NXP Semiconductors
Engineering Bulletin

Document Number: EB00855
Rev. 0, May 2017

Programming Reserved Register Bits for MPC55xx, MPC56xx, and MPC57xx Families

Contents

by: NXP Semiconductors

1 Introduction

This engineering bulletin provides guidance on programming reserved register bits for the MPC55xx, MPC56xx, and MPC57xx families of devices.

2 Programming reserved register bits

Certain device register bits are marked as reserved and should not be modified. Since registers are always written completely and it is not possible to write code that only modifies specific bits, when writing to registers which contain one of more reserved bits, you must do one of the following:

1. Write back the reset value of a reserved bit as documented in the reference manual.

   OR

2. Write back the value which has just been read from that bit of the register.

For example, consider the following register:

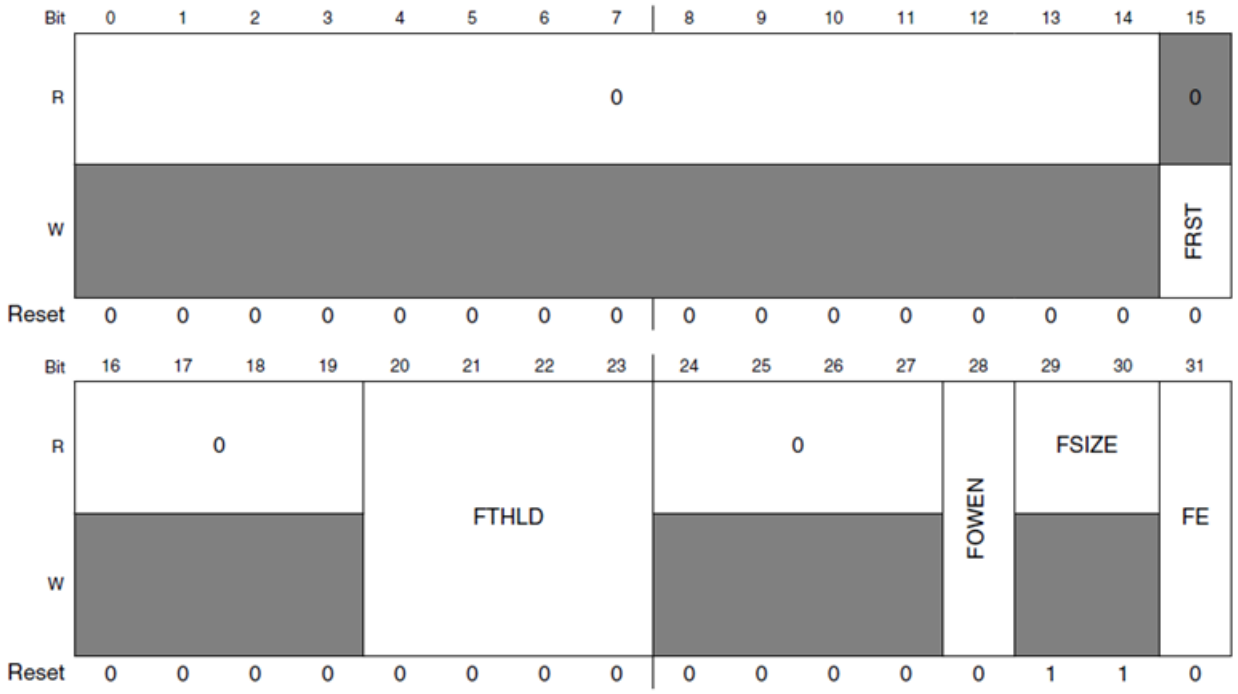

Figure 1. Register with reserved bits

To set FRST to 0x0, FTHLD to 0x4, FOWEN to 0x0, and FE 0x1, you can:

- Use option 1 and fill in the reserved bits with their reset value, thus writing 0x0000_0407 to the register.

  C code representation: register.R = 0x00000407; // write reset value to reserved bits.

- Use option 2 and fill in the reserved bits with the value that has just been read. First, you read the value of the register, suppose it reads as 0x0000_0006. Then, OR it with the bits you want to set, thus writing 0x0000_0407 to the register.

  C code representation: register.R |= 0x00000401; // OR bits to be set with current value

While either of these two options work in most cases, you must always do as specified in the reference manual. The reference manual notes any register bits that require special handling.

Write 1 to clear (w1c) bits also require special attention. Some registers such as interrupt status registers might contain multiple w1c bits along with reserved bits. Writing back the value you just read from a set w1c bit will clear it. Therefore, you must write 0 to any w1c bits you don't intend to clear.

For example, consider the following register with four w1c status bits:

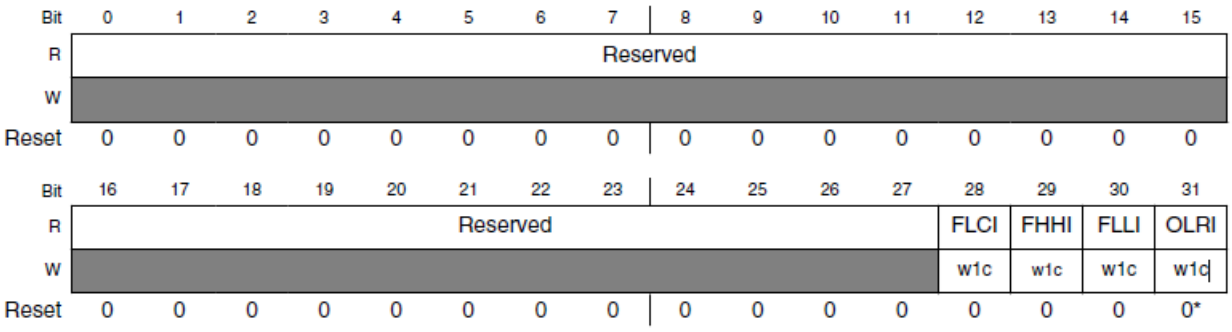

Figure 2. Register with four w1c status bits

Suppose the register currently holds the value 0x0000_000F and you want to clear bit 30 (FLLI). To do so, you must write 0x0000_0002. Writing this value sets the reserved bits to their reset value and preserve the flags you do not intend to clear.

How To Reach Us

Home Page:

nxp.com

Web Support:

nxp.com/support

EB00855
Rev. 0
May 2017