

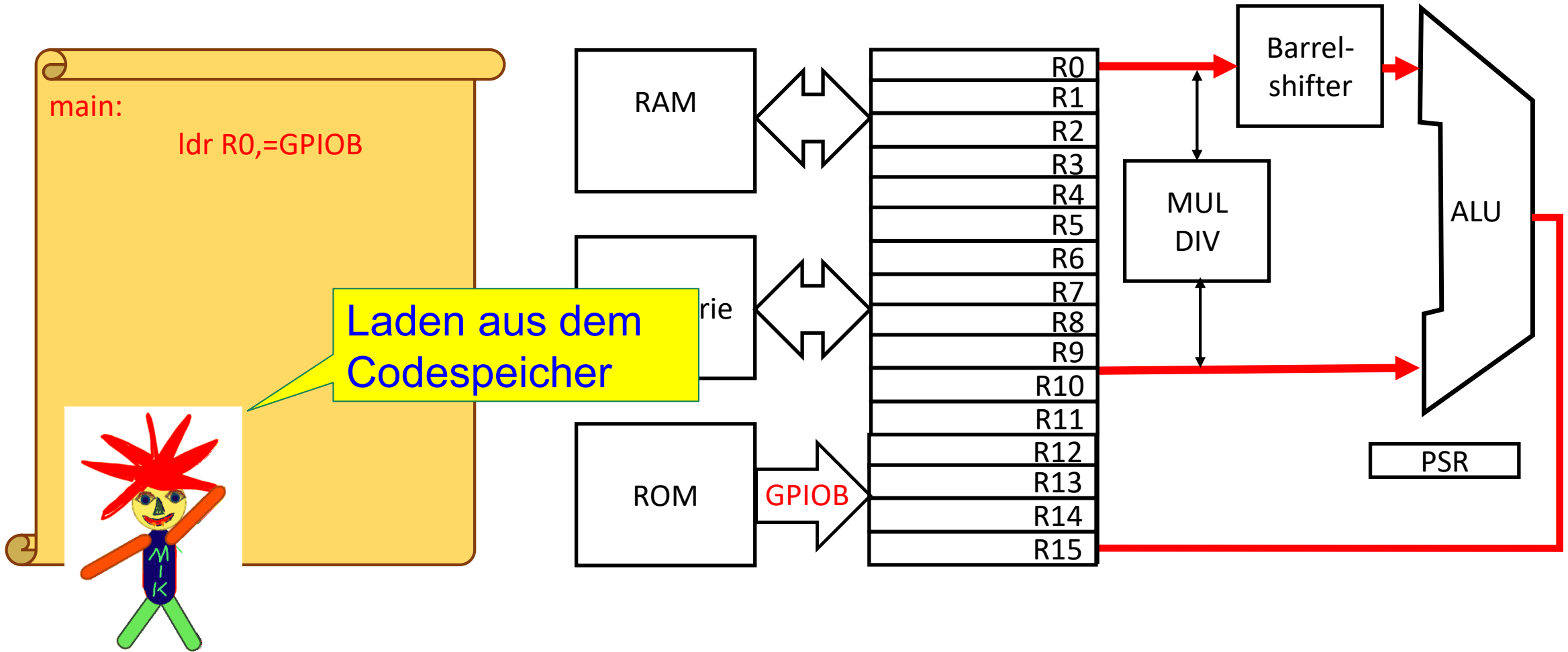
STM32 Load-Store-Architektur



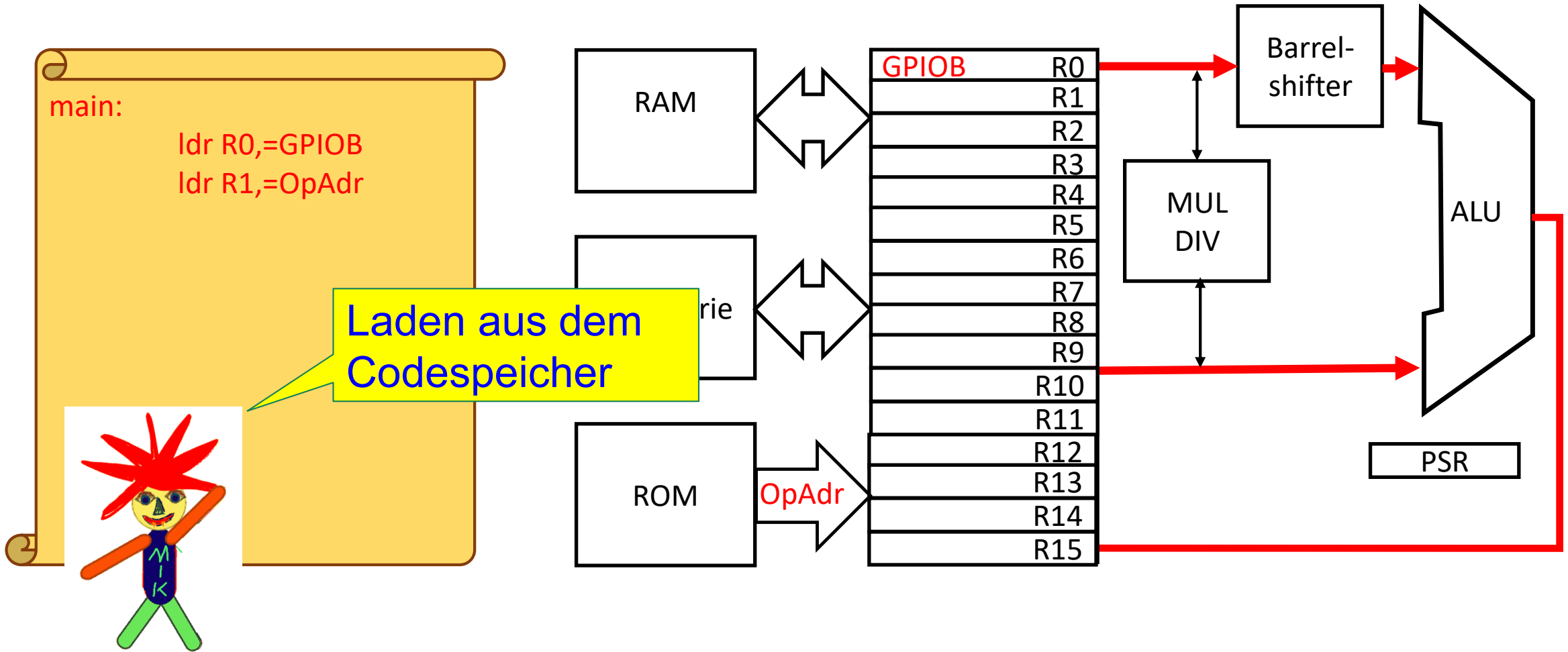
Ich bin Mik, Dein
Mikrocontroller



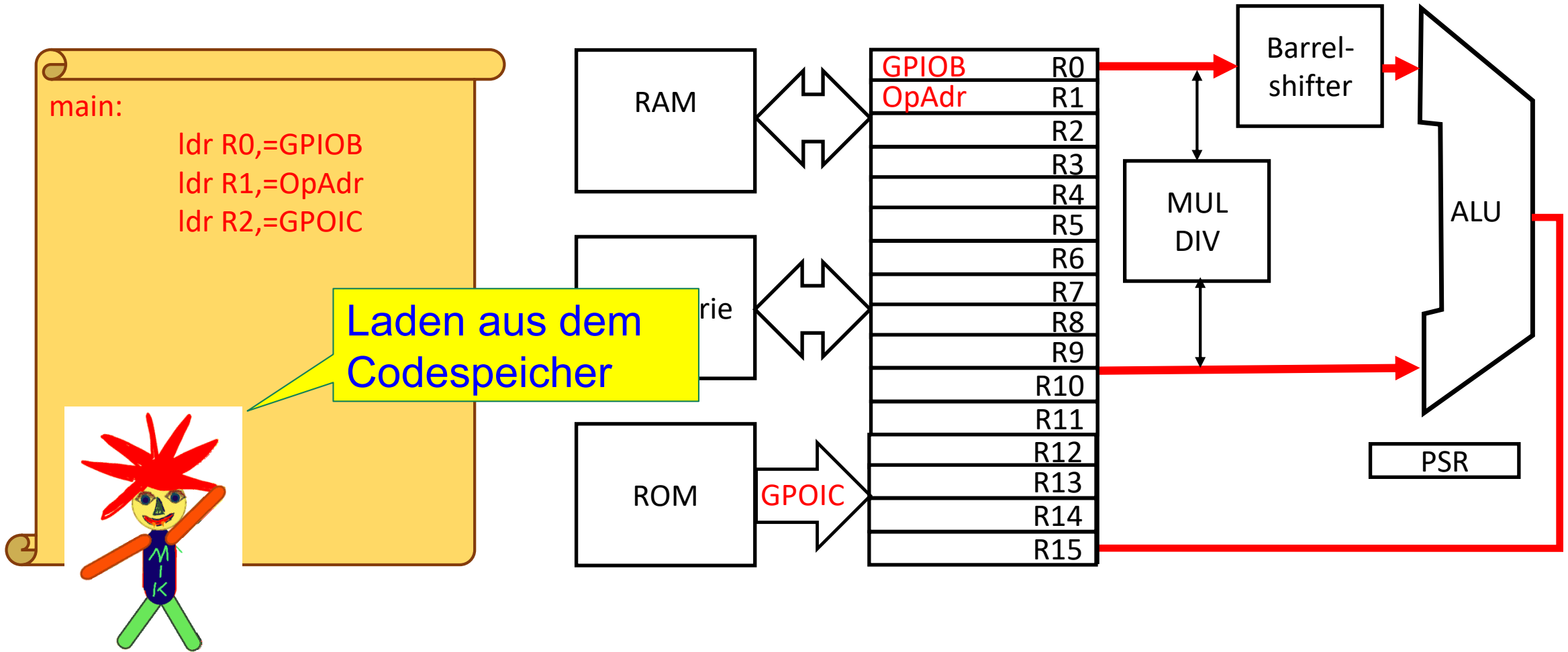
STM32 Load-Store-Architektur



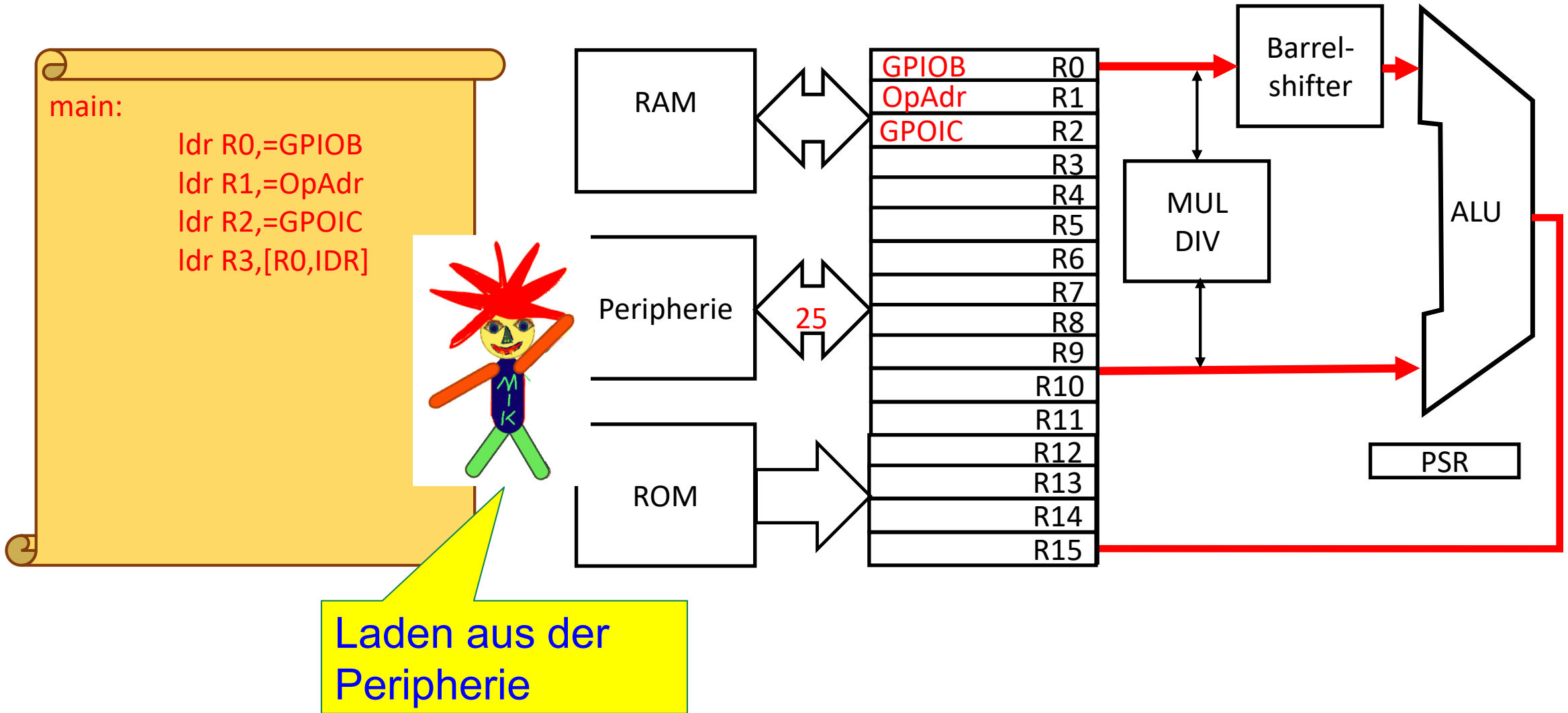
STM32 Load-Store-Architektur



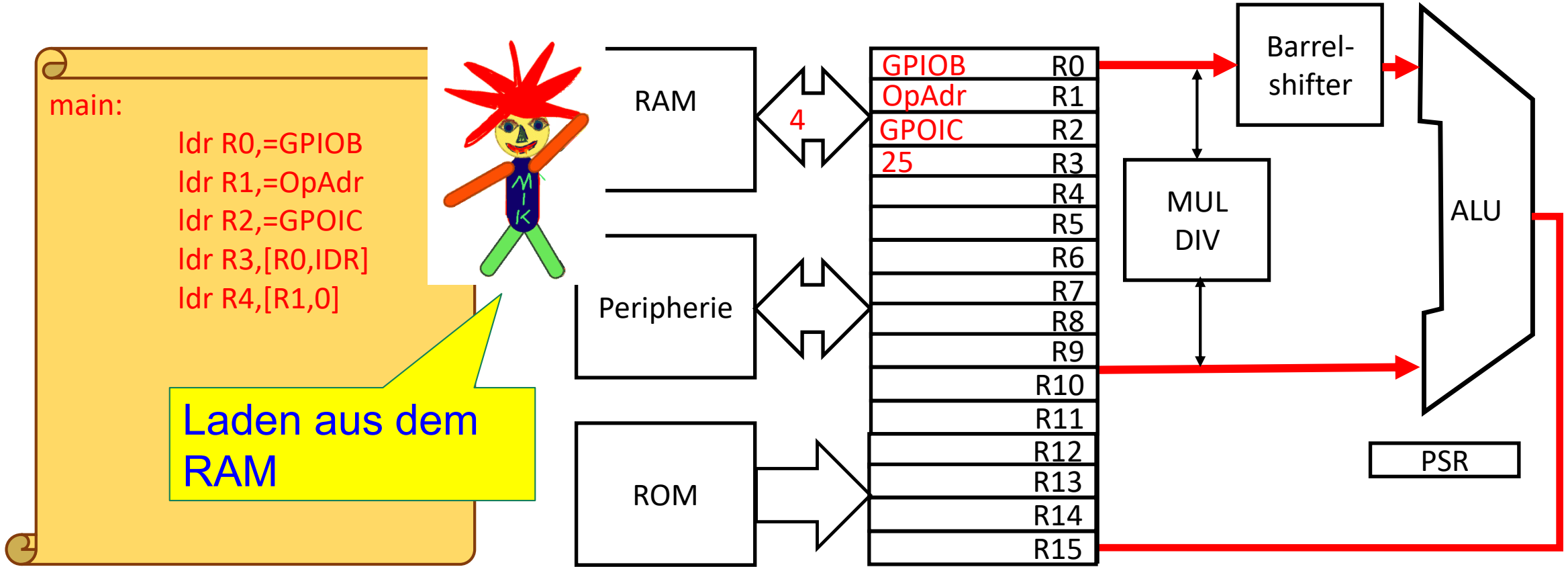
STM32 Load-Store-Architektur



STM32 Load-Store-Architektur



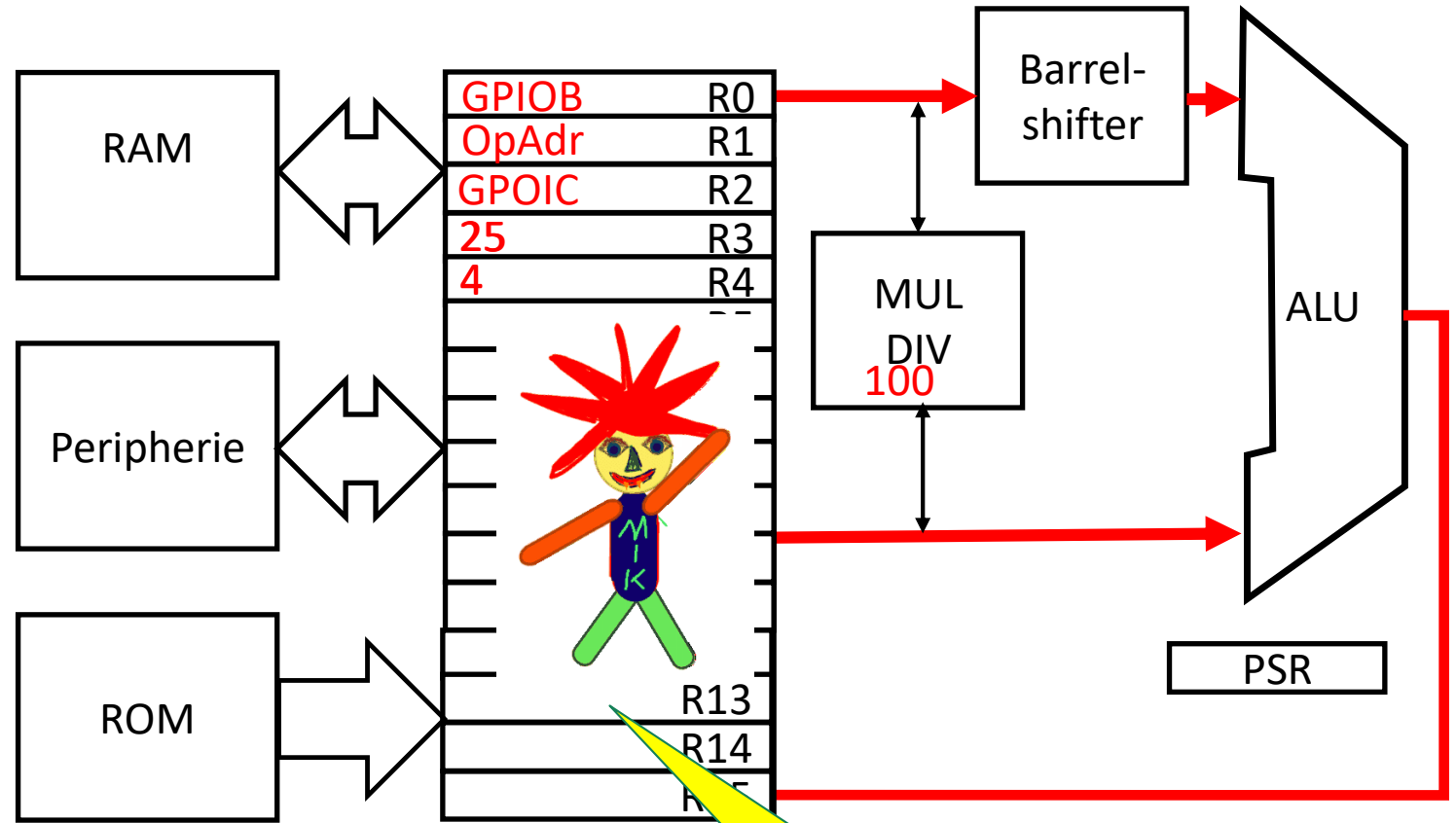
STM32 Load-Store-Architektur



STM32 Load-Store-Architektur

main:

```
ldr R0,=GPIOB  
ldr R1,=OpAdr  
ldr R2,=GPOIC  
ldr R3,[R0,IDR]  
ldr R4,[R1,0]  
mul R5,R3,R4
```



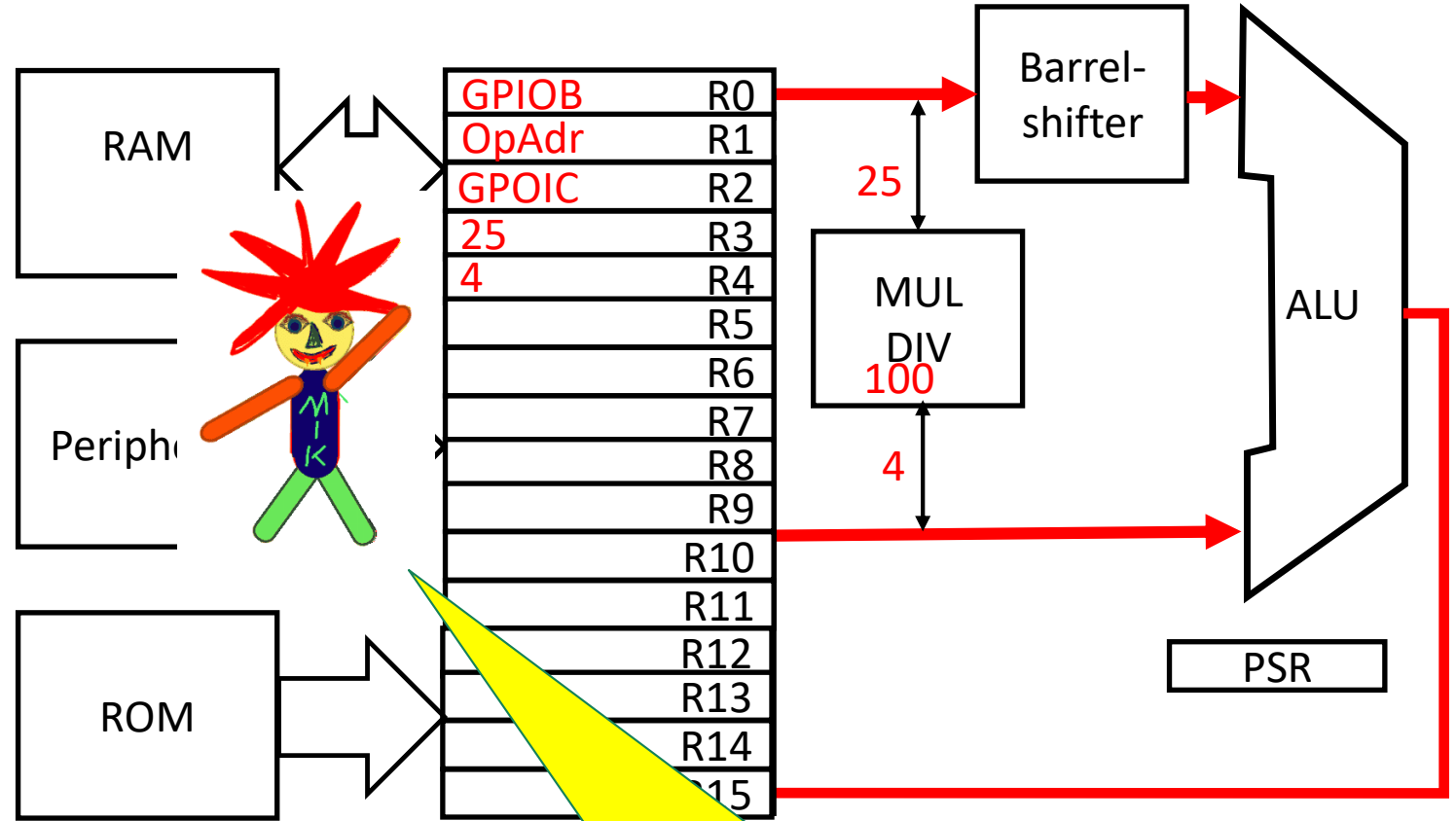
Datenverarbeitung
Multiplikation



STM32 Load-Store-Architektur

main:

```
ldr R0,=GPIOB
ldr R1,=OpAdr
ldr R2,=GPOIC
ldr R3,[R0,IDR]
ldr R4,[R1,0]
mul R5,R3,R4
```



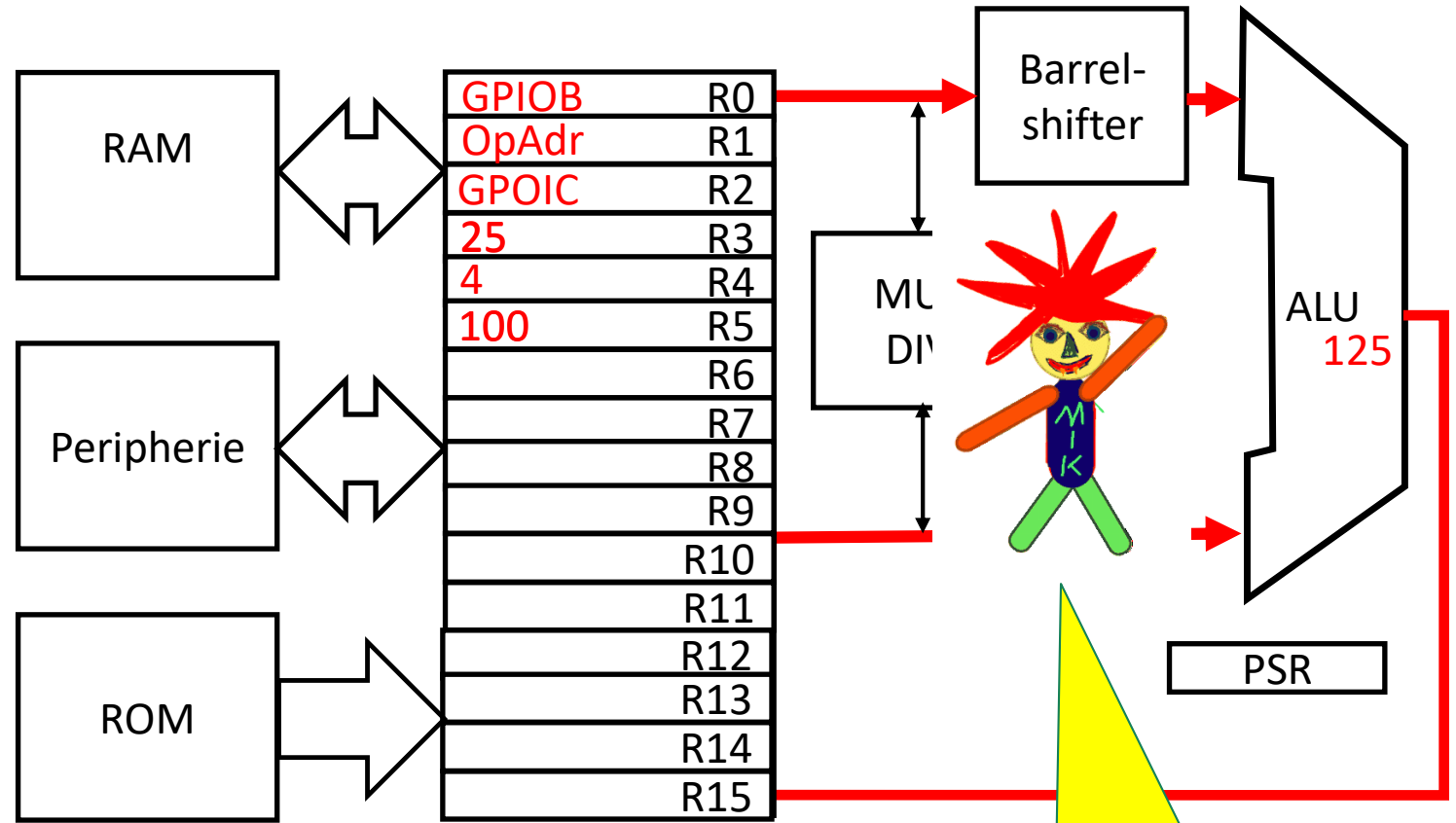
Die Datenverarbeitung erfolgt mit den Registern



STM32 Load-Store-Architektur

main:

```
ldr R0,=GPIOB  
ldr R1,=OpAdr  
ldr R2,=GPOIC  
ldr R3,[R0,IDR]  
ldr R4,[R1,0]  
mul R5,R3,R4  
add R6,R5,R3
```



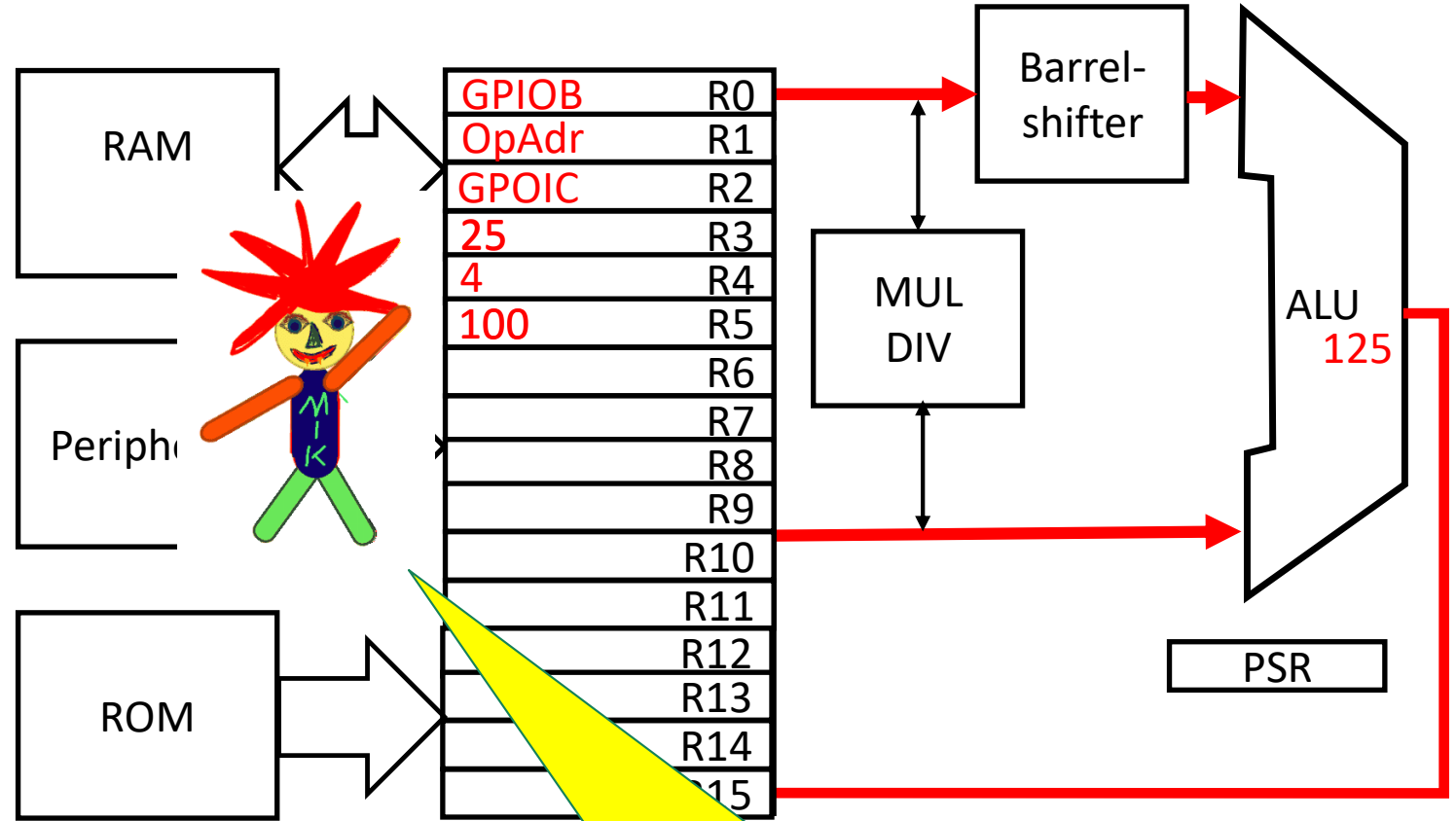
Datenverarbeitung
Addition



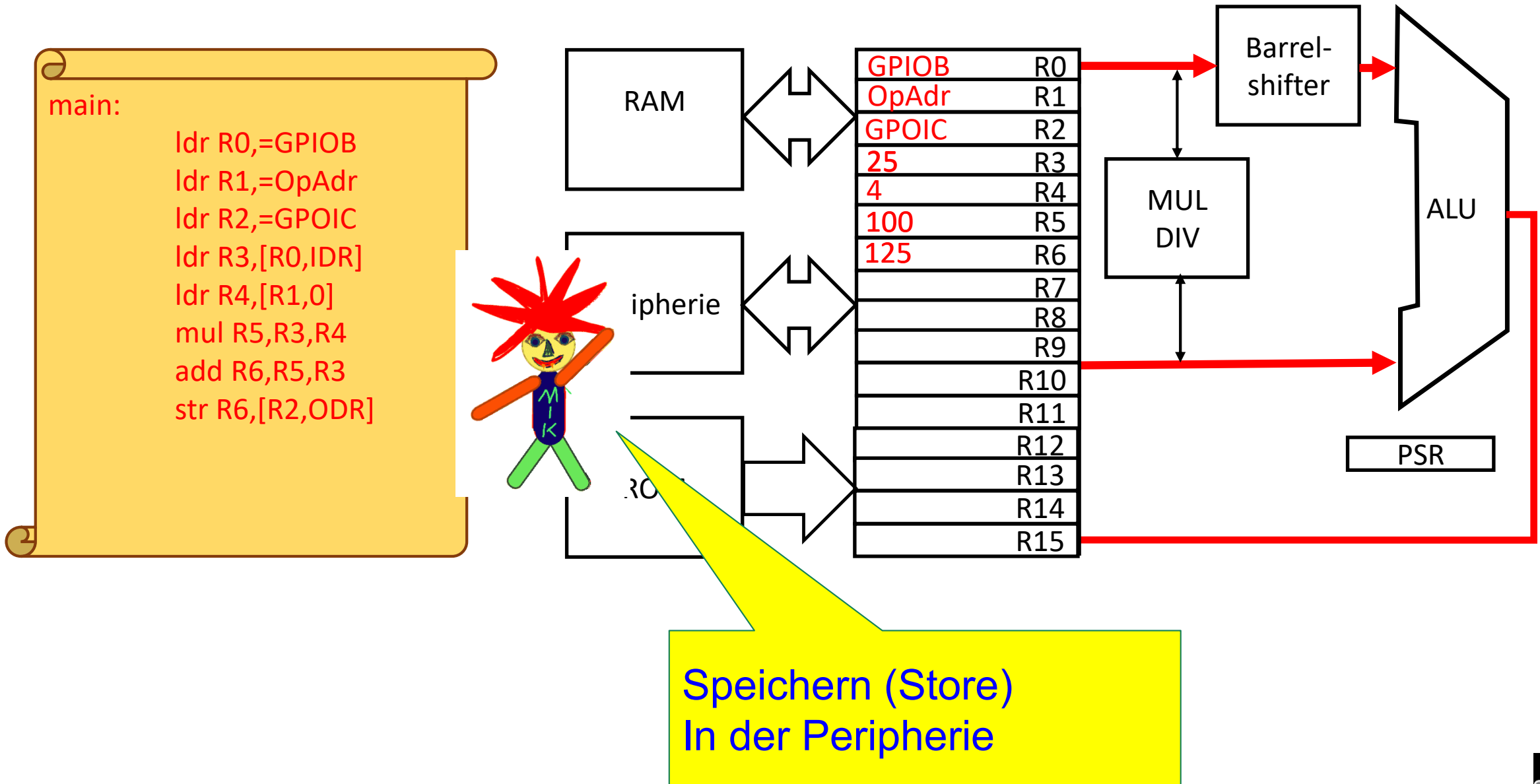
STM32 Load-Store-Architektur

main:

```
ldr R0,=GPIOB  
ldr R1,=OpAdr  
ldr R2,=GPOIC  
ldr R3,[R0,IDR]  
ldr R4,[R1,0]  
mul R5,R3,R4  
add R6,R5,R3
```



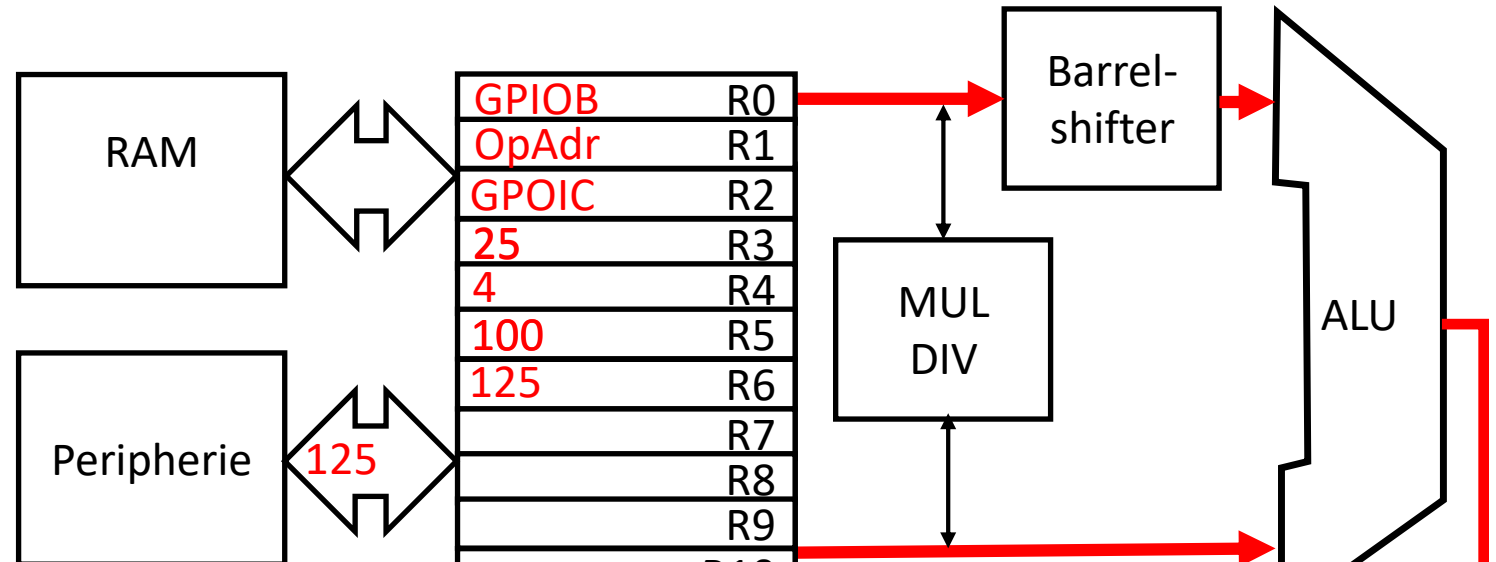
STM32 Load-Store-Architektur



STM32 Load-Store-Architektur

main:

```
ldr R0,=GPIOB
ldr R1,=OpAdr
ldr R2,=GPOIC
ldr R3,[R0,IDR]
ldr R4,[R1,0]
mul R5,R3,R4
add R6,R5,R3
str R6,[R2,ODR]
```



- Bei der Load-Store-Architektur erfolgt die Datenverarbeitung ausschließlich mit den Registern.
- Dazu müssen die Register zunächst mit Werten versorgt werden. (Load)
- Nach der Verarbeitung werden die Register wieder auf die Peripherie oder den Speicher ausgegeben. (Store)

