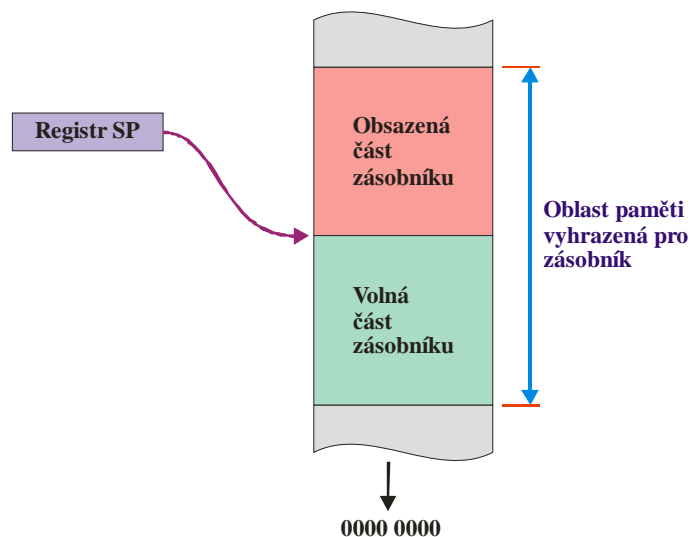


Použití zásobníku

Zásobník podporovaný HW procesoru

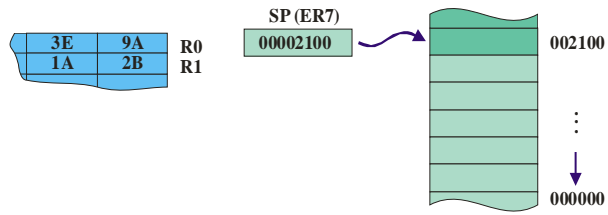
- Procesor má speciální registr SP, který ukazuje na vrchol zásobníku.
- Se zásobníkem se pracuje instrukcemi PUSH a POP.
- Dále je používán při volání podprogramů a obsluze přerušení.



Operace PUSH a POP

- **Procesor má speciální registr SP (Stack Pointer)**

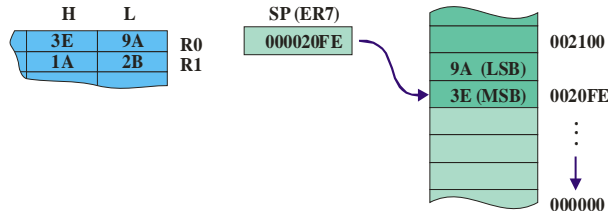
U H8S má funkci SP registr ER7.



- **Operace PUSH**

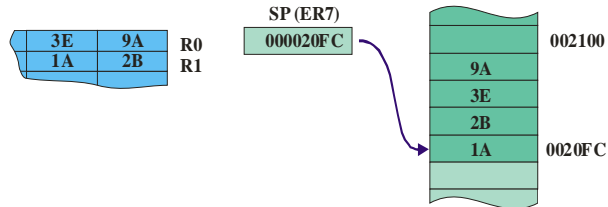
Uložení do zásobníku.

$R_n \rightarrow @-SP$



Příklad:

PUSH R0
PUSH R1



Operace PUSH a POP

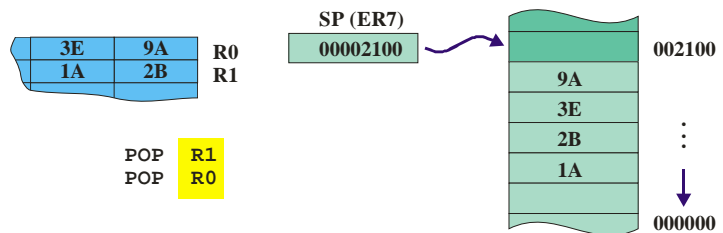
- **Procesor má speciální registr SP (Stack Pointer)**

U H8S má funkci SP registr ER7.

- **Operace POP**

Vyjmutí ze zásobníku.

$@SP+ \rightarrow R_n$



POP R1
POP R0

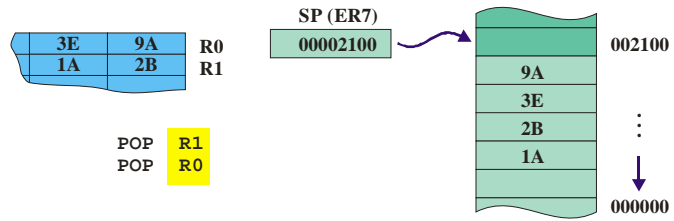
Příklad:

POP R1
POP R0

Operace PUSH a POP

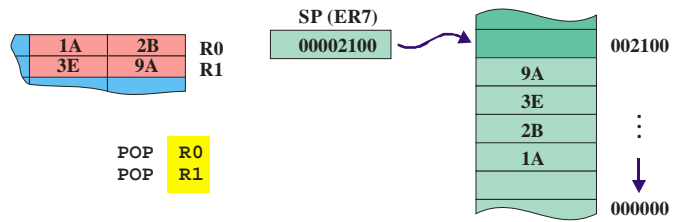
Příklad:

```
POP R1
POP R0
```



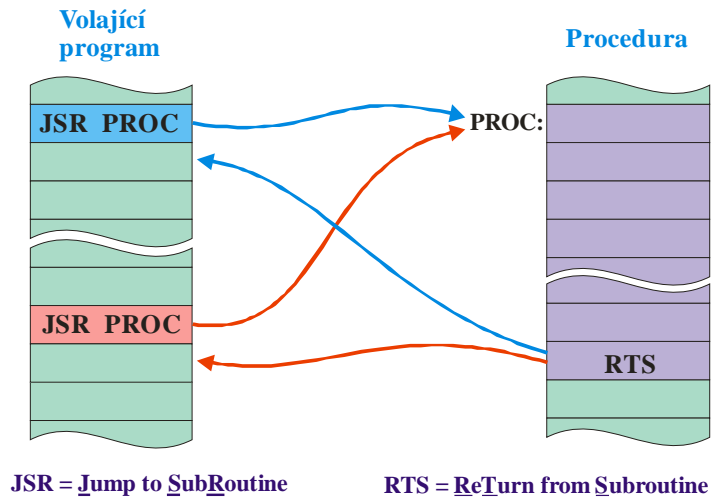
Příklad:

```
POP R0
POP R1
```



Volání podprogramu

- Instrukce RTS musí provést návrat na správné místo (adresu) volajícího programu.



JSR = Jump to SubRoutine

RTS = ReTurn from SubRoutine

Volání podprogramu (1)

• **Instrukce JSR**

Skok do podprogramu.

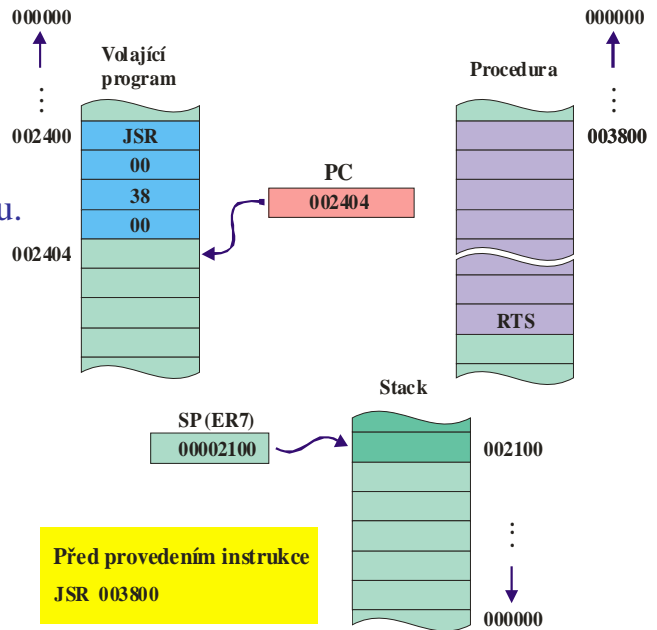
- uloží návratovou adresu do zásobníku,
- provede skok na cílovou adresu.

PC → @-**SP**

<EA> → **PC**

Příklad:

JSR 003800



Volání podprogramu (2)

• **Instrukce JSR**

Skok do podprogramu.

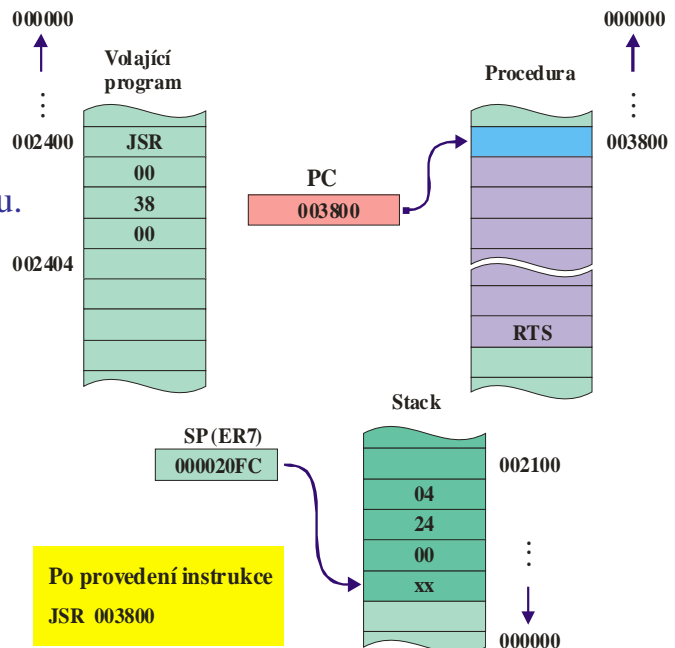
- uloží návratovou adresu do zásobníku,
- provede skok na cílovou adresu.

PC → @-**SP**

<EA> → **PC**

Příklad:

JSR 003800



Návrat z podprogramu

- Instrukce RTS**

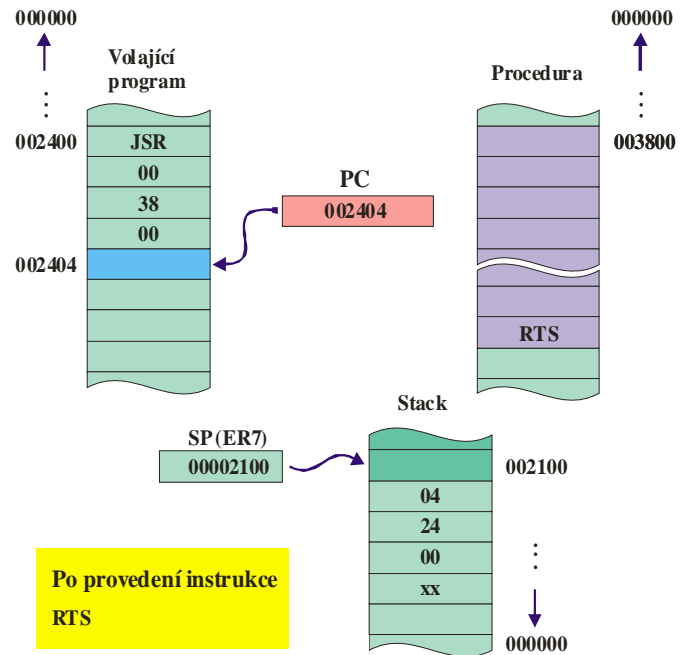
Návrat z podprogramu.

- vyjme obsah vrcholu zásobníku a uloží jej do PC.

@SP+ → PC

Příklad:

RTS



Nastavení SP

- **Inicializace SP je starost programátora.**
- Programátor musí zajistit:
 - Vyhrazení místa pro zásobník.
 - Nastavení SP na dno zásobníku.

```
.data
...
.space 100 ;zasobnik
stck:
...
```

```
.text
...
mov.l #stck,ER7 ;init SP
...
```

Poznámka:

Hodnota symbolické adresy **stck** odpovídá adrese 1. bytu za oblastí, vyhrazenou direktivou **.space 100**.