Gliederung der Vorlesung

1. Einleitung
2. Power Management
3. Parallele I/O
4. Serielle Schnittstelle
    1. I^2C
    2. USART
5. Timer
6. Interrupt Handling
7. Signalverarbeitung
8. Regelungen
    1. Software Interrupt
9. Leitungscodes
10. Weiterführende Themen

Cortex M3 – 3 stage Pipeline

- **Flushing :**

  A flush of the pipeline can occur because of

  - A Branch
  - An exception
  - A breakpoint

- **Branch Pipeline Example:**

Address	Operation	Cycle 1	2	3	4	5	6	7	8	9
0x8000	B 0x8FEC	F	D	E						
0x8002	SUB		F	D						
0x8004	ORR			F						
0x8FEC	AND				F	D	E			
0x8FEE	ORR					F	D	E		
0x8FF0	EOR						F	D	E	

F- Fetch D - Decode E - Execute

Cortex M3 – Programmers Model

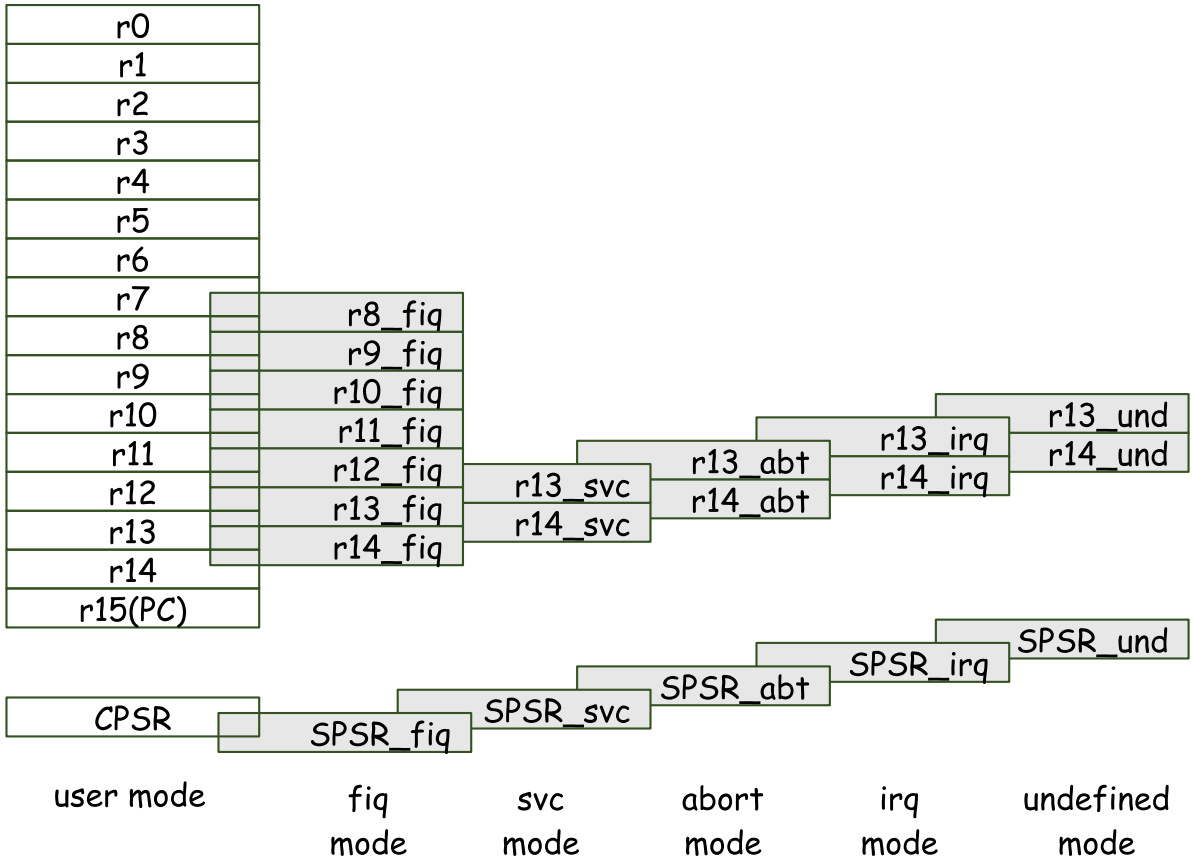
- **Processor modes:**
  - Thread mode
    - Used to execute application software.
    - The processor enters Thread mode when it comes out of reset
  - Handler mode
    - Used to handle exceptions.
    - The processor returns to Thread mode when it has finished exception processing

Cortex M3 – Programmers Model

- **Privilege levels for software execution :**
  - Unprivileged: Software
    - has limited access to the MSR and MRS instructions, and cannot use the CPS instruction
    - cannot access the system timer, NVIC, or system control block
    - might have restricted access to memory or peripherals.
  - Privileged:
    - The software can use all the instructions and has access to all resources.
    - In Thread mode, the CONTROL register controls whether software execution is privileged or unprivileged.
    - In Handler mode, software execution is always privileged.
    - Only privileged software can write to the CONTROL register to change the privilege level for software execution in Thread mode.
    - Unprivileged software can use the **SVC instruction** to make a supervisor call to transfer control to privileged

ARMv4/ARM7TDMI – Register Set und CPU mode

ARMv4/ARM7TDMI AT91

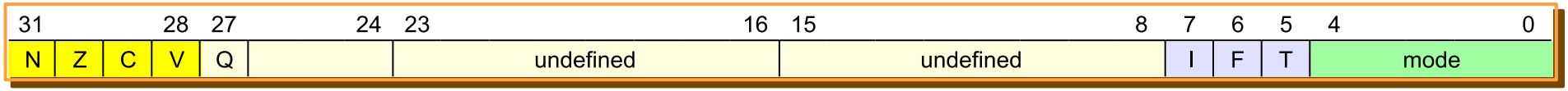


Anwender-programme

Relevant für Exception Handling oder bei der Programmierung auf Systemebene

ARMv4/ARM7TDMI – Register Set und CPU mode

ARMv4/ARM7TDMI AT91



Condition Code Flags

- N = Negatives ALU Ergebnis
- Z = Alu Ergebnis ist Null
- C = Alu Ergebnis erzeugte Carry
- V = Alu erzeugte Overflow

Interrupt Disable Bits

- I = 1, disables IRQ
- F = 1, disables FIQ

T Bit

- T = 0, Prozessor in Arm State
- T = 1, Prozessor in Thumb State

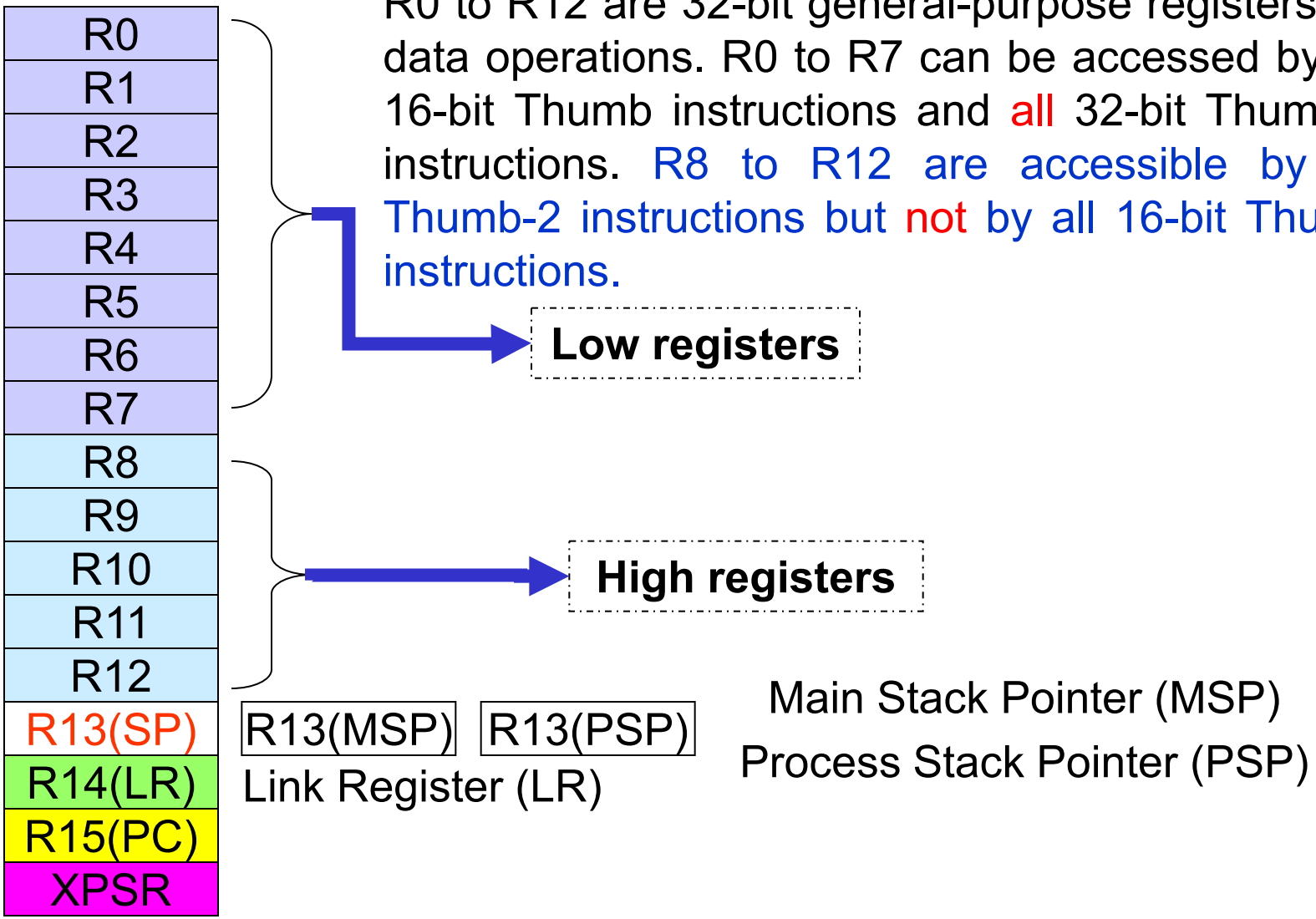
Q Bit

- Cummulative saturation bit for QoS in DSP Context

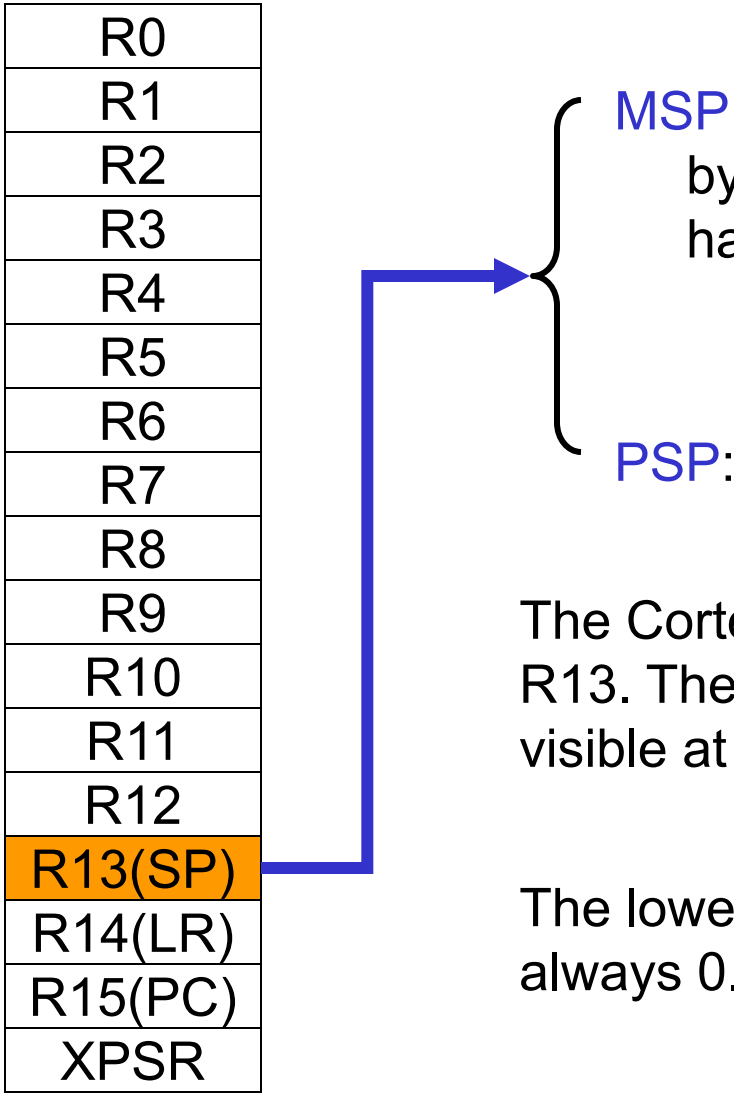
Mode Bits

- Spezifizieren den Prozessor Mode

Cortex M3 – Register Set



Cortex M3 – Register 13 Stack Pointer



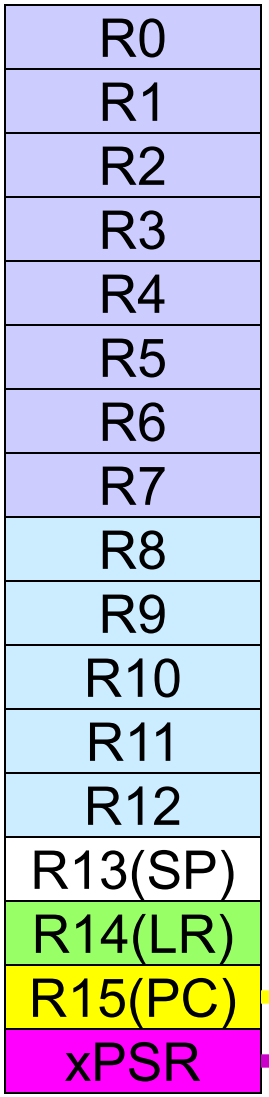
MSP: The default stack pointer; used by the OS kernel and exception handlers

PSP: Used by user application code

The Cortex-M3 contains two stack pointers, R13. They are banked so that only one is visible at a time.

The lowest two bits of the stack pointers are always 0. Word aligned.

Cortex M3 – Register Set



The Program Counter (PC)

The program counter is the current program address. This register can be written to control the program flow.

Special Registers (xPSR)

- Program Status Registers (PSRs)
- Interrupt Mask Registers (PRIMASK, FAULTMASK, BASEPRI)
- Control Register (CONTROL)

Special registers can only be accessed via MSR and MRS instructions; they do not have memory addresses:

```
MRS <reg>, <special_reg>   ; Read special reg.
MSR <special_reg>, <reg> ; write to special
```