ROCKSAVAGE
TECHNOLOGY

# Gpio
# Product User Guide

rocksavagetech.chiselWare.Gpio

IPF certified to level: **0** of 5

Abdelrahman Abbas, Ahmed Elmenshawi, Nick Allison, Jimmy Bright

October 16, 2024

# Contents

# 1 Errata and Known Issues

## 1.1 Errata

None.

## 1.2 Known Issues

None.

# 2 Overview

## 2.1 Features

- Configurable PWM Output
- Timer/Counter Mode
- Event driven Counter
- Watchdog Timer
- RTC
- Interrupts available in all modes

# 3 Introduction

The hardware timer is a versatile and highly configurable peripheral designed for precise time measurement, event generation, and waveform control. It can be configured as a standard timer, RTC, or Watchdog Timer. Each mode generates differnt hardware and registers to cut down on unused registers.

# 4 Port Descriptions

## 4.1 Timer/Counter Mode

The ports for **Timer/Counter** are shown below in Table 1.

| Port Name | Width | Direction | Description |
|---|---|---|---|
| signalOut | 1 | Output | Signal generated by timer/counter |
| irqOutput | 1 | Output | Sent when interrupt is triggered on the Gpio |

Table 1: Timer/Counter Ports Descriptions

## 4.2 Watchdog Timer Mode

The ports for **Watchdog Timer** are shown below in Table 1.

| Port Name | Width | Direction | Description |
|---|---|---|---|
| reset | 1 | Output | System reset signal |

Table 2: WDT Ports Descriptions

## 4.3 Real Time Clock Mode

The ports for **Real Time Clock** are shown below in Table 1.

| Port Name | Width | Direction | Description |
|---|---|---|---|
| irqOutput | 1 | Output | Sent when interrupt is triggered on the Gpio |

Table 3: RTC Ports Descriptions

## 4.4 APB3 Interface

The **APB3 Interface** is a regular APB3 Slave Interface. All signals supported are shown below in Table 2. See the *AMBA APB Protocol Specifications* for a complete description of the signals. The width of several ports is controlled by the following input parameters:

- *dataWidth* is the width of PWDATA and PRDATA in bits
- *addrWidth* is the width of PADDR in bits

| Port Name | Width | Direction | Description |
|---|---|---|---|
| PCLK | 1 | Input | Positive edge clock |
| PRESETN | 1 | Input | Active low reset |
| PSEL | 1 | Input | Indicates slave is selected and a data transfer is required |
| PENABLE | 1 | Input | Indicates second cycle of APB transfer |
| PWRITE | 1 | Input | Indicates write access when HIGH and read access when LOW |

| | | | |
|---|---|---|---|
| PADDR | *addrWidth* | Input | Address bus |
| PWDATA | *dataWidth* | Input | Write data bus driven when PWRITE is HIGH |
| PRDATA | *dataWidth* | Output | Read data bus driven when PWRITE is LOW |
| PREADY | 1 | Output | Transfer ready |
| PSLVERR | 1 | Output | Transfer error |

Table 4: APB Ports Descriptions

# 5 Parameter Descriptions

The parameters for **Gpio** are shown below in Table 3.

| Name | Type | Min | Max | Description |
|---|---|---|---|---|
| dataWidth | Int | 1 | $\leq 32$ | The data width of COUNT, MAX, MIN, PWDATA, and PRDATA. Can be 8, 16, or 32 bits wide |
| addrWidth | Int | 1 | $\leq 32$ | The APB address bus width |
| mode | Int | 1 | $\leq 3$ | Configures the hardware mode.<br>Timer/Counter: 1<br>WDT: 2<br>RTC: 3 |

Table 5: Parameter Descriptions

To instantiate a 16-bit Timer/Counter:

```
// Valid Timer/Counter Instantiation Example
val myTimer = new Timer(
  dataWidth = 16,
  addrWidth = 32,
  mode = 1  )
```

To instantiate a 16-bit Timer/Counter:

```
// Valid Timer/Counter Instantiation Example
val myTimer = new Timer(
  dataWidth = 16,
  addrWidth = 32,
  mode = 1  )
```