## OSOROM Peripheral Reference

The Moroso Project

 $\mathrm{July}\ 29,\ 2015$ 

# Contents

1	Onb	poard Peripherals	4
	1.1	Introduction	4
	1.2	Peripheral Reference	4
	1.3	Timer	4
	1.4	Serial Port	٠
	1.5	Video Controller	ļ
	1.6	SD Card Controller	ļ
	1.7	USB Controller	ļ

### Chapter 1

# Onboard Peripherals

### 1.1 Introduction

The OSOROM will include several onboard peripherals:

- A programmable timer
- A serial port
- A video controller
- An SD card controller
- A USB controller?

Access to all peripherals is through memory mapped registers; each peripheral has its own page of physical address space for control registers.

### 1.2 Peripheral Reference

#### 1.2.1 Register Map

	TIMER_COUNT	0x80000000
Timer	TIMER_TOP	0x80000004
	TIMER_CONTROL	0x80000008
	SERIAL_BAUD	0x80001000
Serial Port	SERIAL_DATA	0x80001004
	SERIAL_CONTROL	0x80001008
Video Controller		0x80002000
SD Controller		0x80003000
USB Controller		0x80004000

#### 1.3 Timer

Each clock tick, the TIMER\_COUNT register is incremented. If this value matches the value in the TIMER\_TOP register, TIMER\_COUNT is cleared and the TIMER\_INT bit in TIMER\_CONTROL is set.

#### 1.3.1 Registers

#### TIMER\_COUNT

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TIMER_COUNT														
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIMER_COUNT														
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

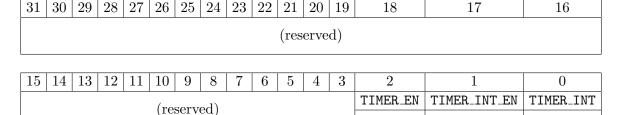
Bits 31:0 TIMER\_COUNT: The current count, incremented each tick and reset to 0 when it matches the value in TIMER\_TOP.

#### TIMER\_TOP

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TIMER_TOP														
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIMER_TOP														
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:0 TIMER\_TOP: The top value for the timer. When TIMER\_COUNT is equal to this value, TIMER\_COUNT will be reset to 0 and the TIMER\_INT bit of TIMER\_CONTROL will be set.

#### TIMER\_CONTROL



rw

rw

rw

- Bit 2 TIMER\_EN: Timer enable bit. The counter will be paused if this bit is set to 0.
- Bit 1 TIMER\_INT\_EN: Timer interrupt enable. If this bit and TIMER\_INT are both set, a timer interrupt will be generated.
- Bit 0 TIMER\_INT: Timer interrupt flag. Set by hardware when TIMER\_COUNT is equal to TIMER\_TOP.

#### 1.4 Serial Port

The serial port contains two internal 8-bit registers, the receive and transmit buffer registers. These cannot be accessed directly, but instead are accessed through the DATA register. When a value is written to the DATA register, it is written through to the transmit buffer. If a transmission is not in progress, a transmission will begin with the value, and the TX\_EMPTY bit

will be set to indicate that the transmit buffer is ready for another write. If a value is written to DATA and a transmission is in progress, the value will be written to the transmit buffer and the TX\_EMPTY bit will be cleared. When the transmission ends, if TX\_EMPTY is clear, a new transmission will begin with the value in the transmit buffer, and the TX\_EMPTY bit will be set.

The TX\_COMPLETE bit will also be set.

Reads from the DATA register read the value in the receive buffer. When a byte is received, the value is stored in this buffer and the RX\_COMPLETE bit is set.

#### 31 30 29 28 27 26 $24 \mid 23$ 20 19 252218 17 16 SERIAL\_BAUD rw rwrw rwrw rw | rw | rw | rw | rw rw | rw rw rw rw

SERIAL\_BAUD

7

6

rw

5

rw rw

 $4 \mid 3$ 

rw

2

rw

1

rw | rw

0

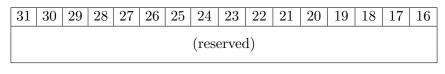
#### SERIAL\_BAUD

Bits 31:0 SERIAL\_BAUD: A value that will determine the baud rate according to some formula we'll figure out later.

8

rw | rw | rw

#### SERIAL\_DATA



	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	(reserved)											DA	TA			
									rw							

Bit 8 P: Reserved for use as a parity bit. Reads as 0.

Bits 7:0 DATA: Data register. See description above.

15 | 14

rw | rw

13

rw

12 | 11

rw

rw

10

rw

#### SERIAL\_CONTROL

31	 22	21	20	19	18	17	16
				(reserved)			

15	$\begin{array}{c c} 15 & \cdots & 6 \\ \hline \text{(reserved)} \end{array}$		5	4	3	2	1	0	
(r			RXC_INT_EN	RX_COMPLETE	TXE_INT_EN	TX_EMPTY	TXC_INT_EN	TX_COMPLETE	
(1			rw	rw	rw	rw	rw	rw	

Bit 5 RXC\_INT\_EN Interrupt enable for RX\_COMPLETE.

Bit 4 RX\_COMPLETE Set by hardware when a receive is complete. Should be cleared by software when the value in DATA is read.

Bit 3 TXE\_INT\_EN Interrupt enable for TX\_EMPTY.

- Bit 2 TX\_EMPTY Cleared by hardware when a value is written to DATA and a transmission is in progress. Set by hardware when a transmission begins.
- Bit 1 TXC\_INT\_EN Interrupt enable for TX\_COMPLETE.
- Bit 0 TX\_COMPLETE Set by hardware when a transmission completes.
  - 1.5 Video Controller
  - 1.6 SD Card Controller
    - 1.7 USB Controller