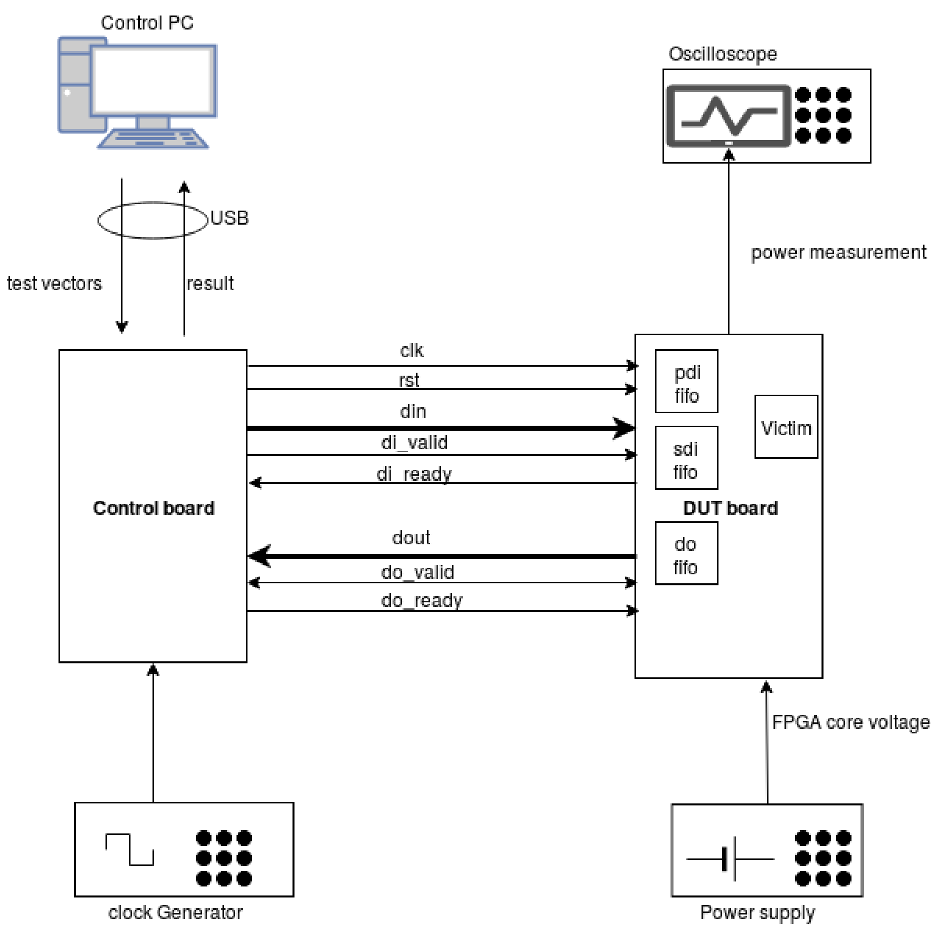
# Data Acquisition

After test vectors have been generated, user can run dataAcquisition.py. The PC will send one test vector at a time to the control board, which sends it to DUT. The control board will trigger the oscilloscope to capture the trace. The process will be repeated unitl all traces are collected.

FOBOS control-DUT protocol

The control receives test vectors from the PC on at a time. It sends the vector to the DUT which uses the header information in the vector to put the data (plaintext, key etc.) into the correct FIFOs. The DUT wrapper then allows the victim algorithm to run by setting the victim reset to zero. The victim then drains the FIFOs (sdi and pdi FIFOs) and stores the output in the dout FIFO. Once the dout FIFO accumulates the expected amount of data, the DUT wrapper sends data to the controller which sends it to the PC.

The following diagram shows the components of FOBOS including the handshake signals used.



**Trigger settings**

The controller can send a trigger to the Oscilloscope once the DUT starts processing the data (ie. di_ready = 0). Or it can be configured to trigger any number of clock cycles after this event occurs.

TRIGGER_WAIT_CYCLES : The number of clock cycles after which the trigger is asserted (after di_ready goes to zero).
TRIGGER_LENGTH_CYCLES : The time the trigger signal is asserted.
TRIGGER_TYPE : possible values: TRG_NORM | TRG_FULL | TRG_NORM_CLK | TRG_FULL_CLK

TRG_NORM : normal trigger mode. in this mode the TRIGGER_WAIT_CYCLES and TRIGGER_LENGTH_CYCLES are applied.

TRG_FULL : Full trigger mode. While DUT is running (between di_ready = 0 and do_valid = 1) the trigger is asserted.

TRG_NORM_CLK : same as TRG_NORM but the trigger signal is anded with the clock.

TRG_FULL_CLK : same as TRG_FULL but the trigger signal is anded with the clock.

CUT_MODE : Controls how the trace retreived from the scope will be processed.

possible values: FULL | TRIG_HIGH

FULL : The trace is cut starting at the rising edge of the trigger to the end of the screen.

TRIG_HIGH : the trace is cut from the rising edge to the falling edge of the trigger ie. the trace where the trigger is high will be saved.

All of there settings are found in confi/acquisitionconfig.txt.

**Data Aquisition Configuration**

Before running the dataAcquisstion,py script, the user must modify the configuration files at config/config.txt and confi/acquisitionconfig.txt
In the config.txt, please set the project name.
Here is sample for acquisitionConfig.txt file. Please refer to FOBOS user guide for information about each parameter.

```
# =============================================
# Global Settings
# =============================================
# =============================================
MEASUREMENT_FORMAT = dat # Default => dat
LOGGING = INFO # INFO|DEBUG
# =============================================
# =============================================
# Control Board Settings
# =============================================
# =============================================
CONTROL_BOARD = Nexys3
TRIGGER_WAIT_CYCLES = 0 #@VICTIM CLOCK
TRIGGER_LENGTH_CYCLES = 1 #@VICTIM CLOCK
TRIGGER_TYPE = TRG_FULL #TRG_NORM | TRG_FULL | TRG_NORM_CLK | TRG_FULL_CLK
CUT_MODE = TRIG_HIGH #FULL | TRIG_HIGH
# =============================================
# =============================================
# Test Data Generation Settings
# =============================================
# =============================================
DATA_FILE     = dinFile.txt
EXPECTED_OUTPUT = 16 # Expected output size in bytes
OUTPUT_FORMAT = hex # Default => hex
NUMBER_OF_ENCRYPTIONS_PER_TRACE = 1
BLOCK_SIZE = 16 # In Bytes
# =============================================
# =============================================
# FOBOS Capture Settings
# =============================================
# =============================================
DUMMY_RUN = NO  #YES/NO
NUMBER_OF_TRACES = 50000
###################################################
######## Signal Alignment Module Parameters ########
###################################################
CAPTURE_MODE = SINGLE # MULTI|SINGLE
TRIGGER_THRESHOLD = 1.0
# =============================================
# =============================================
# FOBOS Oscilloscope Settings
# =============================================
# =============================================
# INTIALIZATION OPTIONS
OSCILLOSCOPE = AGILENT #AGILENT|OPENADC
OSCILLOSCOPE_IP = 192.168.10.10
OSCILLOSCOPE_PORT = 5025
AUTOSCALE = NO   # YES|NO
IMPEDANCE = ONEMEG #FIFTY|ONEMEG
# VOLTAGE AND TIME RANGE OPTIONS
CHANNEL1_RANGE = 0.060V
CHANNEL2_RANGE = 6V
CHANNEL3_RANGE = OFF # ON|OFF|voltage range
CHANNEL4_RANGE = OFF # ON|OFF|voltage range
```

```
TIME_RANGE = 0.000050
TIMEBASE_REF = LEFT
# TRIGGER OPTIONS
TRIGGER_SOURCE =  CHANNEL2
TRIGGER_MODE =  EDGE
TRIGGER_SWEEP = NORM
TRIGGER_LEVEL = 1
TRIGGER_SLOPE = POSITIVE
# ACQUIRE OPTIONS
ACQUIRE_TYPE = NORM # NORM|PEAK|HRES|AVER
ACQUIRE_MODE = RTIM   # RTIM | ETIM| SEG
```

Once the configuration is done, user can run

python dataAcquisition.py

The output will be saved in workspace/<project name>. The traces are stored in a numpy array called rawDataAligned.npy.