2.1.3 Core registers

The processor core registers are:

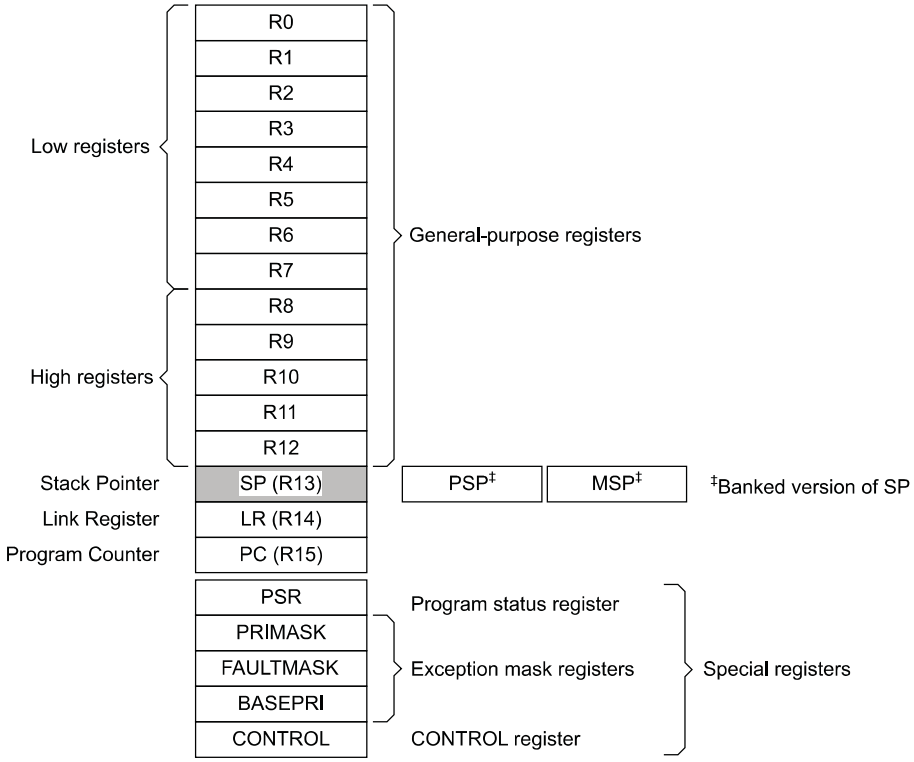


Table 2-2 Core register set summary

Name	Type[a]	Required privilege[b]	Reset value	Description
R0-R12	RW	Either	Unknown	*General-purpose registers* on page 2-4
MSP	RW	Privileged	See description	*Stack Pointer* on page 2-4
PSP	RW	Either	Unknown	*Stack Pointer* on page 2-4
LR	RW	Either	0xFFFFFFFF	*Link Register* on page 2-4
PC	RW	Either	See description	*Program Counter* on page 2-4
PSR	RW	Privileged	0x01000000	*Program Status Register* on page 2-4
ASPR	RW	Either	Unknown	*Application Program Status Register* on page 2-5
IPSR	RO	Privileged	0x00000000	*Interrupt Program Status Register* on page 2-6
EPSR	RO	Privileged	0x01000000	*Execution Program Status Register* on page 2-6
PRIMASK	RW	Privileged	0x00000000	*Priority Mask Register* on page 2-8
FAULTMASK	RW	Privileged	0x00000000	*Fault Mask Register* on page 2-8
BASEPRI	RW	Privileged	0x00000000	*Base Priority Mask Register* on page 2-9
CONTROL	RW	Privileged	0x00000000	*CONTROL register* on page 2-9

a. Describes access type during program execution in thread mode and Handler mode. Debug access can differ.
b. An entry of Either means privileged and unprivileged software can access the register.

General-purpose registers

R0-R12 are 32-bit general-purpose registers for data operations.

Stack Pointer

The *Stack Pointer* (SP) is register R13. In Thread mode, bit[1] of the CONTROL register indicates the stack pointer to use:

- 0 = *Main Stack Pointer* (MSP). This is the reset value.
- 1 = *Process Stack Pointer* (PSP).

On reset, the processor loads the MSP with the value from address `0x00000000`.

Link Register

The *Link Register* (LR) is register R14. It stores the return information for subroutines, function calls, and exceptions. On reset, the processor sets the LR value to `0xFFFFFFFF`.

Program Counter

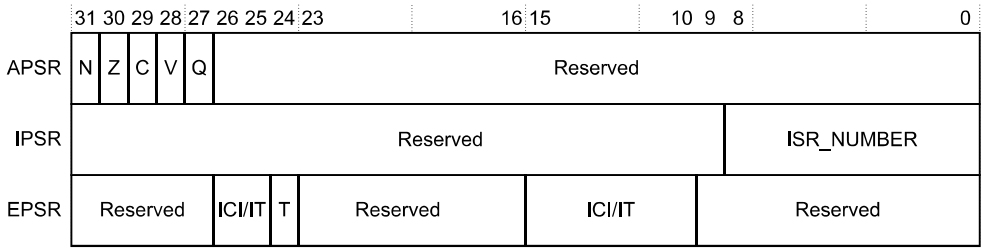
The *Program Counter* (PC) is register R15. It contains the current program address. On reset, the processor loads the PC with the value of the reset vector, which is at address `0x00000004`. Bit[0] of the value is loaded into the EPSR T-bit at reset and must be 1.

Program Status Register

The *Program Status Register* (PSR) combines:

- *Application Program Status Register* (APSR)
- *Interrupt Program Status Register* (IPSR)
- *Execution Program Status Register* (EPSR).

These registers are mutually exclusive bitfields in the 32-bit PSR. The bit assignments are:



Access these registers individually or as a combination of any two or all three registers, using the register name as an argument to the `MSR` or `MRS` instructions. For example:

- read all of the registers using `PSR` with the `MRS` instruction
- write to the APSR N, Z, C, V, and Q bits using `APSR_nzcvq` with the `MSR` instruction.