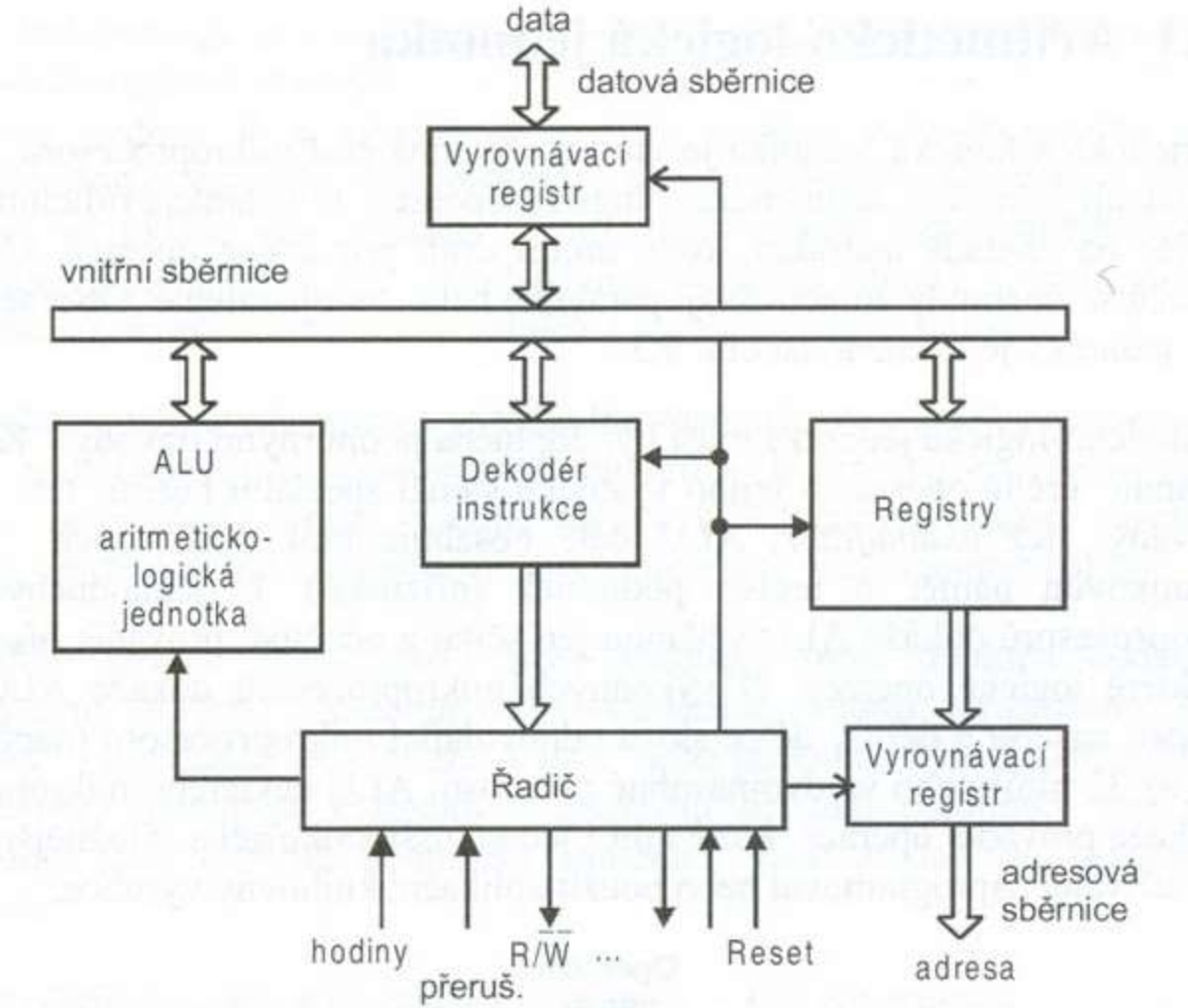
Procesor

MS

Blokové schéma procesoru



Řadič

- Určuje zda je procesor
  - RISC (**Reduced Instruction Set Computing)**
  - CISC (**Complex Instruction Set Computing)**
- V řadiči je zakódovaný strojový kód procesoru potažmo počítače
- Různé řešení
  - Sekvenční automat
  - Mikroprogramový řadič
  - …
- pipelining

Typy instrukcí

- *Aritmetické*
- *Logické*
- *Posuvy a rotace*
- *Přesuny*
- *Instrukce skoků*
  - nepodmíněný - JMP adresa
  - podmíněný - JNZ adresa
- *Řídící instrukce*
- *I/O instrukce*

Registry procesoru x86

- aritmetické
  - AX – akumulátor (řada instrukcí ho má jako implicitní operand)
  - BX – bázový registr (tj. určený pro adresaci)
  - CX – čítač (tj. určený pro počítání cyklů)
  - DX – rozšíření akumulátoru (někdy nazýván data register)
  - SI – source index – index pro zdroj (tj. pro čtení)
  - DI – destination index – index pro cíl (tj. pro zápis)
  - BP – base pointer – určen jako ukazatel na záznam aktivní procedury na zásobníku (tím, že se implicitně spojoval s SS)
  - SP – stack pointer – ukazatel vrcholu zásobníku
- segmentové
  - CS – segment kódu
  - DS – datový segment
  - ES – extra segment
  - SS – zásobníkový (stack) segment
- registr IP (instrukční pointer neboli čítač instrukcí), vždy odkazovaný implicitně
- registr FLAGS (příznaky)

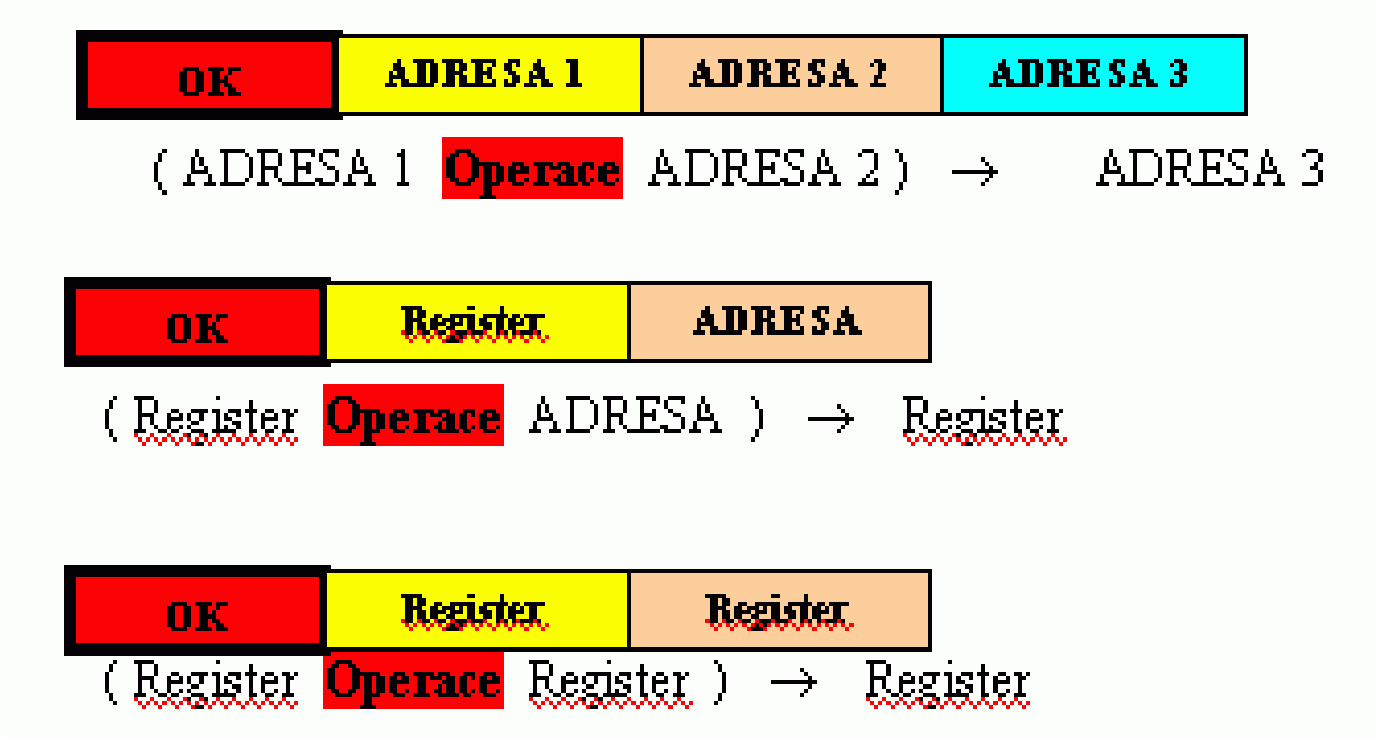
Registr stavový FLAGS

- příznakové bity
- OV (over flow) - přetečení
- Z (zero) - nulový výsledek
- C (Carry) - přenos do vyššího řádu
- AC (Auxillary carry) - přenos při operaci s BCD čísly
- S (sign) - poznaménkový bit
- P (parity) - parita - počítá sudý, lichý počet jedniček

Ukázka instrukcí

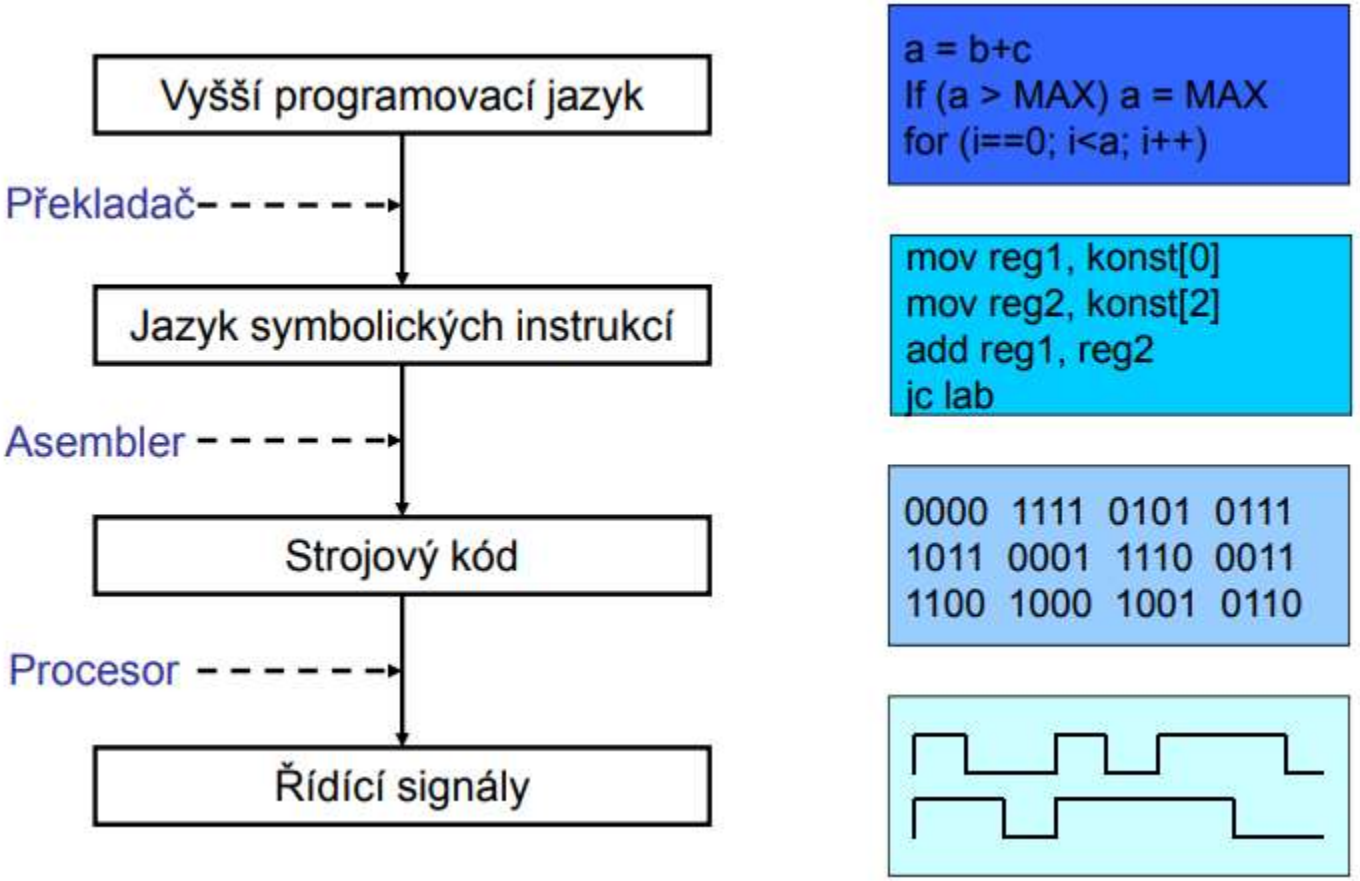
- **add r0, r2**; – sčítání - přičte do registru r0 hodnotu uloženou v registru r2
- **addc r1, r3**; – sčítání s přenosem - přičte do registru r1 hodnotu registru r3 a příznak přenosu C
- **mov 1234h, r0**; – přesun - uloží do paměti na adresu 1234h hodnotu z registru r0
- **mov 1236h, r1**; – přesun - uloží do paměti na adresu 1236h hodnotu z registru r1
- **mov [r7], r0**; – přesun - uloží na adresu určenou registrem r7 hodnotu z registru r0
- **cmp r4, r5**; – porovnání - porovná hodnoty registrů r4 a r5
- **jmpr cc_UGT, 100h**; – podmíněný relativní skok - pokud byla hodnota registru r4 vyšší, pokračuje program o 100h adres dále (PC<-PC+100h)

Ukázka možné struktury instrukcí



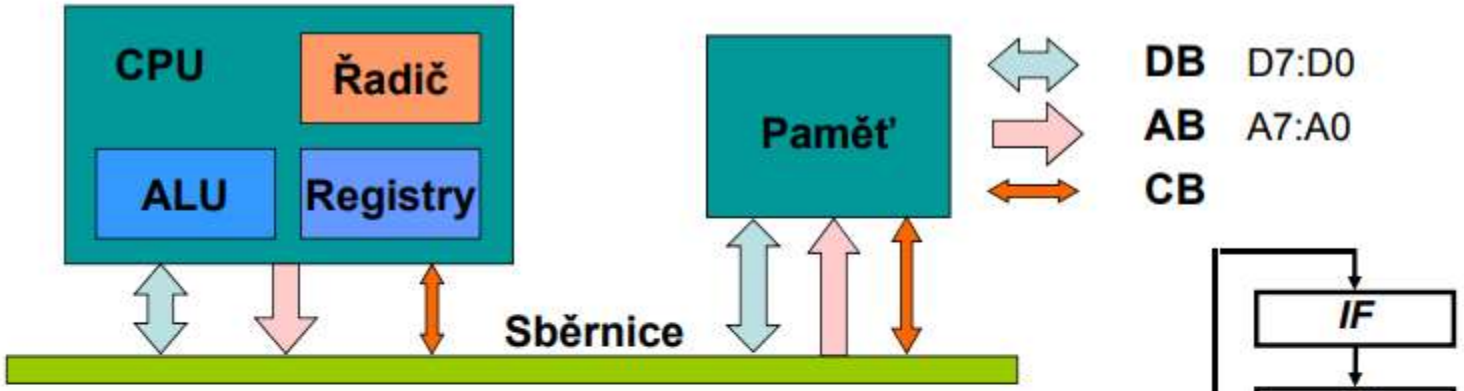
OK – operační kód (binární kód instrukce)

Program


Vyšší programovací jazyk
Překladač
Jazyk symbolických instrukcí
Asembler
Strojový kód
Procesor
Řídící signály
a = b+c
If (a > MAX) a = MAX
for (i==0; i<a; i++)
mov reg1, konst[0]
mov reg2, konst[2]
add reg1, reg2
jc lab
0000 1111 0101 0111
1011 0001 1110 0011
1100 1000 1001 0110

Fáze zpracování instrukce

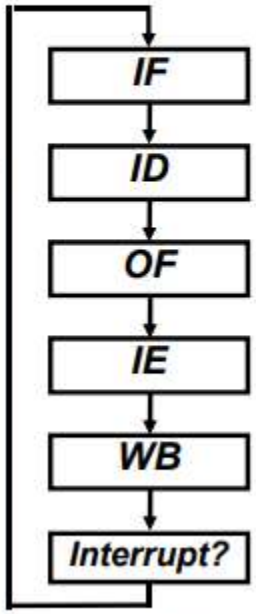
DB – DataBus (datová část sběrnice);
AB – AdressBus (adresová část sběrnice);
CB – ControlBus (řídicí část sběrnice);



CPU musí obsahovat aritmeticko-logickou jednotku (ALU), řadič a registry pro dočasné uchování dat, instrukcí a pomocných proměnných

CPU pro běh programu prochází základním cyklem počítače:

1. Instruction Fetch — IF – načtení instrukce
2. Instruction Decode — ID – dekódování instrukce
3. Operand Fetch — OF – načtení operandu(ů)
4. Instruction Execution — IE – vykonání instrukce
5. Write Back — WB – zapsání výsledku



Dnes

- Mikrokód je u procesorů Intel (Pentium Pro a novějších) možné nahrazovat, čehož využívají výrobci procesorů k opravě chyb. Nový mikrokód se musí po každém zapnutí nahrávat znovu, protože je uložen ve volatilní paměti a mikroprocesor se po vypnutí vrací ke svému původnímu mikrokódu. K aktualizaci mikrokódu může být použit BIOS (v rámci POST testů po zapnutí počítače) nebo je možné ho nahradit i později.

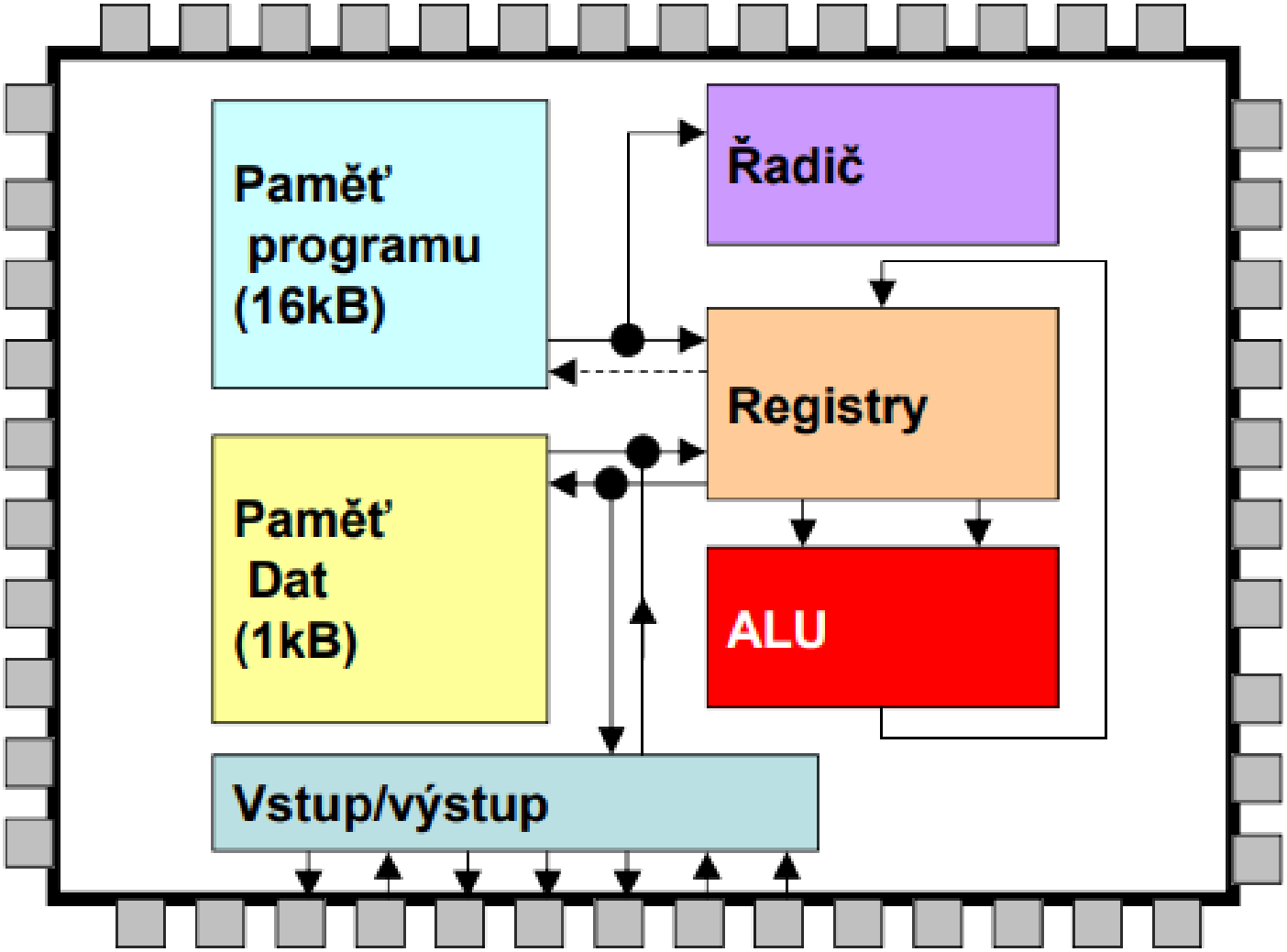
Co je nutné rozhodnout při návrhu

- jaké instrukce
- jak kódované
- jak dlouhé
- jaká adresace operandů
- kolik registrů

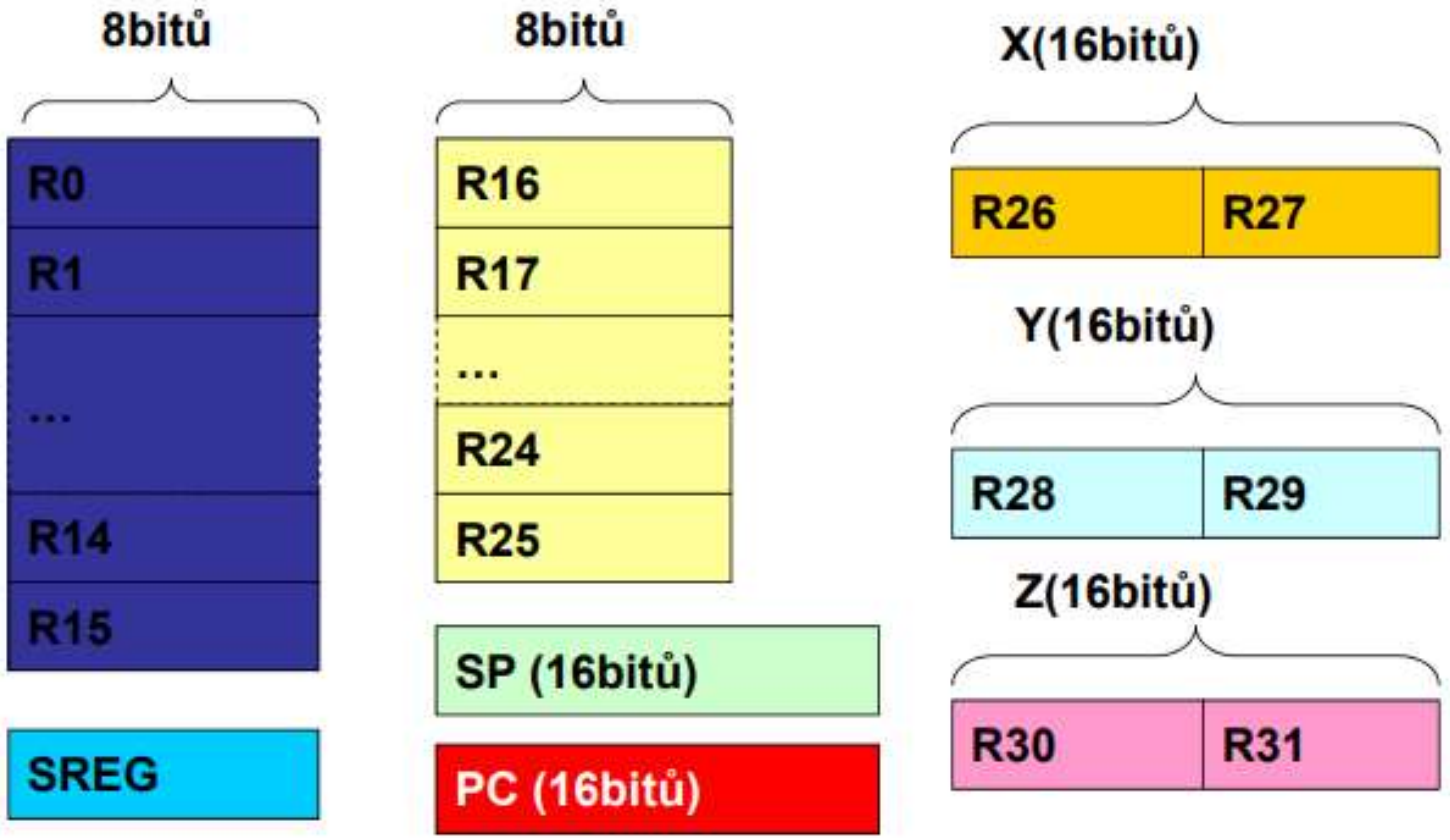
AVR mikropočítač

- Arduino
- Jednoduché

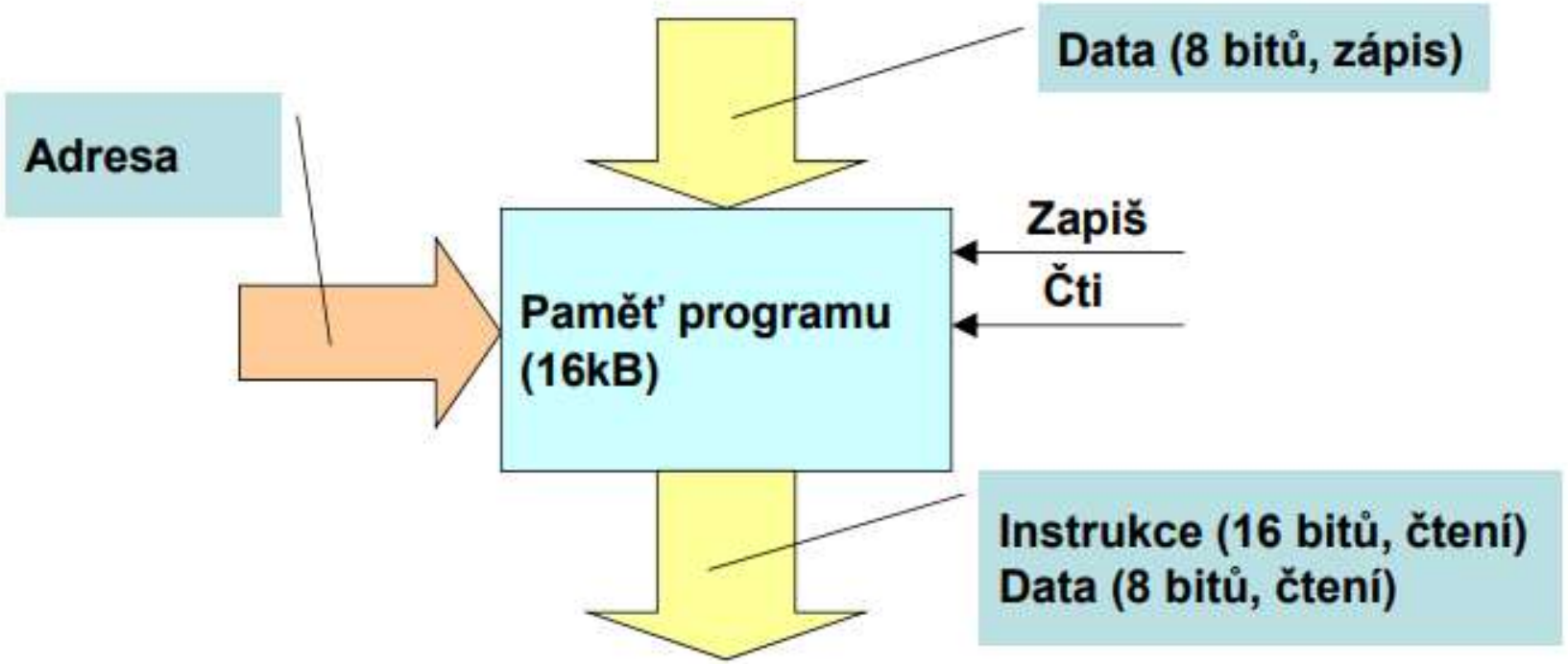
Architektura mikropočítače AVR



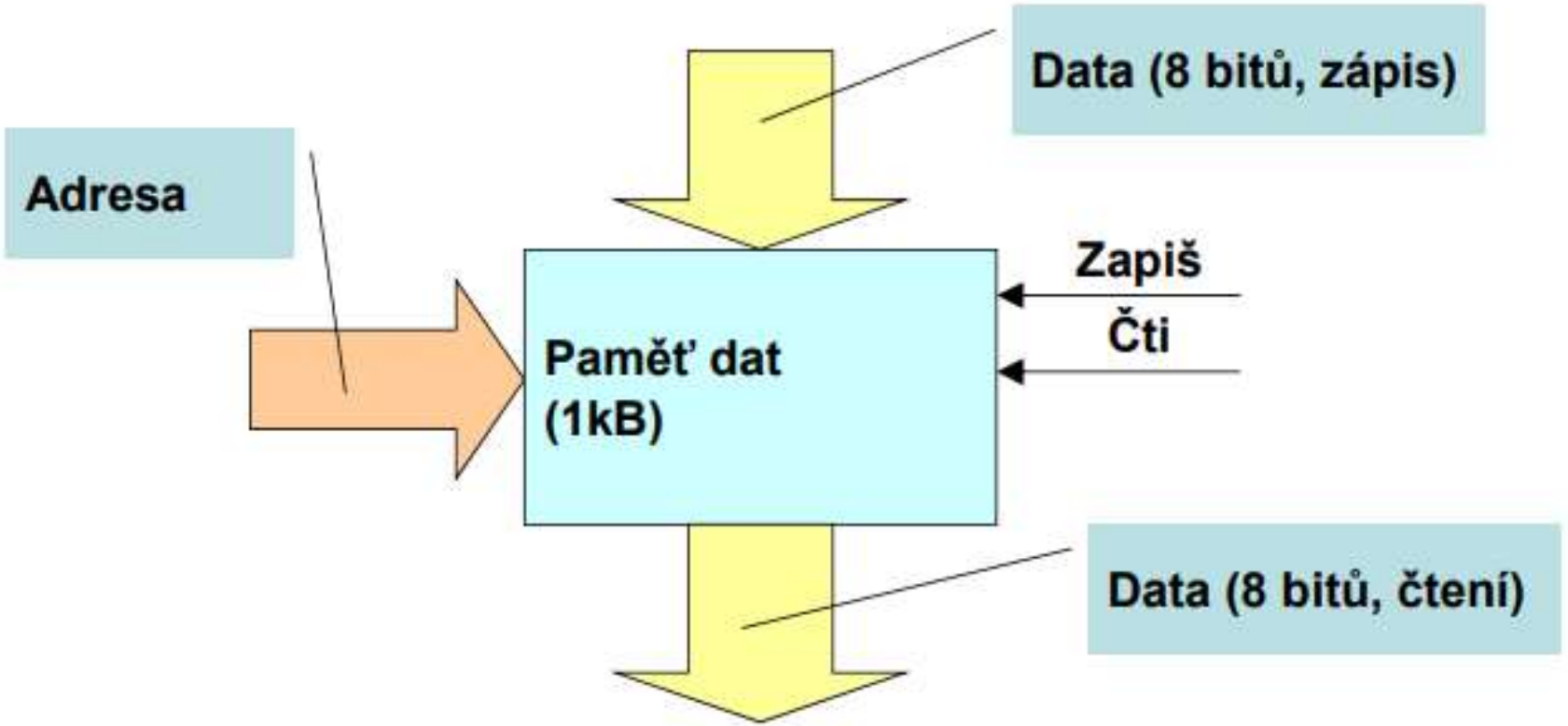
Registry


8bitů
R0
R1
…
R14
R15
SREG
8bitů
R16
R17
…
R24
R25
SP (16bitů)
PC (16bitů)
X(16bitů)
R26
R27
Y(16bitů)
R28
R29
Z(16bitů)
R30
R31

Paměť programu



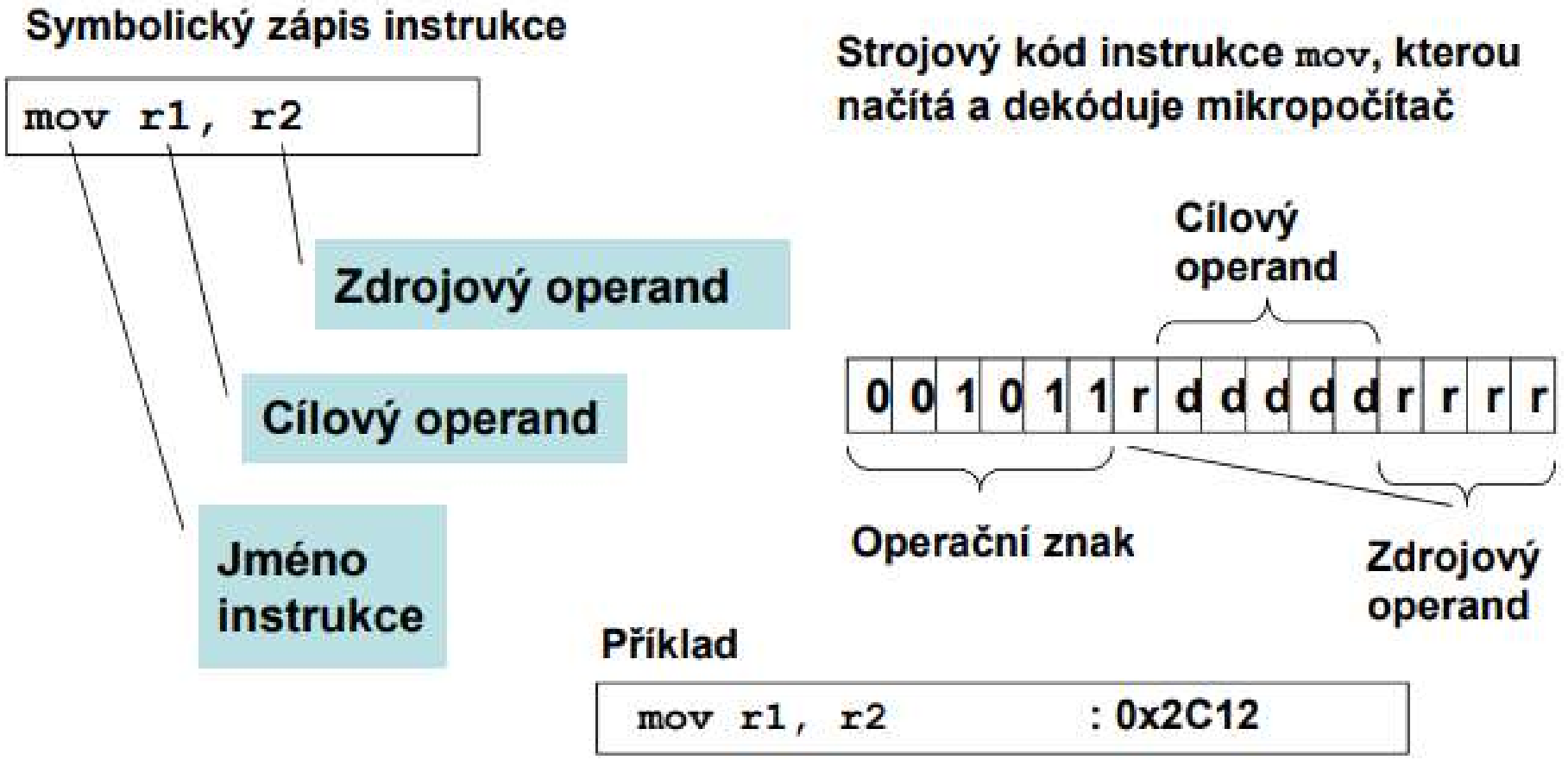
Paměť dat


Data (8 bitů, zápis)
Adresa
Zapiš
Čti
Paměť dat
(1kB)
Data (8 bitů, čtení)

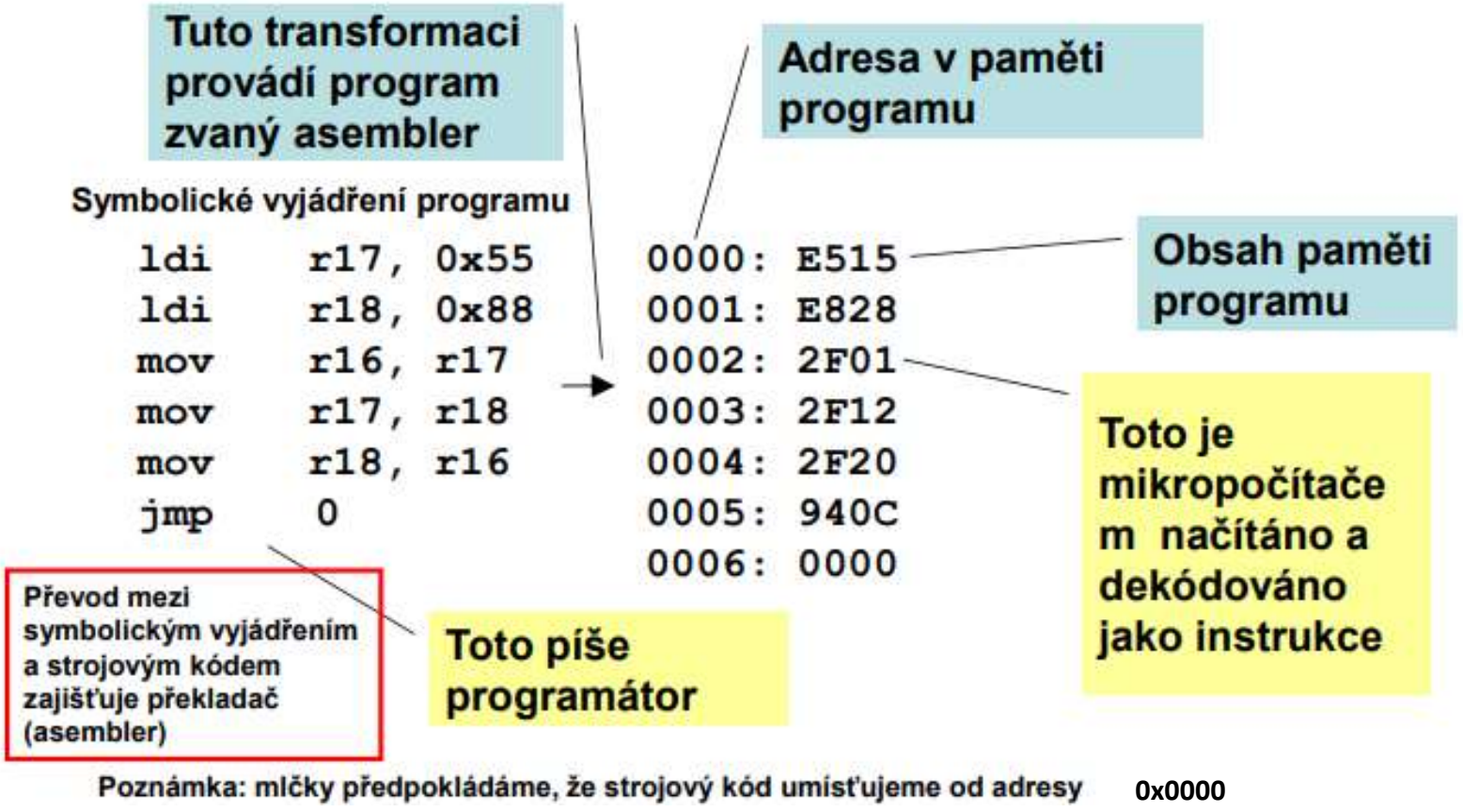
Činnost řadiče

1. Načti instrukci z adresy v PC
2. Dekóduj instrukci
3. Získej operandy
4. Vykonej instrukci
5. Ulož výsledek
6. Urči adresu následující instrukce
7. Pokračuj bodem 1

Příklad instrukce



Příklad programu v symbolickém vyjádření a ve strojovém kódu



Video

- https://www.youtube.com/watch?v=cNN_tTXABUA