

A Bunch-by-Bunch Beam Position Monitor

Fang Liu

**Institute of High Energy Physics,
Chinese Academy of Sciences**

2019-6-5



Introduction



Normal BPM

- Turn-by-Turn Measurement
- Fast Acquisition Measurement
- Slow Acquisition Measurement



doing **Bunch-by-Bunch** measurement



Goal: Resolution of Bunch-by-Bunch : 5um

- Hardware platform of Bunch-by-Bunch **Feedback system**, Bunch-by-Bunch **Beam Current Monitor**; Diagnosing **Beam Lost** and **Catching Tune** of Beam



A Bunch-by-Bunch BPM



■ Electronic System

- **ZYNQ7035**
 - ✓ dual-core ARM Cortex-A9&Kintex®-7 ZYNQ7035
- **4 Channels 500MSPS 14bit ADC**
 - ✓ AD9684, 2 channels
- **1GB DDR3**
- **100/1000Mbps ETHERNET**
- **PCIE GEN2 X4**
- **Input Analog Band Width: 10MHz-2GHz**

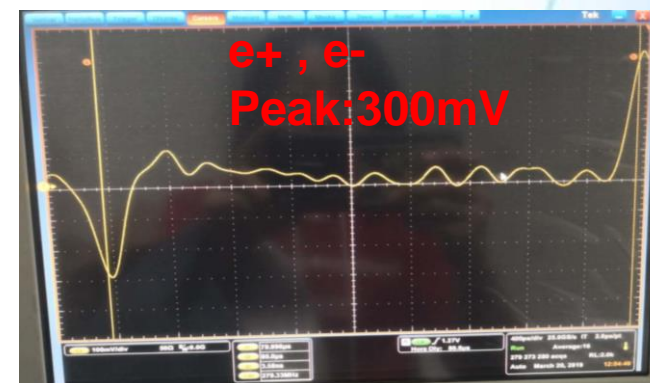


■ A Four Button Pickup

The signals from the pickups of BEPCII are enough for directly data sampling of ADC

■ Delay Module

Adjust to the time to guarantee ADC can get the peak of the Button Signal



Development state



R&D Plan: A ZYNQ Based, Four channels 500Msps ADCs, Bunch-by-Bunch System



Already Down

Hardware : Circuit Design, Soldering and Debugging.

Firmware : ADC control, Clk control, DDR ,Ethernet Bunch-by-Bunch Algorithm

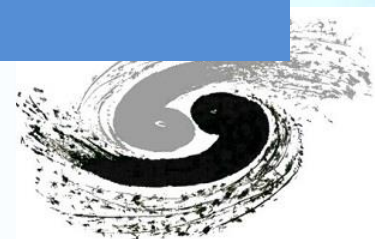
Soft in ZYNQ ARM : Initialization of Ethernet, ADC and Clk. Data Acquisition Control

Need to Do

Develop : Embedded Linux and EPICS on ZYNQ ARM

Improvement: Upgrade Hardware and Firmware

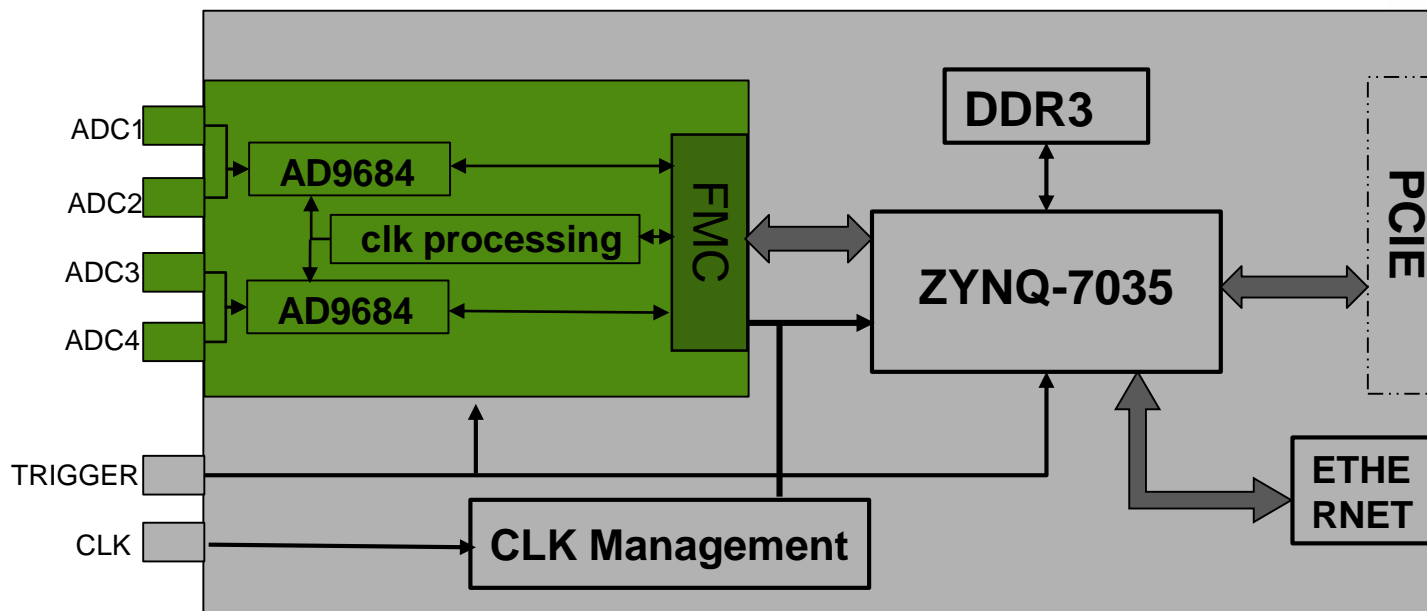
Furthermore: Embedded Timing ?....



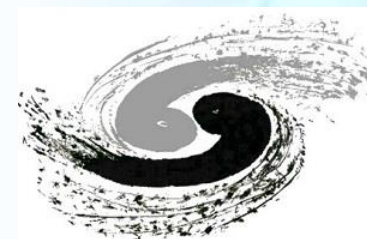
Hardware



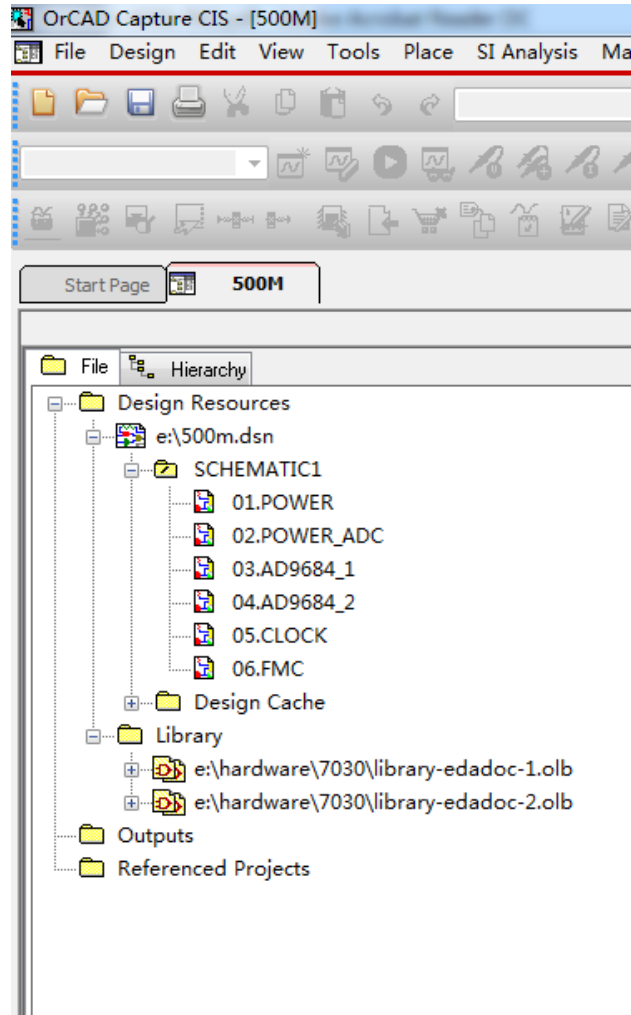
Architecture



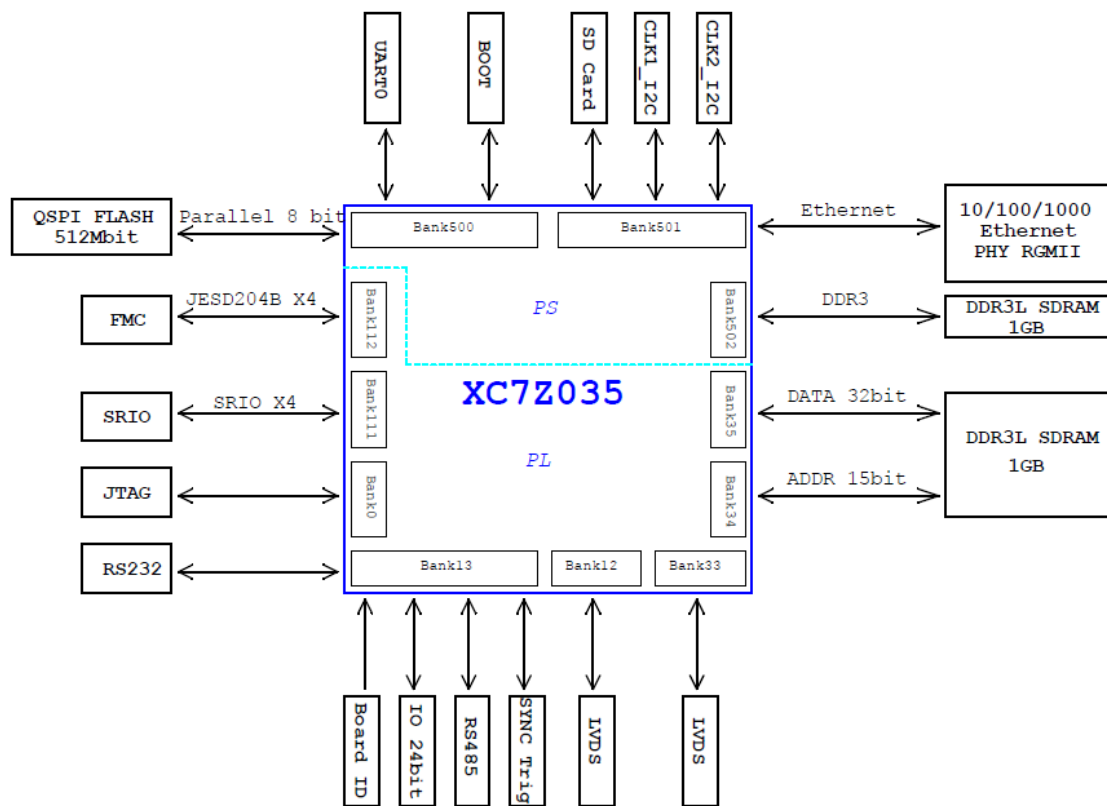
- ❑ Base board and FMC Carrier
- ❑ ZYNQ7035 FPGA and 4 channels 500MSPS 14bit ADC
- ❑ 2 DDR3: one for Data Storage, one for ARM
- ❑ 10/100/1000Mbps Ethernet
- ❑ PCIE GEN2 X4: backplane



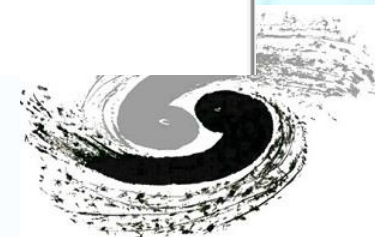
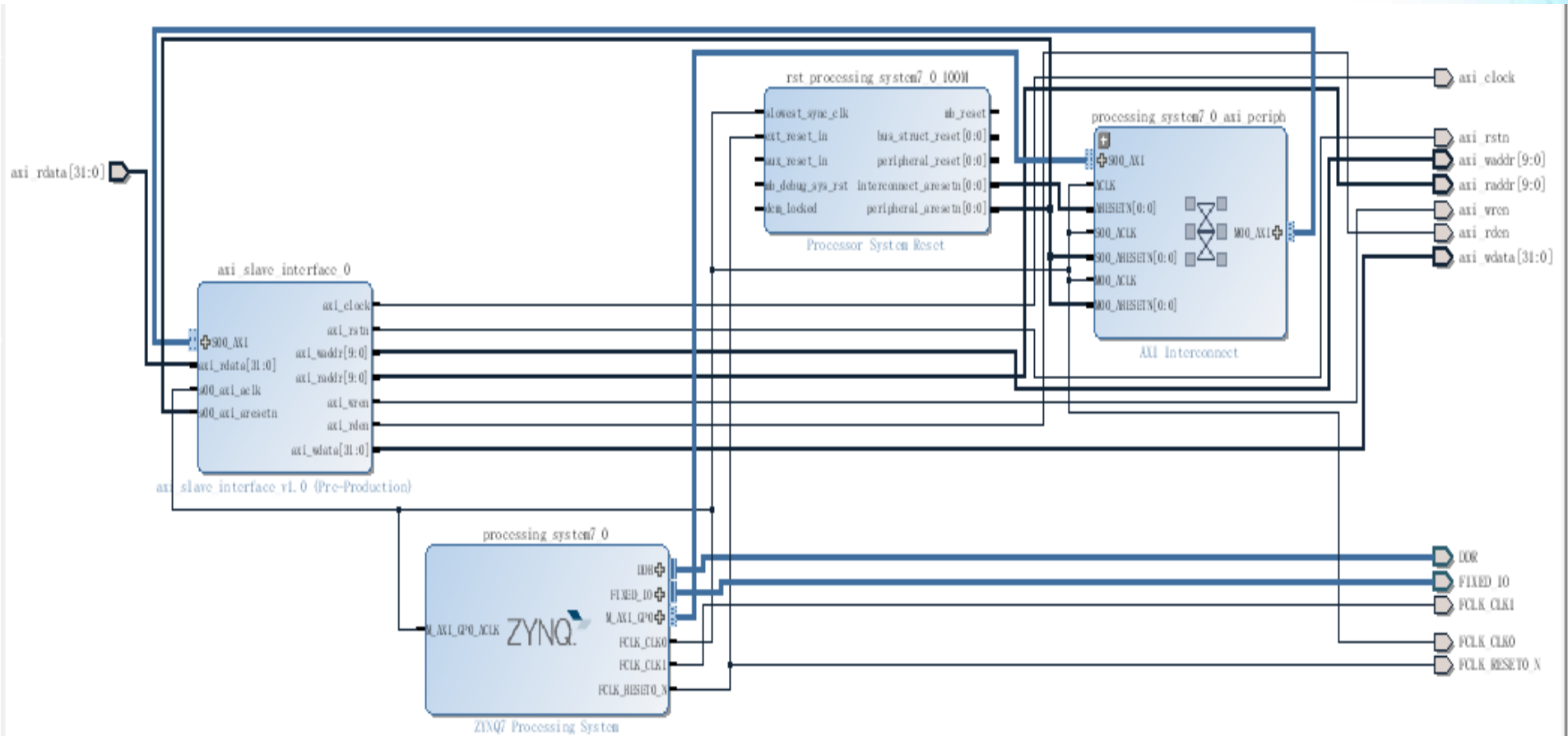
Four Channel 500MSPS ADC FMC Card



Base Board: ZYNQ7035



ARM Core Construction



FPGA Logical in ZYNQ PL

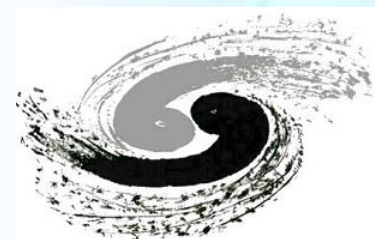
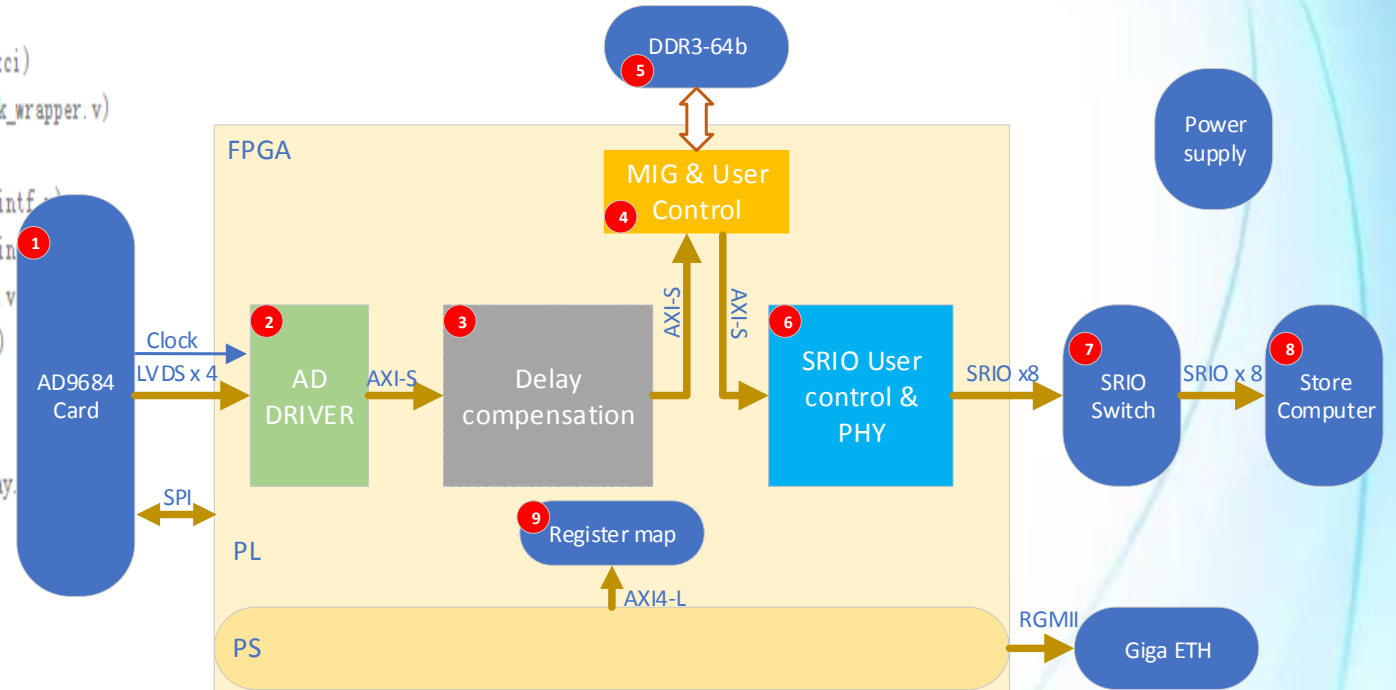


Design Sources (5)

- [-] top_prj (top_prj.v) (10)
 - [+] u_clk_iodelay - clk_iodelay (clk_iodelay.xci)
 - [+] u_top_sdk_wrapper - top_sdk_wrapper (top_sdk_wrapper.v)
 - [+] u_axi_slave - axi_slave (axi_slave.v) (3)
 - [+] u_adc1_lvds_intf - adc_lvds_intf (adc_lvds_intf.v)
 - [+] u_adc2_lvds_intf - adc_lvds_intf (adc_lvds_intf.v)
 - [+] u_adc_sync_gen - adc_sync_gen (adc_sync_gen.v)
 - [+] u_adc_buffer - adc_buffer (adc_buffer.v) (3)
 - [+] u_ila_adc1 - ila_adc (ila_adc.xci)
 - [+] u_ila_adc2 - ila_adc (ila_adc.xci)
 - [+] u_vio_io_delay - vio_io_delay (vio_io_delay.v)
- [+] top_quad (top_quad.v) (1)
- [+] adc_dco_pll (adc_dco_pll.xci)
- [+] ila_trig (ila_trig.xci)
- [+] top_adc (top_adc.v)

Constraints (1)

Simulation Sources (7)



Using PS code configure FPGA



The screenshot displays the Xilinx IDE interface. On the left, the Project Explorer shows a project structure with folders like 'drivers' and files like 'ps7_init.c'. The main editor shows the source code for 'main.c', which includes headers and defines a main function that prints board information and initializes the platform. Below the editor, the 'SDK Log' window shows a series of informational messages from 20:38:55 to 20:39:56, detailing the registration of command handlers, launching of the XSCF server, and successful completion of system reset and FPGA configuration.

On the right side, the 'SDK Terminal' window shows the serial output from the device connected to COM6 at 115200 baud. The output includes connection status, board ID (0x19830411), and vendor IDs for boards ad9510 and ad9684.

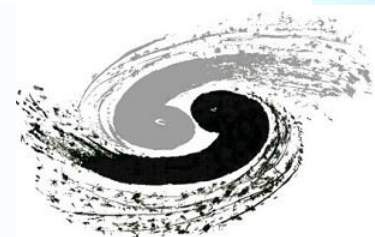
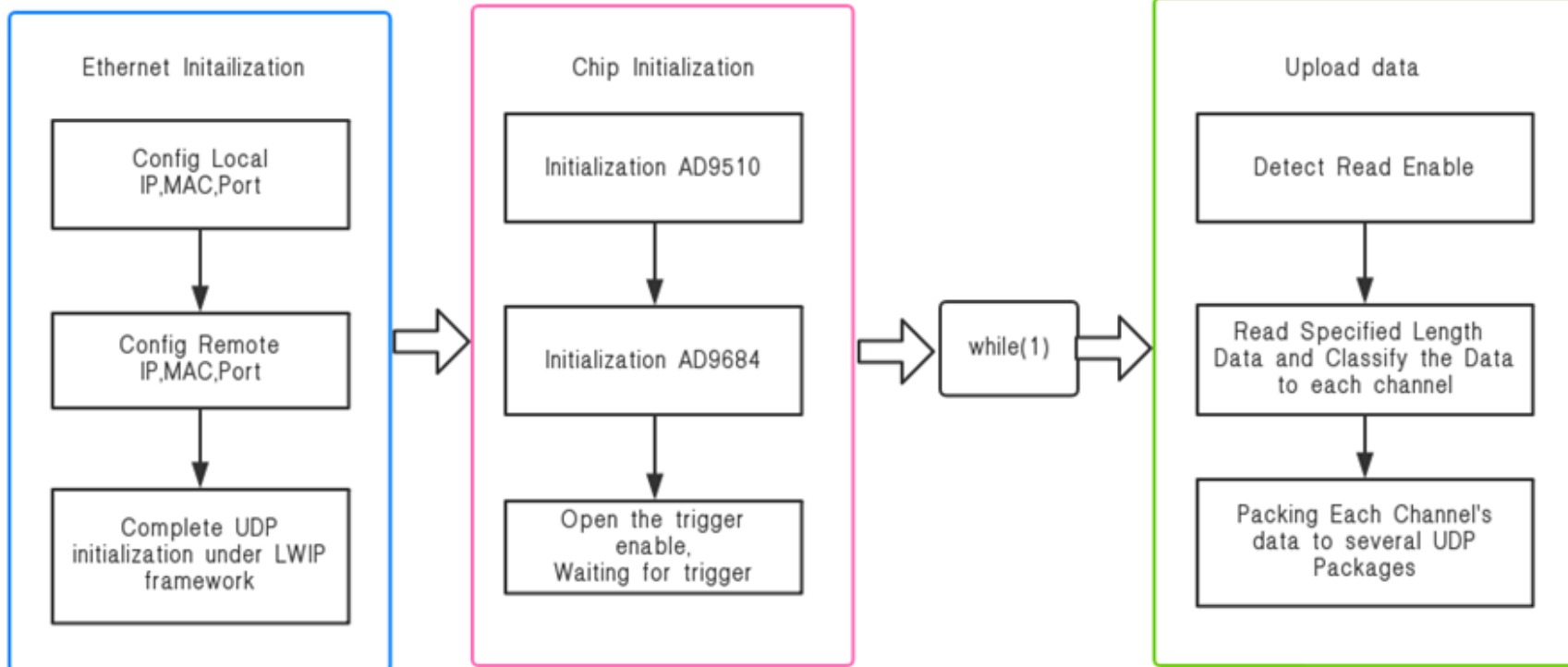
```
Connected to: Serial ( COM6, 115200, 0, 8 )

Connected to COM6 at 115200
Start to send UDP packet ...
Initial Tx Counter : 0 ...

dev id = 0x19830411
ad9510 0x00 : 0x18
ad9684 vender id : 0x456
ad9684 vender id : 0x456
```

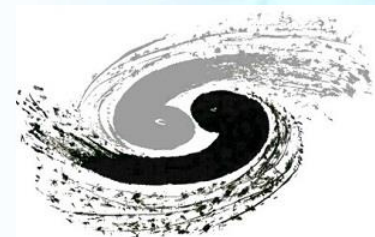
20:38:55 INFO : Registering command handlers for SDK TCF se
20:38:56 INFO : Launching XSCF server: xsct.bat -interactiv
20:39:01 INFO : XSCF server has started successfully.
20:39:01 INFO : Successfully done setting XSCF server connec
20:39:01 INFO : Processing command line option -hwspec E:/A
20:39:01 INFO : Successfully done setting SDK workspace
20:39:26 INFO : System reset is completed.
20:39:55 INFO : FPGA configured successfully with bitstream "E:/AD9684/AD9684_Sync.sdk/top_prj_hw_platform_0/top_prj.bit"
20:39:56 INFO : ps7_init is completed.
20:39:56 INFO : ps7_post_config is completed.

ARM Code in ZYNQ PS





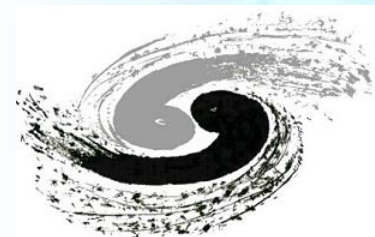
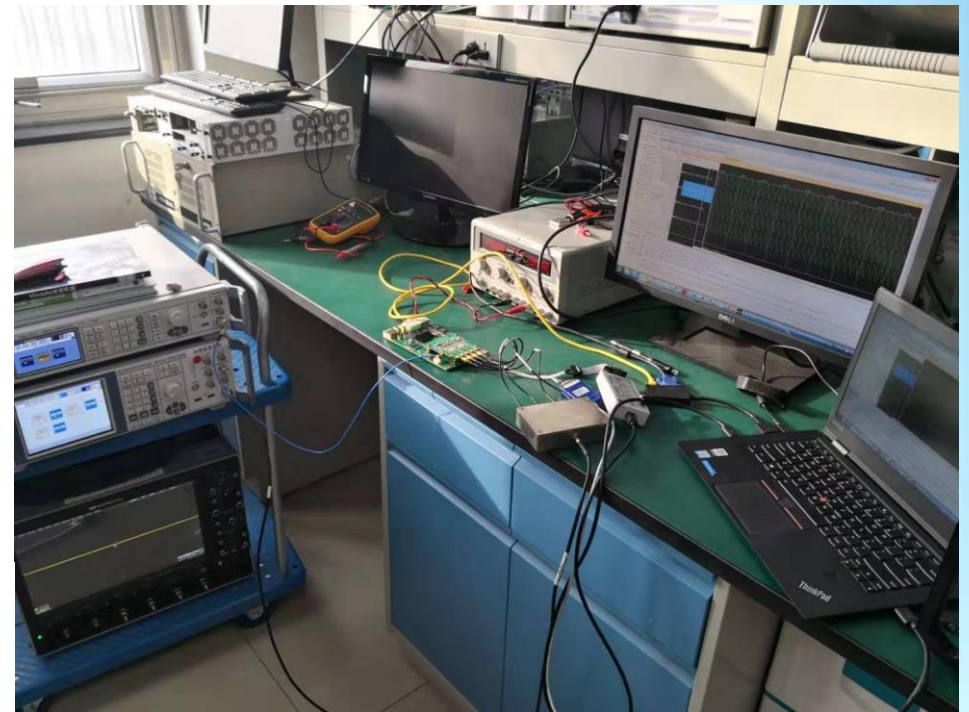
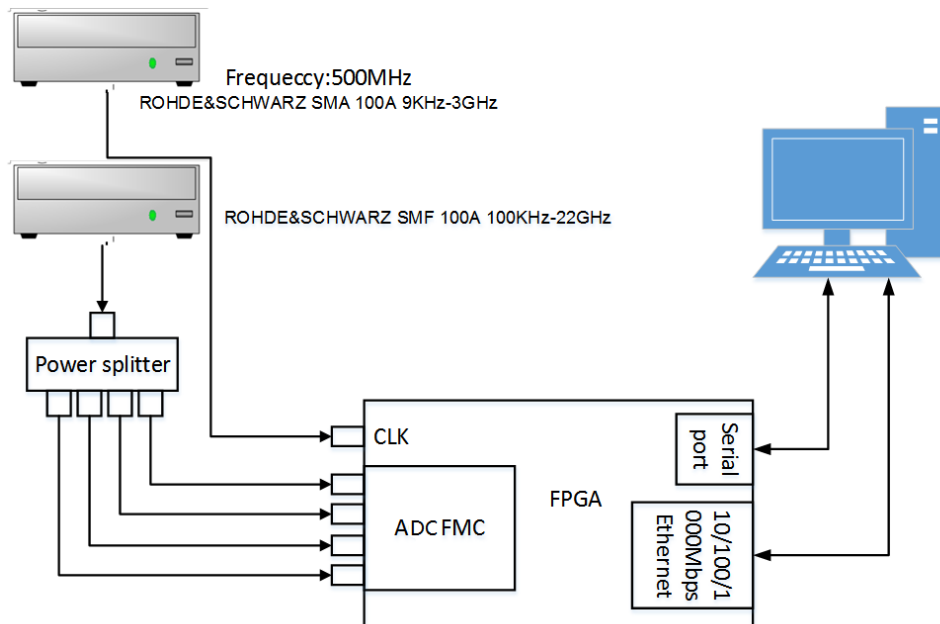
Test in Lab



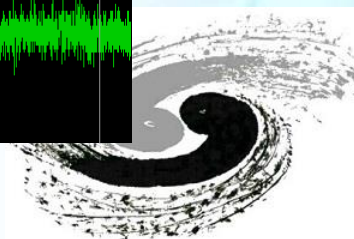
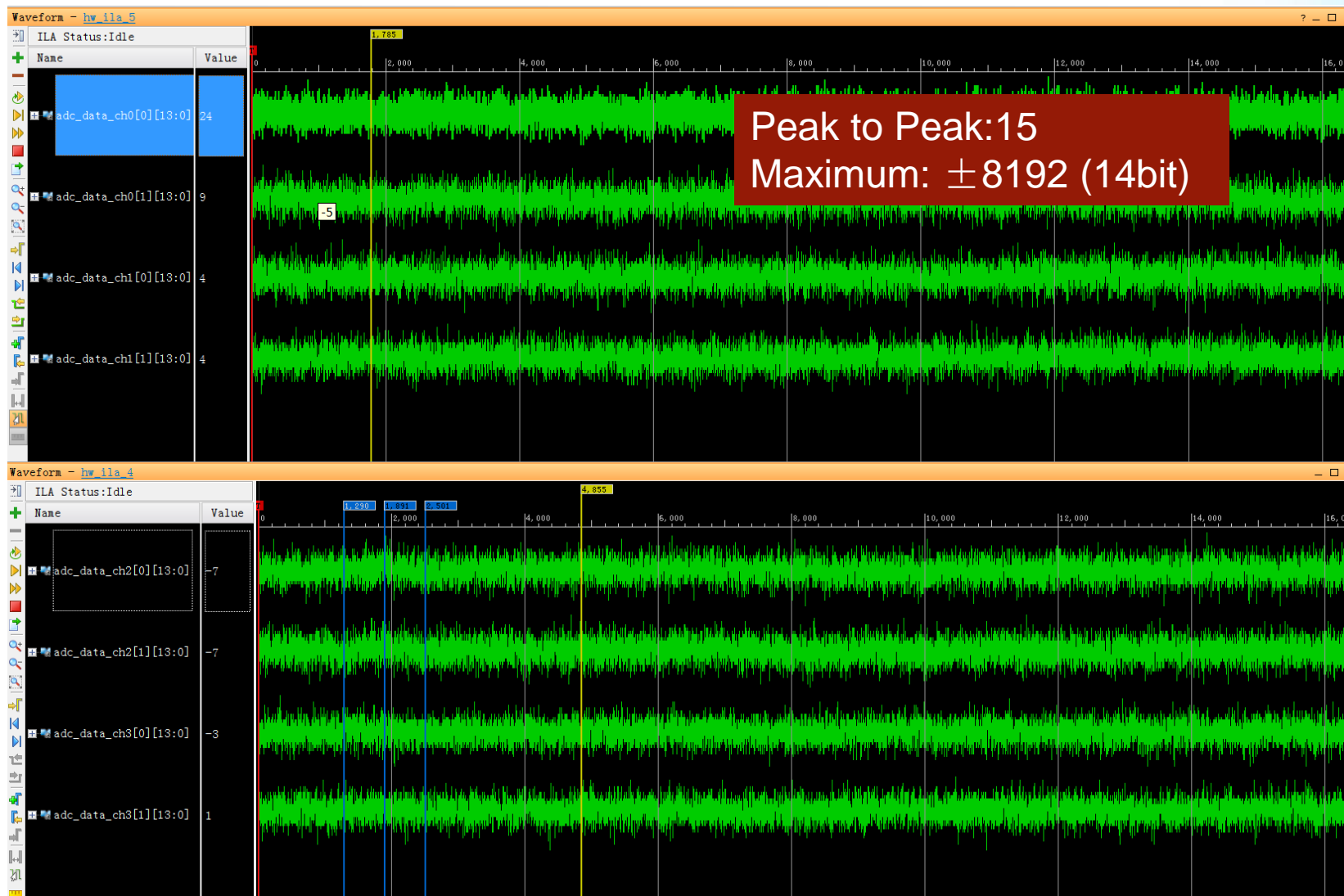
Test Environment

ADC CLK: 500MHz (Convenient to synchronous two signal generator)
(CLK: From ROHDE&SCHWARZ SMA 100A 9KHz-3GHz)

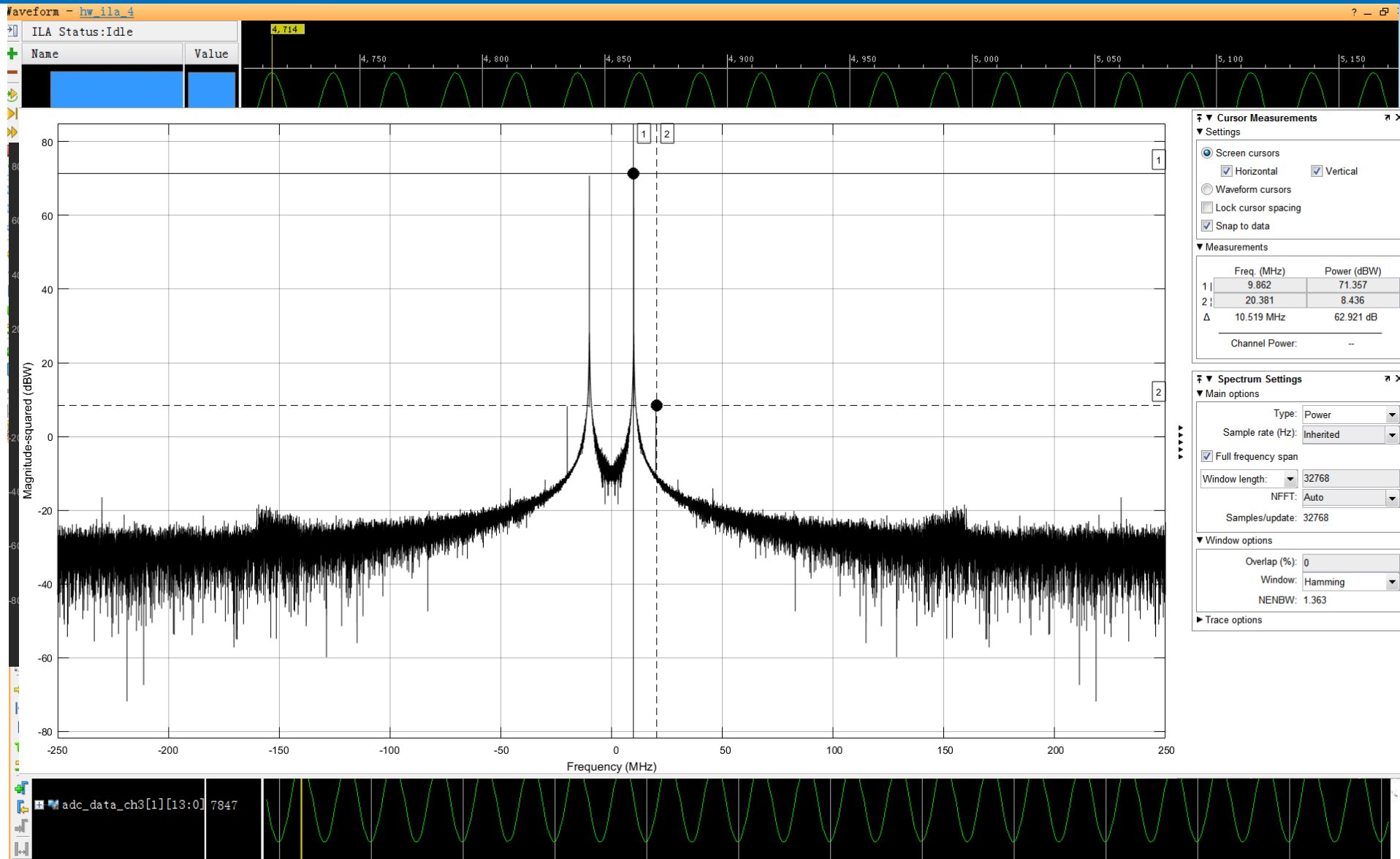
Four Channel ADC Input: One to Four Power Splitter, Signal Source:
(ROHDE&SCHWARZ SMF 100A 100KHz-22GHz)



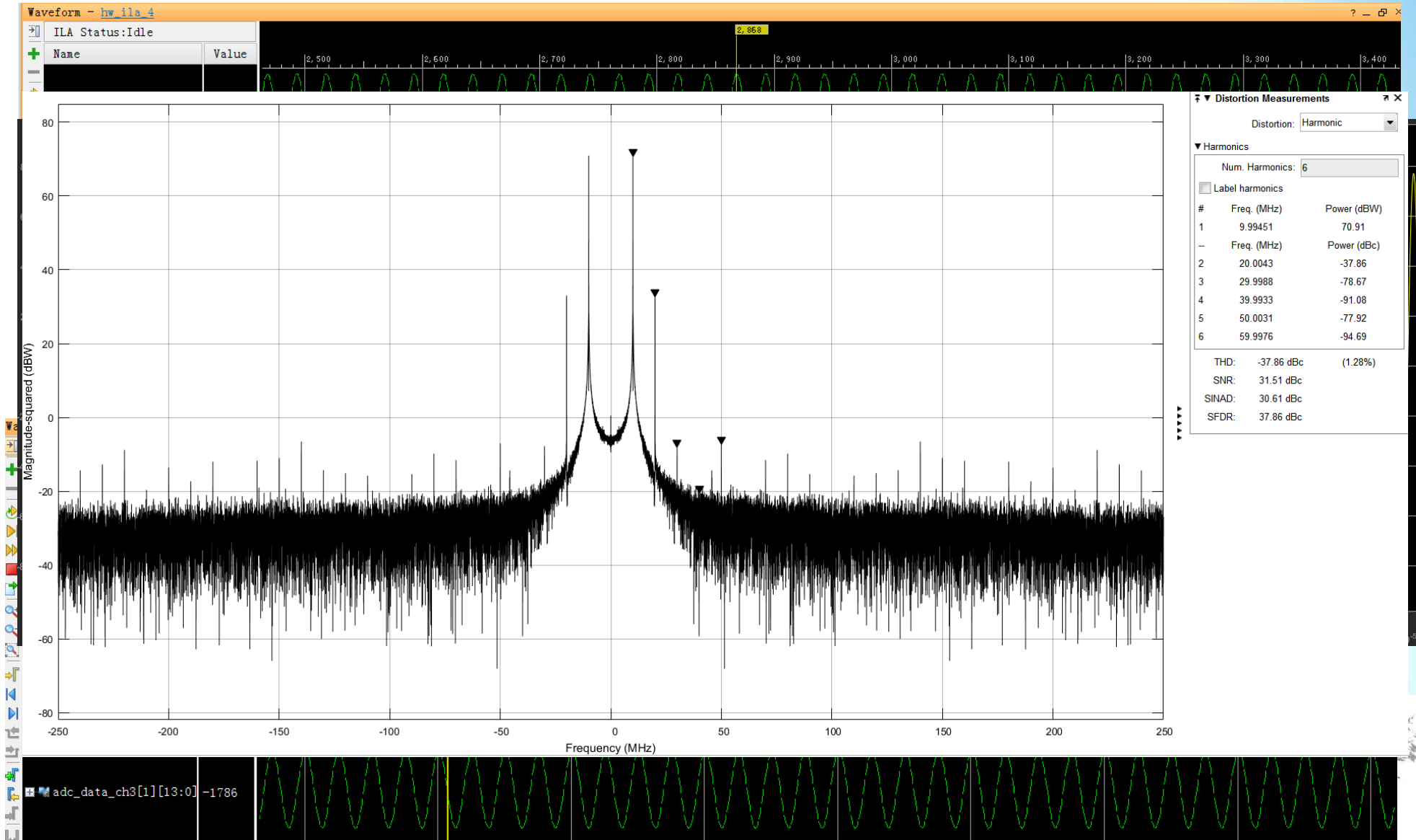
No Input, ADC Raw data



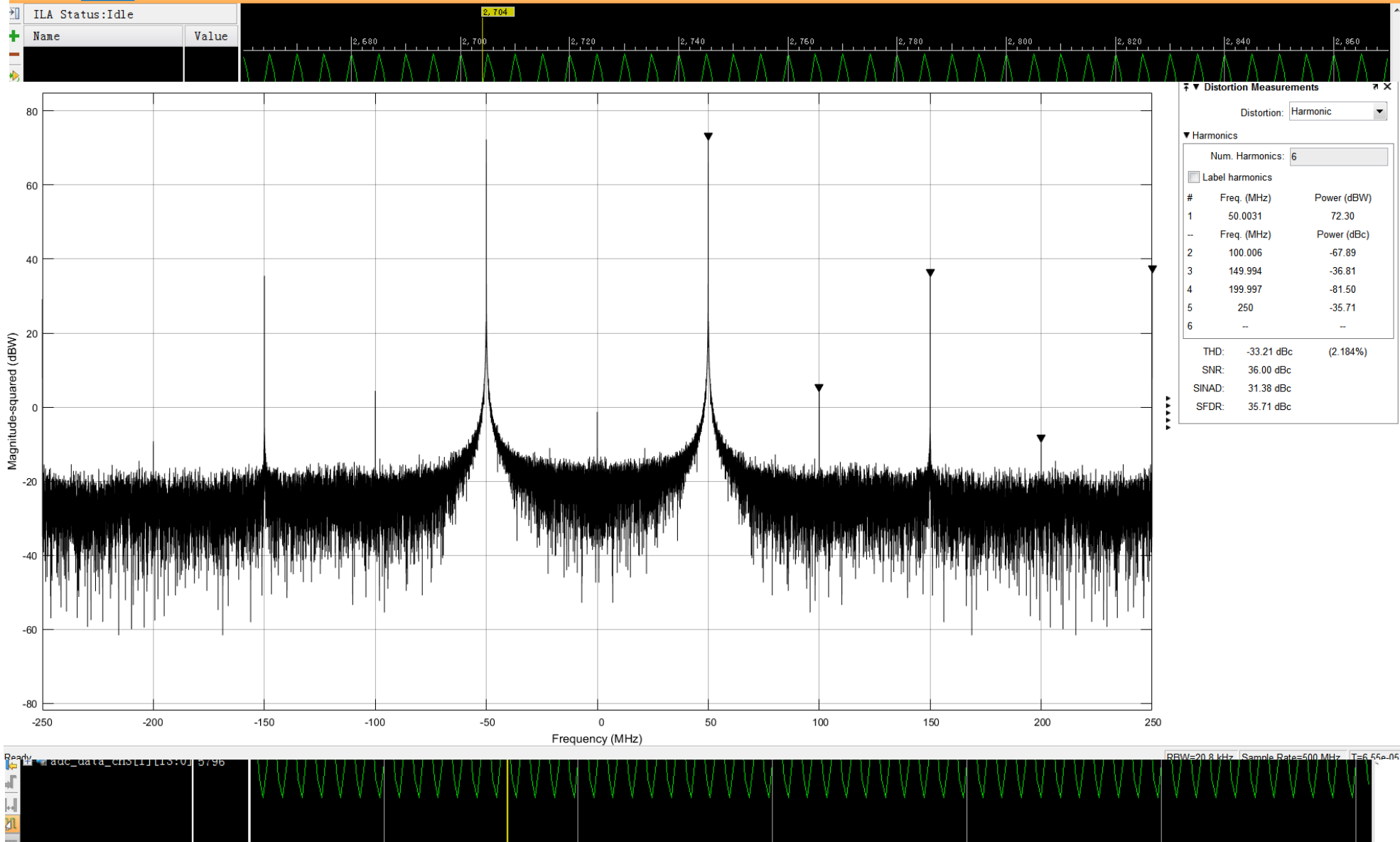
10MHz Input



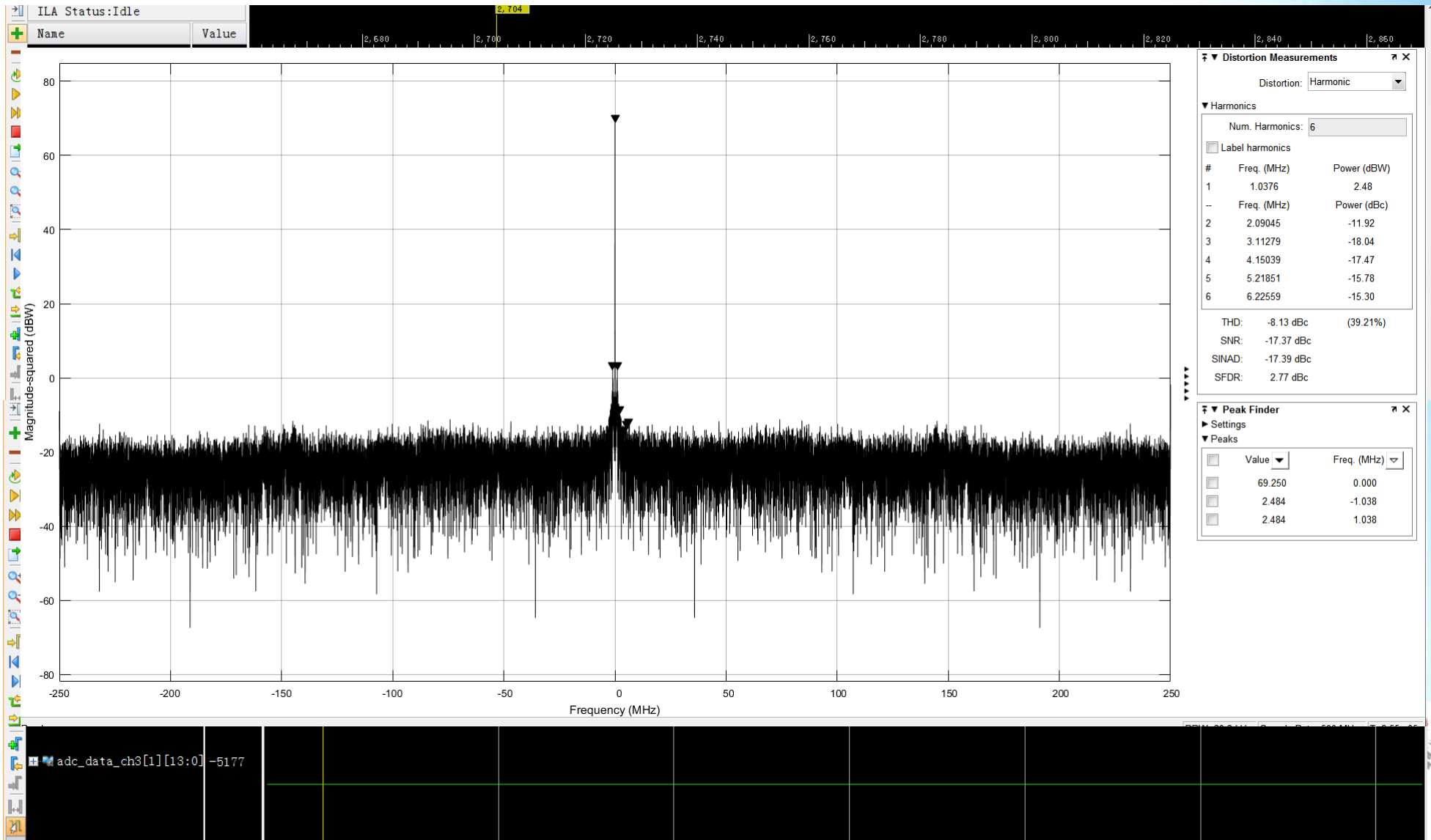
490MHz Input



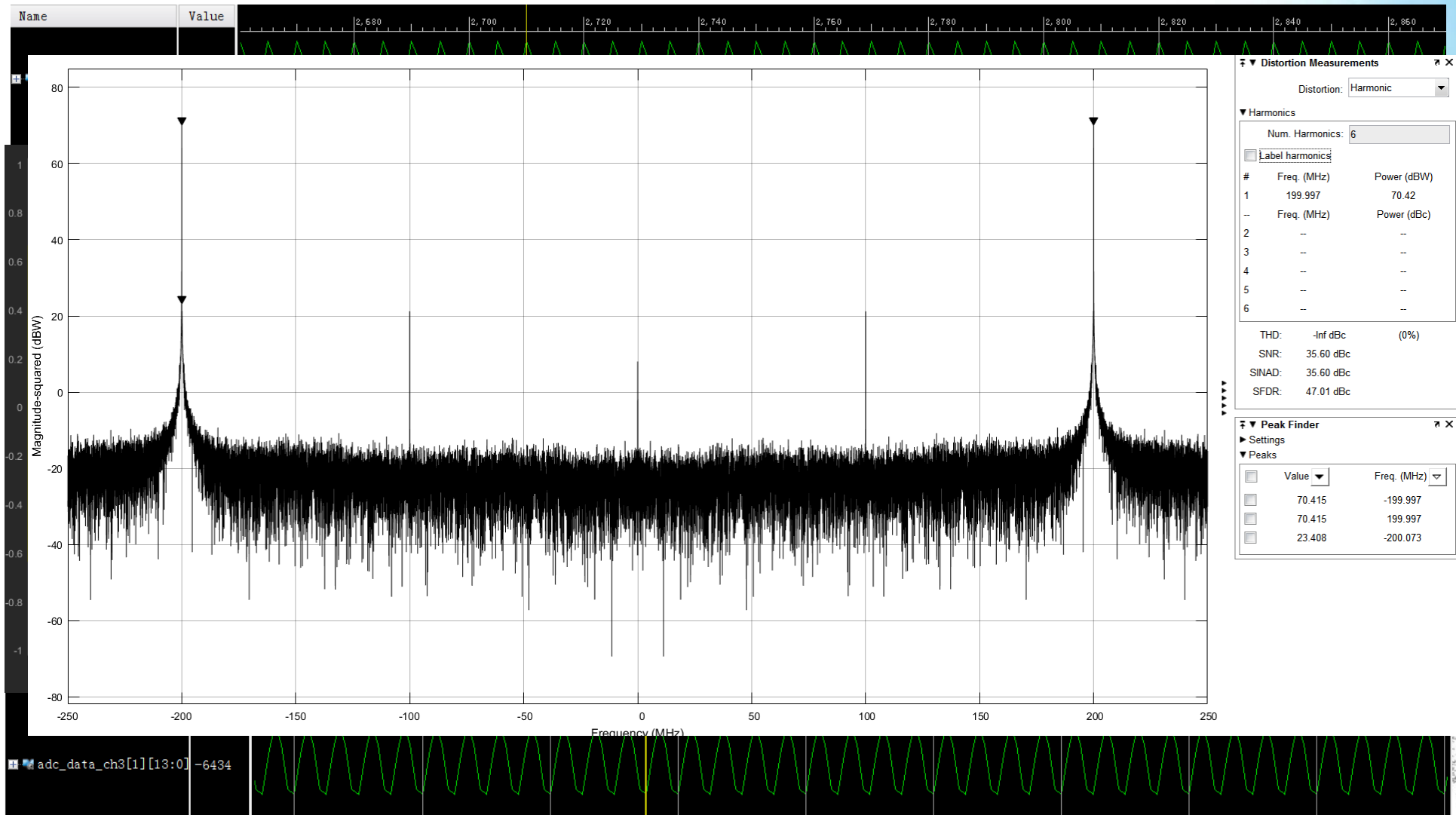
950MHz Input



1.5GHz Input



1.8GHz Input



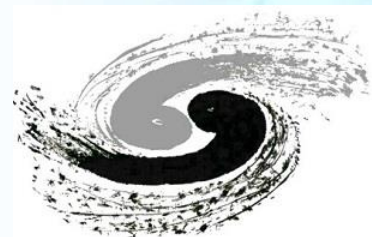
Summary



- ✓ Bunch-by-Bunch BPM system has been designed.
- ✓ Prototype circuit board has been made.
- ✓ Initial board tests have started.

Next Step:

- ✓ **Develop** : Embedded Linux and EPICS on ZYNQ
ARM
- ✓ **Furthermore**: Upgrading Hardware and firmware,
adding embedded Timing module...



Thank you!

