

ELMA I²C to Wishbone bridge

June 25, 2013



Theodor-Adrian Stana (CERN/BE-CO-HT)

Revision history

Date	Version	Change
25-06-2013	1.00	First version

Contents

1	Introduction	1
2	ELMA I²C Protocol	1
3	Implementation	2

List of Figures

1	SysMon write operation	1
2	SysMon read operation	2
3	Main FSM of <i>elma_i2c</i> module	2
4	FSM states when the SysMon writes to the <i>elma_i2c</i>	4
5	FSM states when the SysMon reads from the <i>elma_i2c</i>	4

List of Tables

1	States of <i>elma_i2c</i> FSM	3
---	---	---

List of Abbreviations

FSM	Finite-State Machine
I ² C	Inter-Integrated Circuit (bus)
SysMon	ELMA crate System Monitor board
VME	VERSAmodule Eurocard

3 Implementation

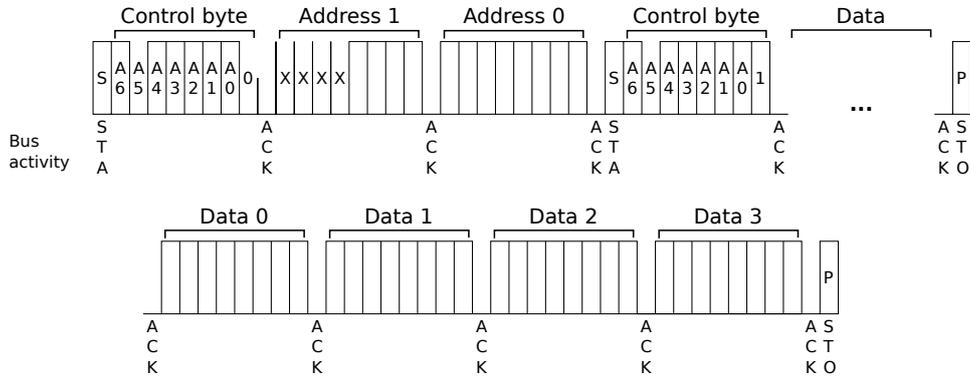


Figure 2: SysMon read operation

3 Implementation

In order to perform low-level I²C transfers, the *i2c_slave* module is instantiated and used within the *elma_i2c* module. The outputs of the *i2c_slave* module [REFERENCE?](#) are used as controls for an eight-state finite state machine (FSM), a simplified version of which is shown in Fig. 3. Table 1 also lists the states of the state machine.

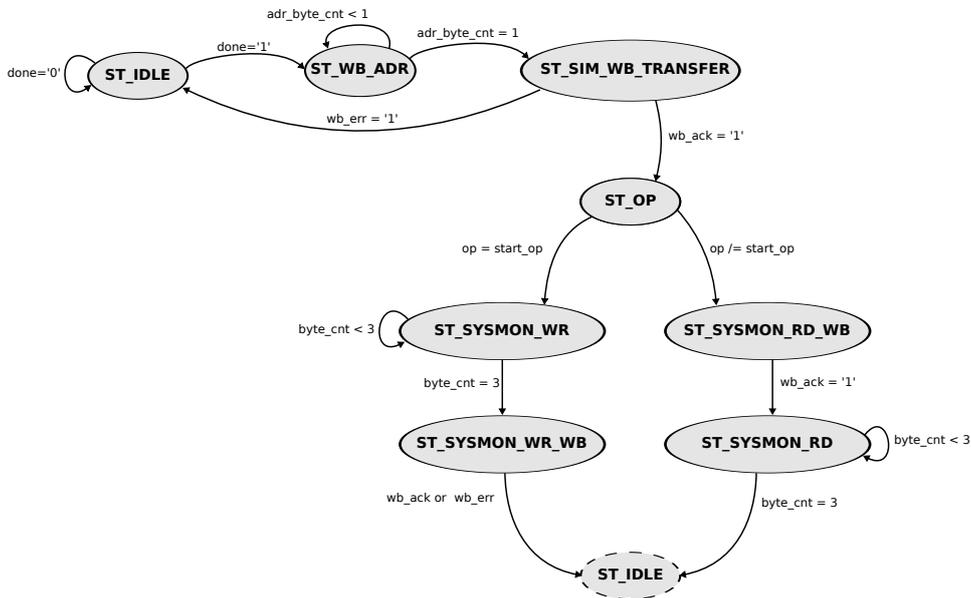


Figure 3: Main FSM of *elma_i2c* module

When the *i2c_slave* module finishes a transfer (signaled by a *done_p_o* pulse), the status is checked and if it is as expected (e.g., a *address good* in the *ST_IDLE* state), the FSM advances to the next state. It should be noted

3 Implementation

that where the SysMon appears in the state names, it indicates what the SysMon action is. For example, if the state of the FSM is *ST_SYSMON_WR*, this means the SysMon is writing and the *elma_i2c* is reading.

Table 1: States of *elma_i2c* FSM

State	Description
ST_IDLE	Wait for the <i>i2c_slave</i> module to receive the I ² C address and go to <i>ST_WB_ADR</i> . The starting value at the <i>op_o</i> output of the <i>i2c_slave</i> module is stored for checking in <i>ST_OP</i>
ST_WB_ADR	Shift in the two address bytes sent via I ² C and go to <i>ST_SIM_WB_TRANSF</i>
ST_SIM_WB_TRANSF	Start a Wishbone read transfer from address received in previous state and go to <i>ST_OP</i> if Wishbone address exists (Wishbone <i>ack</i> received), or <i>ST_IDLE</i> otherwise (Wishbone <i>err</i> received)
ST_OP	Check the <i>op_o</i> output of the <i>i2c_slave</i> module. If different from the value at the start, go to <i>ST_SYSMON_RD_WB</i> state (SysMon is reading from <i>elma_i2c</i>), otherwise continue shifting in bytes (SysMon writing to <i>elma_i2c</i>)
ST_SYSMON_WR	Continue reading up to four bytes sent by the SysMon and go to <i>ST_SYSMON_WR_WB</i>
ST_SYSMON_WR_WB	Perform a Wishbone write transfer to the register with the address obtained in <i>ST_WB_ADR</i>
ST_SYSMON_RD_WB	Perform a Wishbone read transfer from the address obtained in <i>ST_WB_ADR</i> and go to <i>ST_SYSMON_RD</i>
ST_SYSMON_RD	Shift out the four bytes of the Wishbone register when the <i>i2c_slave</i> module successfully finishes a write

To better understand how the FSM operates, Figures 4 and 5 can be consulted, where the state of the FSM is shown during reads and writes from the SysMon.

When reading from the SysMon (Fig. 4), the *elma_i2c* module will wait in the *ST_IDLE* state while the I²C address is sent, then go to the *ST_WB_ADR* state to shift in the address. A Wishbone transfer is simulated and if the address exists (a Wishbone *ack* is received), the first byte is shifted in while in the *ST_OP* state, followed by the next three bytes while in the *ST_SYSMON_WR* state. After that, the register is written to in the *ST_SYSMON_WR_WB* state.

When writing to the SysMon (Fig. 5), the first few steps are the same as for reading from it. The address is shifted in and checked in the Wishbone

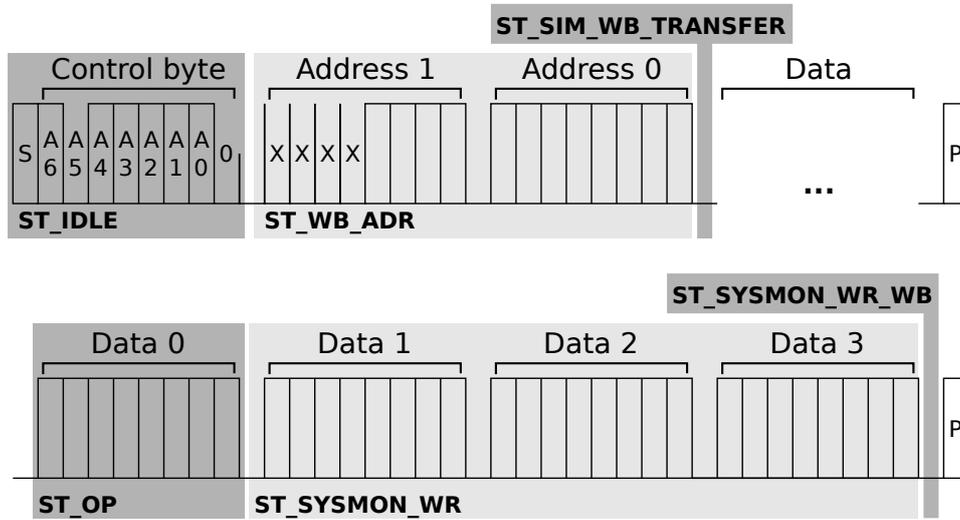


Figure 4: FSM states when the SysMon writes to the *elma_i2c*

transfer simulation state. In the case of a SysMon reading from a board, however, the I²C transfer is restarted and the order is reversed (SysMon starts reading). Thus, while in *ST_OP*, the FSM detects a different value of *op_o* and goes into the *ST_SYSMON_RD_WB* state. Here, an actual Wishbone read transfer is executed, the value of the register is read and sent via I²C in the *ST_SYSMON_RD* state.

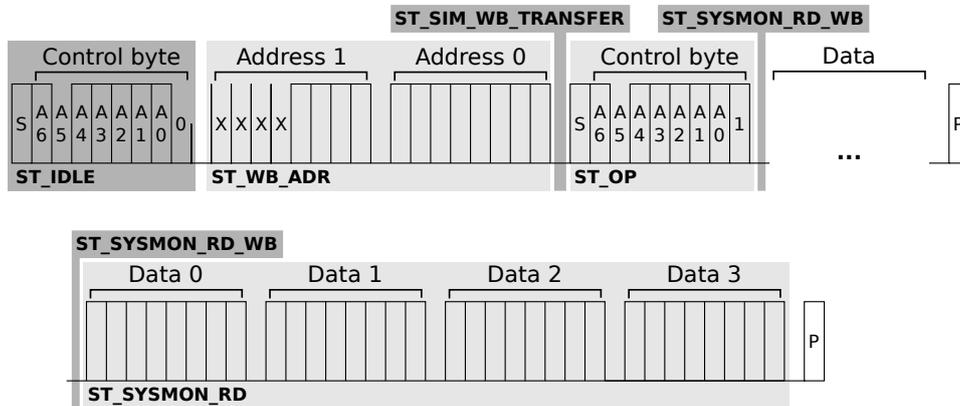


Figure 5: FSM states when the SysMon reads from the *elma_i2c*

References

- [1] ELMA, “Access to board data using SNMP and I2C.” <http://www.ohwr.org/documents/227>.