Freescale Semiconductor
Application Note

Document Number: AN5235
Rev. 1, 03/2016

General C Functions for the REACM2

Covers MPC5777C, MPC5746R, and All REACM2-Equipped Devices

by: Marketa Venclikova

Contents

1 Introduction

The Reaction Module 2 (REACM2) is a dedicated peripheral, designed to control 10 output channels, each channel having three output pins. The REACM2 module receives ADC results either from the eQADC directly, or via the BCTU. Based on the results of ADC conversion the REACM2 drives output signals. In this way closed loop control functionality can be implemented, without any CPU load. A typical REACM2 applications is peak and hold injection current control.

2 Functional overview

The REACM2 utilities reacm2_utils.c/.h comprises of a set of low-level functions for use with the Reaction Module 2. These functions provide the user with an API and eliminate the need for direct access to REACM2 registers, which eases module initialization and modulation control.

Routines can be classified into several groups according to application usage:

- Routines for complex initialization (using init structures)
  - fs_reacm2_module_init
  - fs_reacm2_channel_init
- Run-Time REACM2 module control
  - fs_reacm2_module_enable
    fs_reacm2_module_disable

    - fs_reacm2_hold_timer_enable, fs_reacm2_hold_timer_disable
    - fs_reacm2_shared_timer_enable, fs_reacm2_shared_timer_disable
    - fs_reacm2_clear_adc_ovr_flag
- Run-Time REACM2 channel control
    - fs_reacm2_channel_enable, fs_reacm2_channel_disable
    - fs_reacm2_channel_sw_start, fs_reacm2_channel_sw_stop
- Routines for single module settings (without using init structures)
    - fs_reacm2_set_hold_timer_prs,
    - fs_reacm2_set_shared_timer_prs
    - fs_reacm2_period_generator_config
    - fs_reacm2_period_shift_delay_set
    - fs_reacm2_adc_max_limit_set
    - fs_reacm2_set_modulation_range
    - fs_reacm2_threshold_router_set
- Routines for modulation parameters settings
    - fs_reacm2_modword_set
    - fs_reacm2_modword_bank_status
    - fs_reacm2_modword_countpointer_thrs
    - fs_reacm2_modword_countpointer_shtime
    - fs_reacm2_modword_countpointer_holdtime
- Routines for single threshold and timer settings
    - fs_reacm2_threshold_bank_status
    - fs_reacm2_threshold_set_reg
    - fs_reacm2_sh_time_bank_status
    - fs_reacm2_sh_time_set_reg
    - fs_reacm2_hold_time_bank_status
    - fs_reacm2_hold_time_set_reg

3 Functional description

To start with, Reaction Module 2 has to be initialized by function fs_reacm2_module_init() having REACM2_INIT_T type structure as an input parameter, where all the global parameters for REACM2 initialization are specified. The user can find macro extensions defining these parameters in file reacm2_util.h. After module initialization is complete, each particular REACM2 channel should be initialized. Prior to the first channel initialization, at least one Modulation Control Word has to be set in Modulation Control Word Bank, since one of the input parameters of the routine fs_reacm2_channel_init() is the index of the first Modulation Control Word. The Modulation Control Word specifies the type of modulation that is to be executed on that particular channel. The Modulation Control Word can also be set by calling fs_reacm2_modword_set(), where the input parameter is a pointer to the structure of type MODULATION_WORD_T. After the channel is initialized, it can be enabled by function call fs_reacm2_channel_enable(). In case of software control modulation, the channel starts the modulation as soon as the fs_reacm2_channel_sw_start function() is called and stopped after fs_reacm2_channel_sw_stop() function call. If the channel is set in Timer Control mode, modulation is initiated as soon as the time window appears on the selected timer input.

At runtime, there is the possibility to change module settings using “Routines for single module settings”, or store threshold and timer values into respective banks using “Routines for single threshold and timer settings”.

A detailed description of each REACM2 utility functions can be found in the source code. To ease reading of the documentation, it is generated using DoxyGen in Microsoft Compiled HTML Help (CHM) format.

4 REACM2 initialization example

Users can find an example of the REACM2 utilities usage in reacm2_example.c file which is part of the REACM2 utilities software package.

When an application uses REACM2 utilities, the reacm2_util.h file has to be included in the main application source file. The user has to create two configuration structures, see the reacm2_example.c file. One struture is used to set the overall module parameters, the second one contains channel parameters. The user then has to define a set of thresholds and time variables in form of structures having two members – a value and an index corresponding to its place of storage (the index should not be filled by the user, it is strongly recommended to fill the item with a NAN value and let the indexing to be done automatically). Modulation Control Word definitions are in form of an array of type MODULATION_WORD_T, defining the parameters of the modulation. Definition of the structure types can be found in file reacm2_util.h.

To initialize the Reaction Module, function fs_reacm2_module_init() is called, followed by fs_reacm2_channel_init(). Modulation Control Words storage to REACM2 memory (Modulation Control Word Bank) is performed by fs_reacm2_modword_set() which is called as one of the input parameters of function fs_reacm2_channel_init(). The respective channel (set in timer control mode) is then enabled by function fs_reacm2_channel_enable() and modulation on this channel is started every time the time window appears on the REACM2 input.

5 Summary

This application note serves as an overview of utility functions for Reaction Module 2 that provide ease of implementation when utilizing the REACM2. REACM2 utilities can only be used when programming a device equipped with a REACM2 module. These routines are not backward compatible with the first REACM module.

How to Reach Us:

Home Page:
freescale.com

Web Support:
freescale.com/support

Document Number AN5235
Revision 1, 03/2016