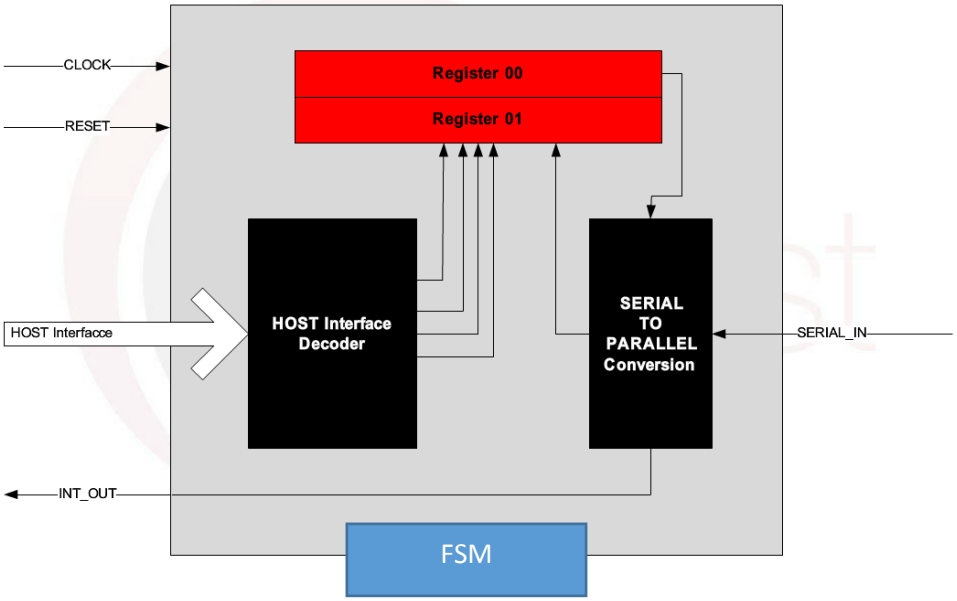
FS

General overview of the DUT

The DUT is a de-serializer that interfaces a single bit input stream on one end and a Host on the other end with a byte wide parallel bus output.



The serial input decoding follows the 8-N-2 UART protocol with a one active low start bit, eight consecutive NRZ data bits and two active low stop bits. This input remains active high when idle or in between words for at least one clock. The DUT stores a successfully received data into an 8-bit register with address 0x01, it then asserts an active high interrupt for the Host to read that data. The interrupt is de-asserted once the byte has been read.

For the Host to read the parallel word once it gets the interrupt, it needs to a) assert an active high request back to the DUT, b) set the R/W bit to 0, c) set the address bus to 0x01 to select the data register and finally d) wait for the DUT to raise its GRANT flag. To avoid protocol violation the R/W bit and address bus must be set by the host on or before the cycle it asserts the request bit.

When the DUT receives the host's request assertion it will read the selected register, assert the active high GRANT flag back to the host and put the data on the read bus. A read transaction is considered complete when the host's request and the DUT grant are both active high at the same time. At this point the interrupt and the request bits are de-asserted by the DUT and Host respectively.

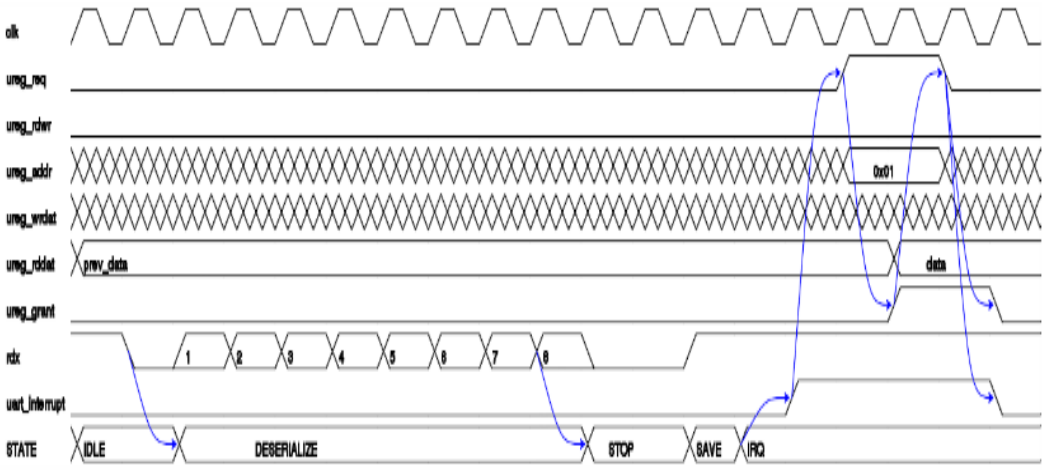


Figure 1: Timing Diagram: RXD and Read Cycle

The DUT can be enabled or disabled by the Host by toggling bit 1 of the register at address 0x00. The Host first needs to assert the request and R/W flags, the data vector to write and the address (0x00). The DUT will act on the request and acknowledge back a successful write by asserting the GRANT flag back to the host.

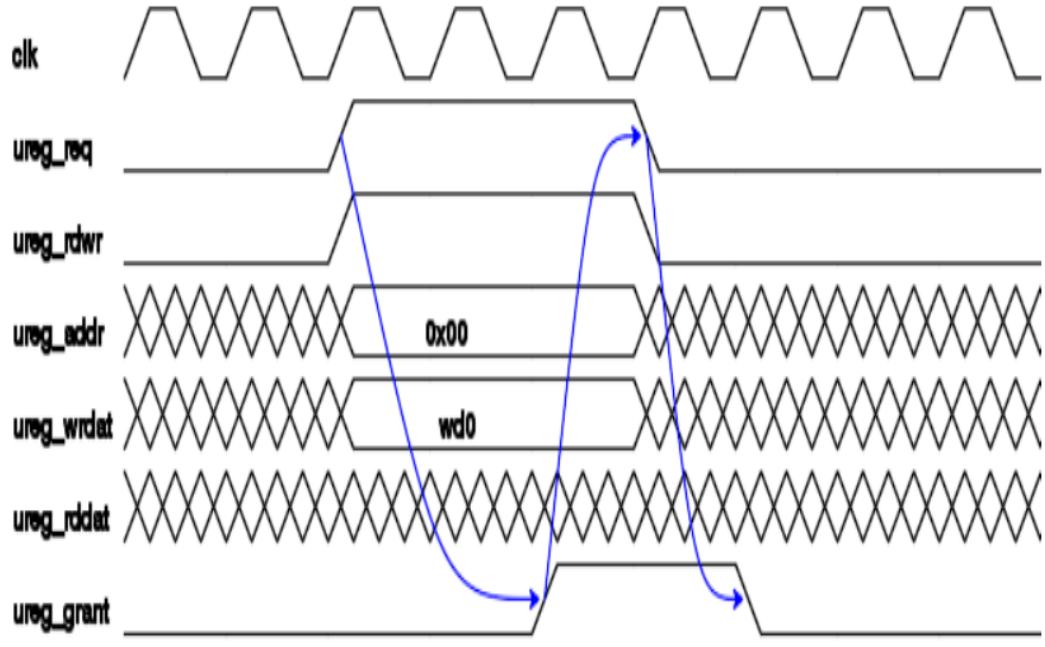


Figure 2: Timing diagram: Write Cycle

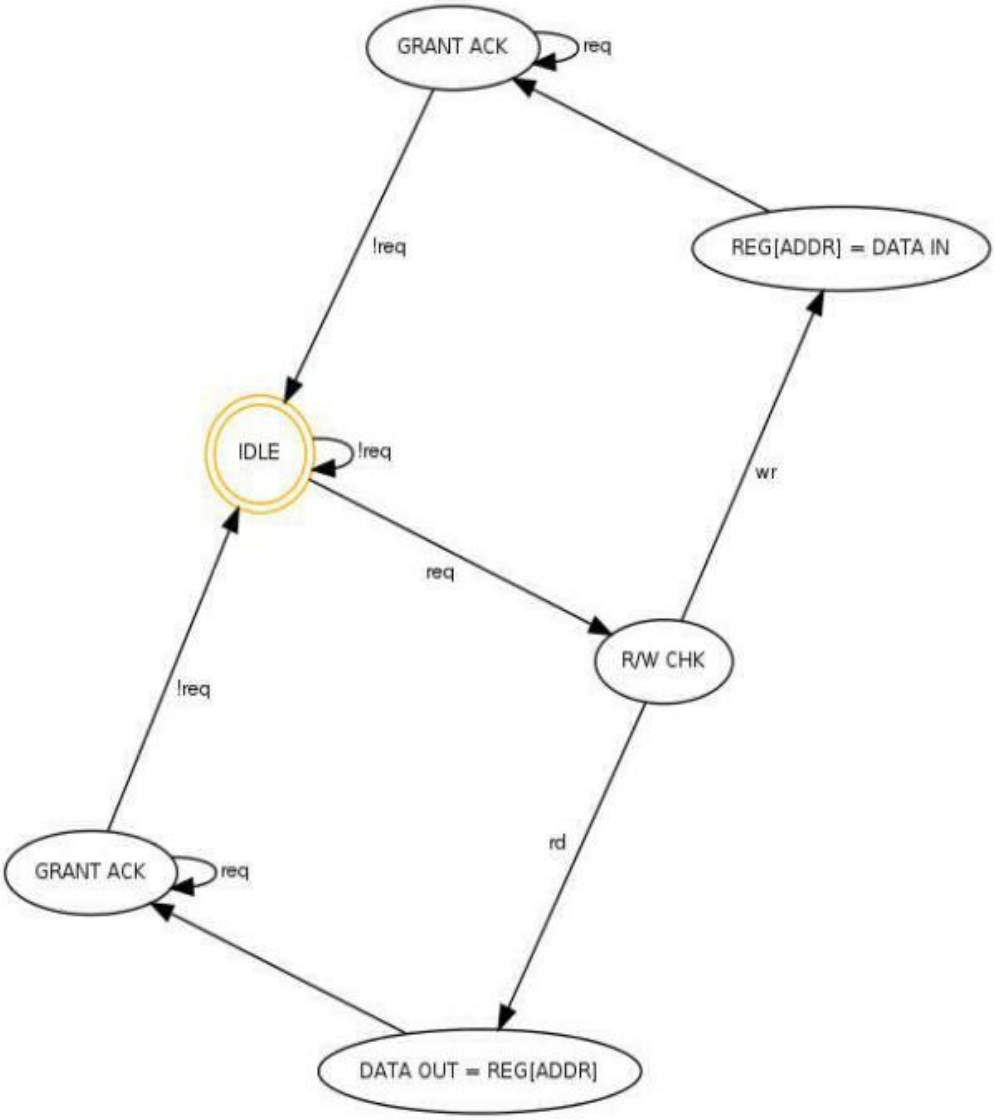


Figure 3: R/W Operation Flow