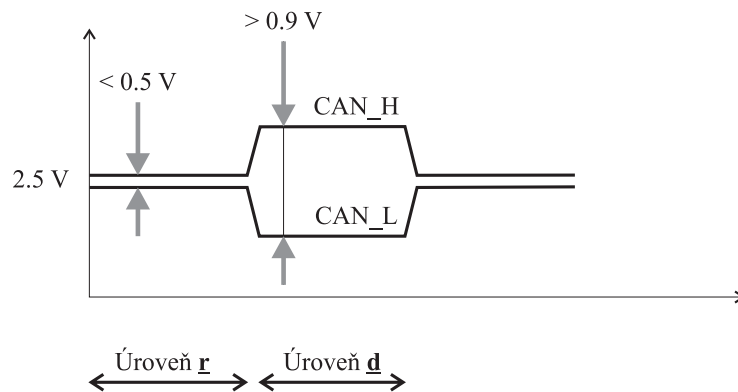
je tento stav interpretován jako logická hodnota **r**. Je-li

$$U_{\text{CAN}_H} \geq U_{\text{CAN}_L} + 0.9V \text{ ,}$$

je tento stav interpretován jako logická hodnota **d**. Klidové napětí na vodičích CAN\_H i CAN\_L je 2.5V.



**Obr. 20** Přenosové médium CAN podle ISO 11898.



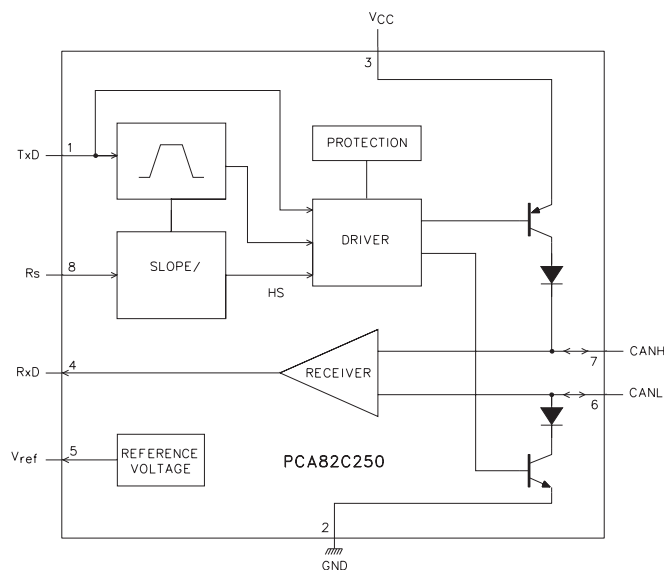
**Obr. 21** Repräsentace dominantní a recesivní úrovně.

Maximální přenosová rychlost sběrnice CAN je závislá mj. i na délce vedení. Pro krátké spoje může dosahovat až 1Mbit/sec.

Délka vedení	Odpor vodičů na 1m délky	Max. přenosová rychlost
< 40m	<70mΩ/m	1Mbit/sec
< 300m	<60 mΩ/m	500kbitů/sec
< 600m	<40 mΩ/m	100kbitů/sec
< 1000m	<26 mΩ/m	50kbitů/sec

**Tab. 2** Maximální rychlost sběrnice CAN v závislosti na délce vedení.

Ideové vnitřní uspořádání budiče CAN podle ISO 11898 je na Obr. 22. Pro realizaci budiče jsou určeny speciální obvody, např. Philips 82C250, SGS-Thomson L9615, Unitrode UC5350 nebo Bosch CF150B.



**Obr. 22** Blokové schéma budiče Philips PCA82C250.

## 5.3 Linková vrstva

### 5.3.1 Formát rámce

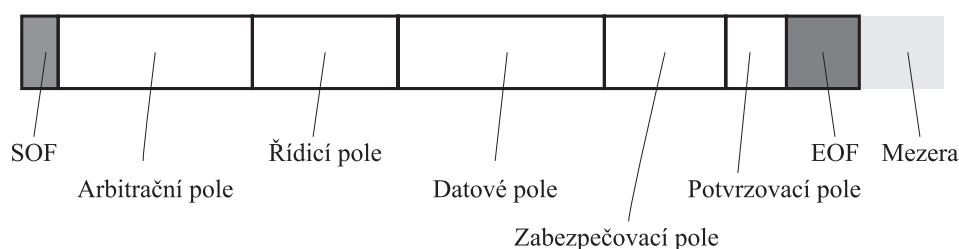
Přenosy na sběrnici CAN jsou prováděny po rámcích. Kromě datového rámce mohou stanice v případě potřeby vyslat také tzv. CAN Remote Frame, který má význam žádosti o data a dále dva typy chybových rámců. Datové rámce mohou být v závislosti na implementované specifikaci CAN (CAN 1 nebo CAN 2) buďto standardní, tj. s 11bitovým identifikátorem nebo rozšířené s 29bitovým identifikátorem. Formát datového rámce je znázorněn na Obr. 23 a Obr. 24. Podrobněji popíšeme formát standardního rámce (tj. rámce s 11bitovým identifikátorem).

Rámec je tvořen 7 poli:

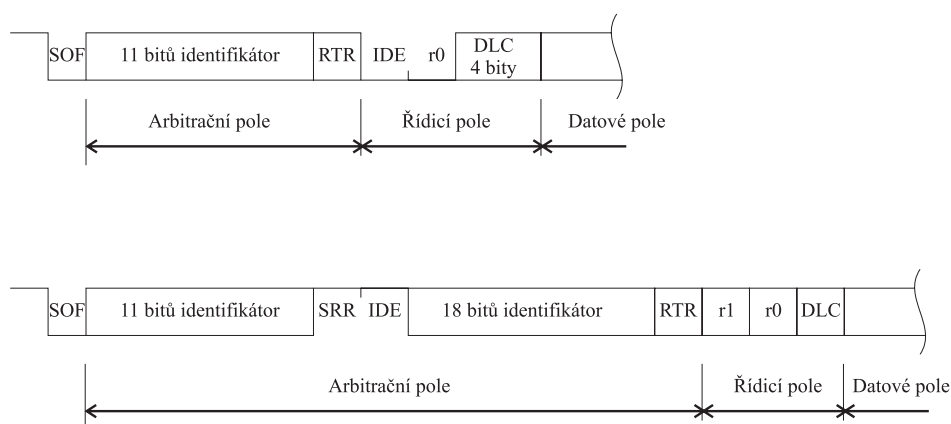
- Na začátku každého rámce je Start bit (SOF), který je vždy v úrovni **d**.
- Arbitrační pole, je tvořené identifikátorem zprávy (11 bitů) a bitem RTR (viz dále).
- Za arbitračním polem následuje řídicí pole, které obsahuje bit IDE, r0 a čtyři bity DLC. Bit IDE slouží k rozlišení mezi standardním (IDE = **d**) a rozšířeným (IDE = **r**) formátem zprávy. Bit r0 je rezervován pro další použití a musí být vysílán jako d. Bity DLC obsahují informaci o délce datového pole. Ta může být v rozsahu 0 (DLC = **dddd**) až 8 (DLC = **rddd**) bytů.
- Datové pole které může mít délku 0 až 8 bytů.

- Zabezpečovací pole je tvořené 15bitovým CRC kódem a jedním zakončovacím bitem. CRC kód je odvozen z polynomu (15,14,10,8,7,4,3). Zabezpečení se počítá ze všech předcházejících polí, včetně SOF.
- Potvrzovací pole je tvořené jedním bitem ACK a jedním zakončovacím bitem. Vysílač vysílá oba bity potvrzovacího pole jako r. Přijímající stanice, která správně přijala celý rámec, indikuje tuto situaci vysláním obou bitů potvrzovacího pole v úrovni d. Tímto způsobem může vysílač zjistit, že alespoň jedna stanice správně přijala vyslanou zprávu.
- Zakončovací pole (EOF) je tvořeno 7 bity, které jsou vysílány v úrovni r.

Jednotlivé rámce jsou od sebe odděleny mezerou (Intermission) v délce nejméně 3 bitových intervalů. V této mezeře je linka v úrovni r.



**Obr. 23** Formát datového rámce CAN.



**Obr. 24** Rozdíl mezi standardním (nahore) a rozšířeným (dole) datovým rámcem.

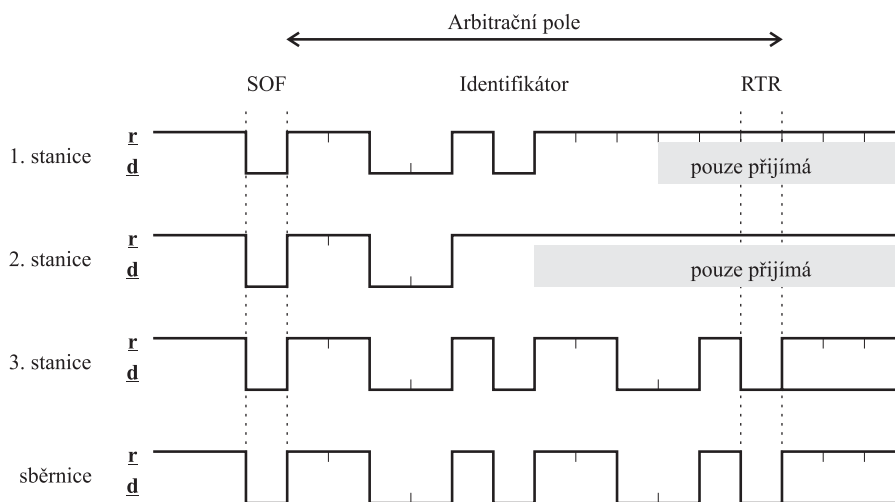
### 5.3.2 Arbitrace

Arbitrační mechanismus používaný sběrnici CAN je označován jako CSMA/CD+AMP (Carrier Sense Multiple Access with Collision Detection and Arbitration on Message Priority). Využívá se přitom vlastnosti fyzické vrstvy, která dovoluje současné vysílání několika stanic, přičemž výsledná logická úroveň na sběrnici je dána pravidly popsány v odstavci 5.2.1.

Jednotlivé stanice mohou zahájit vysílání, je-li sběrnice v klidovém stavu. Při vysílání musí každá stanice neustále sledovat skutečnou logickou úroveň na sběrnici. Pokud daná stanice vysílala úroveň r a na sběrnici je zjištěna úroveň d, je to příznak kolize dvou nebo více stanic.



V takovém případě stanice, které nesoulad úrovní zjistila, musí ukončit vysílání a bezprostředně přejít do přijímacího režimu (viz Obr. 25).



**Obr. 25** Arbitrace na sběrnici CAN.

K arbitraci se na sběrnici CAN používá především identifikační pole jednotlivých rámců, ale v případě jejich shody i bit RTR. Vhodným přiřazením identifikačních čísel jednotlivým údajům (např. datům z různých čidel) lze proto volit prioritu jejich vysílání. Dojde-li ke kolizi datového rámce s požadavkem na tato data, vysílaným jinou stanicí, liší se oba rámce v úrovni bitu RTR. Protože datový rámeček má RTR = d, zatímco rámeček s požadavkem má RTR = r, má datový rámeček vyšší prioritu. V tomto případě vlastně stanice požadující určitá data přijme tato data dříve, než je schopna vyslat požadavek na ně.

### 5.3.3 Rámeček s požadavkem na vyslání dat

Potřebuje-li některá stanice určitá data, která jejich zdroj na sběrnici z nějakého důvodu nevysílá, může o ně tato stanice požádat vysláním zvláštního rámce. V identifikačním poli tohoto rámce je identifikace požadovaných dat (tedy vlastně adresa příjemce) a bit RTR (Remote Transfer Request) je nastaven na úroveň r. Datové pole má délku 0 bytů. Tento rámeček by měl být akceptován stanicí, která je schopná odpovědět vysláním datového rámce s požadovanými daty.

### 5.3.4 Detekce a zpracování chyb

Každá stanice na sběrnici CAN by měla být schopna detekovat následující typy chyb:

1. Chyba vkládaných synchronizačních bitů. Chyba nastane v situaci, kdy po vyslání (příjmu) 5 bitů určité logické úrovně není přijat bit opačné úrovně (viz odstavec 5.2.2). Chyba vkládaných bitů není indikována v potvrzovacím a zakončovacím poli rámce.
2. Bitová chyba. Chyba nastane v situaci, kdy stanice vysílá určitou logickou úroveň a na sběrnici je zjištěna úroveň opačná. Tato chyba není indikována v bitech, které jsou používány k arbitraci a dále v potvrzovacím a zakončovacím poli rámce.
3. Chyba CRC. Chyba nastane v situaci, kdy byl zjištěn rozdíl mezi přijatou hodnotou v poli CRC a hodnotou vypočtenou přijímající stanicí.

4. Chyba potvrzení. Nastane v situaci, kdy vysílající stanice nedostala potvrzení bitem ACK, tj. když vyslaná zpráva nebyla správně přijata ani jednou stanicí.
5. Chyba formátu. Je indikována stanicí, která zjistí nesprávnou logickou úroveň v zakončovacích bitech potvrzovacího a zabezpečovacího pole nebo v zakončovacím poli rámce.

Každá stanice obsahuje dva čítače, REC a TEC, které čítají počet chyb při příjmu resp. při vysílání. Při detekci chyby je v závislosti na jejím typu přičtena k REC resp. k TEC určitá hodnota. Při úspěšném příjmu nebo vyslání rámce je naopak obsah příslušného čítače snížen. Z hlediska reakce na detekované chyby může být stanice v několika stavech:

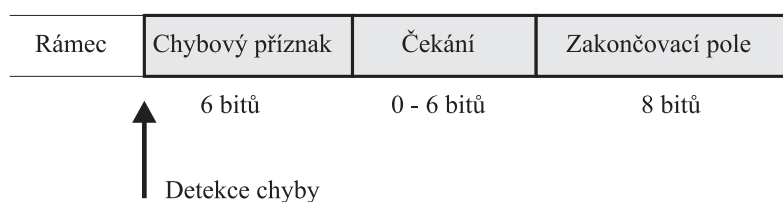
- Je-li obsah obou čítačů  $< 127$ , je stanice v aktivním stavu a při detekci chyby vysílá aktivní chybový rámec.
- Je-li obsah některého z čítačů  $> 96$ , je tato situace indikována nadřazenému procesoru nastavením příslušného bitu ve stavovém registru řadiče CAN.
- Je-li obsah TEC nebo REC  $\geq 127$ , je stanice v pasivním stavu a při detekci chyby vysílá pasivní chybový rámec.
- Je-li obsah TEC  $> 255$ , je stanice ve stavu „odpojena“. Její nové připojení je možné až po restartu.

Při zjištění chyby vysílá stanice podle situace buď aktivní nebo pasivní chybový rámec.

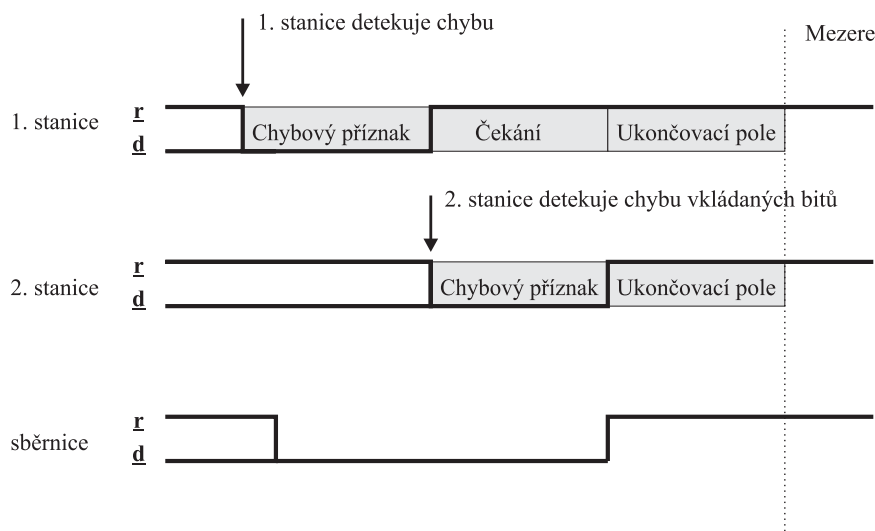
### Aktivní chybový rámec

Formát aktivního chybového rámce je na Obr. 26. Na začátku chybového je pole „chybový příznak“, které sestává ze 6 bitových intervalů v úrovni d. Jeho vysláním je vyvolána „chyba vkládání synchronizačních bitů“ i u ostatních stanic, které proto reagují vysláním chybového rámce (tzv. globalizace chyby). Tím může být trvání úrovně d na sběrnici prodlouženo o dalších max. 6 bitových intervalů (viz Obr. 27).

Po odvyslání 6 bitů d v chybovém rámci vysílá stanice r a čeká, dokud sběrnice skutečně nedosáhne úrovně r. Tohoto stavu by mělo být dosaženo po maximálně 6 bitových intervalech, ve kterých vysílají chybové rámce ostatní stanice. Dosáhne-li sběrnice úrovně r, vysílá stanice zakončovací pole chybového rámce, které sestává z 8 bitů v úrovni r.



**Obr. 26** Formát chybového rámce. Aktivní rámec vysílá chybový příznak v úrovni d, pasivní rámec vysílá chybový příznak v úrovni r.



**Obr. 27** Globalizace chyby na sběrnici CAN.

### Pasivní chybový rámec

Pasivní rámec vysílá stanice v situaci, kdy její čítač chyb dosáhne hodnoty  $\geq 127$ . Formát pasivního chybového rámce je na Obr. 26. Rozdíl mezi pasivním a aktivním chybovým rámcem spívá v tom, že pasivní chybový rámec má v poli „chybový příznak“ úroveň  $\underline{r}$ , která sama nemění logické úrovně na sběrnici. Tím nedojde k přerušení vysílání ostatních stanic.

### Rámec přetížení

Rámec přetížení může vyslat stanice v situaci, kdy není schopna z nějakého důvodu (např. přetížení) zahájit příjem rámců. Rámec přetížení má stejný formát jako aktivní chybový rámec, ale může být vyslán pouze poté, co na sběrnici bylo ukončeno vysílání zakončovacího pole datového nebo chybového rámce. Rámec přetížení tak nepřerušuje vysílání žádného datového rámce. Čítače chyb se nemění.