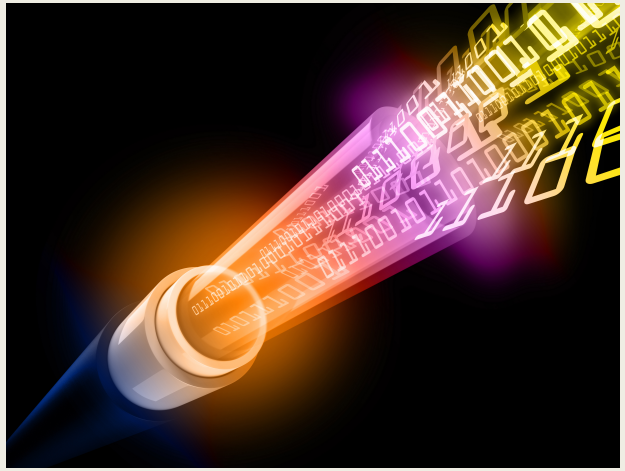
SAVRAN | ZUMBO

BUFFERED FIBER OPTIC COMMUNICATION



Overview

- Engineer a communication infrastructure between two lab kits that:
  - *Exploits visible light*
  - *Is prone to a certain span of interruption!*
- Interesting: Because you “see” the data transmission

Solution

- Use AV02 Fiber Optic Receiver and Transmitter Pairs
- Implement a buffering system on the receiving end
- Create packetizing model for labkit communication

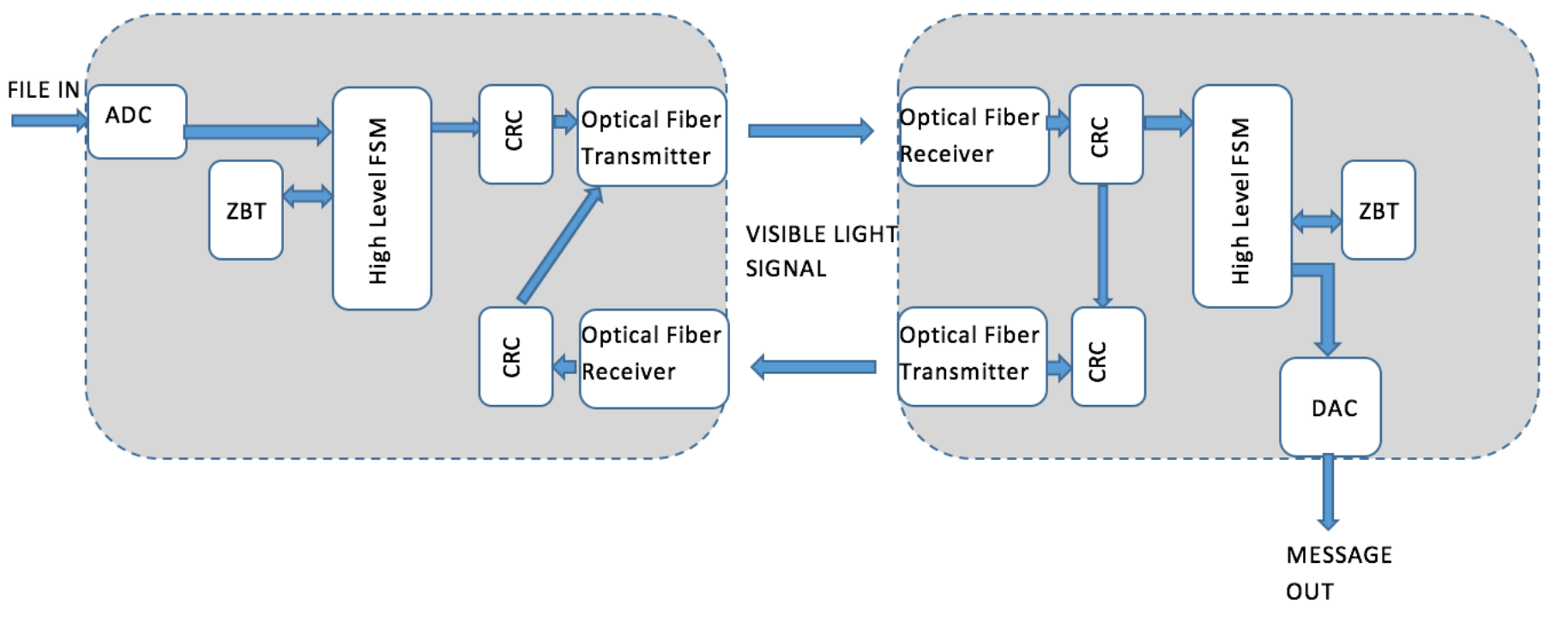
High Level Specs

- Transmitting Lab Kit (FPGA):
  - *Takes in data from AC97 Line in*
  - *Packetizes the data*
  - *Writes these packets to the ZBT (not a frat) Memory*
  - *Transmits it to the Receiving FPGA*
  - *In case of interruption or transmission error:*
  - *Resends the data starting from the last packet of failure*

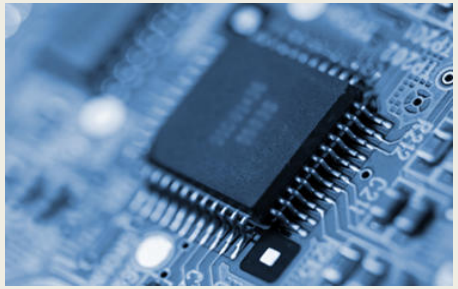
- Receiving FPGA:
  - *Receives the data serially from the TFPGA*
  - *CRCs every packet and rerequests an erronous packet from TFPGA*
  - *Meanwhile it also halts its “receiving” process*
  - *Writes packets to ZBT (buffer) and plays the audio with lag (which is why it is robust to a certain extent!)*

Did you notice most of the components behave as FSMs? :)))))))

HIGH LEVEL BLOCK DIAGRAM

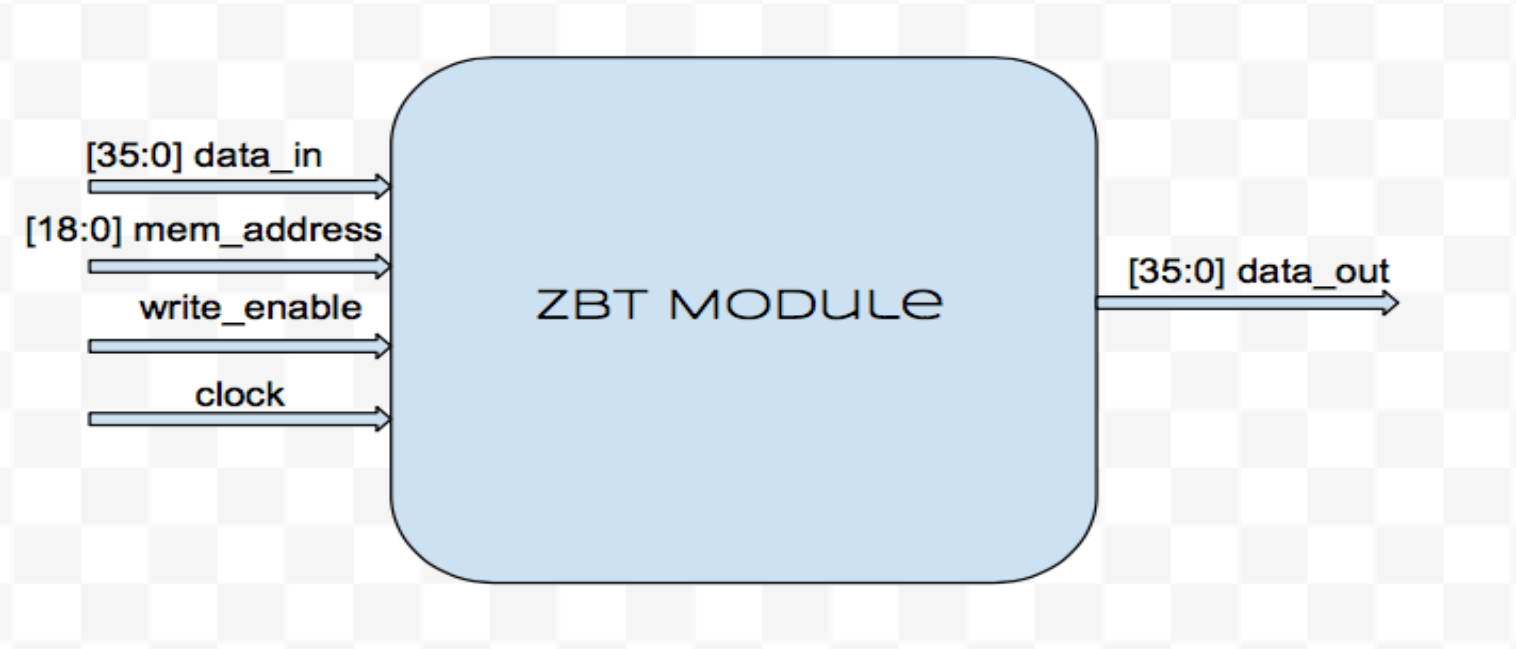


MODULES OVERVIEW

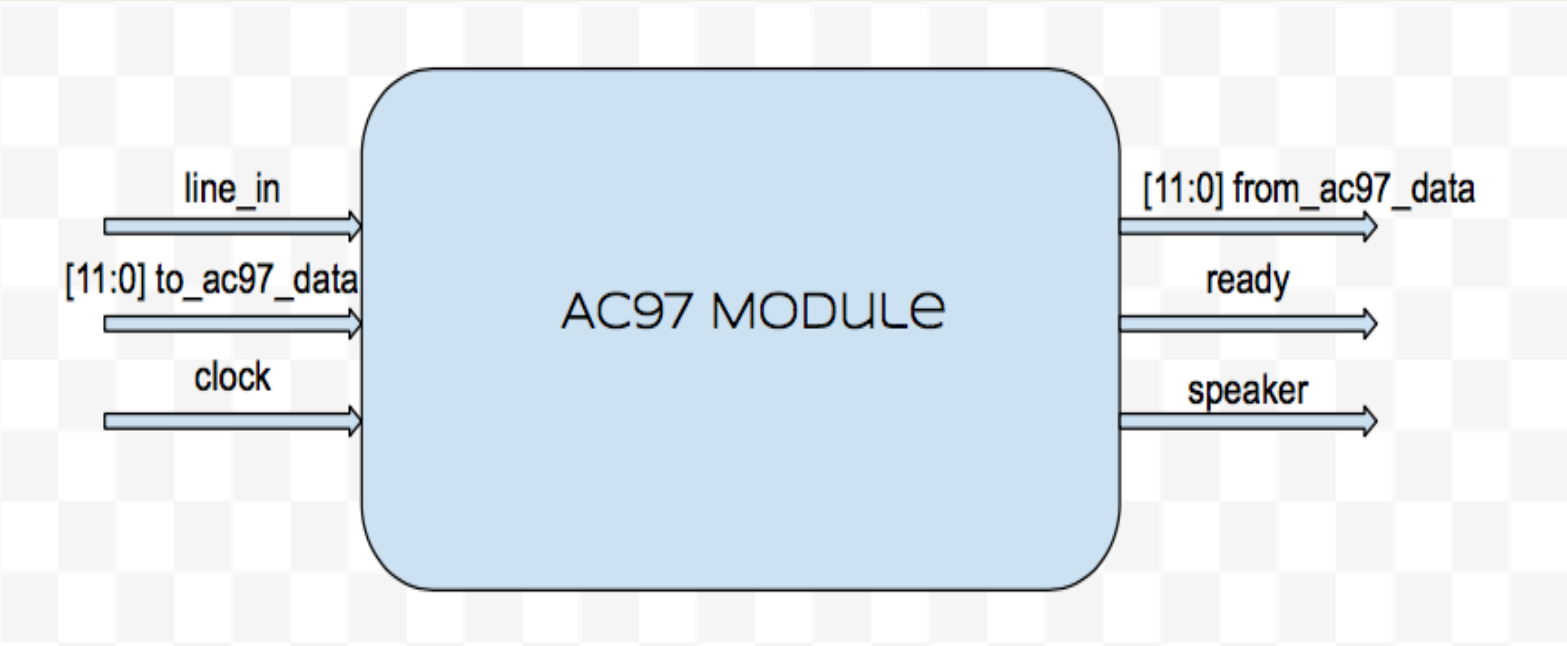


- ZBT
- AC97
- CRC
- Transmitter
- Receiver
- High-level-FSM (transmitting and receiving FPGA)

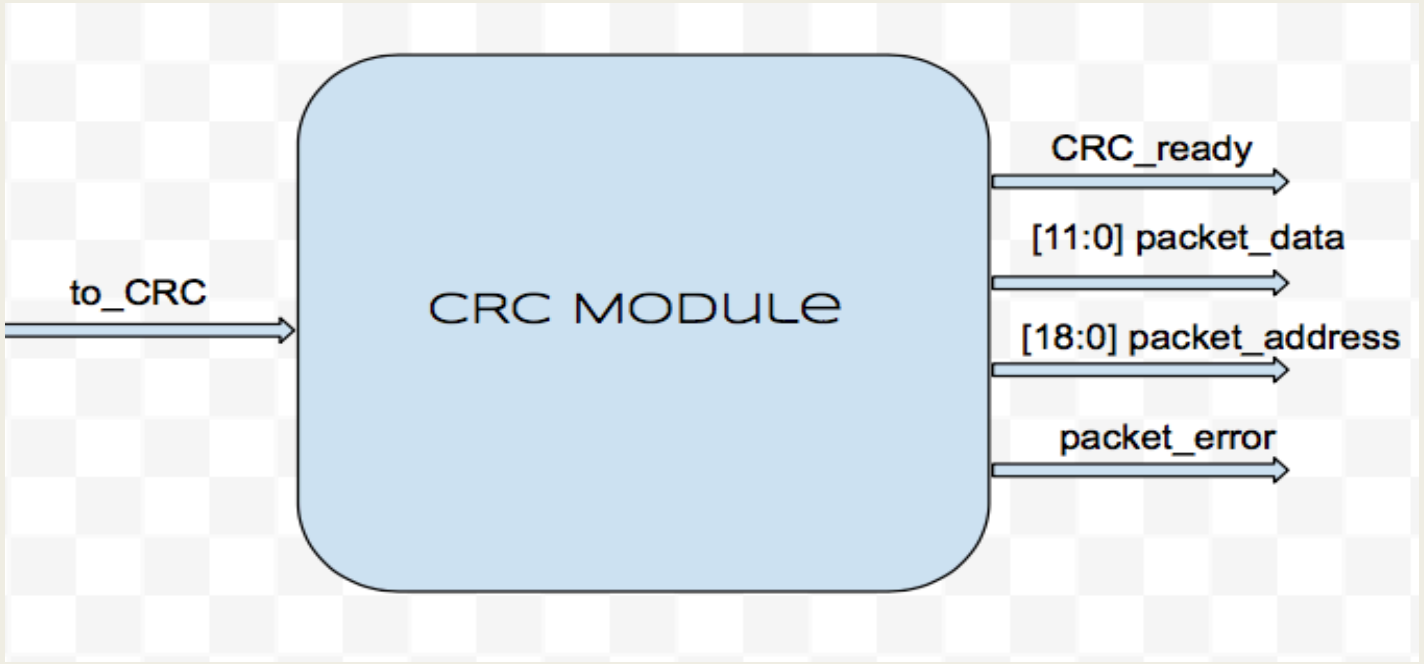
- **ZBT memory:** 512k x 36-bit wide bus (2^19 memory addresses)

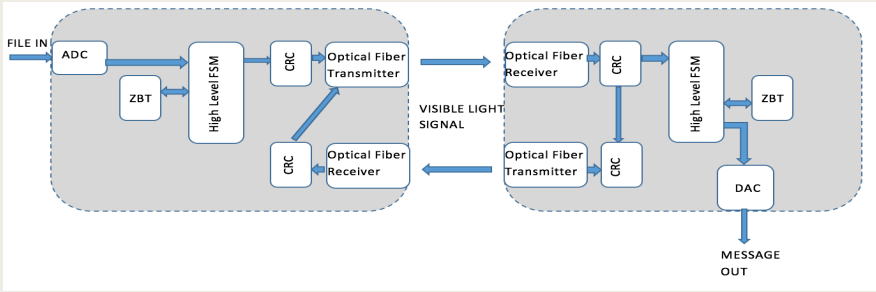


- **AC97 module:** sampling audio signal at 48kHz with 12-bit depth
  - *Packets: 3 X 12-bit samples fit into 36-bit memory bus, index by 12*
  - *3 samples/address * 512k addresses / 48kHz samples per second = ~32 seconds of audio*

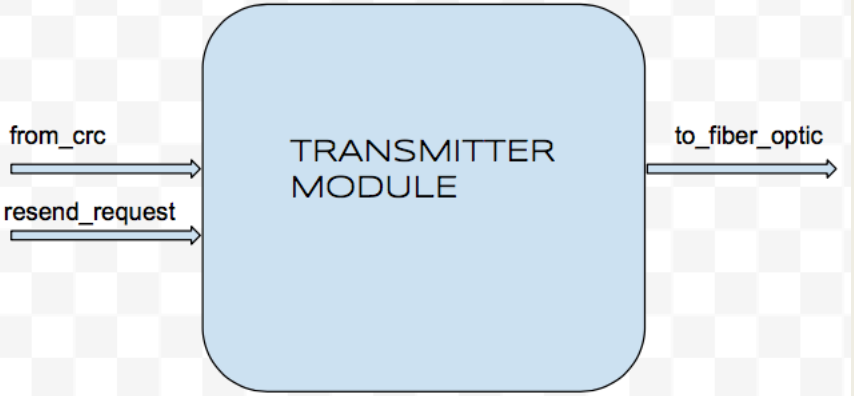


- **Cyclic Redundancy Check:** Verifies packet integrity, serial input, parallel output
  - *Can write packet data to memory even if it's incorrect, we can just overwrite it later*
  - *This saves clock cycles*

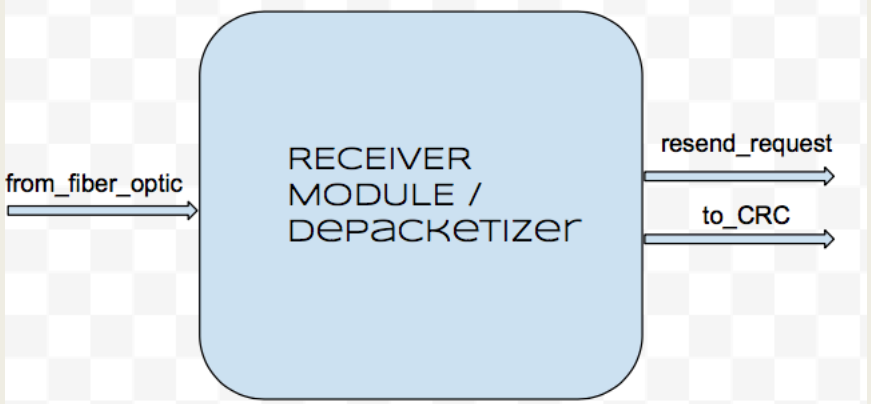




- **Transmitter Module** (Not the entire labkit!)



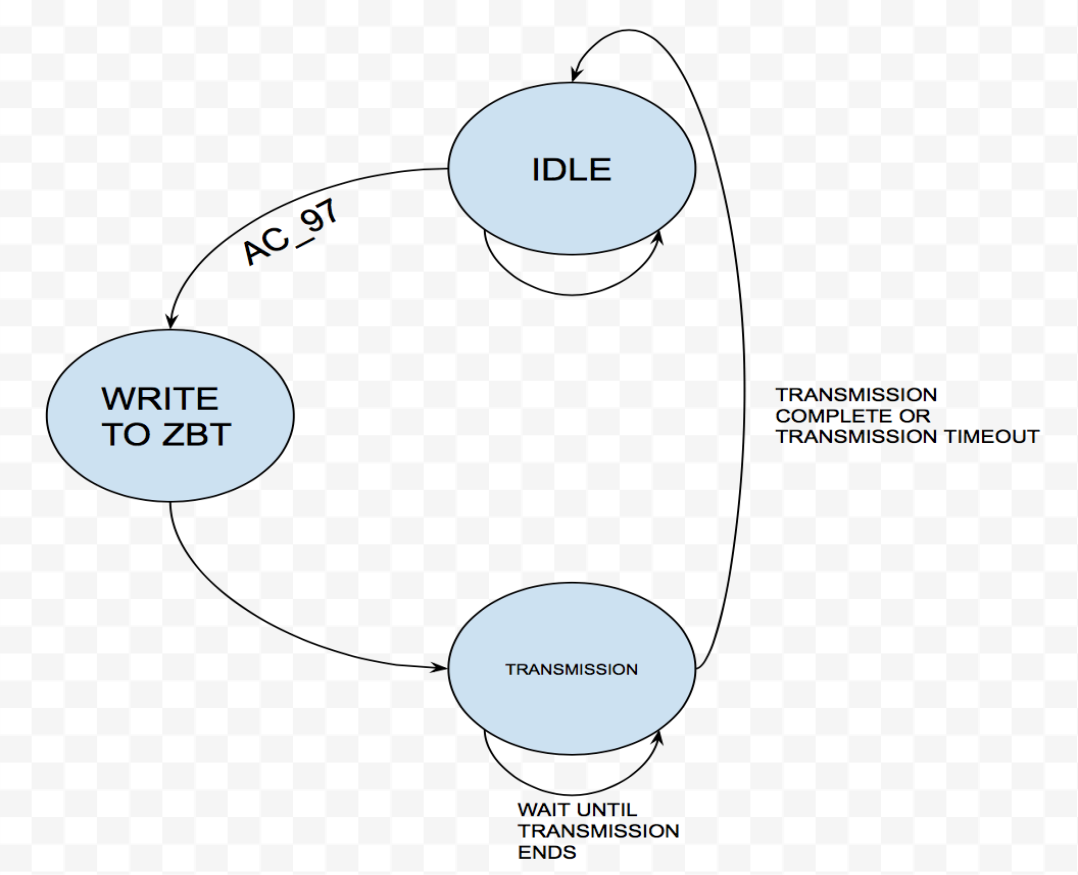
- **Receiver Module** (Not the entire labkit!)



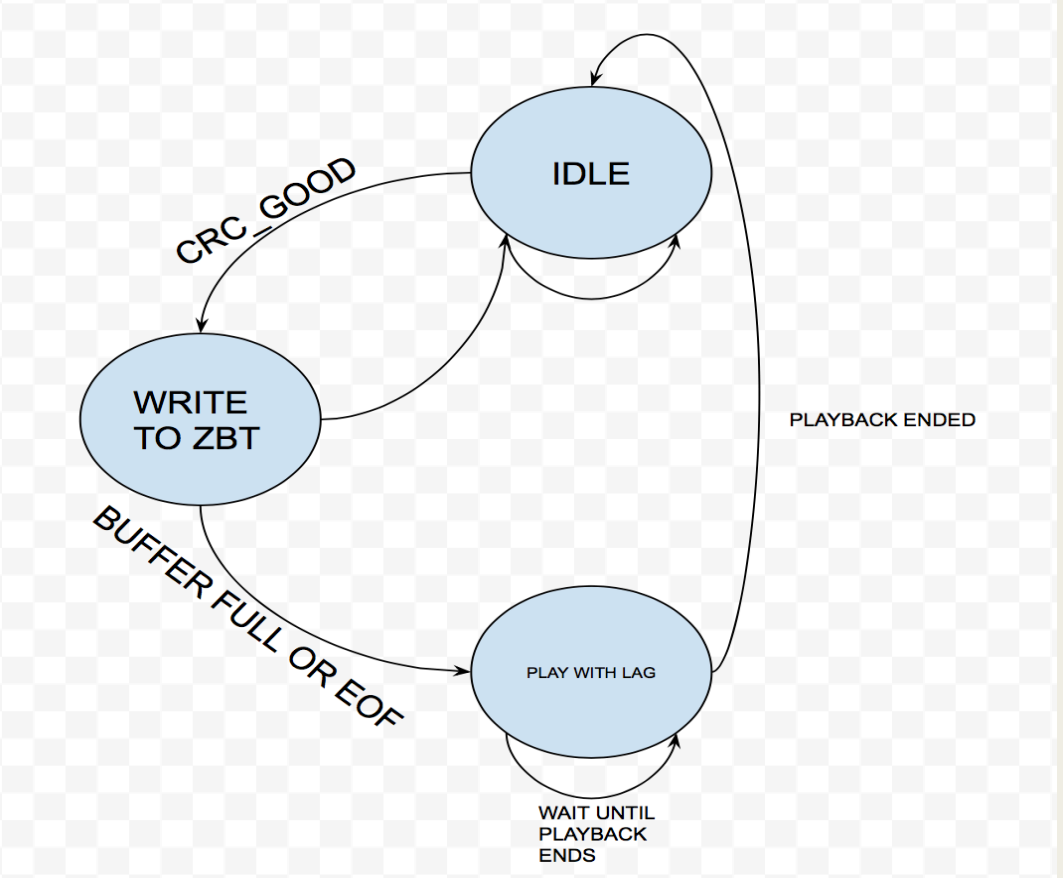
Packetizing Scheme

FSM?

- Transmitter FPGA FSM



- Receiver FPGA FSM



Timeline

Development\Week of	11/7	11/14	11/21	11/28
Integration of AV02 Fiber Optic Transmitter and Receiver				
Modification/Testing ADC				
Integration of built-in DAC and simple communication testing				
ZBT Memory Module				
Building/Testing Packetizer				
Implementing CRC				
Development of FSM				
Testing FSM and Communication				
Debugging and More Testing				
Stretch Goals