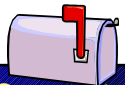
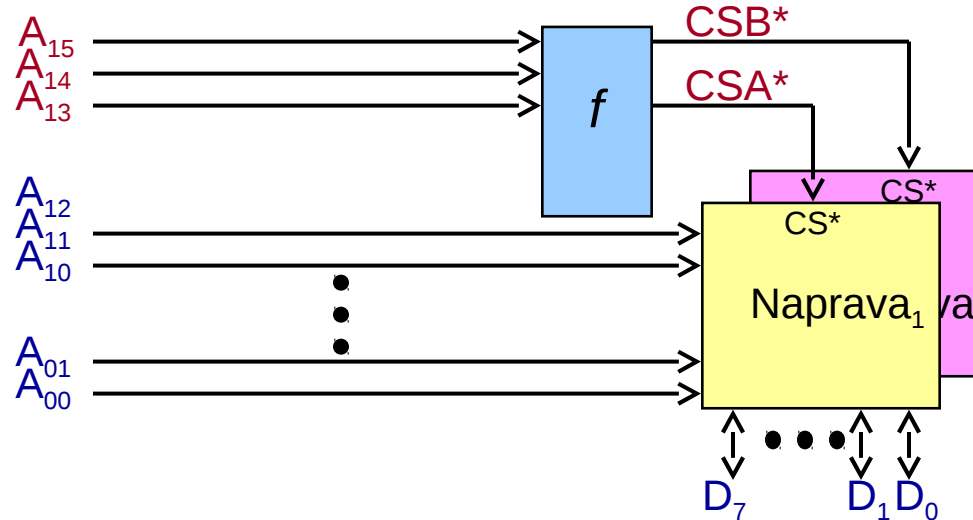


*Naslovno dekodiranje –  
postavljanje naprav v naslovni prostor*

# Izbira čipa (CS)

- Kako priključimo dve (ali več) naprav na vodilo?
  - Naenkrat mora biti izbran samo en čip (ali nobeden)
  - Za izbiro uporabimo naslednje signale:
    - R/W\*, Naslov ( $A_0$ - $A_{15}$ )
- Uporabni so biti, ki niso povezani na naslovne signale naprav ( $A_{15}$ - $A_{13}$ )
- $CSA^*$  in  $CSB^*$  sta torej funkciji  $A_{15}$ - $A_{13}$



Comments, Questions?

# Izbirna funkcija (CS)

CSA\* aktiven ob  $A_{15}--A_{13} = 001$

CSB\* aktiven ob  $A_{15}--A_{13} = 011$

CSA\* bo aktiven, kadar je naslov oblike:

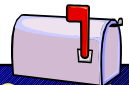
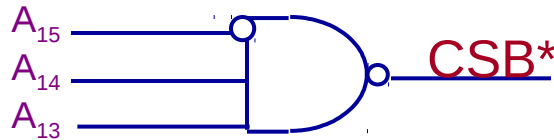
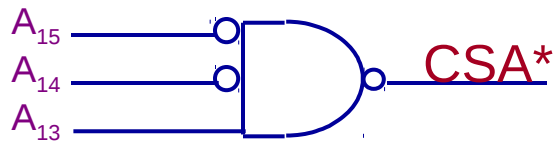
001x xxxx xxxx xxxx ; torej:

0010 0000 0000 0000 - 0011 1111 1111 1111 --> \$2000-\$3FFF

CSB\* bo aktiven, kadar je naslov oblike:

011x xxxx xxxx xxxx ; torej:

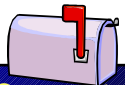
0110 0000 0000 0000 - 0111 1111 1111 1111 --> \$6000-\$7FFF



Comments, Questions?

# Popolno naslovno dekodiranje

- **Popolno naslovno dekodiranje:**
  - Pri izbiri naprave upoštevamo **vse naslovne bite**.
  - Upoštevati je potrebno tudi vse neuporabljene bite in zagotoviti, da je na njih prava kombinacija 0 in 1 (ničel in enic).
  - Vsaka naprava zaseda **svoje območje naslovnega prostora**
- **Postopek:**
  - V pomnilniško sliko najprej postavimo notranje pomnilnike, ki so fiksni.
  - Nadaljujemo z napravami, ki zasedejo veliko n.p.
  - Na koncu v sliko umestimo manjše naprave



Comments, Questions?

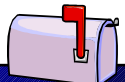
# Popolno naslovno dekodiranje

256 bajtov: notranji RAM (8 naslovnih bitov, \$0000 - \$00FF)  
 64 bajtov: notranji registri (6 naslovnih bitov, \$1000 - \$103F)  
 512 bajtov: notranji EEPROM (9 naslovnih bitov, \$B600 - \$B7FF)

Notranji  
pomnilniki

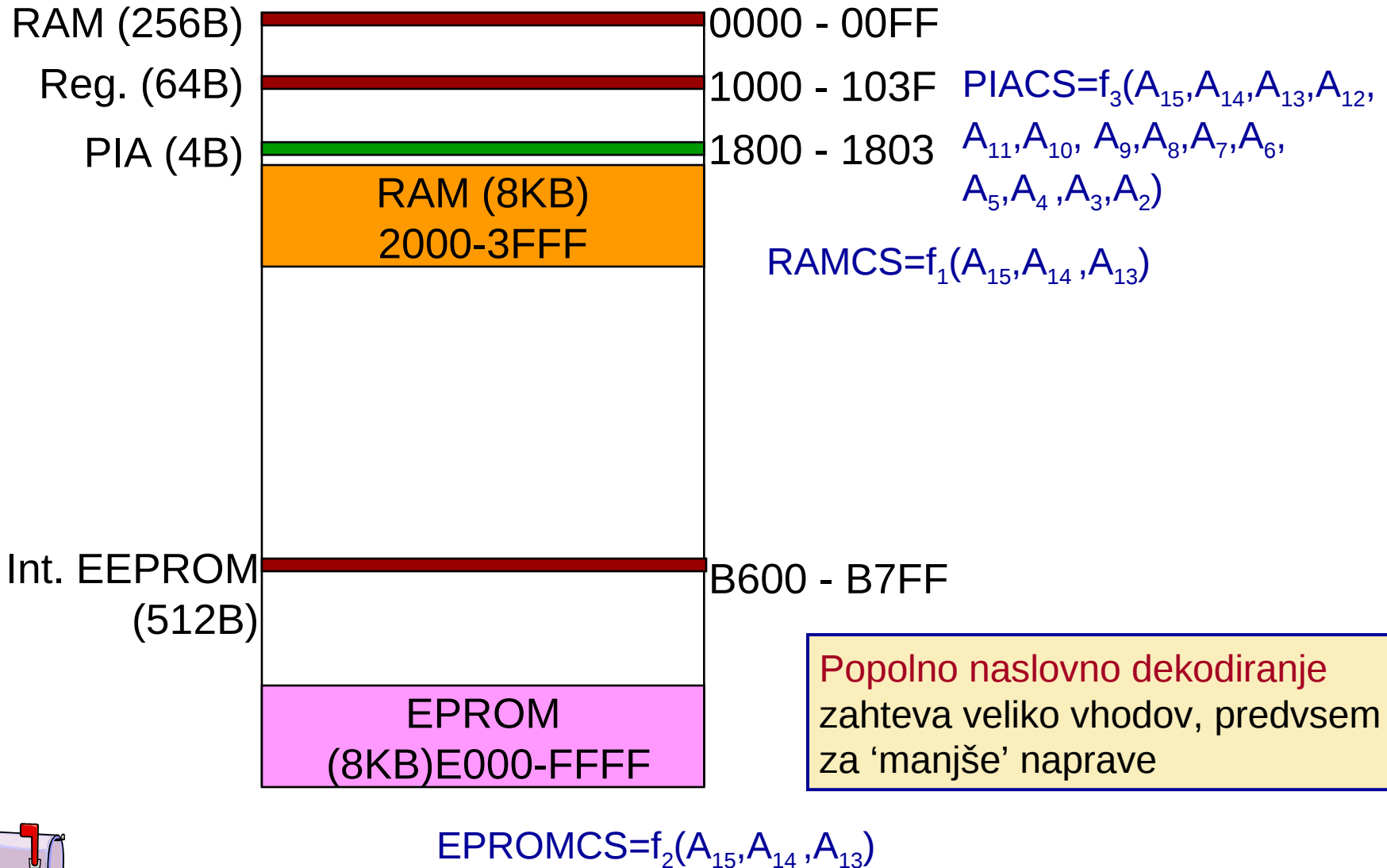
8KB RAM (13 naslovnih bitov) \$2000 - \$3FFF  
 8KB EPROM(RAM) (13 naslovnih bitov) \$E000 - \$FFFF  
 4 B PIA (2 naslovna bita) \$1800 - \$1803

	Naslovni biti															
	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
RAM	0	0	0	0	0	0	0	0	X	X	X	X	X	X	X	X
Registri	0	0	0	1	0	0	0	0	0	0	X	X	X	X	X	X
EEPROM	1	0	1	1	0	1	1	X	X	X	X	X	X	X	X	X
RAM	0	0	1	X	X	X	X	X	X	X	X	X	X	X	X	X
EPROM	1	1	1	X	X	X	X	X	X	X	X	X	X	X	X	X
PIA	0	0	0	1	1	0	0	0	0	0	0	0	0	0	X	X

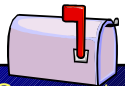


Comments, Questions?

# Pomnilniška slika



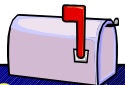
Popolno naslovno dekodiranje  
zahteva veliko vhodov, predvsem  
za 'manjše' naprave



Comments, Questions?

## P.N.D. - povzetek

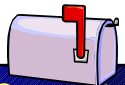
- **Popolno naslovno dekodiranje:** vsak fizični naslov ima natanko en logični naslov
  - Funkcije CS imajo veliko vhodov
  - Naslovni prostor za naknadno dodajanje naprav
- **Dekodiranje naslovov je zapleteno**
  - Običajno zahteva **veliko dekodejev** (čipov)



# Nepopolno naslovno dekodiranje

- Umestimo **2KB RAM** in a **2KB ROM** v naslovni prostor
- Dovolj je, če uporabimo samo **najpomembnejši bit** za izbiro RAM/ROM ( $A_{15}$ )
  - Če je  $A_{15} = 1$ , aktiviraj RAM, če je 0, aktiviraj ROM
  - Potrebujemo samo en negator!!
- Kaj z neuporabljenimi biti ( $A_{14}$ - $A_{11}$ )?
  - Stanje  $A_{14}$ - $A_{11}$  je **poljubno** – lahko ali 0 ali 1
  - Logika jih preprosto ignorira

	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
ROM	0					x	x	x	x	x	x	x	x	x	x	x
RAM	1					x	x	x	x	x	x	x	x	x	x	x



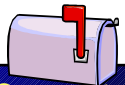
Comments, Questions?



# Več naslovov

- Kaj je npr. z naslovoma:
  - \$1100 = 00**0**1 0001 0000 0000
  - \$3100 = 00**1**1 0001 0000 0000
- Razlikujeta se samo na mestu  $A_{13}$ 
  - Vendar se bit  $A_{13}$  ne upošteva (ni dekodiran)
    - Naslovni dekodler ne zazna razlike!!
    - Na obeh **logičnih** naslovih je isti **fizični** naslov!!

	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
ROM	0					x	x	x	x	x	x	x	x	x	x	x
RAM	1					x	x	x	x	x	x	x	x	x	x	x



Comments, Questions?

# Slike

	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
ROM	0					X	X	X	X	X	X	X	X	X	X	X
RAM	1					X	X	X	X	X	X	X	X	X	X	X

From 0000 to 1111

Obstaja **16 kombinacij** bitov  $A_{15}$ ,  $A_{14}$ ,  $A_{13}$ , and  $A_{12}$

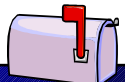
Torej obstaja **16 slik** (logičnih naslovov) za vsak fizični naslov

Slike \$8014:

**\$8014** \$8814 \$9014 \$9814 \$A014 \$A814 \$B014 \$B814  
\$C014 \$C814 \$D014 \$D814 \$E014 \$E814 \$F014 \$F814

Vseeno je, kateri naslov uporabljamo...

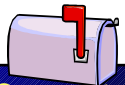
**Osnovni naslov** dobimo, če presledke v tabeli nadomestimo z ničlami



Comments, Questions?

# Nepopolno naslovno dekodiranje

- **Nepopolno naslovno dekodiranje:**
  - Pri izboru naprave upoštevamo *minimalno število* naslovnih bitov
    - Ostali biti se ne upoštevajo
  - Ista naprava se v naslovnem prostoru pojavi na **več območjih**
- **Postopek:**
  - V pomnilniško sliko najprej umestimo notranje pomnilnike
  - Sledijo veliki pomnilniki
  - **Upoštevamo čim manj bitov**



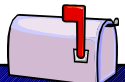
Comments, Questions?

# Nepopolno naslovno dekodiranje

Osnovno območje    Zasedeno območje

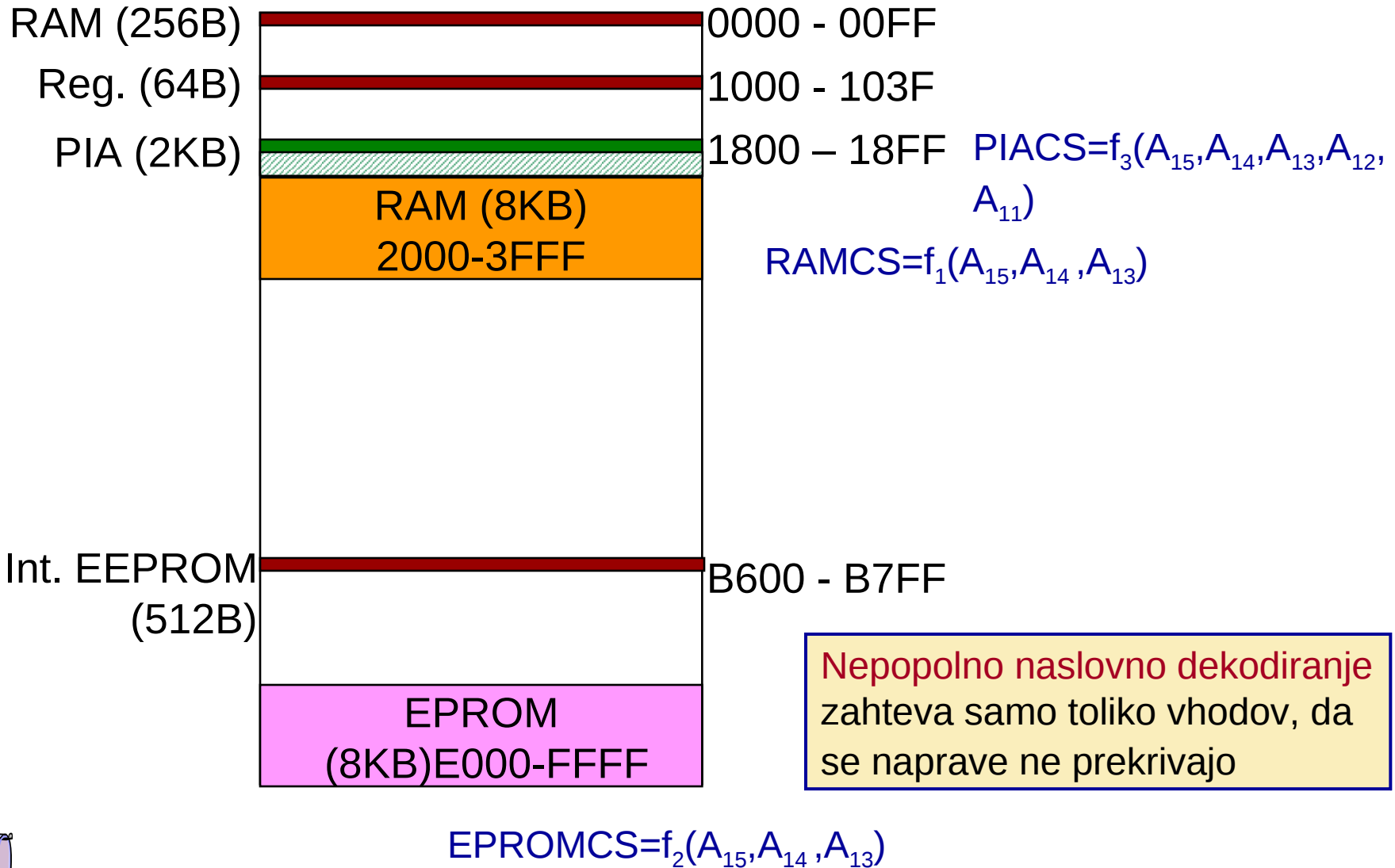
8KB RAM	(13 naslovnih bitov)	\$2000 - \$3FFF	\$2000 - \$3FFF
8KB EPROM(RAM)	(13 naslovnih bitov)	\$E000 - \$FFFF	\$E000 - \$FFFF
4B PIA	(2 naslovna bita)	\$1800 - \$1803	\$1800 - \$1FFF (2KB)

	Naslovni biti															
	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
RAM	0	0	0	0	0	0	0	0	X	X	X	X	X	X	X	X
Registri	0	0	0	1	0	0	0	0	0	0	X	X	X	X	X	X
EEPROM	1	0	1	1	0	1	1	X	X	X	X	X	X	X	X	X
RAM	0	0	1	X	X	X	X	X	X	X	X	X	X	X	X	X
EPROM	1	1	1	X	X	X	X	X	X	X	X	X	X	X	X	X
PIA	0	0	0	1	1										X	X

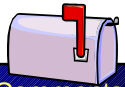


Comments, Questions?

# Pomnilniška slika



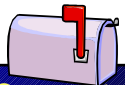
Nepopolno naslovno dekodiranje zahteva samo toliko vhodov, da se naprave ne prekrivajo



Comments, Questions?

## N.N.D. – Povzetek

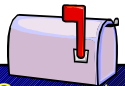
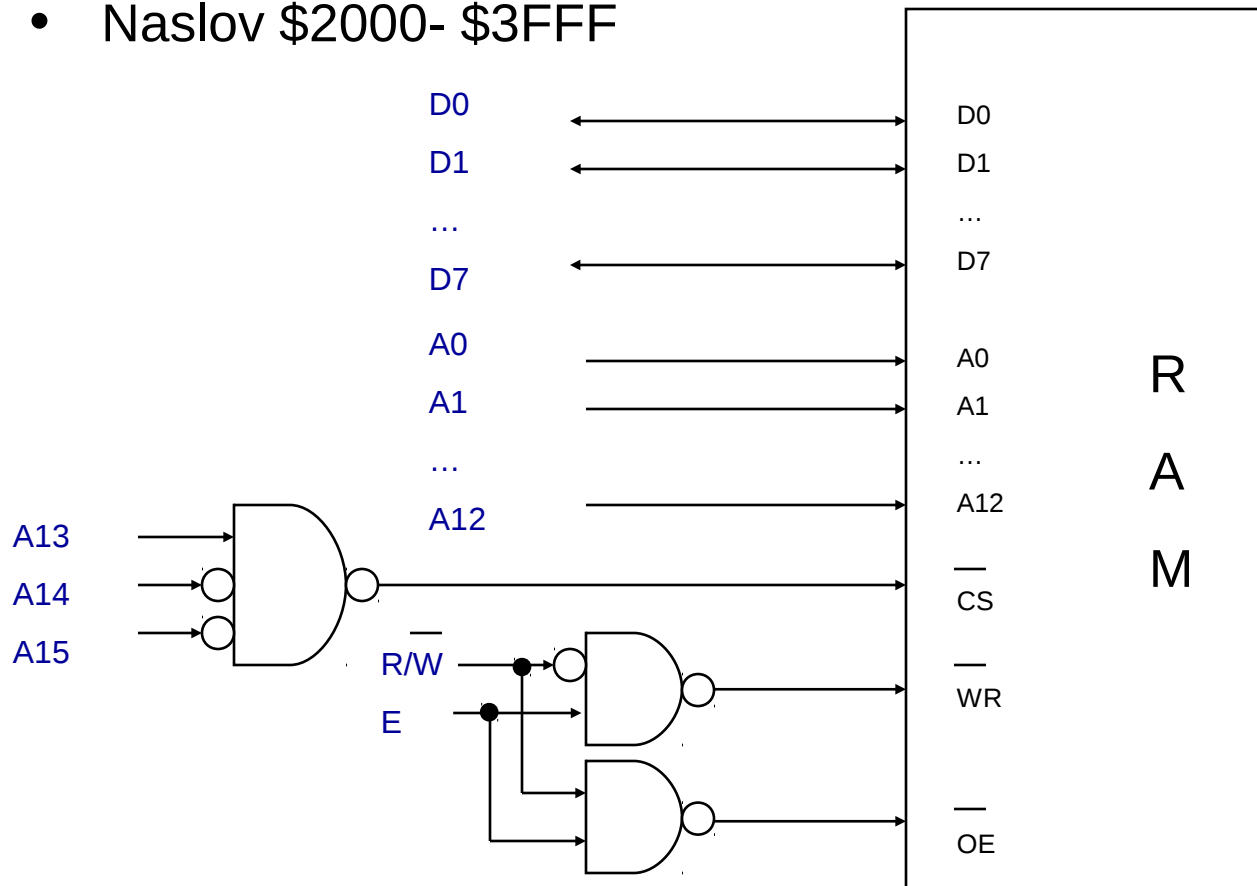
- **Za nepopolno naslovno dekodiranje je značilno bolj preprosto dekodiranje naslovov**
  - Uporabimo samo toliko bitov, kot je potrebno
- **Vendar imajo posamezne naprave v naslovnem prostoru več slik**
  - Do istih fizičnih naslovov pridemo preko različnih logičnih naslovov, kar je nepregledno.
- **Večkratne kopije naprav porabijo veliko naslovnega prostora**
  - Omejena razširljivost



Comments, Questions?

# Priključitev 8KB RAM

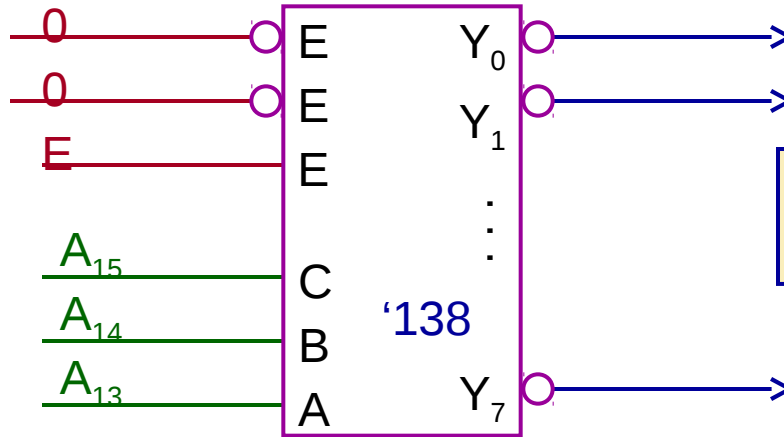
- Na procesorje iz družine 6800
- Naslov \$2000- \$3FFF



Comments, Questions?

# Uporaba dekoderjev

Omogočimo ko je **E** v stanju 1

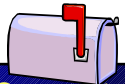
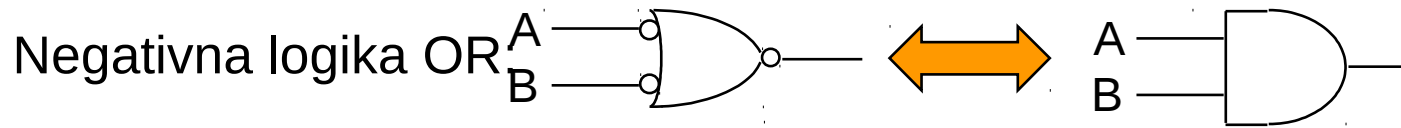


Izhodi so funkcije **CS\*** za pomnilnike in druge naprave

Dekodiramo tri najpomembnejše bite naslova

Naslovni prostor razdelimo v **osem različnih območij** velikosti **8KB**.

Za večja območja: 'OR' dveh izhodov za 16KB območje

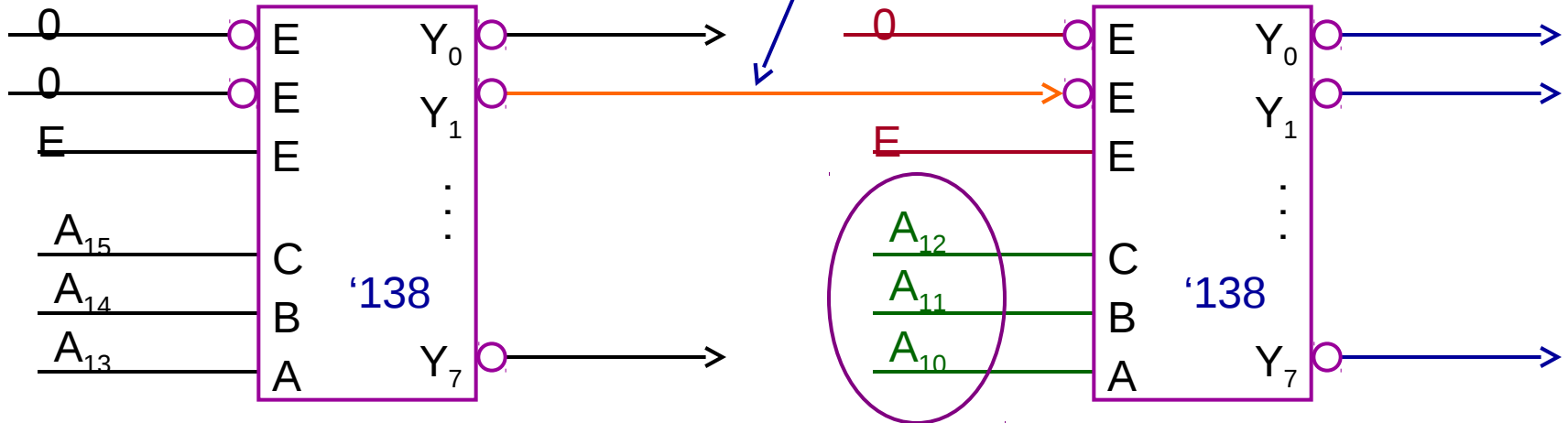




# Kaskadna vezava

Izbor 8KB območij

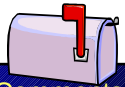
8KB: 2000 - 3FFF



Naslednji trije biti

Osem 1KB območij  
2000 - 3FFF

Sedem 8KB območij in osem 1KB  
območij



Comments, Questions?