

Návrh paměti

požadované velikosti a vlastností

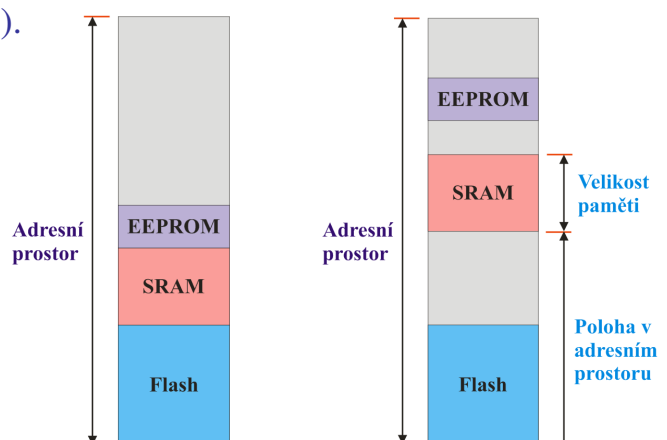
K.D. - přednášky POT

1

Návrh paměti

- Požadované vlastnosti:
 - Typ pamětí (SRAM, Flash, ...).
 - Velikost pamětí (počet slov \times délka slova).
 - Umístění pamětí v adresním prostoru.

Požadované vlastnosti jsou specifikovány při návrhu systému.



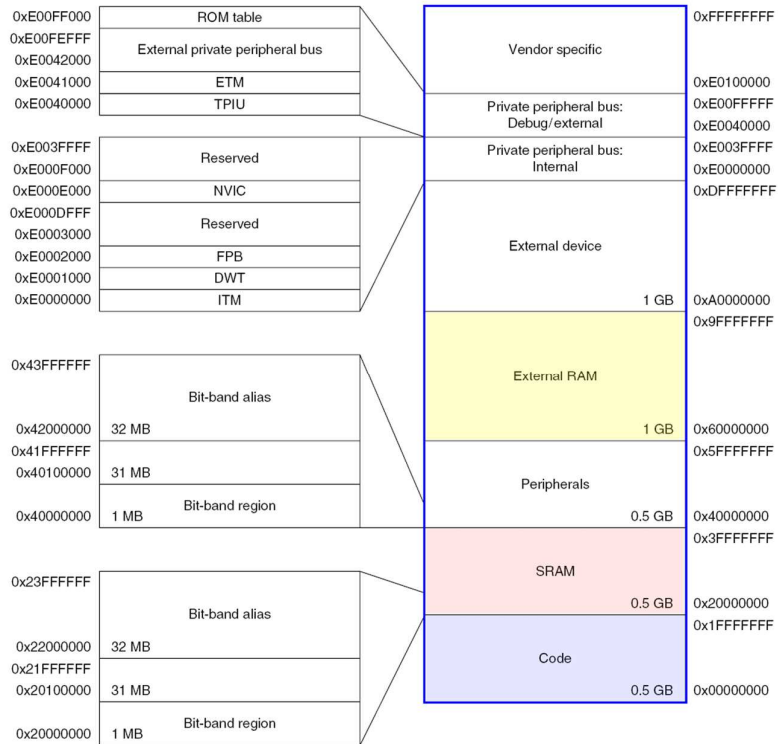
Různé možnosti umístění pamětí v adresním prostoru

K.D. - přednášky POT

2

Návrh paměti

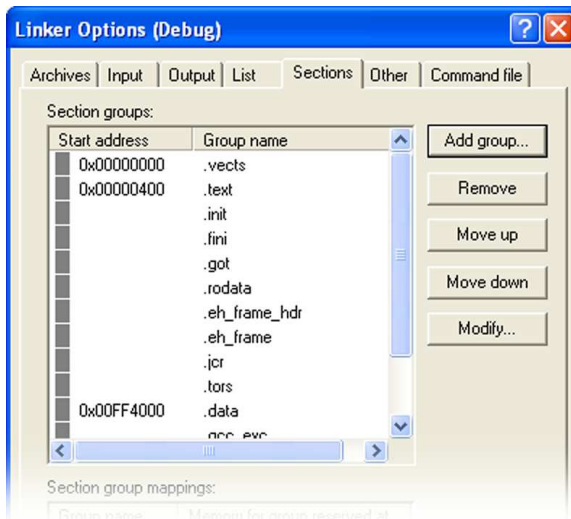
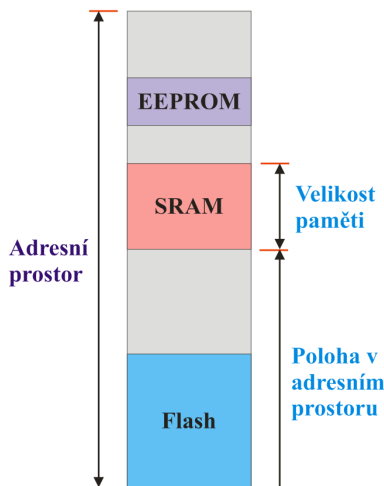
Mapa adresního prostoru Cortex M3



K.D. - přednášky POT

Mapa adresního prostoru

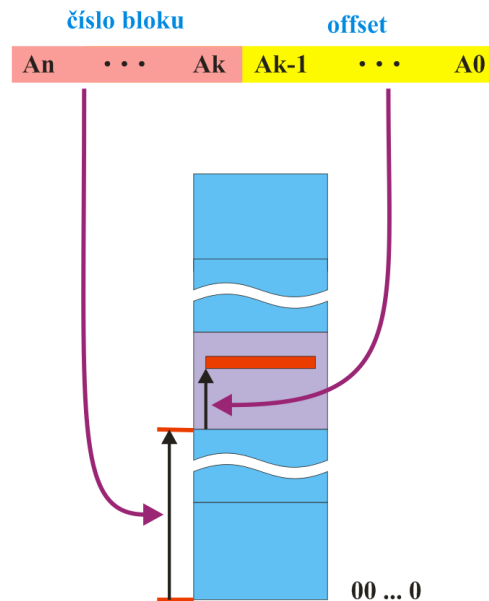
- Rozložení paměť v adresním prostoru = mapa adresního prostoru.
- Znalost mapy adresního prostoru je nutná při vytváření programového vybavení.



K.D. - přednášky POT

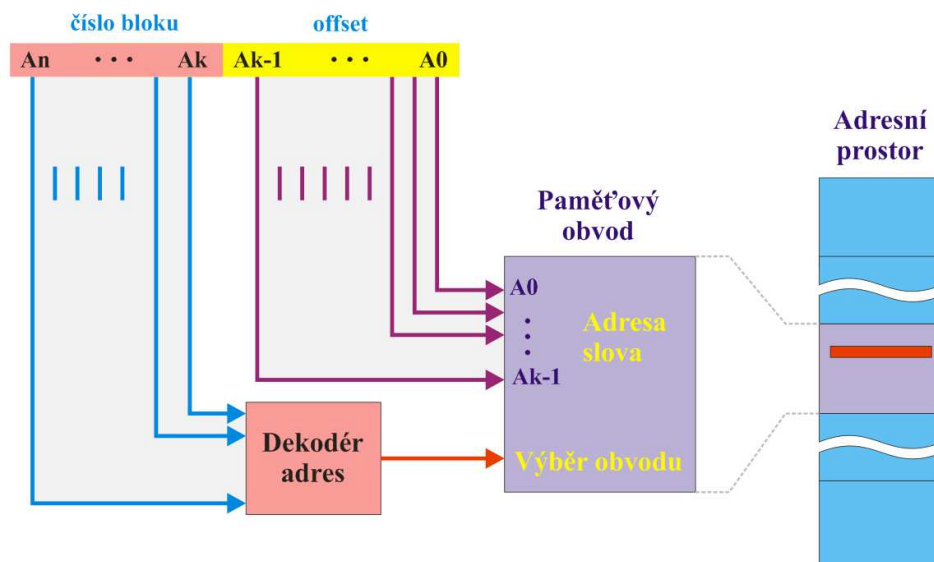
Interpretace adresy

- Ve kterémkoliv místě lze adresu rozdělit na číslo bloku a offset uvnitř bloku.
- Velikost bloku určuje délku příslušné části adresy - offsetu (na obrázku 2^k slov).
- Poloha bloku v adresním prostoru je dána vyšší částí adresy.
- Počet bloků je dán délkou vyšší části adresy (na obrázku 2^{n-k+1} bloků).



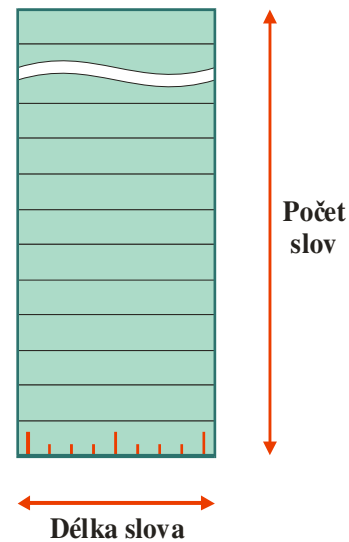
Interpretace adresy – fyzická paměť

- Velikost bloku (= paměťového obvodu) určuje délku příslušné části adresy – offsetu.
- Poloha bloku v adresním prostoru je dána vyšší částí adresy.



Kapacita paměti

- Kapacita paměti = počet slov \times délka slova.
 - Udává se někdy v počtu bitů (např. 64 Mb).
- Častěji s ohledem na organizaci jako součin počet slov \times délka slova, např. $256k \times 8$ (256 k slov po 8 bitech).



Základní počty

- **1 byte = 8 bitů.**
- **1 kbit (kb) = 2^{10} bitů = 1024 bitů.**
- **1 kByte (kB) = 2^{10} bytů = 1024 bytů.**
- **1 MByte (MB) = 2^{20} bytů = 2^{10} kB = 1 048 576 bytů.**
- **1 GByte (GB) = 2^{30} bytů = 2^{20} kB = 2^{10} MB = 1 073 741 824 bytů.**
 - Má-li paměť kapacitu 1 kB, tj. 2^{10} bytů, je k jejímu adresování zapotřebí 10 adresních bitů (předpokládáme-li bytovou organizaci paměti).
 - Má-li paměť kapacitu 2 kB, tj. 2^{11} bytů, je k jejímu adresování zapotřebí 11 adresních bitů.
 - atd.

Základní počty

$$2^m \times 2^n = 2^{m+n}$$

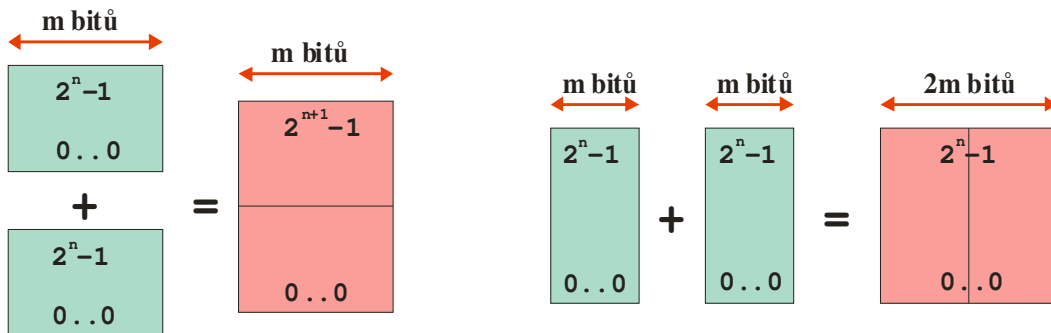
$$128 \text{ k} = 128 \times 2^{10} = 2^7 \times 2^{10} = 2^{17}$$

$$2^{24} = 2^4 \times 2^{20} = 16 \times 2^{20} = 16 \text{ M}$$

2^0	=	1
2^1	=	2
2^2	=	4
2^3	=	8
2^4	=	16
2^5	=	32
2^6	=	64
2^7	=	128
2^8	=	256
2^9	=	512
2^{10}	=	1024

Základní úlohy zvětšení kapacity paměti

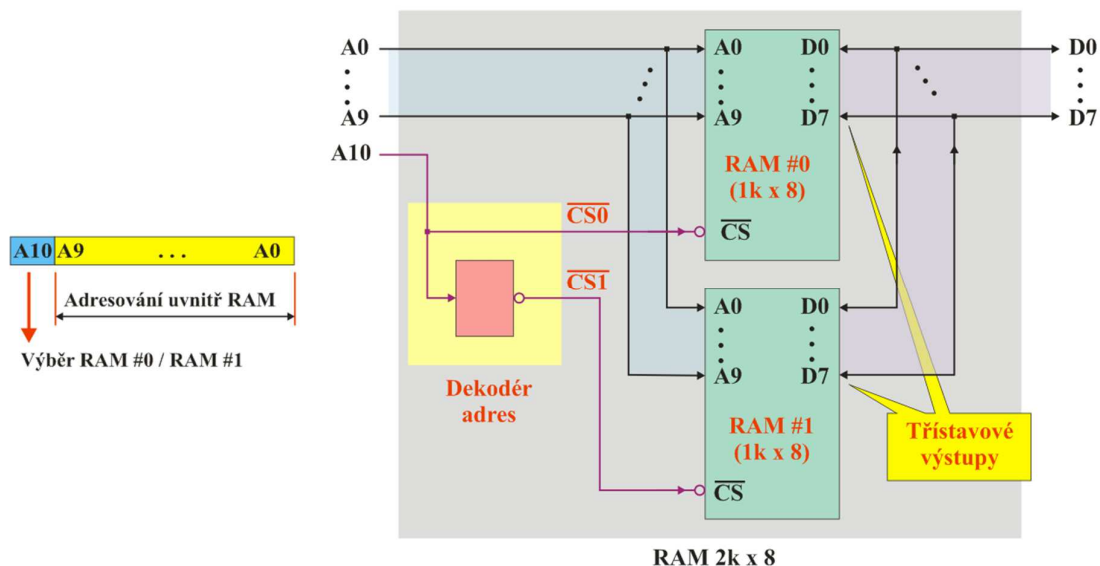
- Zvětšení počtu slov
- Zvětšení délky slova



Zvětšení počtu slov

Příklad: Sestavení paměti $2k \times 8$ ze dvou obvodů $1k \times 8$.

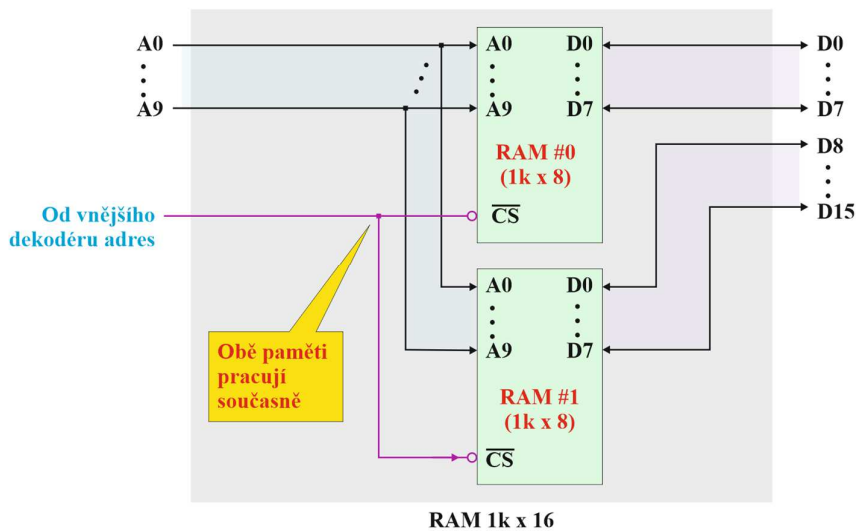
- $1k = 2^{10} \Rightarrow$ adresování 10 adresními vodiči.
- $2k = 2^{11} \Rightarrow$ adresování 11 adresními vodiči.



Zvětšení délky slova

Příklad: Sestavení paměti $1k \times 16$ ze dvou obvodů $1k \times 8$.

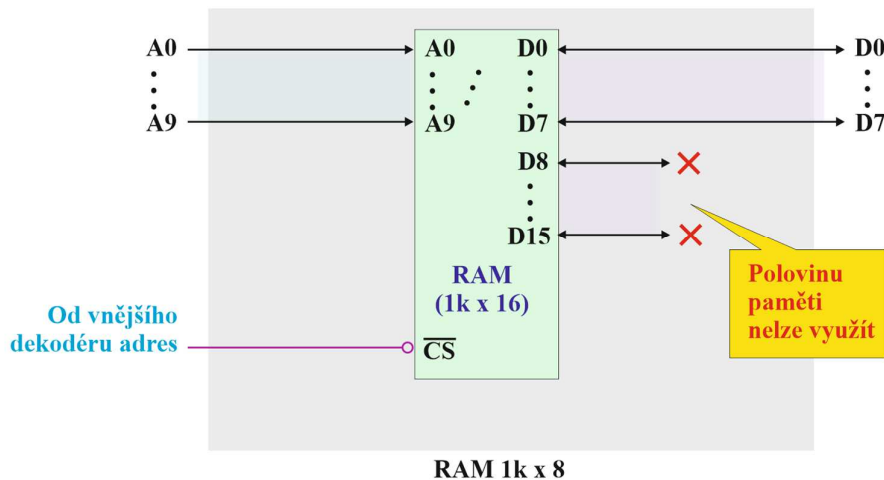
- $1k = 2^{10} \Rightarrow$ adresování 10 adresními vodiči.



Zmenšení délky slova

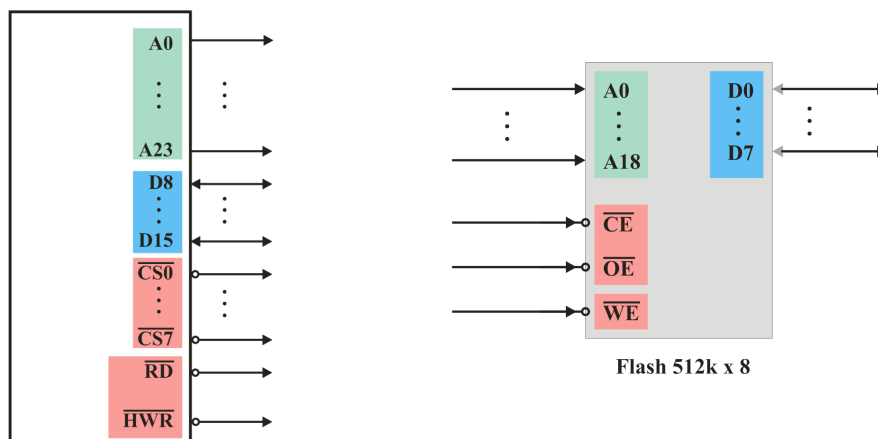
Příklad: Konstrukce paměti $1k \times 8$ z obvodu $1k \times 16$.

- Využije se pouze část paměti.
- \Rightarrow Prakticky má taková konstrukce pochybný význam (!).



Připojení Flash 512 k \times 8 k H8S - zadání

- Úkol: připojit Flash 512 kB k H8S tak, aby byla umístěna v adresním prostoru od adresy 000 000.
 - Paměť má kapacitu 512 kB ($= 2^{19}$ kB) \Rightarrow A18 – A0 adresují uvnitř paměti.
 - A23 – A19 musíme zavést do dekodéru adres.



Připojení Flash 512 k × 8 k H8S - analýza

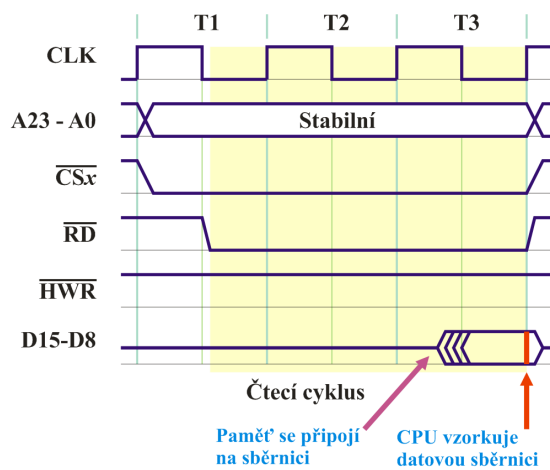
- Paměť má kapacitu 512 kB ⇒ A18 – A0 adresují uvnitř paměti.
- A23 – A19 musíme zavést do dekodéru adres.



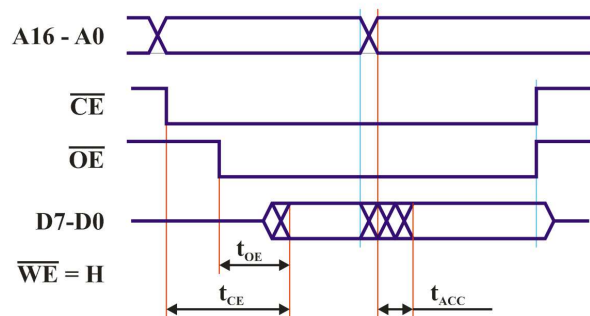
FFF	FFF
F80	000
17F	FFF
100	000
0FF	FFF
080	000
07F	FFF
000	000

Umístění Flash 512 k v adresním prostoru

Signály na sběrnici H8S a na vývodech Flash



Sběrnice H8S



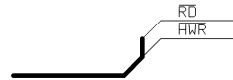
Flash

Připojení Flash 512 k × 8 k H8S – krok 1

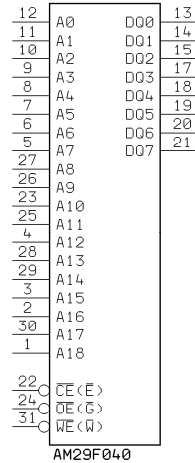
- Výchozí situace - schéma

Datová sběrnice [D15 - D8]

Adresní sběrnice [A23 - A0, /CS0 - /CS7]



Řídicí sběrnice [/RD, /HWR, /WAIT]



Připojení Flash 512 k × 8 k H8S – krok 2

Datová sběrnice [D15 - D8]

Adresní sběrnice [A23 - A0, /CS0 - /CS7]

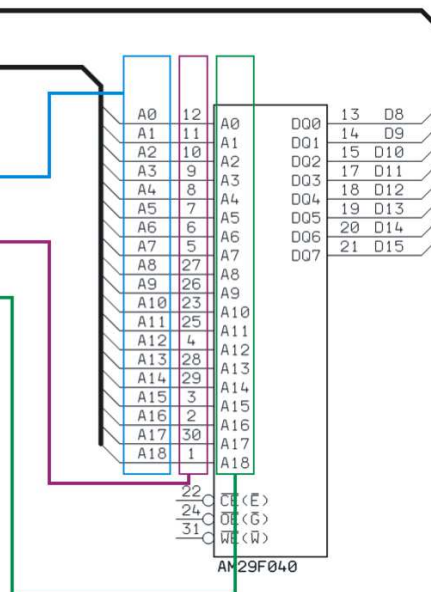


Řídicí sběrnice [/RD, /HWR, /WAIT]

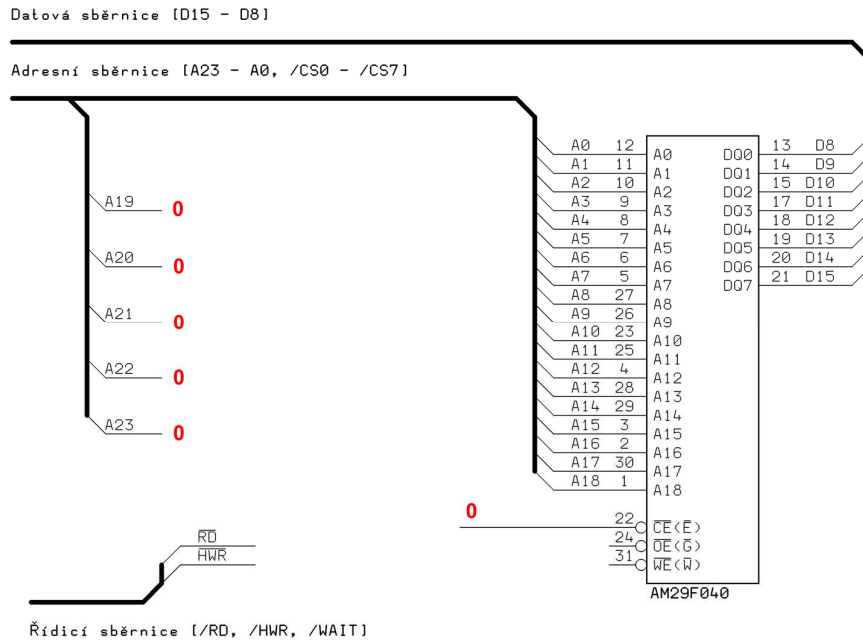
Signál na sběrnici

Číslo pinu paměti

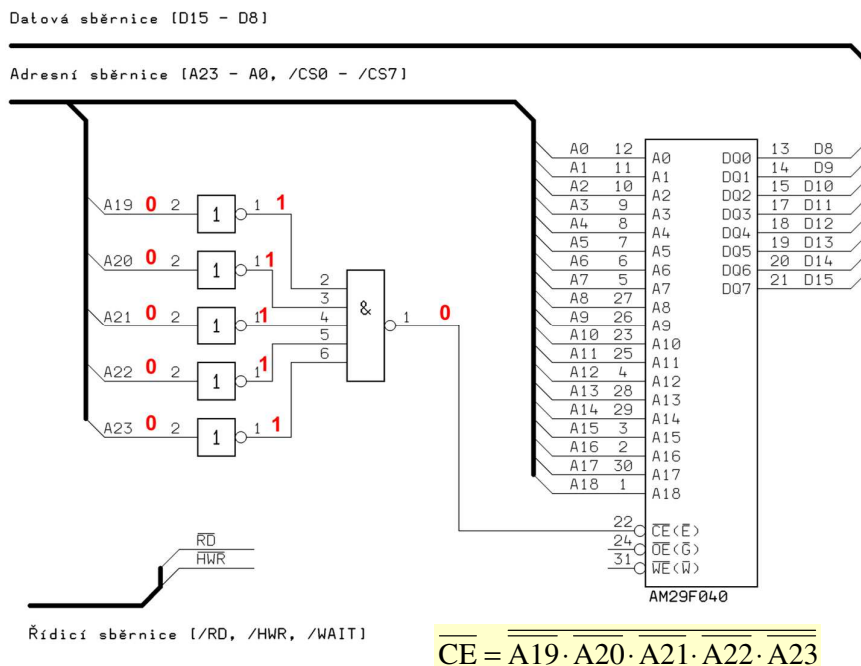
Označení pinu (vývodu) paměti



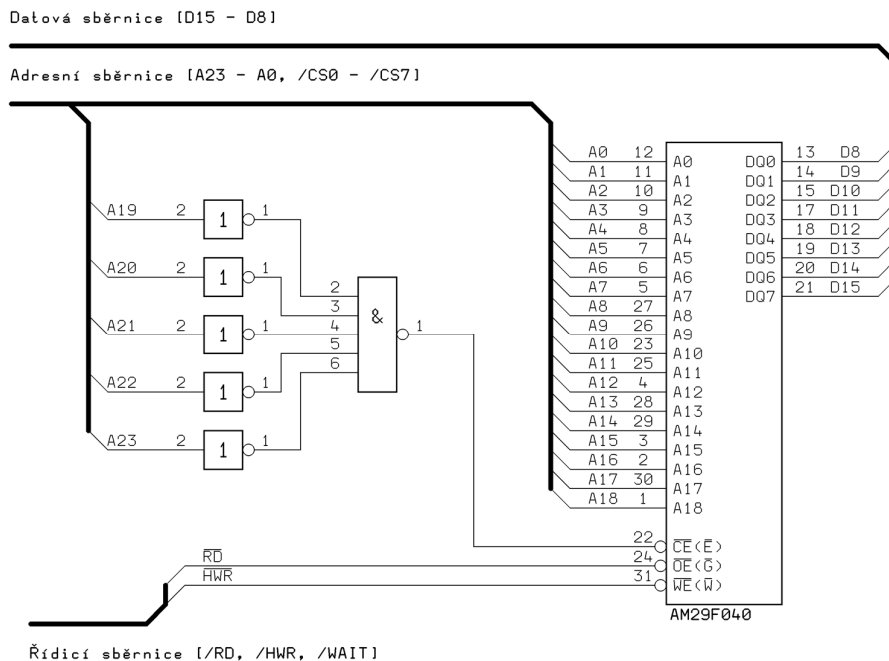
Připojení Flash 512 k × 8 k H8S – krok 3



Připojení Flash 512 k × 8 k H8S – krok 4



Připojení Flash 512 k × 8 k H8S – krok 5



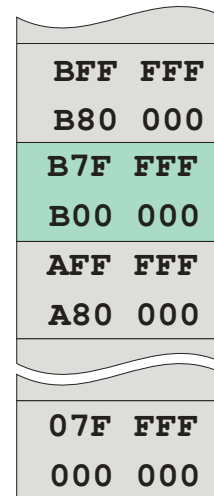
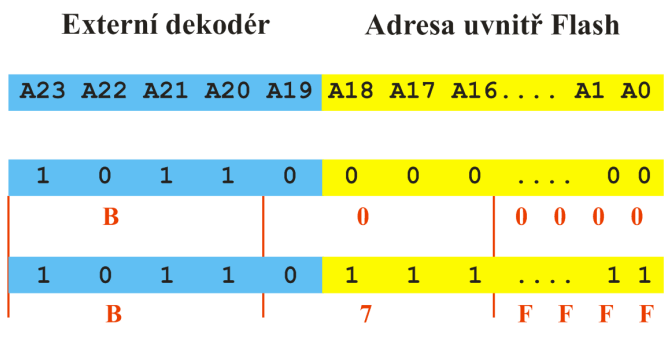
Připojení Flash 512 k × 8 k H8S – jiné zadání

- Úkol: připojit Flash 512 kB k H8S tak, aby byla umístěna v adresním prostoru na adrese **B00 000 – B7F FFF**.

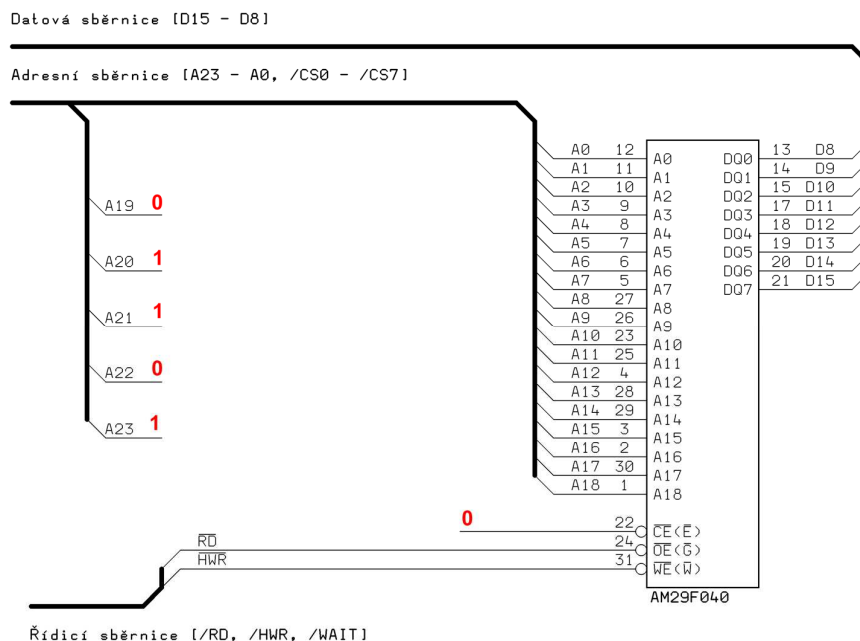
BFF FFF
B80 000
B7F FFF
B00 000
AFF FFF
A80 000
07F FFF
000 000

Připojení Flash 512 k × 8 k H8S - analýza

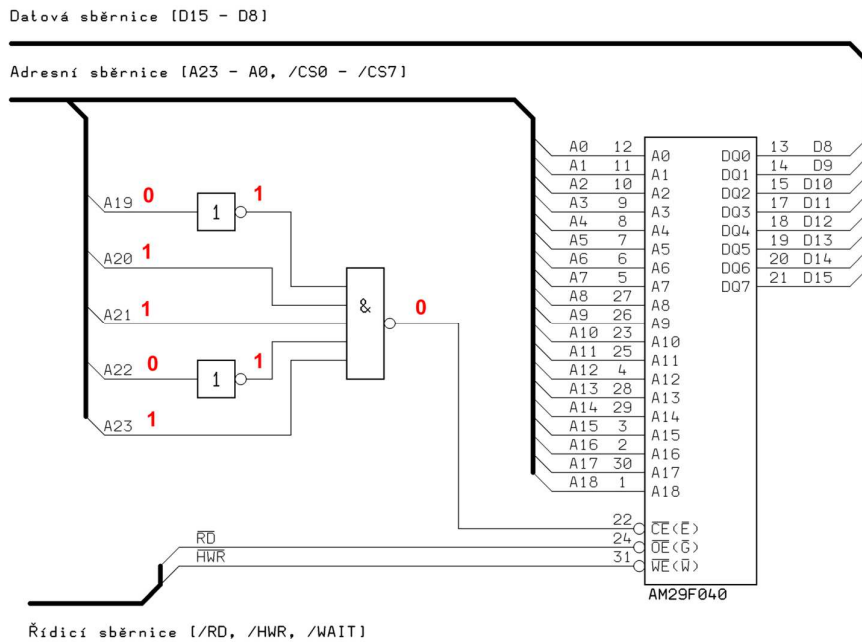
- Paměť má kapacitu 512 kB ⇒ A18 – A0 adresují uvnitř paměti.
- A23 – A19 musíme zavést do dekodéru adres – musí na nich být hodnota **10110**.



Připojení Flash 512 k na adrese B00 000 – B7F FFF – krok 1

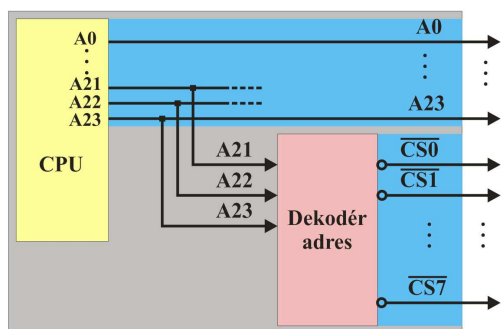


Připojení Flash 512 k na adrese B00 000 – B7F FFF – krok 2



Interní dekodér adres v H8S

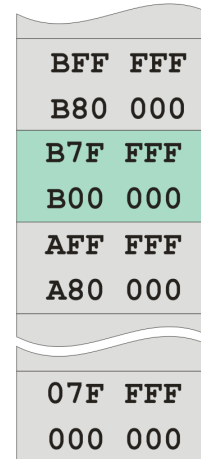
- Podle kombinace na A23 – A21 aktivuje některý z /CS0 - /CS7.



A23	A22	A21	/CS0	/CS1	/CS2	...	/CS7
0	0	0	L	H	H	...	H
0	0	1	H	L	H	...	H
0	1	0	H	H	L	...	H
...		
1	1	1	H	H	H	...	L

Použití interního dekodéru adres v H8S

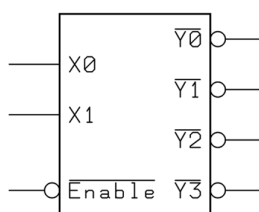
- Paměť má kapacitu 512 kB ⇒ A18 – A0 adresují uvnitř paměti.
- A23 – A21 jsou dekodovány přímo v H8S. Musí na nich být kombinace **101** ⇒ bude aktivní /CS5.
- A20 – A19 musí mít hodnotu **10** – dekodujeme je v **externím dekodéru**.



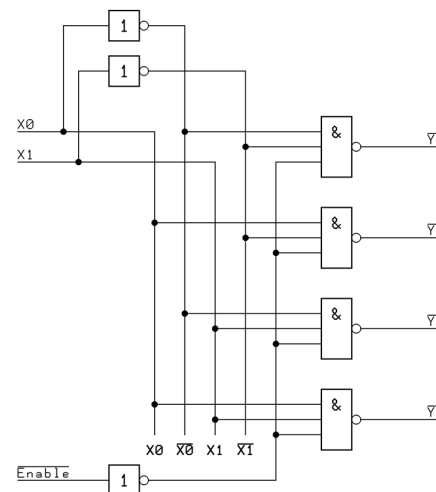
Dekodováno v H8S			Externí dekodér		Adresa uvnitř Flash						
A23	A22	A21	A20	A19	A18	A17	A16	...	A1	A0	
1	0	1	1	0	0	0	0	...	0	0	
B					0			0 0 0 0			
1	0	1	1	0	1	1	1	...	1	1	
B					7			F F F F			

Externí dekodér

- Je-li /Enable = H, jsou všechny výstupy H (tj. neaktivní).
- Je-li /Enable = L, je aktivní některý z výstupů /Y0, ... , /Y3 podle kombinace na X1 X0.

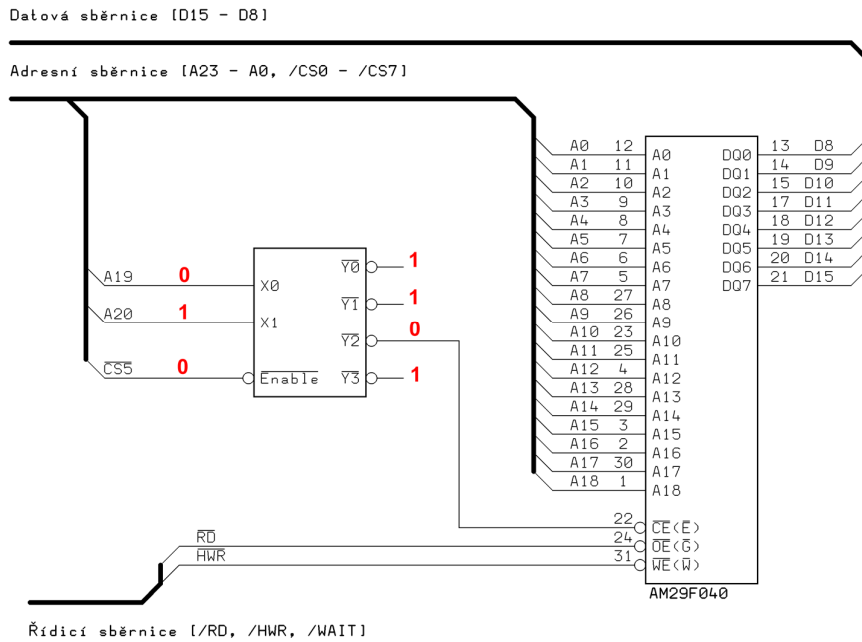


X1	X0	/Enable	/Y0	/Y1	/Y2	/Y3
x	x	H	H	H	H	H
L	L	L	L	H	H	H
L	H	L	H	L	H	H
H	L	L	H	H	L	H
H	H	L	H	H	H	L



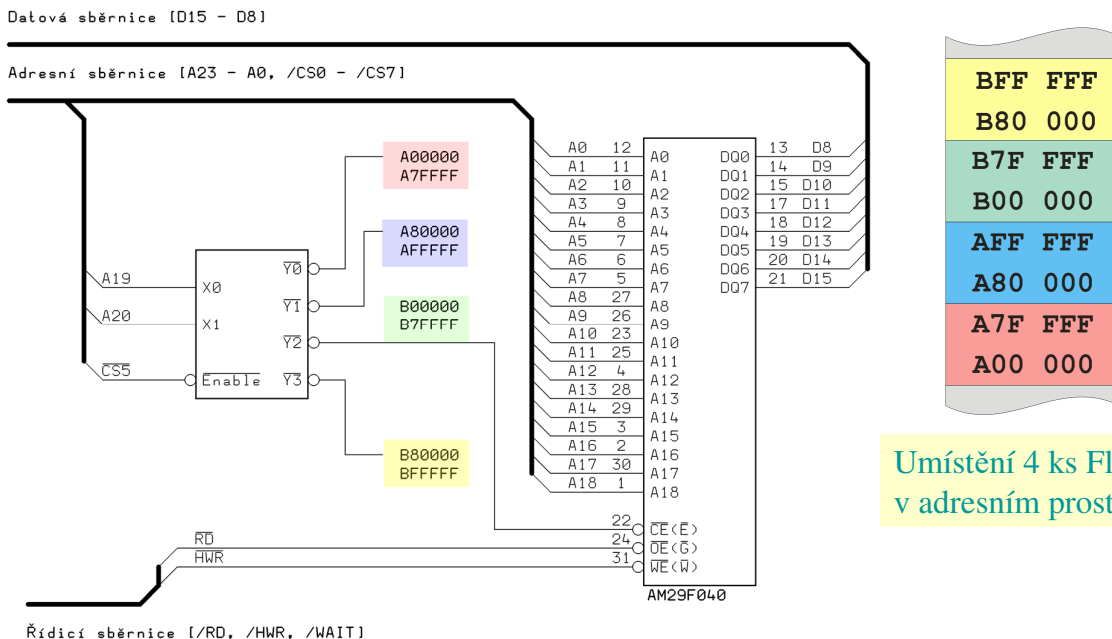
Např. SN74HC139

Zapojení s externím dekodérem



Další využití externího dekodéru

- Výstupy /Y0, /Y1, /Y2 a /Y3 z dekodéru adres lze použít pro připojení 4 ks Flash 512 kB ⇒ celkem 2 MB Flash.



Návrh paměti – SRAM 256 kB - zadání

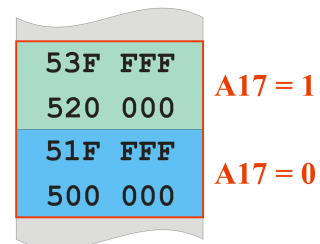
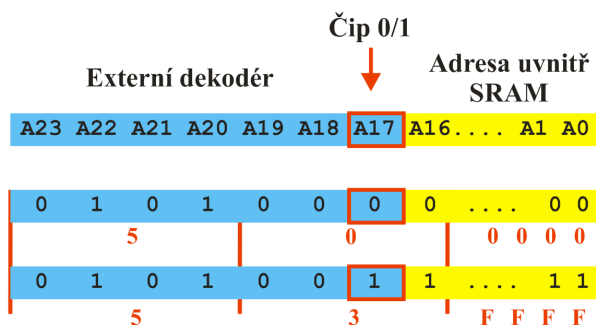
• Úkol:

- Sestavit SRAM 256 kB z obvodů 128 kB (128 k × 8).
- Paměť má být v adresním prostoru umístěna od adresy 500 000.

Návrh paměti – SRAM 256 kB

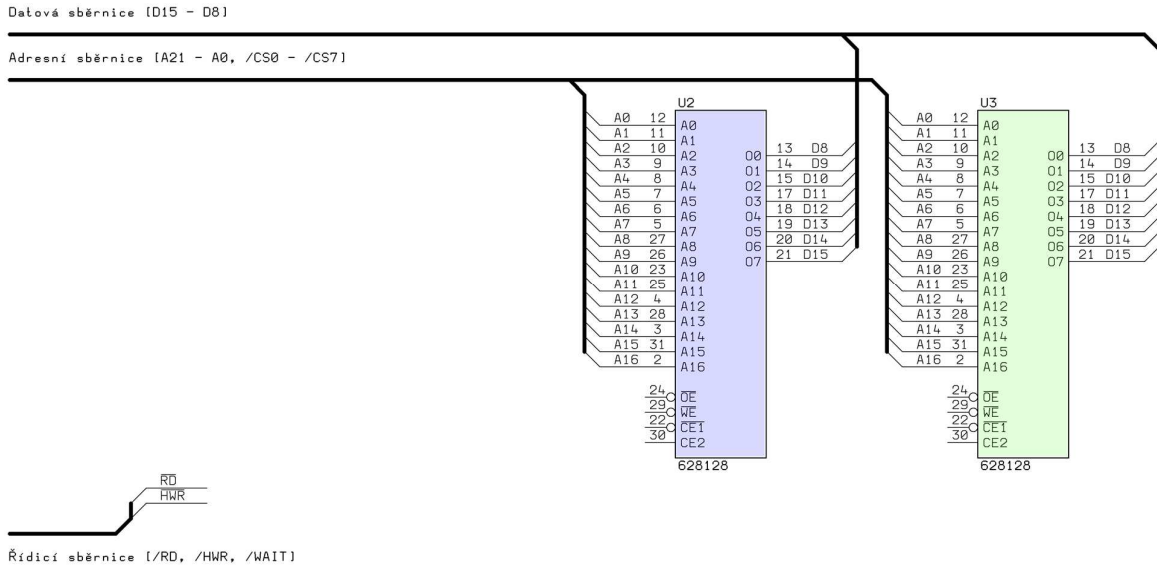
• Další úvahy:

- K dosažení potřebné kapacity jsou zapotřebí dva paměťové obvody.
- Paměťový obvod o kapacitě 128 kB má 17 adresních vodičů (A16 – A0).
- Vodič A17 vybírá jeden ze dvou použitých obvodů.
- Vodiče A23 – A18 zavedeme do dekodéru adres – musí na nich být kombinace **010100**.

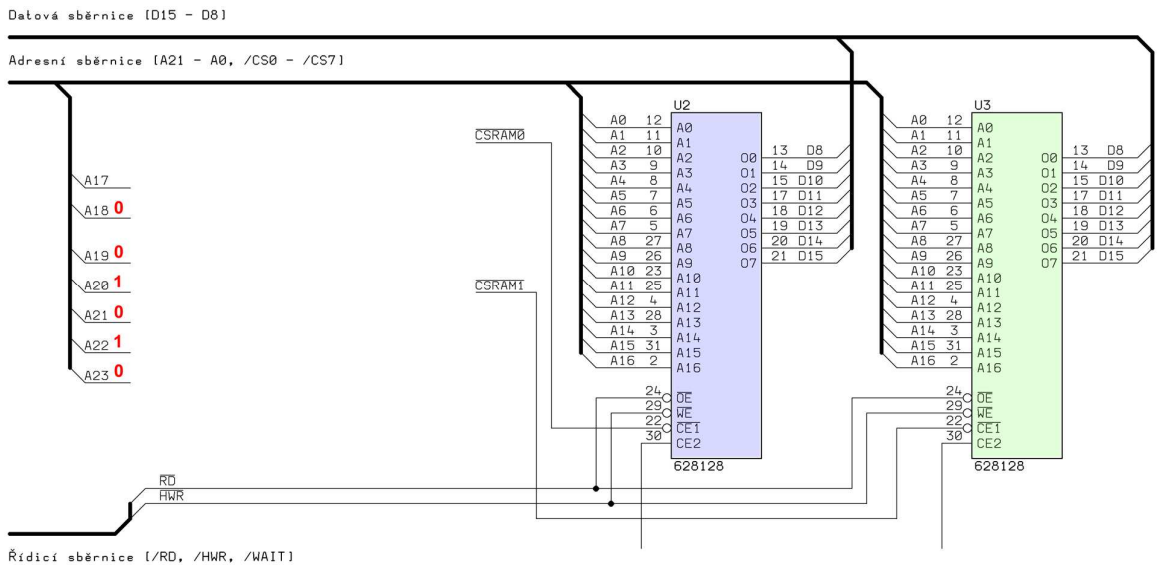


Umístění 2 ks paměťových obvodů v adresním prostoru

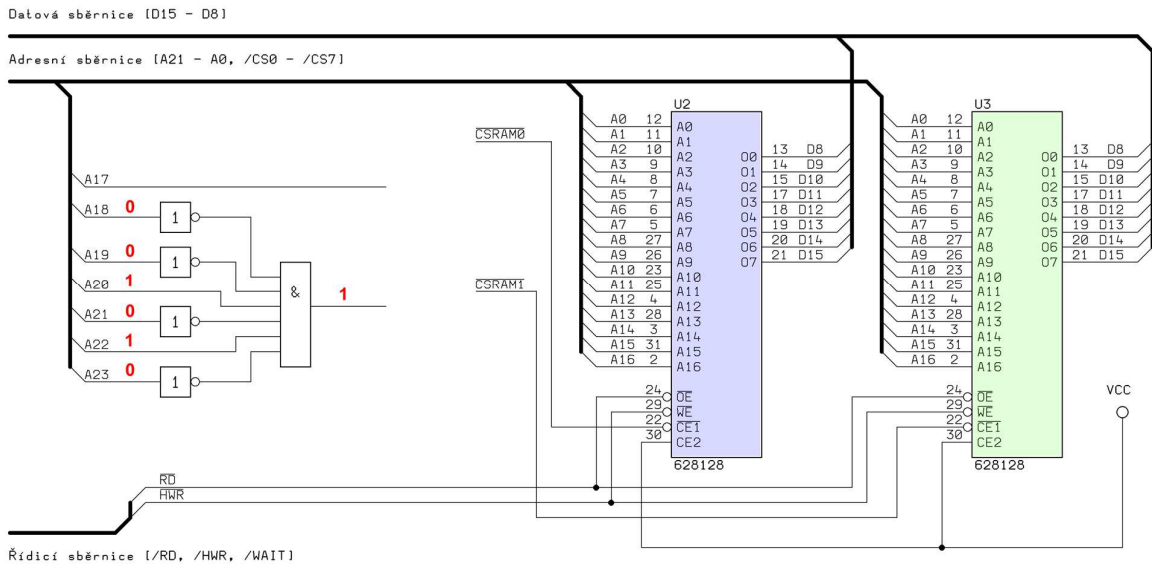
Návrh paměti – SRAM 256 kB – 1. krok



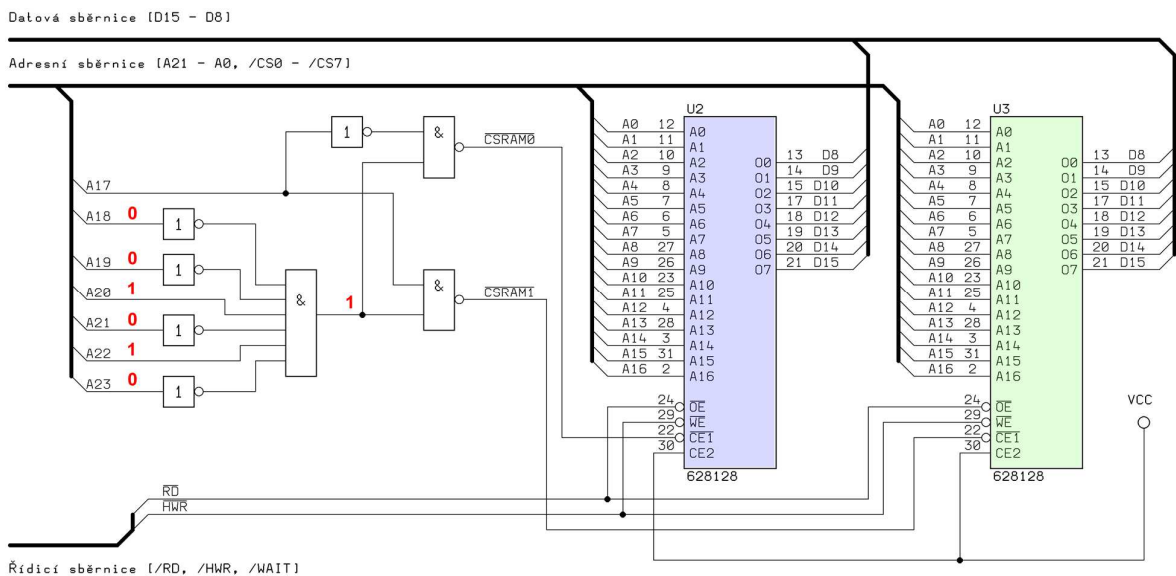
Návrh paměti – SRAM 256 kB – 2. krok



Návrh paměti – SRAM 256 kB – 3. krok



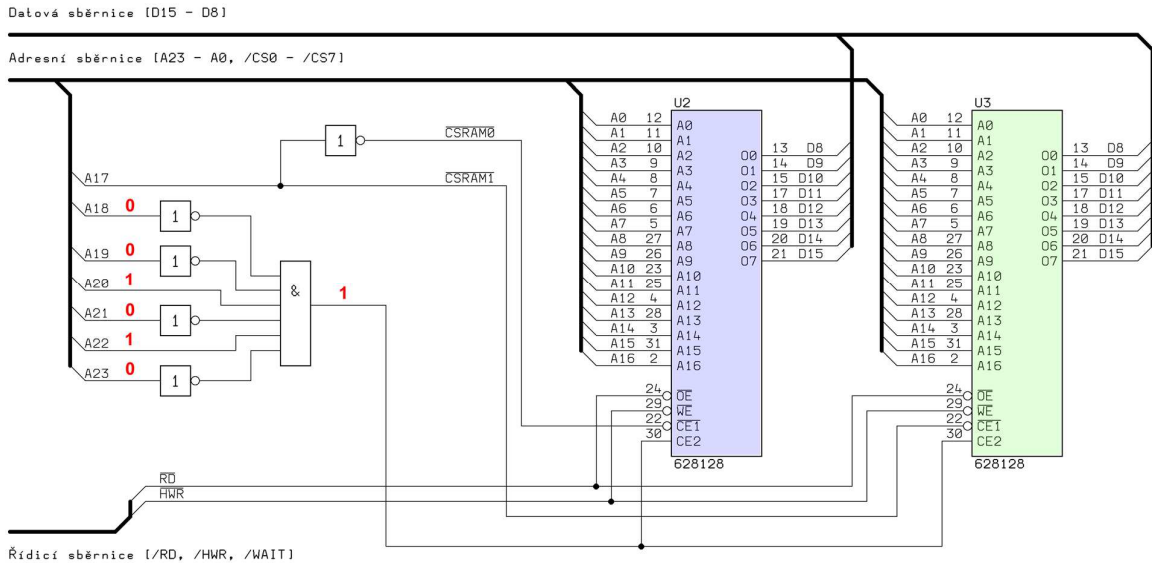
Návrh paměti – SRAM 256 kB – 4. krok



$$CSRAM0 = \overline{(A23 \cdot A22 \cdot A21 \cdot A20 \cdot A19 \cdot A18)} \cdot A17$$

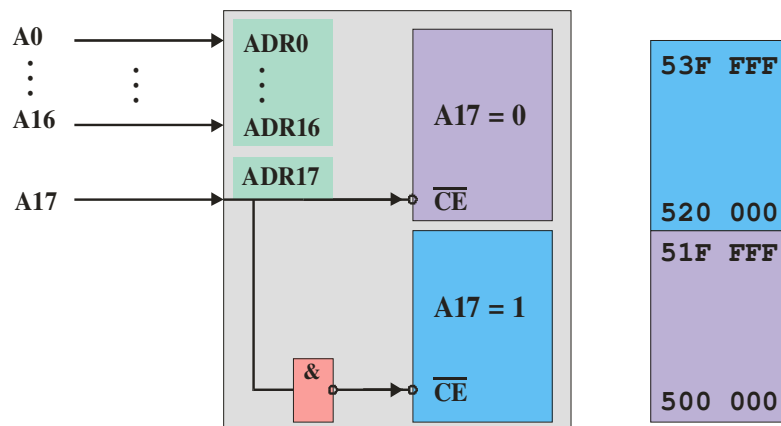
$$CSRAM1 = \overline{(A23 \cdot A22 \cdot A21 \cdot A20 \cdot A19 \cdot A18)} \cdot A17$$

Návrh paměti – SRAM 256 kB – jiné řešení



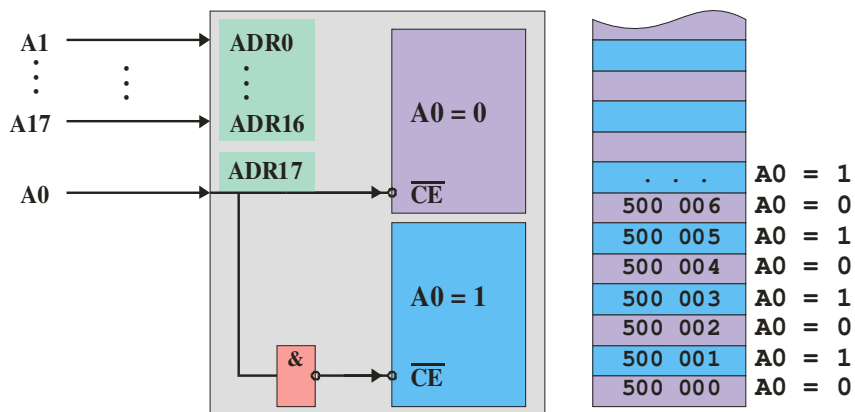
Mapa adresního prostoru paměti 256 kB

- Paměť 256 kB lze považovat za ucelený blok.
- Vodič A17 vybírá jeden ze dvou paměťových obvodů ⇒ Každý paměťový obvod pokrývá *souvisle* část bloku o velikosti 128 kB.



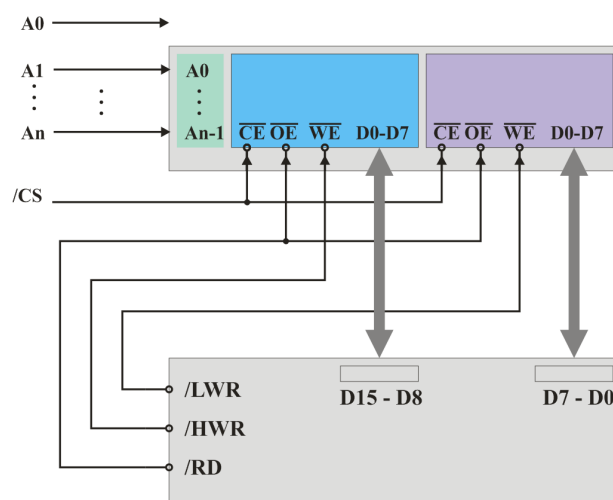
Jiná varianta zapojení SRAM

- Paměť 256 kB lze považovat za ucelený blok.
- Vodič A0 z adresní sběrnice je připojen na vstup ADR17 paměťového bloku \Rightarrow A0 vybírá jeden z paměťových obvodů.
- Paměťový prostor pokrytý jednotlivými obvody není souvislý.



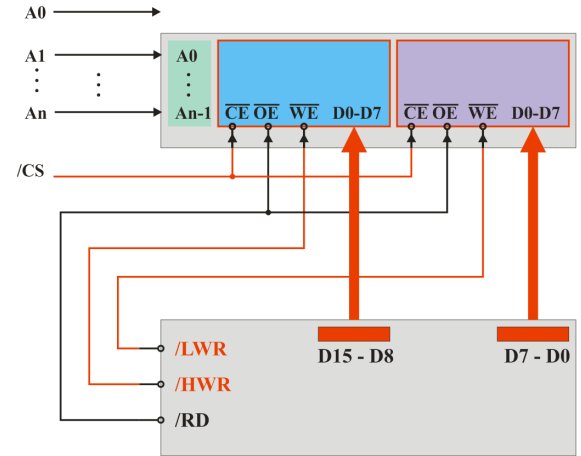
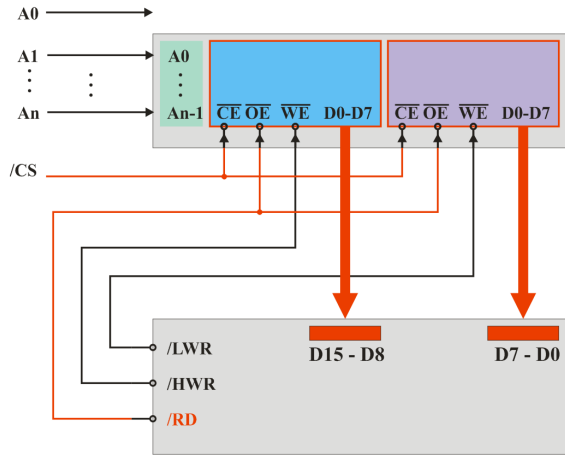
Práce H8S s 16bitovou pamětí (1)

- Vodič A0 je využit pouze interně v H8S, do paměti se nezapojuje.
- 16bitové přístupy vyžadují „word alignment“.
- Při 8bitovém čtení se čte z paměti celé slovo. Podle A0 se horní nebo dolní část nevyužije.
- Při 8bitovém zápisu se zapisuje pouze dolní nebo horní polovina slova signálem /LWR nebo /HWR.



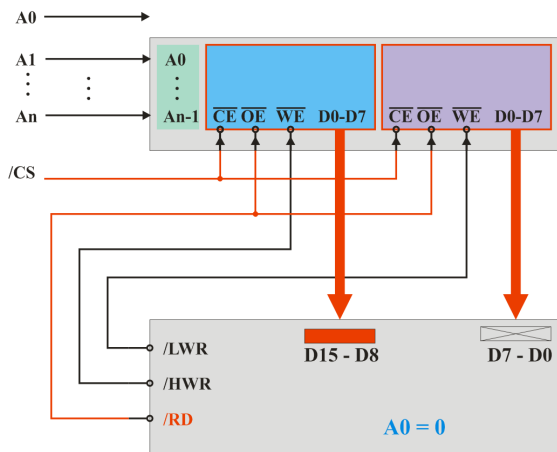
Práce H8S s 16bitovou pamětí (2)

- 16bitové čtení a zápis.

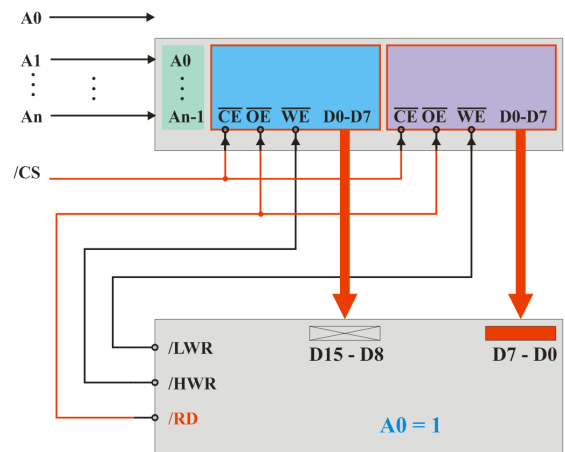


Práce H8S s 16bitovou pamětí (3)

- 8bitové čtení.



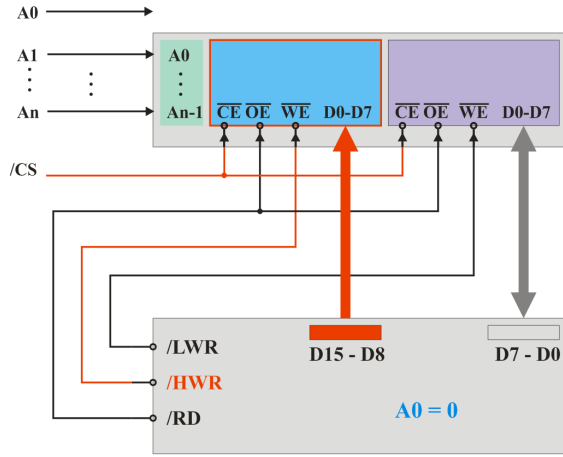
Sudá adresa



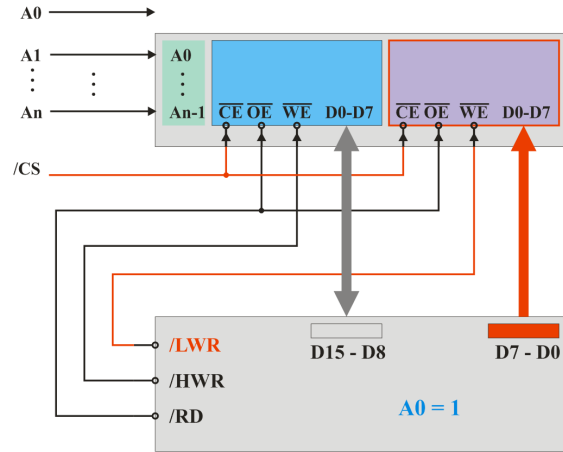
Lichá adresa

Práce H8S s 16bitovou pamětí (4)

- 8bitový zápis.

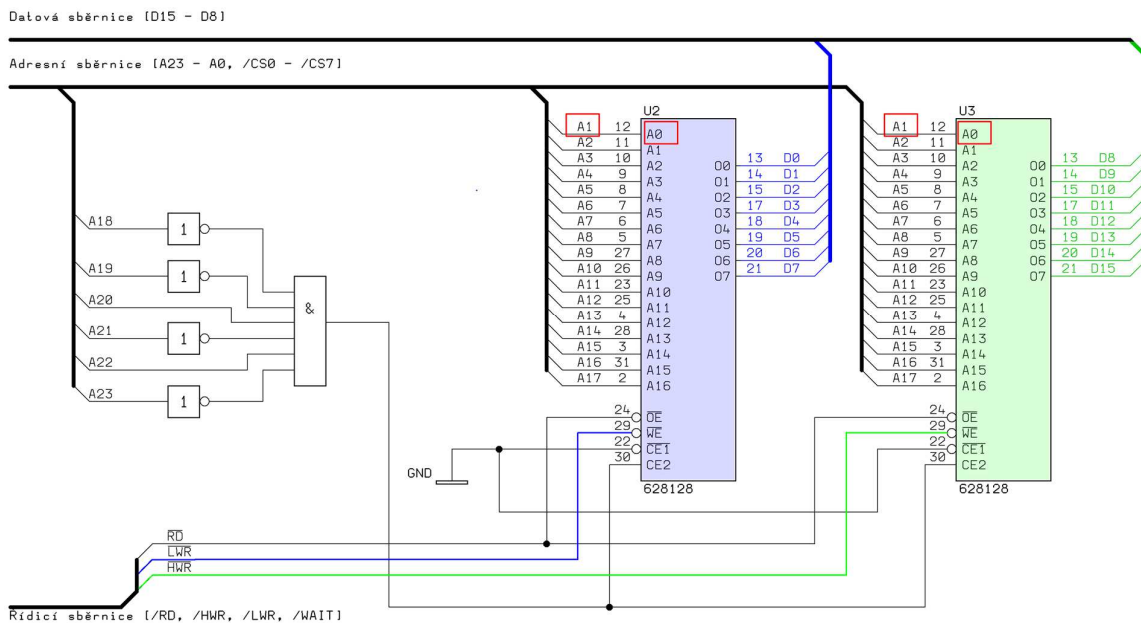


Sudá adresa



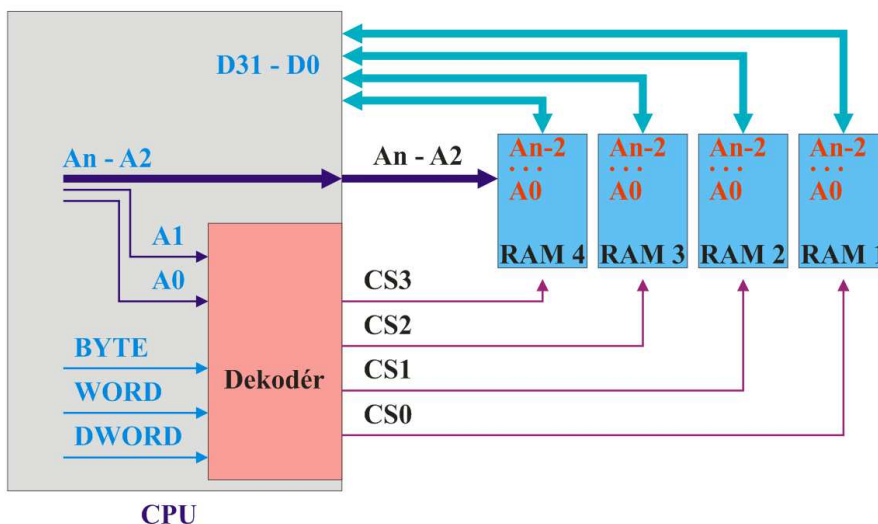
Lichá adresa

Zapojení 16bitové paměti



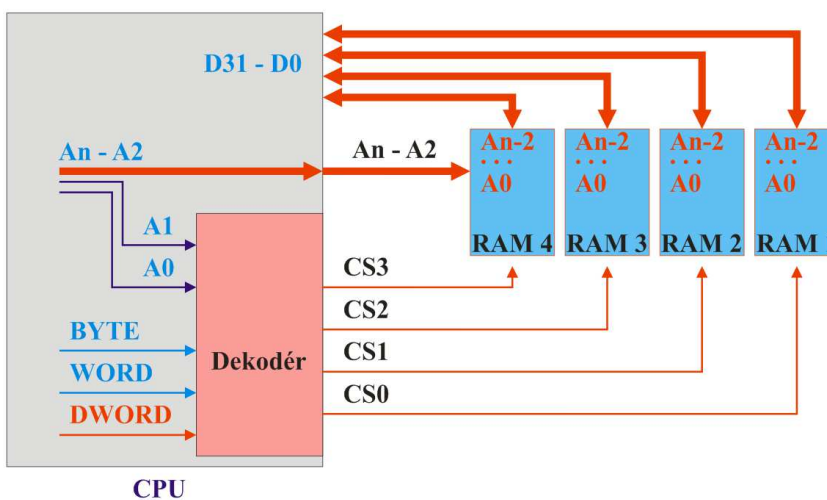
Připojení pamětí – 32bitová sběrnice (1)

- Adresní vodiče A1 a A0 jsou interně dekodovány v CPU.
- Paměť je sestavena ze 4 ks 8bitových obvodů.



Připojení pamětí – 32bitová sběrnice (2)

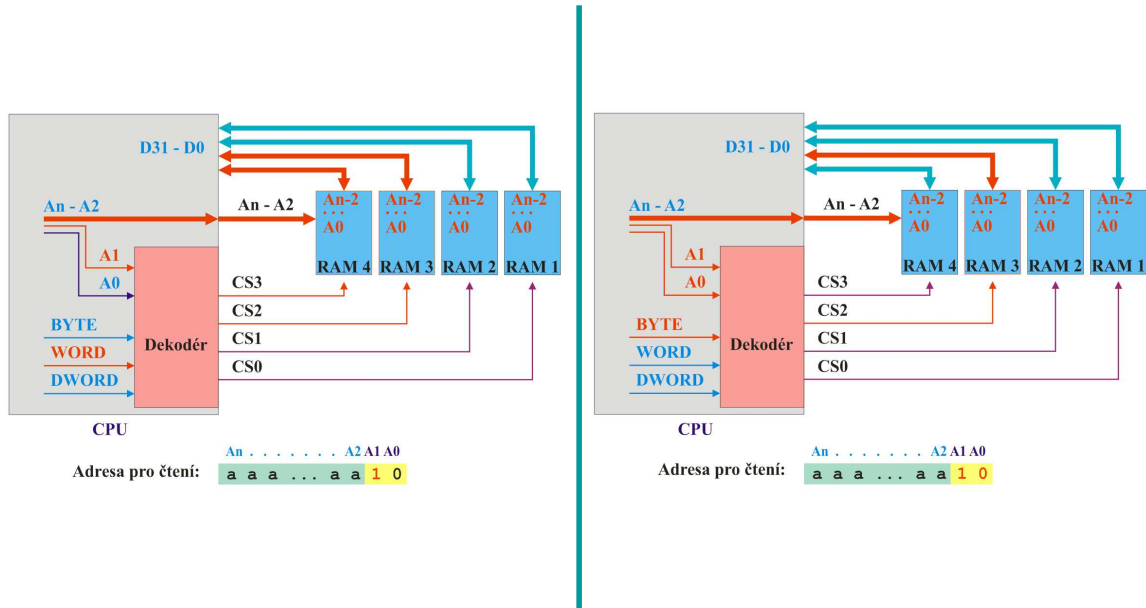
- Čtení 32bitového slova.



Adresa pro čtení: $A_n \dots A_2 A_1 A_0$
 $a \ a \ a \ \dots \ a \ a \ 0 \ 0$

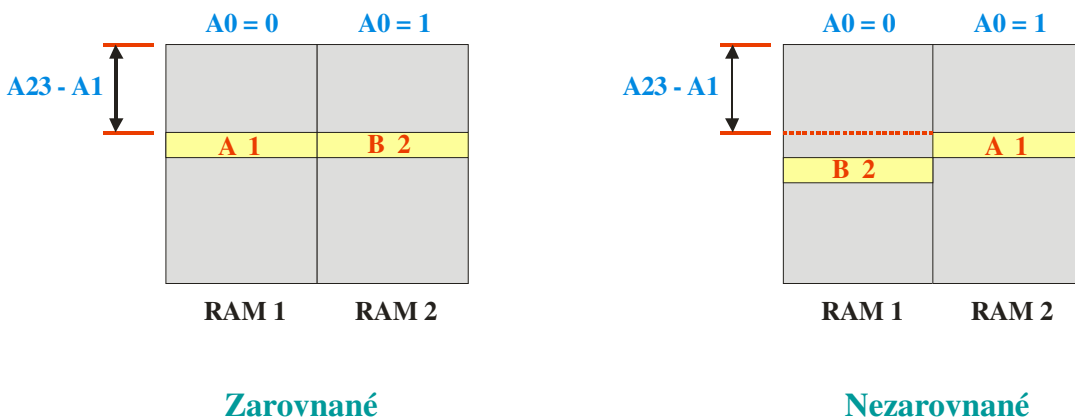
Připojení pamětí – 32bitová sběrnice (3)

- Příklad čtení 16bitového a 8bitového slova
 - Adresa = **aaa ... aaa10** .



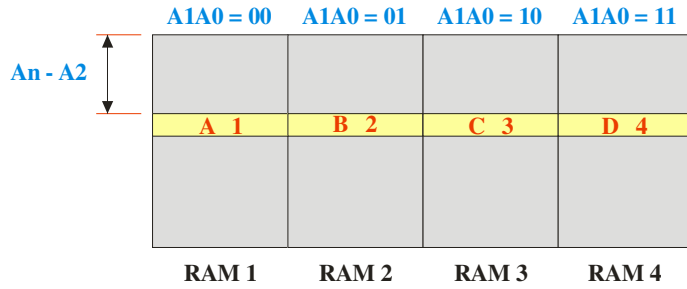
Zarovnání dat v paměti (word)

- Zarovnání „Word Alignment“:
 - Celé slovo má stejnou adresu $A_n - A_1$.
 - Uložené slovo = **0xA1B2** .

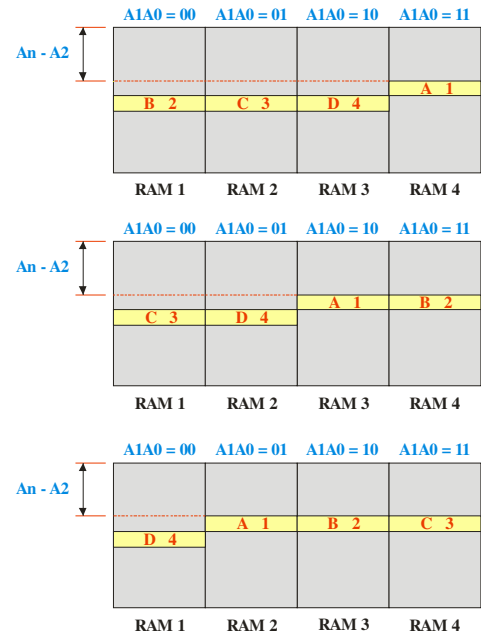


Zarovnání dat v paměti (dword)

- Zarovnání „DWord Alignment“:
 - Celé slovo má stejnou adresu $A_n - A_2$.
 - Uložené slovo = **0xA1B2C3D4** .



Zarovnané

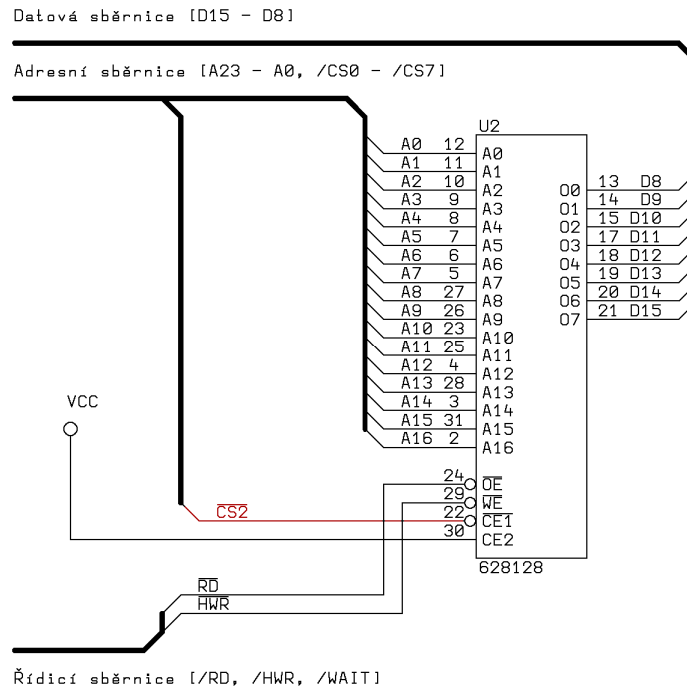


Nezarovnané

Neúplné dekódování adresy (1)

- Některé adresní vývody procesoru jsou nezapojeny \Rightarrow jejich hodnota může být libovolná.
- Každá paměťová buňka má několik adres (zrcadlení), které se liší v adresních bitech odpovídajících nezapojeným adresním vývodům.
- Příklad:** Paměť SRAM 128 kB zapojená podle následujícího schématu:

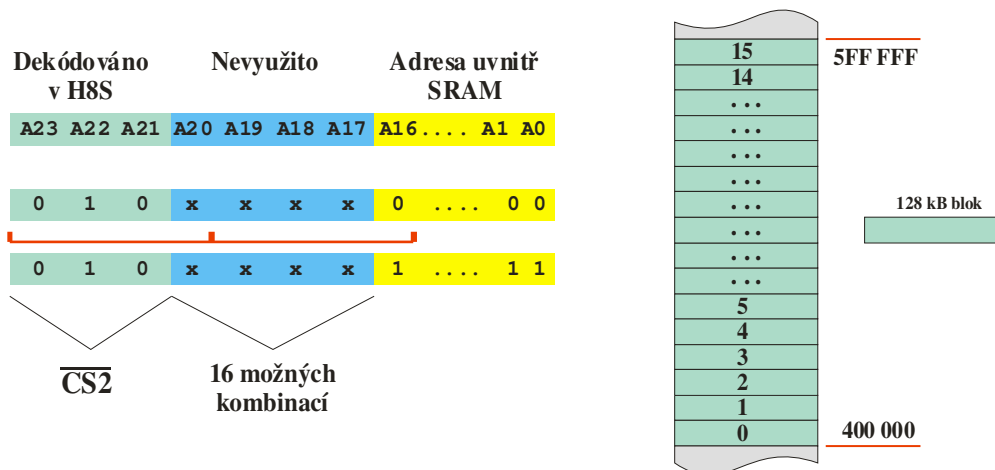
Neúplné dekódování adresy (2)



Neúplné dekódování adresy (3)

Komentář k předchozímu schématu:

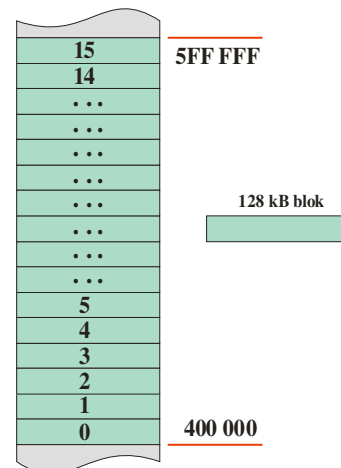
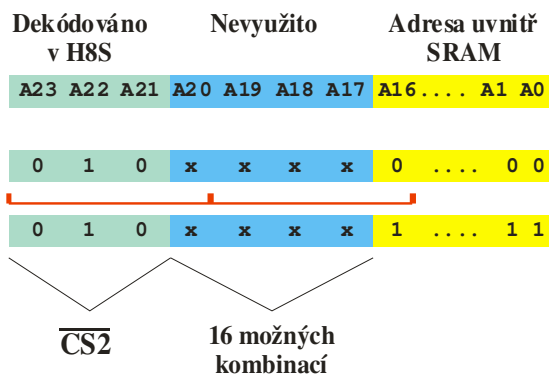
- Adresní vodiče A0 – A16 jsou využity v paměti SRAM.
- Adresní vodiče A21 – A23 jsou dekódovány v procesoru H8S. Paměť SRAM je připojena na /CS2 ⇒ vodiče A21 – A20 musí mít hodnotu 010.
- Adresní vodiče A17 – A20 nejsou využity ⇒ každá paměťová buňka má 16 různých adres, které se liší hodnotou A17 – A20.



Neúplné dekódování adresy (4)

Některé ze 16 možných adres slova na relativní adrese 0:

- 0100 0000 0000 0000 0000 0000 (0x400000)
- 0101 1110 0000 0000 0000 0000 (0x5E0000)
- 0101 1000 0000 0000 0000 0000 (0x580000)
- 0100 0110 0000 0000 0000 0000 (0x460000)
- ...

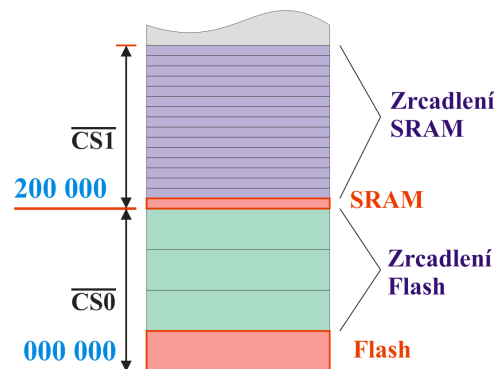


Použití neúplného dekódování (1)

Jednoduchý návrh paměťového systému: použijí se pouze signály /CSn bez dalších součástek.

Příklad:

- Zapojení má obsahovat Flash 512 kB + SRAM 128 kB.
- Flash má začínat na adrese 0x000000 ⇒ použijeme /CS0.
- Na poloze SRAM nezáleží ⇒ použijeme např. /CS1.
⇒ začátek SRAM bude na adrese 0x200000.



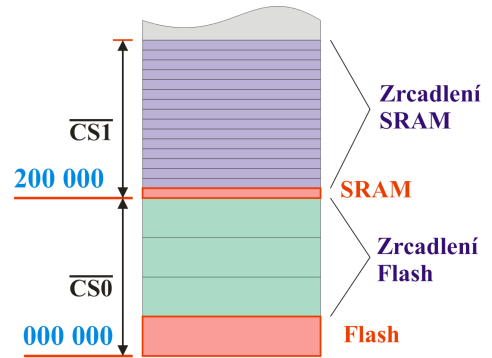
Použití neúplného dekódování (2)

Dekódováno v H8S				Nevyužito				Adresa uvnitř SRAM			
A23	A22	A21	A20	A19	A18	A17	A16	...	A1	A0	
0	0	1	x	x	x	x	x	0	...	0	0
0	0	1	x	x	x	x	x	1	...	1	1

CS1
16 možných kombinací
Dekódování adresy SRAM

Dekódováno v H8S				Nevyužito				Adresa uvnitř Flash			
A23	A22	A21	A20	A19	A18	A17	A16	...	A1	A0	
0	0	0	x	x	0	0	0	...	0	0	
0	0	0	x	x	1	1	1	...	1	1	

CS0
4 možné kombinace
Dekódování adresy Flash



Použití neúplného dekódování - zapojení

