SR5 E1 line – Register protection overview

Introduction

The purpose of this document is to give an overview of the register protection mechanism (REG PROT) present in the SR5 E1 line products listed in the following table. It explains how to use the REG PROT feature and provides the necessary information and a reference code to use the register protection mechanism.

Table 1. Device list

Device	Device list
SR5E1	SR5E1E3, SR5E1E7

1 Overview

The SR5E1 system architecture presents four peripheral bridges (AHB1, AHB2, APB1, and APB2) that support a hardware mechanism to protect the configuration registers of allocated peripherals on each bridge from unwanted modifications.

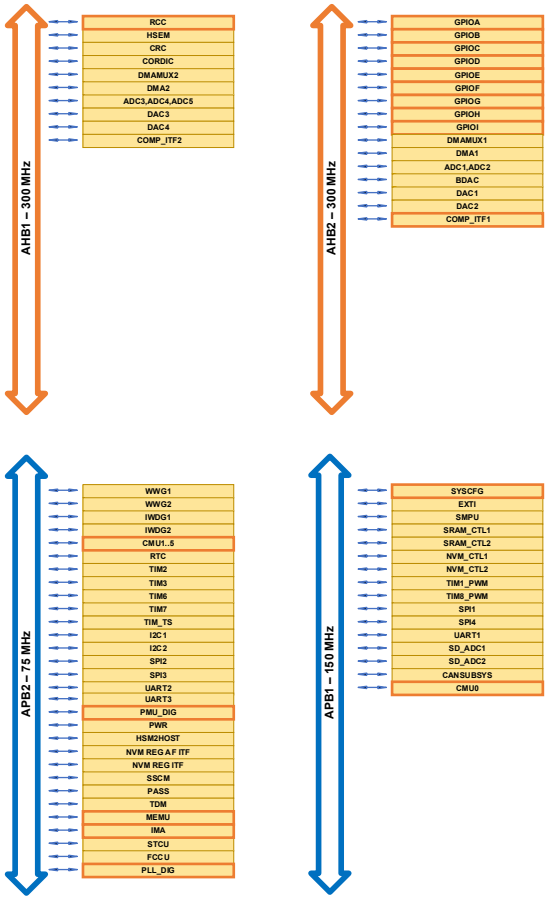
The protection mechanism works like a firewall for the write operations, based on the lock status of the register.

For this reason, the register protection feature is also referred to as register hardware lock.

Note: The registers are protected with a 32-bit granularity.

The following figure shows the allocation of peripherals available on bridges of the SR5E1 device.

Figure 1. SR5E1: peripheral allocations on bridges



2 REG_PROT

2.1 Enabling mechanism

The register protection mechanism (REG PROT) is not enabled by default.

The feature is activated by programming a dedicated device configuration format record (REG_HWLOCK_DCF) in the UTEST DCF records area. An enable bit is available for each AHB and APB bridge, giving the possibility to lock APB1, APB2, AHB1, and AHB2 independently.

During the system boot, the device configuration is loaded, and the protection mechanism is actually activated.

Note: The REG_HWLOCK_DCF is programmable only once. Therefore, if enabled, the register protection mechanism cannot be disabled.

The enabling of the register protection mechanism on a peripheral bus bridge activates the protection for all the peripherals on the bus, which have the mechanism available. The following table shows the actual protected peripherals on each bridge of the SR5E1 device. Refer to the section “REG_PROT configuration” of the SR5E1 reference manual (see Reference documents) for the whole list of protected registers for each IP.

Table 2. SR5E1: protected peripherals on bridges

REG_HWLOCK DCF			
HWLOCK4 APB Bridge2	HWLOCK3 APB Bridge1	HWLOCK2 AHB Bridge2	HWLOCK1 AHB Bridge1
SYSCFG CMU0	CMU1..5 MEMU PMU_DIG PLL_DIG IMA	GPIOx (x = A…I)	RCC

2.2 Two-step writing mechanism

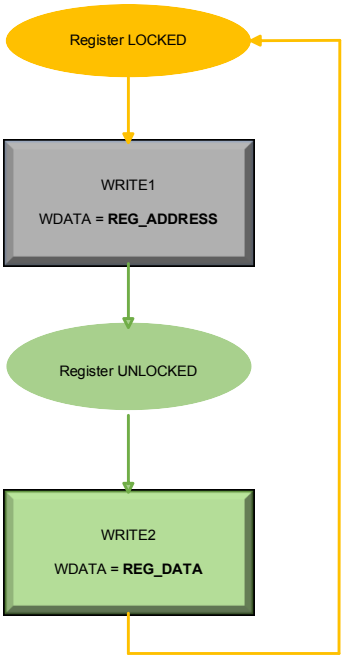
The register protection mechanism (REG PROT) has the main goal to protect registers from unwanted modification. This means modifications are possible only using a special procedure.

The modification of a protected register consists of a two-step write operation.

1. The first write (**WRITE1**), to unlock the register for write, requires providing as data the actual address of the register.
2. The second write (**WRITE2**) is the effective write to the register with the desired data.

Note: As shown in Figure 2. Two-step writing mechanism, the lock of the register is reactivated right after the WRITE2 takes effect.

Figure 2. **Two-step writing mechanism**



Note: To unlock the register protection mechanism, the writing of the register address and the writing of the actual data must be done by the same bus master. An erroneous unlock procedure leads to a hard fault error.

2.3 Usage example

The example below shows how to configure the REG PROT mechanism on the APB2 bridge and how to modify one register (EXTICR_3_0) of one peripheral (SYSCFG) connected to this bus.

Table 3. **Example: SYSCFG.EXTICR_3_0 configuration**

SYSCFG register name	Address	Value
EXTICR_3_0	0x42000008	0x1 = EXTI0 mux out is PA0 (default value) 0x2 = EXTI0 mux out is PB0

To configure the register protection mechanism for the APB2 bridge, the user must program the REG_HWLOCK_DCF as follows:

- **REG_HWLOCK_DCF.HWLOCK4 = 0x1,**

The write access to the configuration registers of peripherals SYSCFG and CMU0 is already active after the system boot.

To activate the REG PROT mechanism, the user must issue a system reset. Just after the system reset, the SYSCFG and CMU0 peripherals are immediately protected from unwanted modification.

In order to change the value of the register the two-step write mechanism must be used:

WRITE1:

```
/* Register locked: register address required as data. */
```

```
SYSCFG.EXTICR_3_0 = SYSCFG_BASE + 0x08U;
```

WRITE2:

```
/* Register unlocked: write access granted. */
```

```
SYSCFG.EXTICR_3_0 = 0x2U;
```

The next write in the register to program a different value will lead to a hard fault error.

```
/* Register locked: write access results in an error. */
```

```
SYSCFG.EXTICR_3_0 = 0x1U;
```

Revision history

Table 4. **Document revision history**

Date	Revision	Changes
31-Jul-2023	1	Initial release.

Contents