SpartanMC

Configurable Parallel Input for 1 to 18 Bit (port_in)

Table of Contents

List of Figures

List of Tables

Configurable Parallel Input for 1 to 18 Bit (port_in)

The input port module provides up to 18 input signals. These inputs can be used for switches, buttons etc.. Furthermore, they can be configured to generate interrupts (triggerd by a raising or falling edge) as SoC input. Each input pin provides seperate configuration bits.

1. Module Parameters

Table 1: PORT_IN module parameters

Parameter	Default Value	Descripton
BASE_ADR		Start address of the memory mapped peripheral registers. The value is taken as offset to the start address of the peripheral memory space. **This parameter is set by jConfig automatically.**
PORT_WIDTH	18	Number of input bits.

2. Interrupts

An interrupt signals will be generated for each enabled PORT_IN bit. To delete the interrupt flag a read access on PIN_IN_DAT, PIN_IN_IE or PIN_IN_EDGSEL is required.

3. Peripheral Registers

3.1. Input Port Register Description

The input port peripheral provides four 18 bit registers which are mapped to the SpartanMC address space e.g. 0x1A000 + BASE_ADR + Offset.

Table 2: PORT_IN registers

Offset	Name	Access	Description
0	PIN_IN_DAT	read	Register for incomming data.

Offset	Name	Access	Description
1	PIN_IN_IE	read/ write	Enables the interrupts on PIN_IN_DAT register. After system reset all PIN_IN_IE bits are initialized with zero.
2	PIN_IN_EDGSEL	read/ write	Specify the input edge which triggers the interrupt (0 = falling edge, 1 = raising edge) After system reset all PIN_IN_EDGSEL bits are initialized with zero.
3	PIN_IN_IR_STATUS	read	Register for interrupt flags. If set to one it indicates an interrupt on the corresponding input pin. The interrupt flag will be deleted with a read access on all other module registers except this one. After system reset all PIN_IN_IR_STATUS bits are initialized with zero.

3.2. PORT_IN C-Header for Register Description

```
#ifndef __PORT_IN_H
#define __PORT_IN_H

#ifdef __cplusplus
extern "C" {
#endif

#ifndef __PORT_IN_F_H

#define PORT_INBIT_0   (1<<0)
#define PORT_INBIT_1   (1<<1)
#define PORT_INBIT_2   (1<<2)
#define   PORT_INBIT_3   (1<<3)
#define PORT_INBIT_4   (1<<4)
#define PORT_INBIT_5   (1<<5)
#define PORT_INBIT_6   (1<<6)
#define PORT_INBIT_7   (1<<7)
#define PORT_INBIT_8   (1<<8)
#define PORT_INBIT_9   (1<<9)
#define PORT_INBIT_10   (1<<10)
#define PORT_INBIT_11   (1<<11)
#define PORT_INBIT_12   (1<<12)
#define PORT_INBIT_13   (1<<13)
#define PORT_INBIT_14   (1<<14)
#define PORT_INBIT_15   (1<<15)
#define PORT_INBIT_16   (1<<16)
#define PORT_INBIT_17   (1<<17)

#endif
```

```
typedef struct port_in {
   volatile unsigned int data;   // (r)   (r = reset-interrupt)
   volatile unsigned int ie;   // (r/w) (r = reset-interrupt)
   volatile unsigned int edgsel;   // (r/w) (r = reset-
interrupt)
   volatile unsigned int ir_stat;   // (r)
} port_in_regs_t;

#ifdef __cplusplus
}
#endif

#endif
```