Capstone Project 2

Week 6 (32) – Semester 233

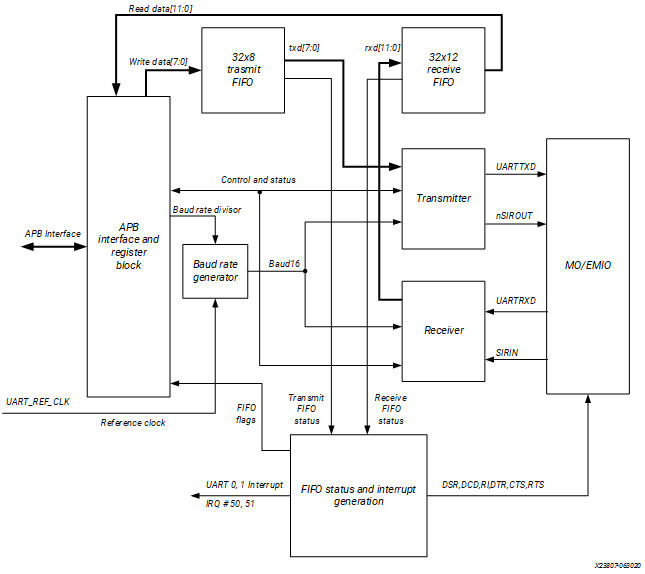
Student: Vo Viet Hung – 2051076

**Implementation of AHB-APB and APB-UART bridges**

# The problem: APB UART

***Figure 1: APB UART main operating block***

In last week, I have finally completed most important tasks is that finished the APB UART block. First, I have finalized and tested the key components of this block, including the baud rate generator signal out to the finite state machine (FSM) for data reception (RX) and transmission (TX), the FIFO buffers for RX and TX, and the SHIFT registers with little adjustment. Moreover, I used register for direct out the data needed relate to PADDR. On the other hand, with RX FSM and its shift register block (where I have reduced the IDLE state because I thought this could be ignored), I have adjusted the signal so that the operating of TX and RX will not overlap each other during operation (in the previous design, I let them operating simultaneously and it have troubled with write in and read out data in case that data have not completed). Overall, these components have been integrated and thoroughly tested to ensure accurate operation. Additionally, I am checking and debugging some other case to ensure that necessary output signals of the APB UART during the process stables and meets design requirements. This work helps ensure that the UART communication system via the APB operates efficiently and reliably.

* 1. **Summary**

***Figure 1.1.1: General Block Diagram of APB UART***

In the past week, I continued with addressing issues within the APB UART block to fix its functionality and stability. The three significant blocks of improvement were the shift register of RX, RX FSM and its FIFO block.

***Shift Register of RX:***

- Issue: In the previous design, the input signals (***UART RXD***) handling was not as well as I expected, leading to this reason was the input signal were used as a block of data of 8 bits instead of serial of data, because I mimicked the operation of UART RX as same as TX. Then, I re-read some examples related to UART RXD in which they used a string of data input signal instead of the register of 8 bits UART RXD data.

- Solution: By reading some examples of UART RX side, I have adjusted the input signal from a block of 8 bits data only into a serial of data (12 bits 1 bit for start, 8 for data, 1 parity and 1 stop). (I set up data go into shift register of RX randomly for testing as following ***Figure 1.1.2***).

***RX FIFO Block Enhancements:***

- Issue: The initial design of this RX FIFO block has troubled with inputting signal, in which I have not used an external signal named **RXen** (same as UART TX FIFO. This signal takes place as a flag to indicate which method is using, if **RXen** is on, it will let the data come from UART RXD get into, prepare for next step of reading data out. The rest changed are same as UART TX as I have done last week is that I also used two signals out name rx\_ptr\_add\_i and rx\_ptr\_add\_o to detect the status of FIFO whether is full or not.

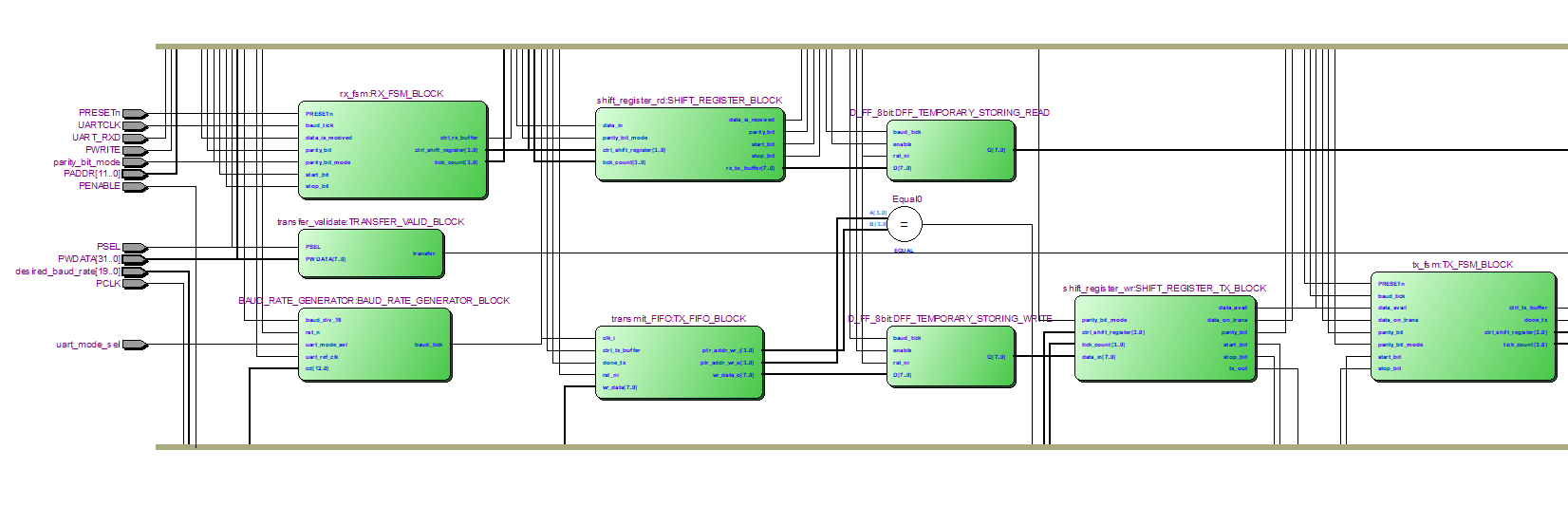
- Solution: First, I resized the RX FIFO block so now the data is just stored for only 16 8 bits data instead for easier compiling. Secondly, I have added an additional input signal to indicate the read method is required. Finally, took out address pointer of index inside FIFO, same as the previous week (WEEK 31) for indicating the status of FIFO.

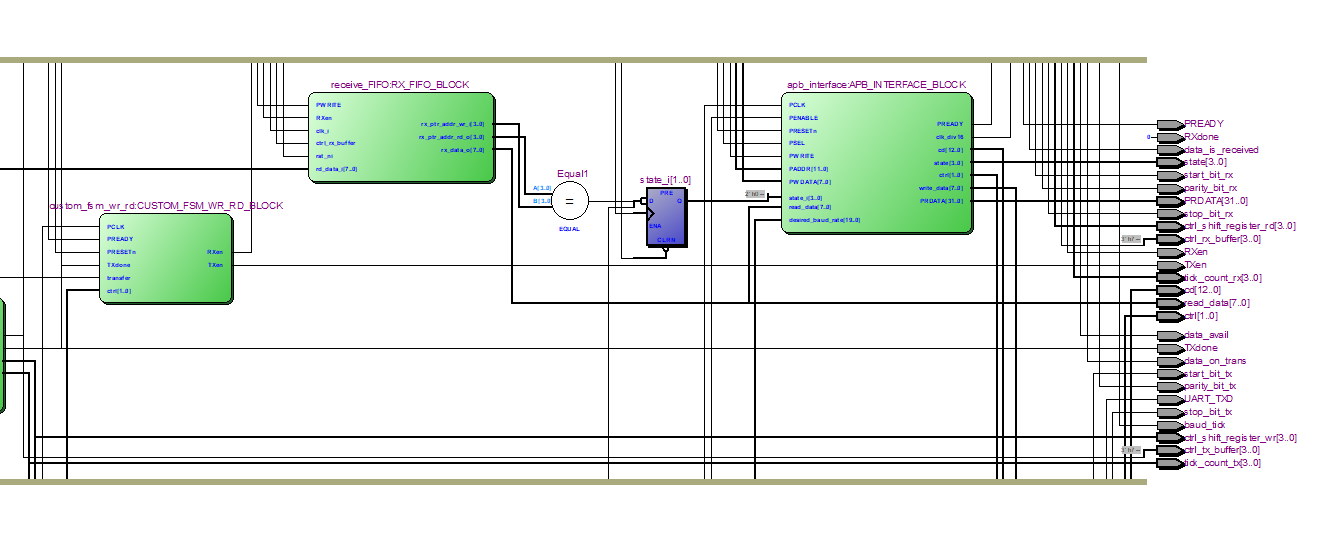
***RX FSM Block:***

- Issue: In the previous design, there was lack of signal-controlled states (lead to cannot generate out data, I met this problem because in the previous design, I have not checked its operation), so I did tiny adjustments related to this block are firstly I have considered about the IDLE state of its operation, then during my simulation, I saw no different much between with this state and without it. So, I did delete it to optimize operation. Then I did add external signal (RXen) same as RX FIFO and others to indicate the current operating method is reading or writing (Transmit or Receive).

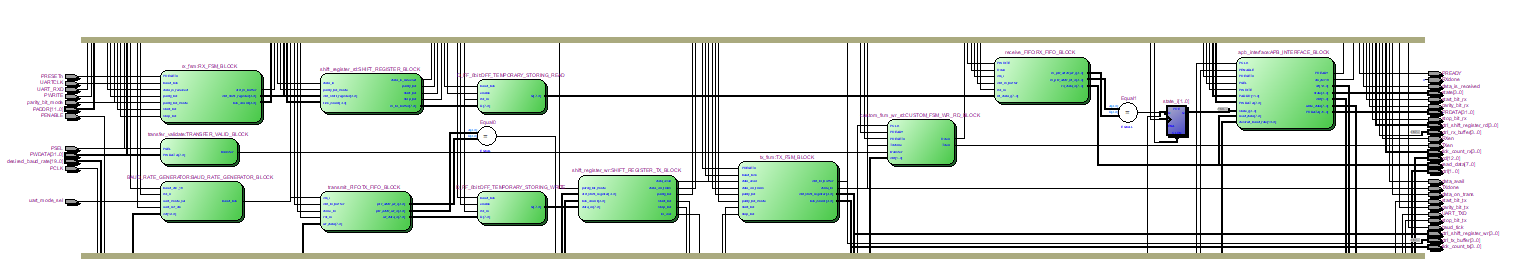
- Solution: First, I added signal into the RX FSM block so now it only works during! PWRITE signal (not constantly receive data from nowhere), then I tried to reduce the state used in FSM for optimizing and others.

The other parts are just related to the APB UART where I changed for debugging and solving incorrect data come out and also did more simulation for ensuring its operation and stability.

And following is the nearly final figure of APB UART that I have done and on checking: 

***Figure 1.1.2: The first part of RTL view of APB UART Block design reference***

***Figure 1.1.3: The part of RTL view of APB UART Block design reference***

****

***Figure 1.1.4: The full view of RTL of APB UART Block design reference***

***Explanation***: For testing fields and other simulations figure, I don’t add to most of my report is because during each testing, I have changed a lot and I missed to record during work time. I am very sorry for this non-sense explanation but it **real**. I promised to send you in the next report.

* 1. **Planning for next week working**

For the next week, If the testing goes well, I would combine the AHB Slave (AHB-APB Bridge) together with APB UART and run whole simulation of it. If the test field and other things such as simulation, optimized, etc goes on well, I would ask you for some testing features for my final report and furthermore would be my graduation.

**----------------Thanks for reading my Journal-------------**