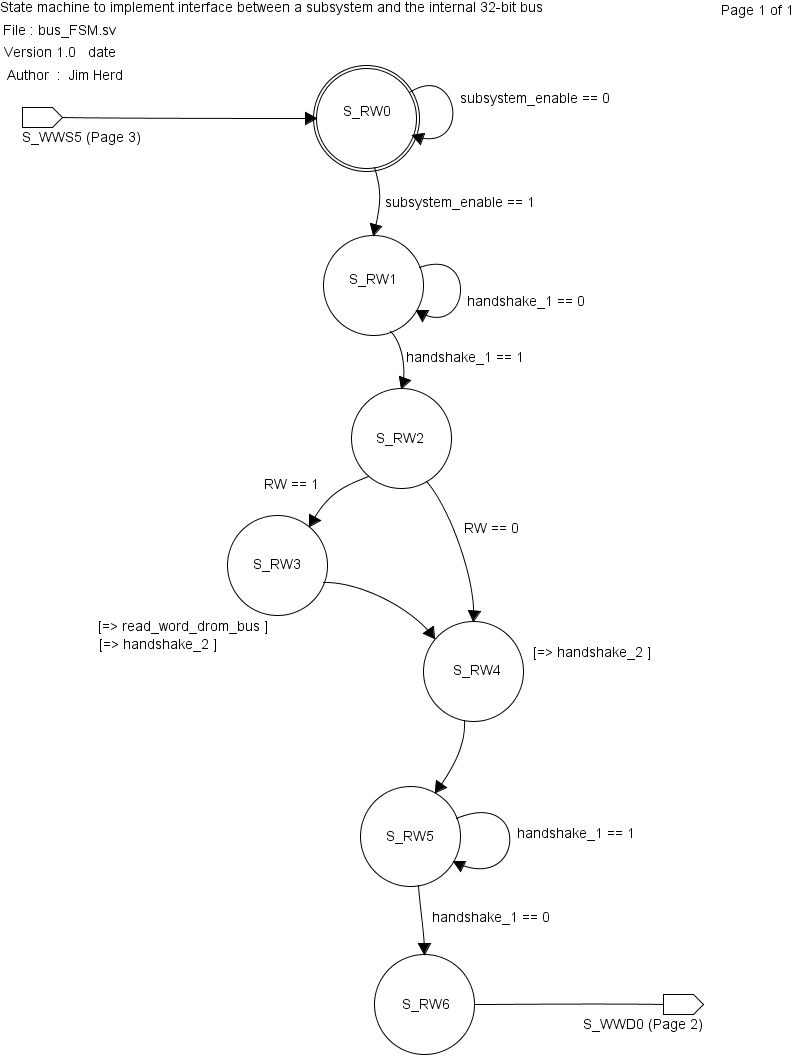
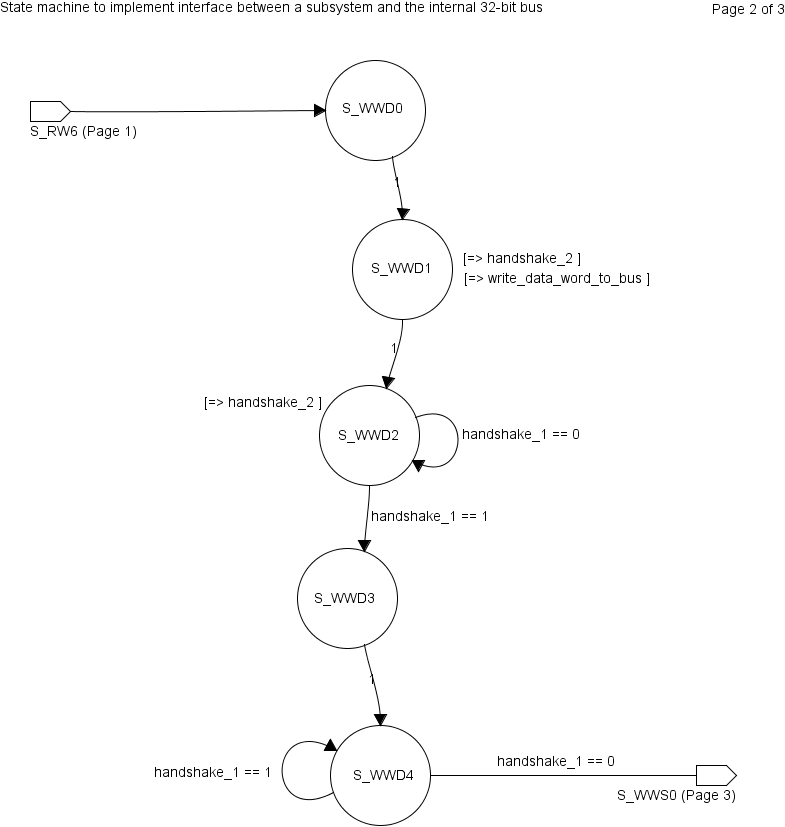
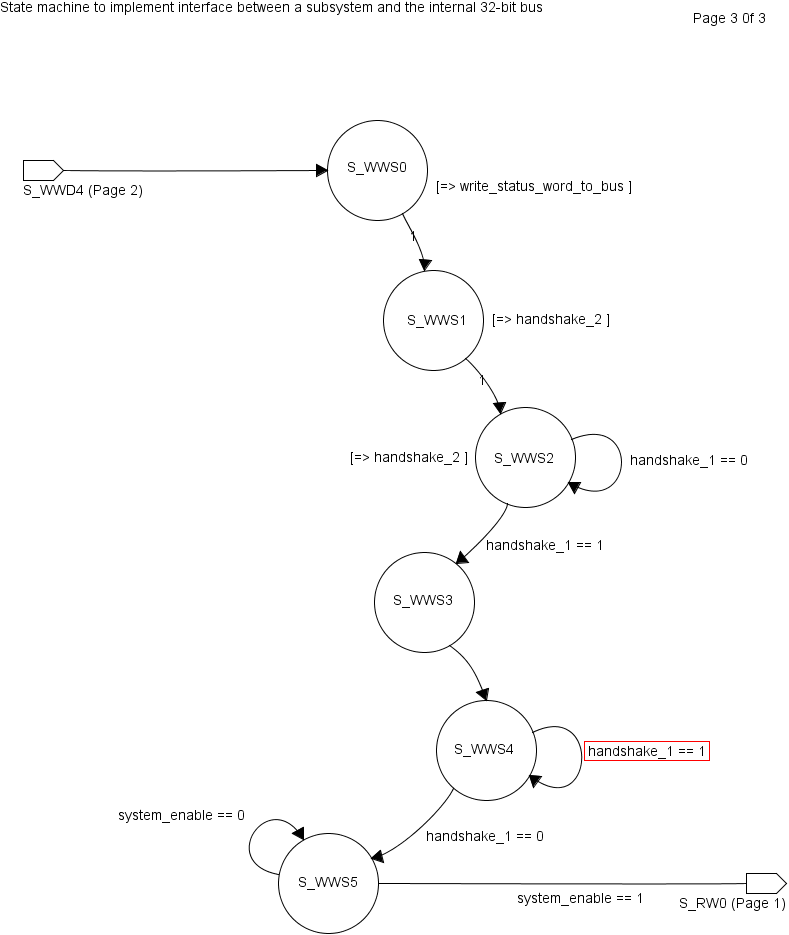
|  |  |  |  |
| --- | --- | --- | --- |
| FSM name : | bus\_FSM.sv | Date: | Feb 2021 |
| Section : | 1 of 1 | Author | Jim Herd |
| Notes : |  | | |







State Objects

|  |  |
| --- | --- |
| **States** | Notes |
| **S\_RW0** | Initial state.  Wait for ‘system-enable’ signal to be asserted. This implies that the subsystem has been addressed. |
| **S\_RW1** | Wait for ‘handshake\_1’ signal to be asserted. This is the first phase of the handshake of the data across the 32-bit bus. |
| **S\_RW2** | Check whether this transaction is a READ from a SUBSYSTEM REGISTER or a WRITE to a SUBSYSTEM REGISTER.  0 == WRITE  1 == READ |
| **S\_RW3** | Assert READ\_WORD\_FROM\_BUS signal to input 32-bit value to addressed register |
| **S\_RW4** | Assert ‘handshake\_1’ signal to indicate read/write of word is complete |
| **S\_RW5** | Wait for ‘handshake\_2’ signal to be de-asserted to indicated that handshake has been accepted by the uP. |
| **S\_RW6** | One clock delay to allow everything to settle. This completes the uP to FPGA transfer part of the command execution process.  \*\*\* may not be needed. To be checked. |
|  |  |
| **S\_WWD0** | Assert signal to write a 32-bit data value on the bus. This signal will remain asserted through states **S\_WWD1** and **S\_WWD2**. |
| **S\_WWD1** | Assert ‘handshake\_2’ to indicate first part of the transfer handshake. |
| **S\_WWD2** | Wait for signal ‘handshake\_1’ to indicate that uP has detected handshake. |
| **S\_WWD3** | One clock delay to allow everything to settle.  \*\*\* may not be needed. To be checked. |
| **S\_WWD4** | Wait for ‘handshake\_1’ signal to be de-asserted to indicate that handshake has been completed.  Go to state **S\_WWSO** if to send status word to uP if compile time parameter ‘INCLUDE\_32\_BIT\_STATUS\_RETURN’ has been defined. Otherwise go to state **S\_WWS5**. |
|  |  |
| **S\_WWS0** | Assert signal to write a 32-bit status value on the bus. This signal will remain asserted through states **S\_WWD1** and **S\_WWD2**.  This status transfer phase may be disabled. |
| **S\_WWS1** | Assert ‘handshake\_2’ to indicate first part of the transfer handshake. |
| **S\_WWS2** | Wait for signal ‘handshake\_1’ to indicate that uP has detected handshake. |
| **S\_WWS3** | One clock delay to allow everything to settle.  \*\*\* may not be needed. To be checked. |
| **S\_WWS4** | Wait for ‘handshake\_1’ signal to be de-asserted to indicate that handshake has been completed. |
| **S\_WWS5** | Wait for ‘system\_enable’ signal to be de-asserted before returning to initial state. |

Input objects

|  |  |  |
| --- | --- | --- |
| **Inputs** | Type | Notes |
| **subsystem\_enable** | binary | Asserted if subsystem is addressed. |
| **RW** | binary | == 0 ignore 32-bit value  == 1 read 32-bit word from bus |
| **handshake\_1** | binary | Handshake signal from bus |

Output objects

|  |  |  |
| --- | --- | --- |
| **Outputs** | Type | Notes |
| **handshake\_2** | binary | Handshake signal to bus |
| **write\_data\_word\_to\_bus** | binary | Write addressed 32-bit register value to bus to be transferred to uP. |
| **write\_status\_word\_to\_bus** | binary | Optional.  Write 32-bit status value to bus to be transferred to uP. |