The OBI to AHB-Lite Master Adaptor is Intrinsix custom logic to translate an OBI transaction to an AHB-Lite transaction.

HADDR_M[31:0]

The adapter supports transactions to addresses that are 32 bits wide. Note that this signal is only updated (and valid) when performing an AHB transaction.

HBURST_M[2:0]

Only single bursts (**HBURST_M** = 3’b000) are generated.

HMASTLOCK_M

This signal is tied low (**HMASTLOCK_M** = 1’b0).

HPROT_M[3:0]

Non-bufferable, non-cacheable data accesses are supported (**HPROT_M** = 4’b0011, 4’b0001). The privilege bit (**HPROT_M[1]**) is driven by an input signal that indicates its current privilege mode. This signal is only updated when performing an AHB transaction.

HSIZE_M[2:0]

Byte, half-word, and word transfers are supported (**HSIZE_M** = 3’b000, 3’b001, 3’b010). Note that half-word transfers will be half-word aligned and word transfers will be word aligned. This signal is only updated (and valid) when performing an AHB transaction.

The following table shows the appropriate decoding of **data_be_o** used to generate **HSIZE_M**. **HSIZE_M[2]** will be tied to 0, and for simplicity, invalid transfers will be implemented as **HSIZE_M[1:0]** = 2’b00.

Table 1: HSIZE Decoding of DATA_BE_O

data_be_o				HSIZE_M			Description
Bit 3	Bit 2	Bit 1	Bit 0	Bit 2	Bit 1	Bit 0	
0	0	0	0	0	0	0	No transfer
0	0	0	1	0	0	0	Byte transfer
0	0	1	0	0	0	0	Byte transfer
0	0	1	1	0	0	0	Half-word transfer
0	1	0	0	0	0	0	Byte transfer
0	1	0	1	0	0	0	Invalid transfer
0	1	1	0	0	0	0	Invalid transfer

0	1	1	1		0	0	0		Invalid transfer
1	0	0	0		0	0	0		Byte transfer
1	0	0	1		0	0	0		Invalid transfer
1	0	1	0		0	0	0		Invalid transfer
1	0	1	1		0	0	0		Invalid transfer
1	1	0	0		0	0	1		Half-word transfer
1	1	0	1		0	0	0		Invalid transfer
1	1	1	0		0	0	0		Invalid transfer
1	1	1	1		0	1	0		Word transfer

HTRANS_M[1:0]

Only transfer types IDLE and NONSEQUENTIAL are supported (**HTRANS_M** = 2'b00, 2'b10).

HWDATA_M[31:0]

The OBI side can perform byte, half-word, or word write transfers. For byte lanes that are inactive, their contents are tied to 0. This signal is only updated (and valid) when performing an AHB transaction.

HWRITE_M

The **HWRITE_M** signal asserts for a write transfer (**HWRITE_M** = 1'b1) and de-asserts for a read transfer (**HWRITE_M** = 1'b0).

HRDATA_M[31:0]

The OBI side can initiate byte, half-word, or word read transfers.

HREADY_M

When a transfer has finished on the bus, **HREADY_M** is high (**HREADYOUT_M** = 1'b1). It is driven low (**HREADY_M** = 1'b0) to extend a transfer. If this signal is de-asserted following a bus transaction generated by the processor, the processor will stall.

HRESP_M

HRESP_M is low when a transfer is OKAY (**HRESP_M** = 1'b0). When high, the transfer has an error (**HRESP_M** = 1'b1).

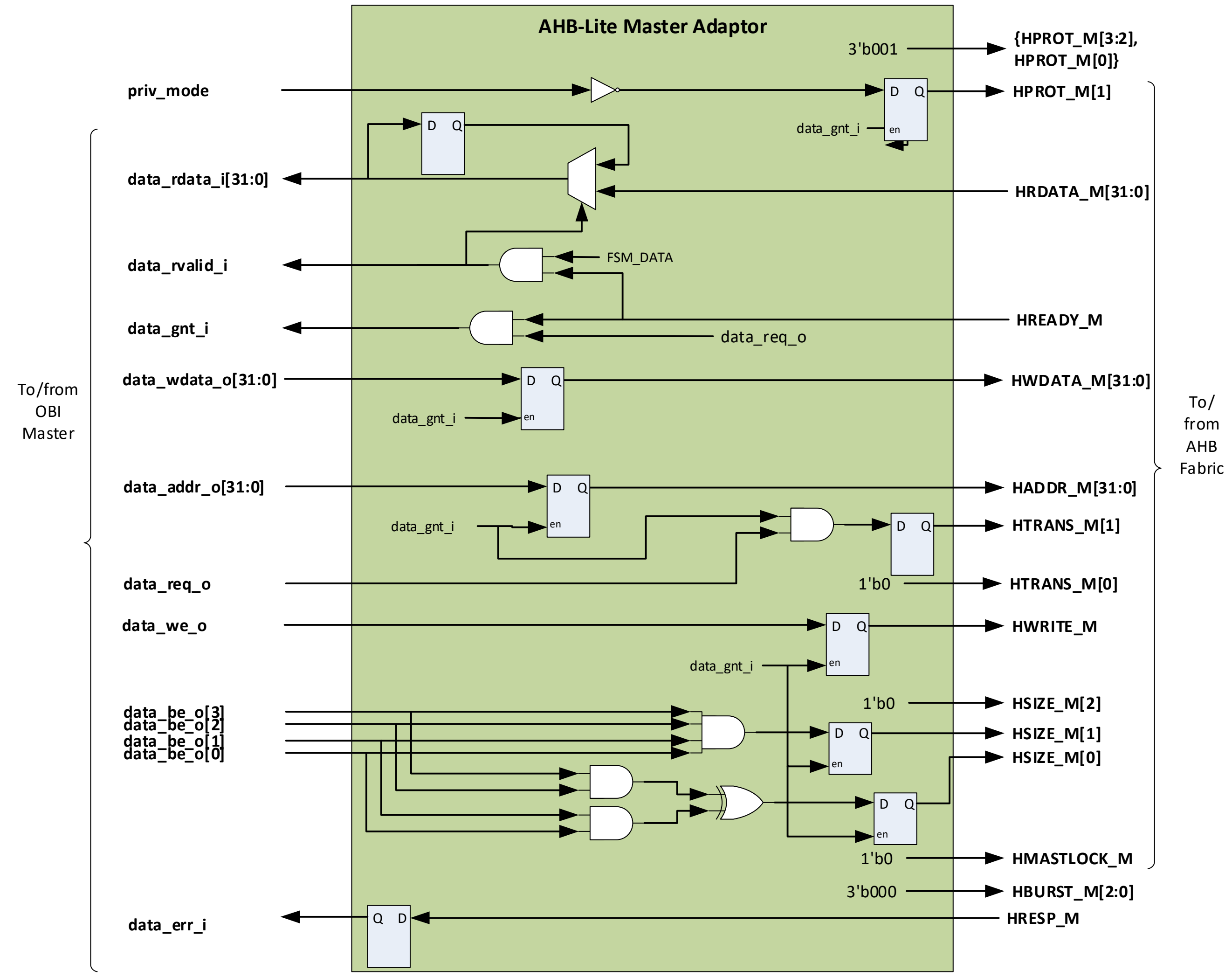

Figure 1: AHB-Lite Master Adaptor Block Diagram

The interface between the OBI side transaction protocol and the AHB-lite protocol is implemented with a state machine that can be summarized by Figure 2.

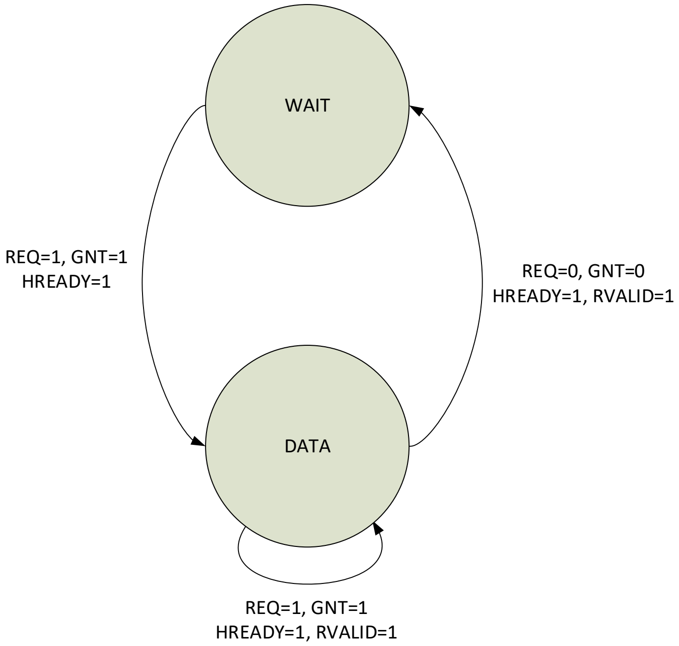

FIGURE 2: AHB-LITE MASTER STATE MACHINE TRANSITIONS

WAIT->DATA

Out of reset, the AHB-lite state machine is in the WAIT state. When the OBI side initiates a request, a grant will either be asserted on the same cycle, if HREADY is also asserted, or the grant will be asserted on the next cycle in which HREADY is asserted. While the grant is issued, **data_addr_o** is passed through to **haddr_m** and on a write transaction **data_wdata_o** can be captured into a register as **hwdata_m** on the following cycle. The cycle while the grant is given can be treated as the address phase of the AHB non-sequential transaction.

DATA->WAIT

When the transaction is complete, HREADY and RVALID will be asserted. On this cycle, if there is not another request initiated by the OBI side, then the state machine will return to the wait state.

DATA->DATA

When the transaction is complete, HREADY and RVALID will be asserted. On this cycle, if there is another request initiated by the OBI side, then a grant will be issued in the same cycle and the state machine will remain in the DATA state for the subsequent transaction. Just like in the WAIT state, the cycle while the grant is asserted can be treated as the address phase of the subsequent transaction.