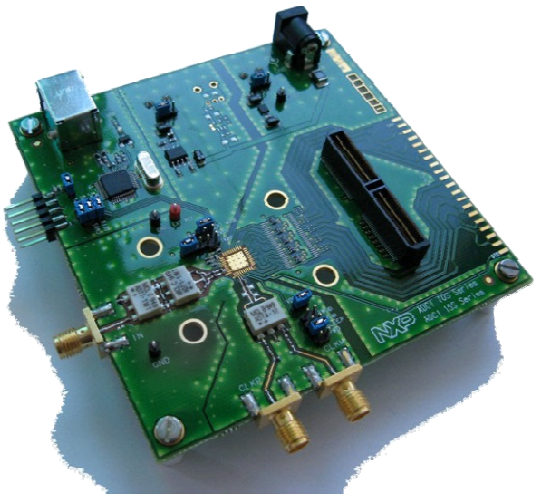
# Quick start ADC1610S series (F1 or F2 versions)

## Demonstration board for ADC1610S series

**Rev. 5 — January 2011** **Quick start**

### Document information

| Info | Content |
|---|---|
| **Keywords** | PCB2131-1, Demonstration board, ADC, Converter |
| **Abstract** | This document describes how to use the demonstration board for the analog-to-digital converter ADC1610Sseries. |
| **Overview** |  |

## Revision history

| Rev | Date | Description |
|---|---|---|
| 1 | 20081001 | Initial version. |
| 2 | 20090518 | Update |
| 3 | 20090610 | Add SPI software description. |
| 4 | 20100519 | Add HSDC extension module acquisition system description. |
| 5 | 20110120 | Update with latest software tool. |

# 1. Overview of the ADC1610S demo board

## 1.1 ADC1610S F1 series (CMOS digital outputs)

Figure below presents the connections to measure ADC1610S.

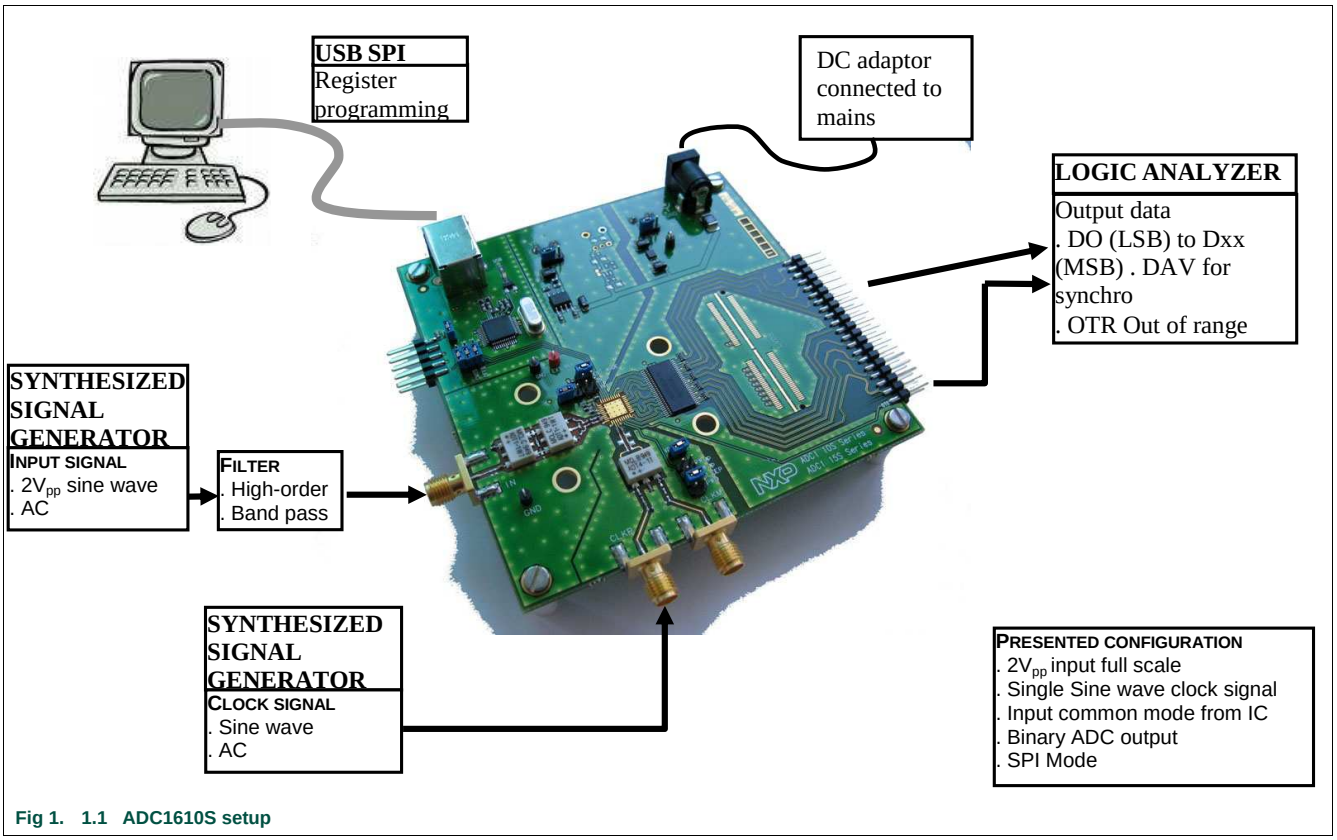



**Fig 1. 1.1 ADC1610S setup**

## 1.2 ADC1610S F2 series (LVDS/DDR digital outputs)

Figure below presents the connections to measure ADC1610S.

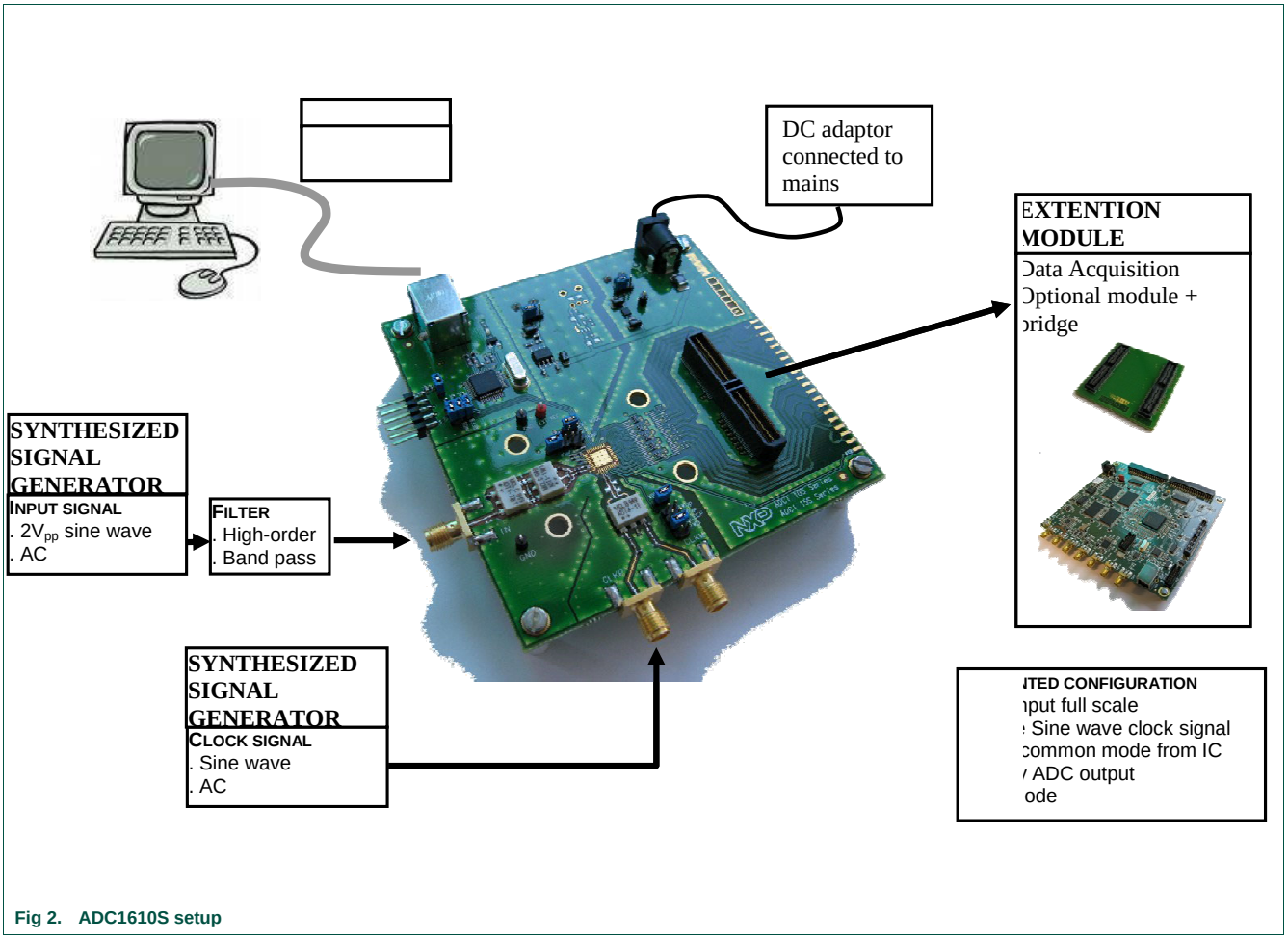
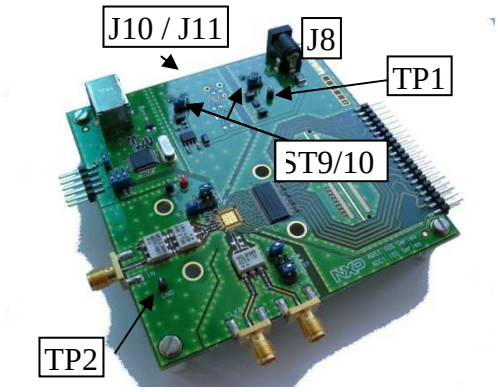



**Fig 2. ADC1610S setup**

## 1.3 Power supply

The board is powered either with a 3 $V_{DC}$ and 1.8/3 $V_{DC}$ power supplies or a 5V DC adaptor.

**Table 1. Power supply**

| Name | Function | View |
|---|---|---|
| J8 | 2.1 Jack connector – 5VDC<br>**Change ST9 and ST10 position accordingly** |  |
| J10 | +3V green connector – Power supply 3 $V_{DC}$ | |
| J11 | **Change ST9 and ST10 position accordingly**<br>CMOS version<br>+1.8V green connector – Power supply 1.8 $V_{DC}$<br>LVDS DDR version<br>+3V green connector – Power supply 1.8 $V_{DC}$ | |
| TP1 | GND test point | |
| TP2 | GND test point | |

## 1.4 Input signals (IN, CLK)

The input clock signal can be either a sine wave or a LVCMOS signal.

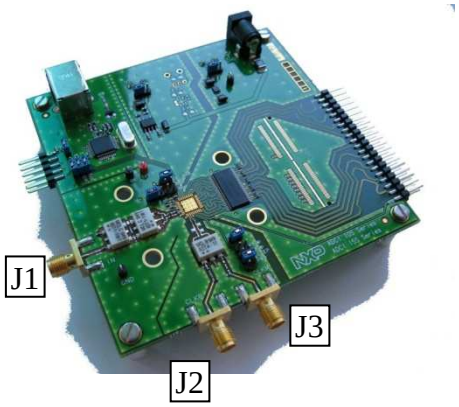
To ensure a good evaluation of the device, the input signal and the input clock must be synchronized together.

Moreover, the input frequency (Fi, MHz) and the clock frequency (Fclk, Msps) should follow the formula:

$$\frac{Fi}{Fclk} = \frac{M}{N}$$

, where M is an odd number of period and N is the number of samples.

**Table 2. Input signals**

| Name | Function | View |
|---|---|---|
| J1 | IN connector – Analog input signal (50Ω matching) |  |
| J2 | CLKP connector – Single ended clock input signal (50Ω matching), with a transformer. | |
| J3 | CLKM connector – Grounded on that demoboard | |

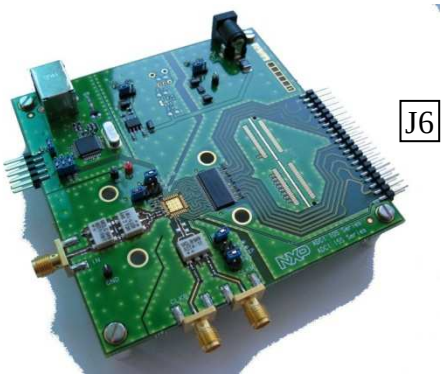


## 1.5 Output signals in CMOS version (D0 to D15, DAV, OTR)

The digital output signal is available in binary, 2's complement or gray format.

A Data Valid Output clock (DAV) is provided by the device for the data acquisition.

Table 3. **Output signals**

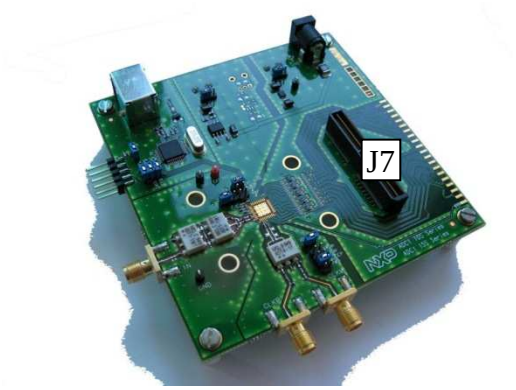
| Name | Function | View |
|---|---|---|
| J6 | Array connector – ADC digital output (D0 to D15), OTR and Data Valid (DAV) |  |

## 1.6 Output signals in LVDS DDR version

The digital output signal is available in binary, 2's complement or gray format.

A Data Valid Output clock (DAV) is provided by the device for the data acquisition.

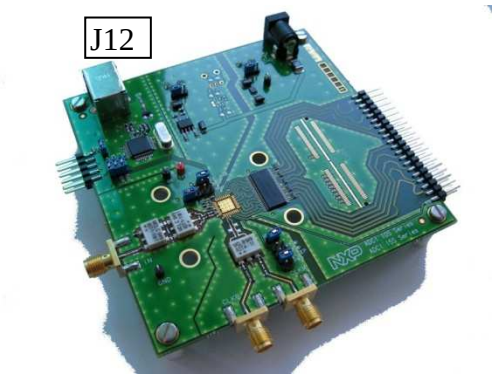
Table 4. **Output signals**

| Name | Function | View |
|---|---|---|
| J7 | Samtec QTH connector – ADC digital output (D0 to D15) and Data Valid (DAV) |   |

## 1.7 SPI Mode

The ADC1610S can be **controlled** either by a Serial Peripheral Interface (SPI) or by PIN**.**

Table 5. **SPI Interface**

| Name | Function | View |
|---|---|---|
| J12 | USB connector – SPI interface |  |



## 1.8 SPI program

For more details on how to control device with SPI, refer to section 3.3.

## 2. HSDC extension module: acquisition board

The figure 4 shows an overview of the extension module HSDC-EXTMOD01/DB acquisition board:

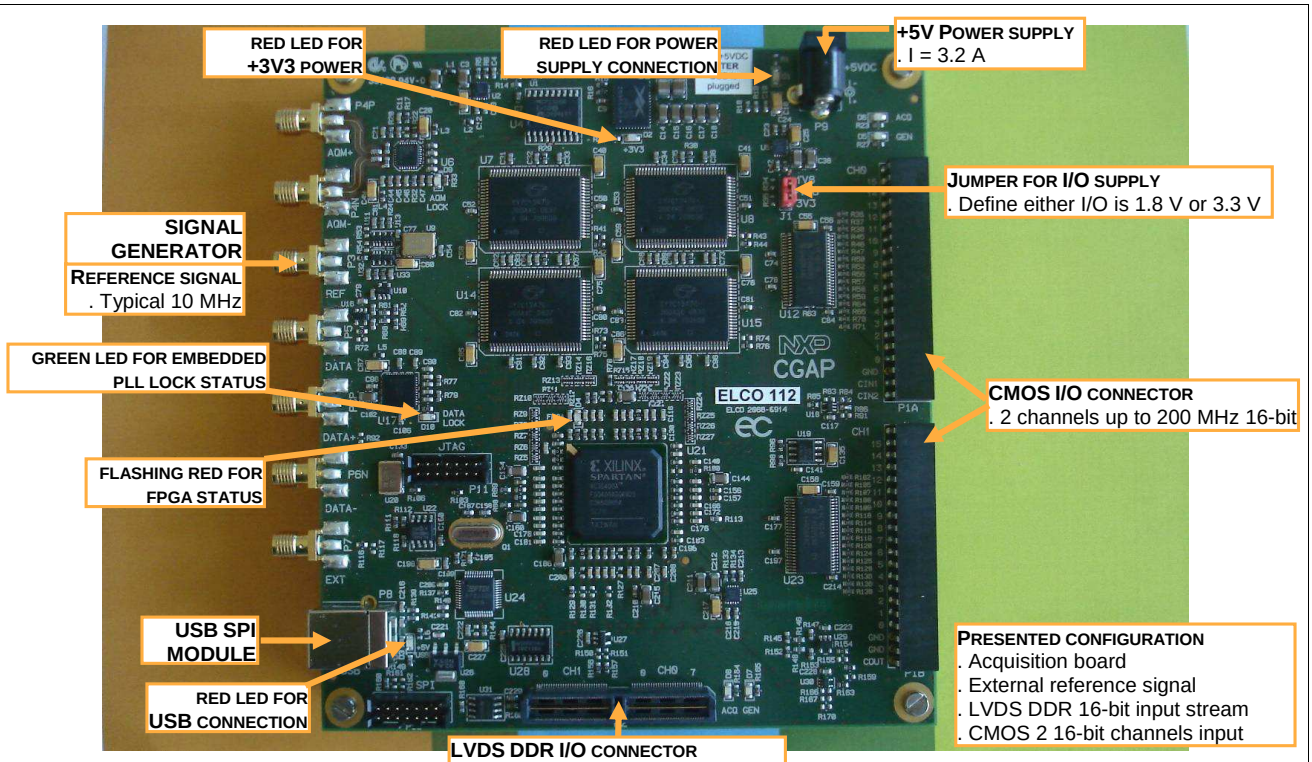



**Fig 3. HSDC extension module: acquisition board**

The HSDC extension module is intended for acquisition/generation and clock generation purpose. When connected to an ADC demo-board it is intended as an acquisition system for digital output bits delivered by ADC, either CMOS (HE14 P1 connector) or LVDS DDR (SAMTEC QTH_060_02 P2 connector).

The board brief specification is shown below:

- 32 MB memory size for acquisition pattern;
- 2 16-bit channels CMOS up to 200 MHz;
- 16-bit LVDS DDR input data stream up to 320 MHz;
- On-board or external reference for signal generation.

In this section the specific requirement for the use with ADC1610S demo-board will be shown.

For more details on the HSDC-EXTMOD01/DB, please contact dataconverter-support@nxp.com.

## 2.1 HSDC extension module: hardware initialization

Before using the generation board, make sure that you connect the USB cable **prior to** the supply.

When USB and power cable are connected, the HSDC-EXTMOD will light 3 red LEDs.

The green LED close to the PLL is only when it is locked (see section 3.3).

The red LED close to FPGA reports normal behavior when flashing ¼ on, ¾ off. Any other flashing behavior reports a failure at initialization (see section 3.3).

## 2.2 HSDC extension module: CMOS connector description

The figure 5 shows a brief description of the hardware connection on the HE14 connector.

For proper use of the acquisition board, make sure that resistor R86 (0 Ω) is connected while R84 is removed.

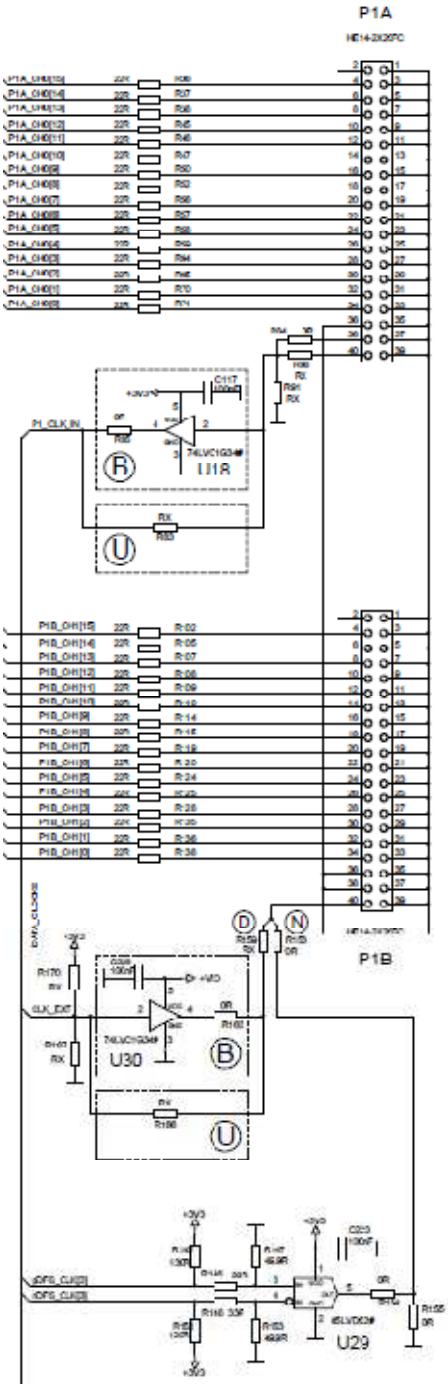



**Fig 4. HSDC extension module: HE14 CMOS hardware schematic overview**

# 3. Combo 1610S and HSDC extension module

## 3.1 ADC1610S setup CMOS outputs

The figure 24 below shows an overview of the whole system ADC1610S+HSDC extension module with CMOS outputs configuration for which connection is straightforward, together with a supply extension module (release A) for the ADC1610S demo-board:

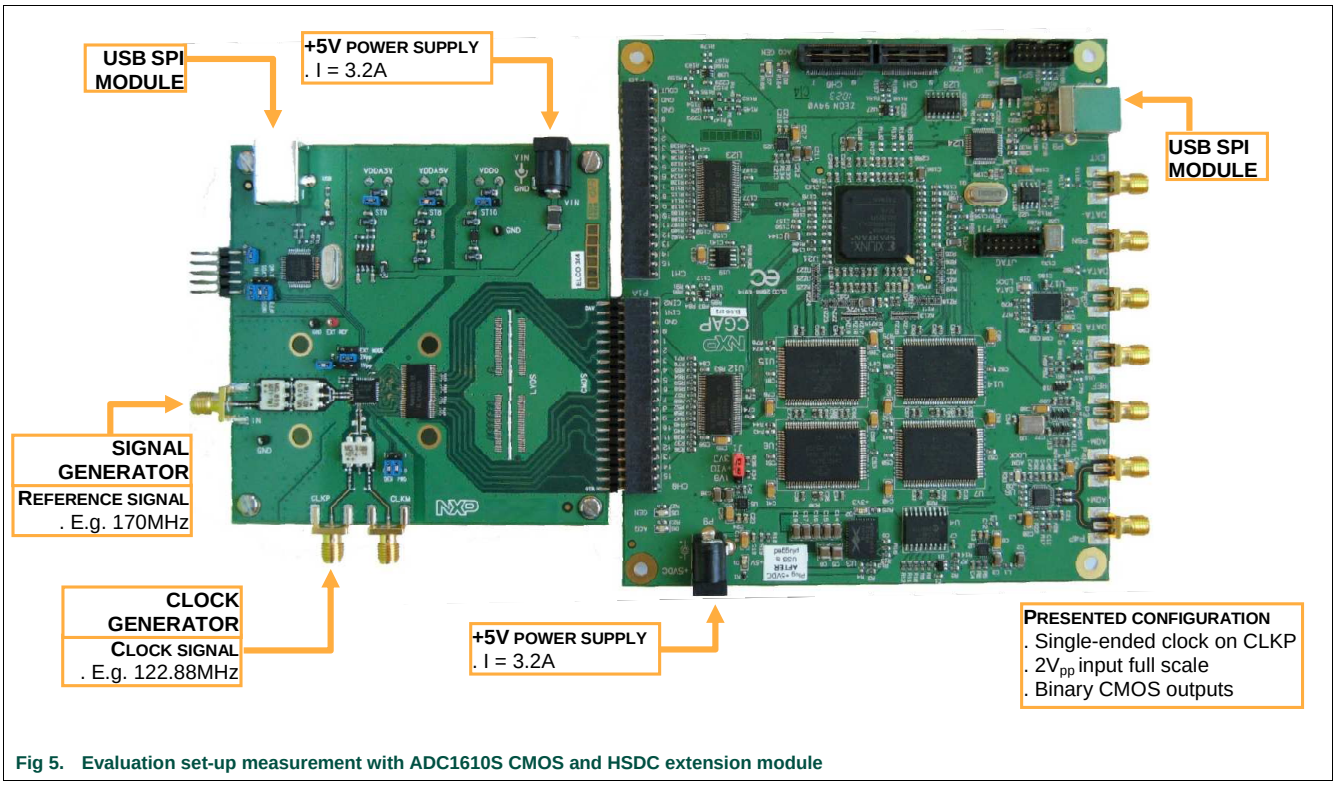



**Fig 5. Evaluation set-up measurement with ADC1610S CMOS and HSDC extension module**

## 3.2 ADC1610S setup LVDS/DDR outputs

The figure 24 below shows an overview of the whole system ADC1610S+HSDC extension module with CMOS outputs configuration for which connection is straightforward, together with a supply extension module (release A) for the ADC1610S demo-board:

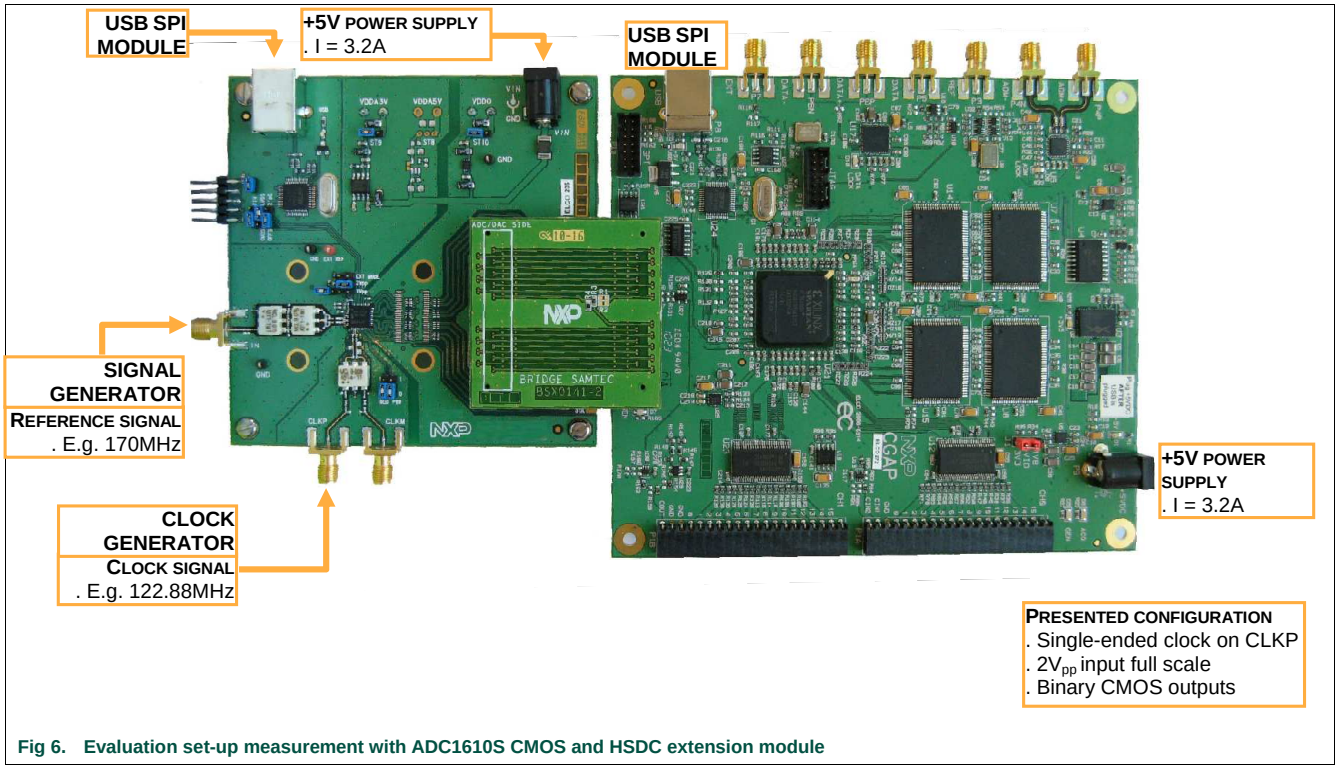



**Fig 6. Evaluation set-up measurement with ADC1610S CMOS and HSDC extension module**

## 3.3 ADC Software tool

Run the application “SW_ADC_1_r02.exe”. This application will allow:

- the user to control features on our high speed ADC through the SPI interface available on any ADC1610S series;
- As well as performing any online data acquisition to evaluate the performances of the ADC1610S series.

At start-up, the program will detect any board connected to your system and display information as can be seen on following window:

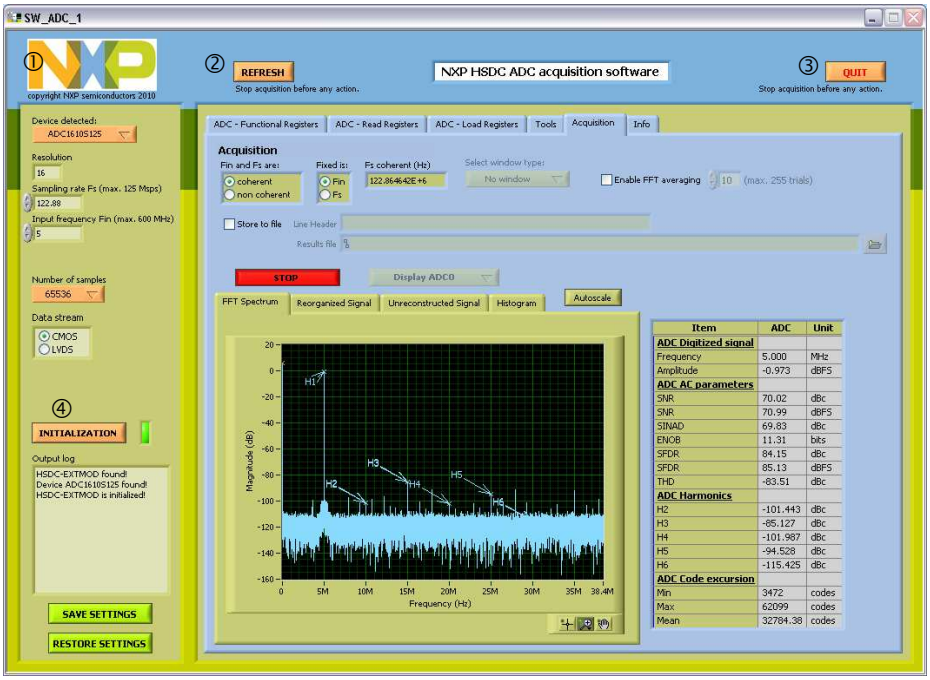



**Fig 7. SW_ADC_1_r02: start-up screen**

①: “NXP Banner Button” will display your default internet browser to the NXP data converter home page;

②: “REFRESH” allows you to poll your system for any hardware change. It will reset any board connected to your system;

③: “QUIT” allows you quit the application;

④: “INITIALIZATION” allow you to initialize the HSDC-EXTMOD board prior to any acquisition task.

In the example above, the HSDC-EXTMOD has been detected, as well as ADC1610S125.

At this moment, make sure that 4 LEDs are visible on the HSDC-EXTMOD (2 close to power plug, 1 for USB and 1 close to FPGA).

The “Info” page gives more details on the current hardware configuration for the HSDC-EXTMOD board:

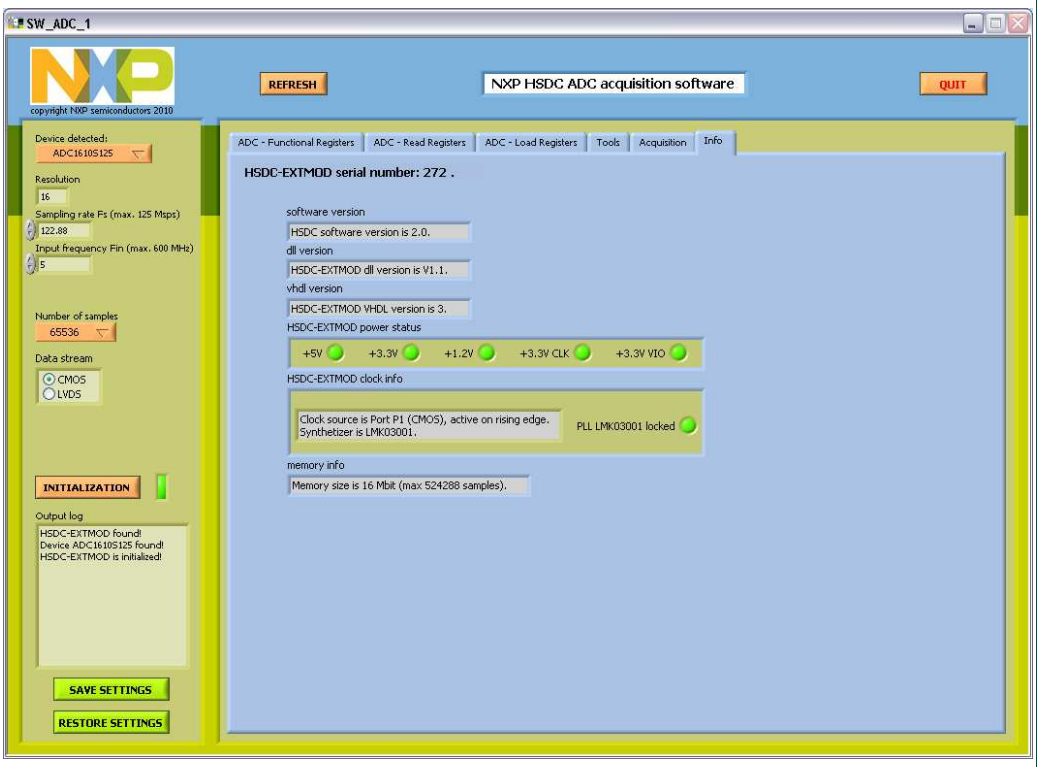



**Fig 8. SW_ADC_1_r02: “Info” page**

The HSDC-EXTMOD is not yet initialized, so the embedded PLL (LMK03001 in this example) is not locked. Initialization is only required for acquisition purpose.

### 3.3.1 ADC SPI programming Functional Registers page

The page displays all SPI registers for ADC1610S series:

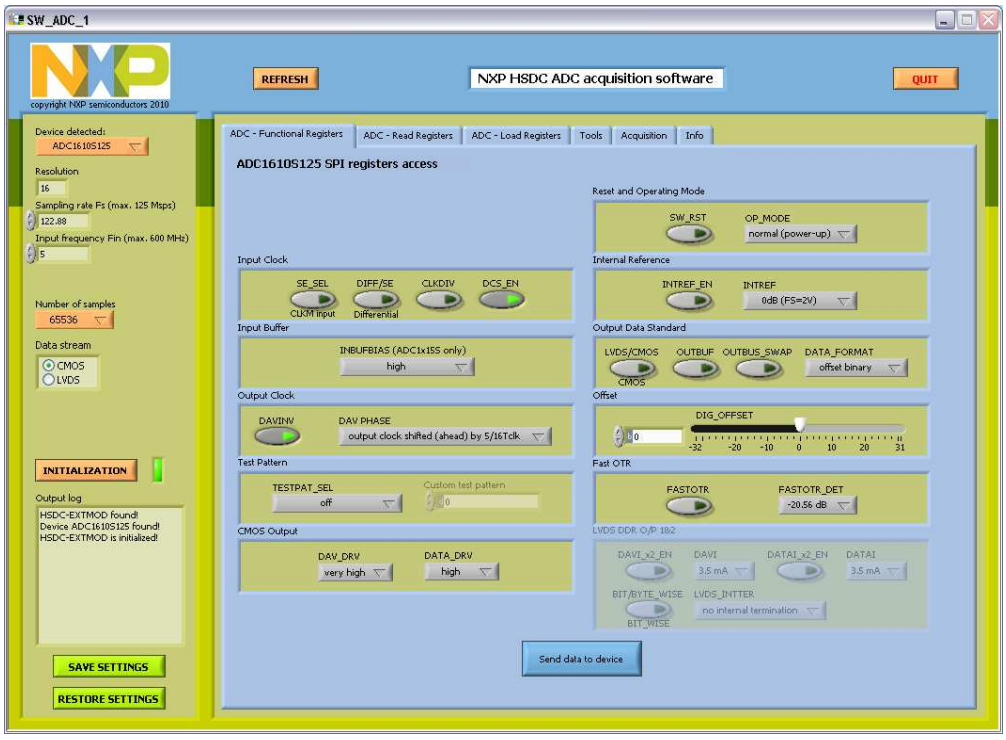



**Fig 9. SW_ADC_1_r02: “ADC - Functional Registers” page**

Perform any settings and then click on the “Send data to device” button to update the device registers.

### 3.3.2 ADC SPI programming Read Registers page

This page can be used to read all registers by clicking on the “Read all registers” button and will display the result in the table below:

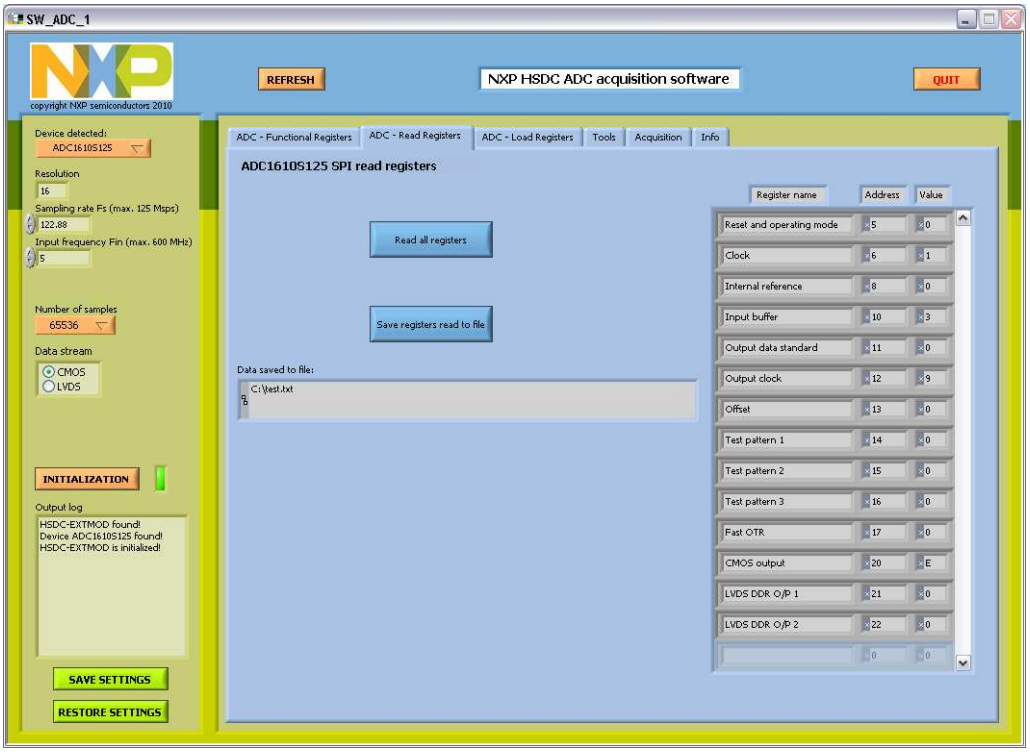



**Fig 10. SW_ADC_1_r02: “ADC - Read Registers” page**

When all registers have been read, it is possible to save the data to a text file. The settings are saved in a table-like format as shown below:

**Table 6. Typical saving on text file**

*Content of file is shown as table format*

| Column 1 | Column 2 |
|---|---|
| Address | Value |
| 05 | 00 |
| 06 | 01 |
| 08 | 00 |
| 10 | 03 |
| 11 | 00 |
| 12 | 09 |
| 13 | 00 |
| 14 | 00 |
| 15 | 00 |
| 16 | 00 |
| 17 | 00 |

| Column 1 | Column 2 |
|---|---|
| 20 | 0e |
| 21 | 00 |
| 22 | 00 |

Note that all data are saved in hexadecimal format.

Click on the “Save registers read to file” button to select the file to store data to. Make sure that you store your file with “.txt” extension, this will allow you to re-use the file on the “ADC - Load Registers” page.

### 3.3.3 ADC SPI programming Load Registers page

This page allows downloading configuration data to the device registers:

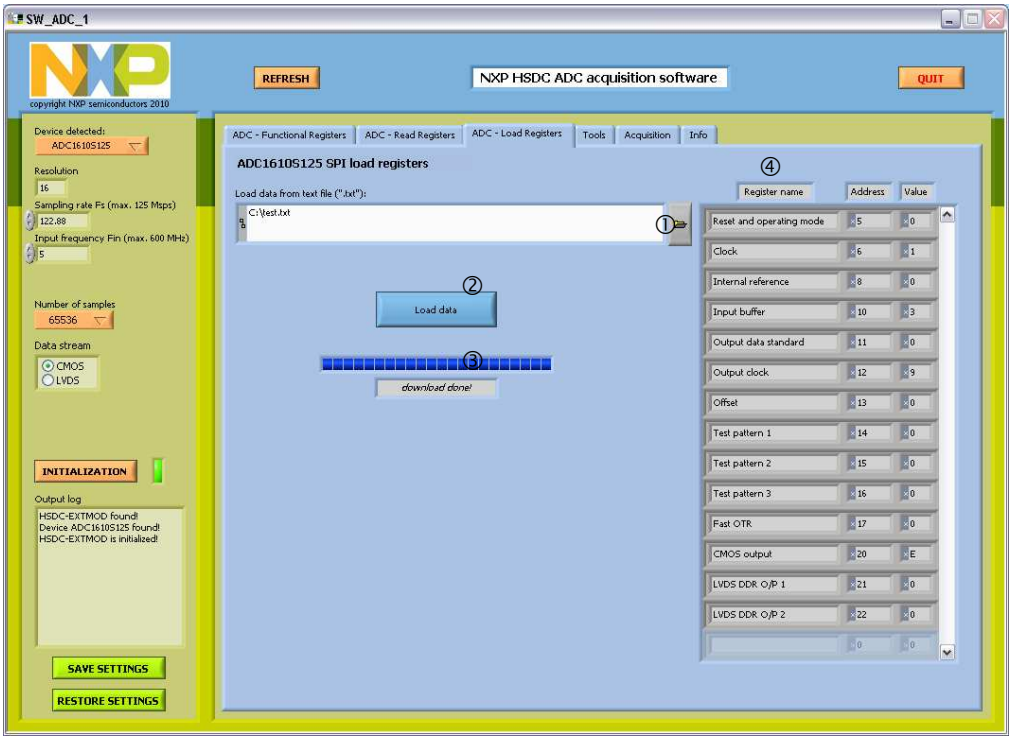



**Fig 11. SW_ADC_1_r02: “ADC - Load Registers” page**

It is not necessary to have a file that has the whole set of registers listed. The only restriction is regarding the formatting of the file as given in section 3.3.2.

Note: this page cannot be used to download data saved during the comparison process.

To download settings onto device registers, follow the procedure below:

- Browse to select your file (button ①);
- Click on “Load data” button②.

A message on field ③ and a progress bar will inform about the status of the operation until message “download done!” is seen. The table ④ is updated with the current values downloaded at the fly as can be seen on figure 13.

### 3.3.4 Tools page

This page allows the user to calculate the coherent frequencies values involved of the acquisition process. It gives an indication where the 6 first harmonics are located in the Nyquist zone.

Enter your analog and sampling frequencies in field①. Indicate the number of samples to be acquired②, as well as the fixed parameter for the coherency calculation (Fs in our example above③). Press “UPDATE GRAPH” to look at the frequency plan, it gives also the real Fin frequency (Refer to appendix A.1 for more details on coherency calculation):

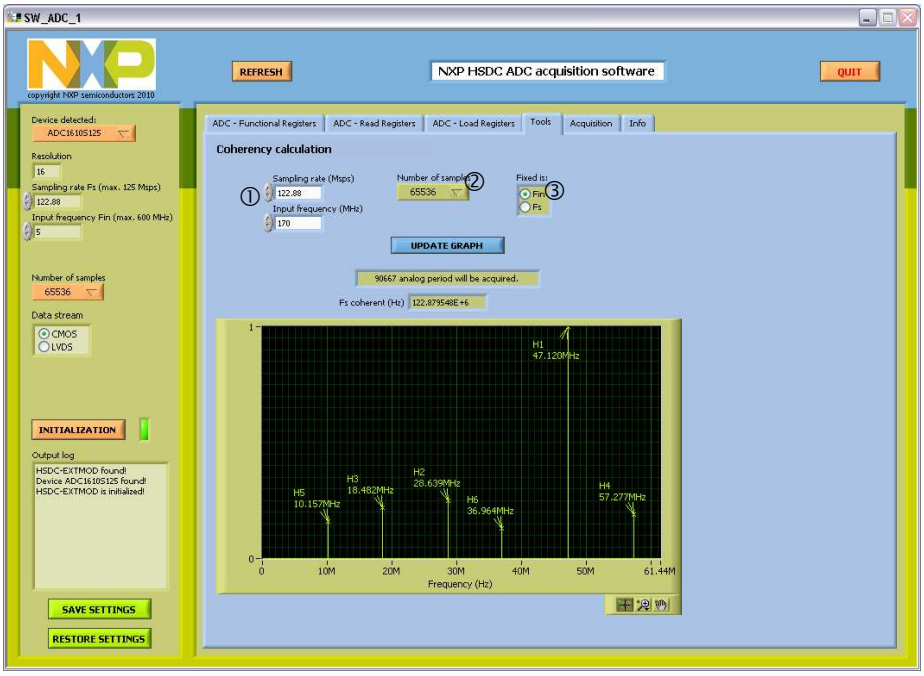



**Fig 12. SW_ADC_1_r02: “Tools” page**

Note: The level of the harmonics shown does not reproduce the behavior of the ADC; they are only given as indication for location.

### 3.3.5 Acquisition page

This page will acquire data to evaluate the high dynamic performance of the device:

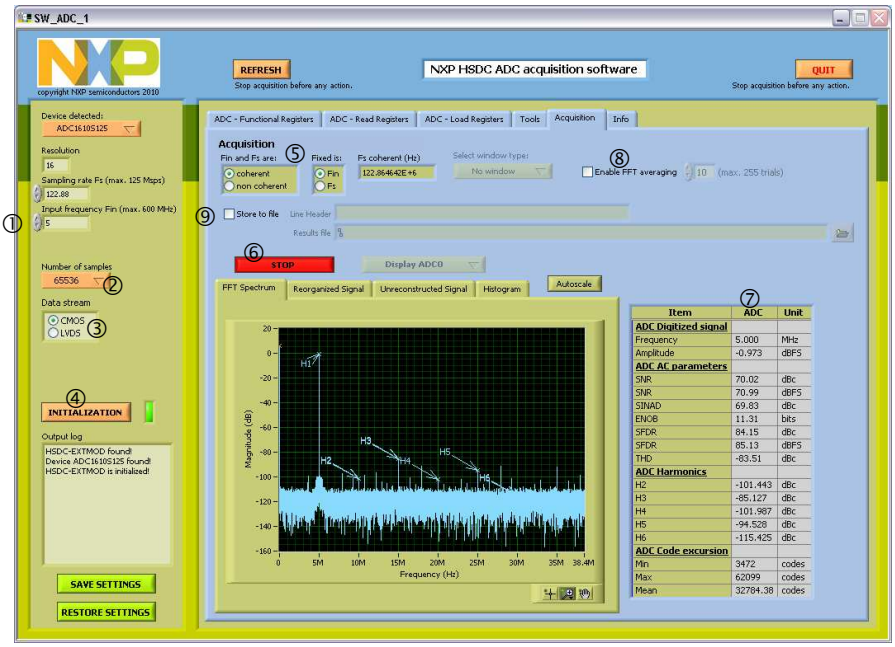



**Fig 13. SW_ADC_1_r02: "Acquisition" page**

Before proceeding to any acquisition, the user needs to do the following entries:

- the sampling frequency Fs: 122.88 Msps in our example (field ①);
- the input frequency Fin: 5 MHz in our example (field ①);
- the number of samples to be acquired 65536 in our example (field ②);
- indicate whether it is CMOS or LVDS DDR (field ③);
- Press the "INITIALIZATION" button④. It will initialized the HSDC-EXTMOD board:
  - FPGA is ready (red LED is flashing ¼ on and ¾ off) ;
  - PLL embedded is locked (green LED is on);
- indicate whether Fin or Fs are coherent or not (field ⑤):
  - if signals are coherent, selected which Fin or Fs are fixed for the calculation (see appendix A.1);
  - If signals are not coherent, select the window for FFT processing to apply (the Blackman window gives better results).
- Press the "ACQUIRE" button ⑥ to display the results from the FFT processing. The results fields ⑦ will be updated automatically.
- press "STOP" ⑦ button to stop acquisition;
- field ⑧ allows to do FFT averaging over up to 255 trials, suitable for small signal analysis;
- Field ⑨ allows storing dynamic results to text file. Click on the check box, enter a header as a comment and browse to indicate where to store data file. Table 7 shows how data are stored:

**Table 7. Dynamic results as stored in a text file**

*Content of file is shown as table format*

| Name | Fin | Fs | Vin | ENOB | SINAD_C | SNR_C | SNR_FS | SFDR_C | SFDR_FS | THD | H2 | H3 | H4 | H5 | H6 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | (MHz) | (MHz) | (dBFS) | - | (dBc) | (dBc) | (dBFS) | (dBc) | (dBFS) | (dBc) | (dBc) | (dBc) | (dBc) | (dBc) | (dBc) |
| ADC1610S test | | | | | | | | | | | | | | | |
| **ADC0** | **5.00** | **122.88** | **-0.96** | **11.28** | **69.79** | **69.67** | **70.75** | **84.62** | **85.58** | **-85.27** | **-104.62** | **-100.12** | **-103.67** | **-86.57** | **-112.7** |

Note that while acquisition is running, any other action (ADC SPI programming, quit or refresh) is not possible. Stop acquisition first before proceeding to any other task.

### 3.3.5.1 FFT spectrum

The first graph to be displayed is the FFT spectrum of the digital pattern acquired:

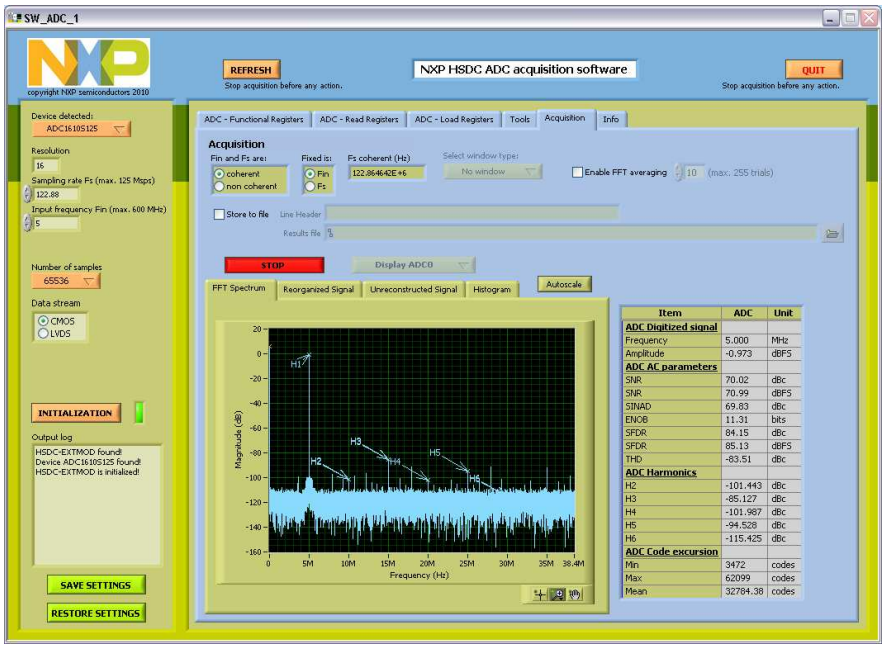



**Fig 14. SW_ADC_1_r02: "Acquisition" page, FFT graph**

Press the "Autoscale" button to display the whole content.

### 3.3.5.2 Reorganized signal

The reorganized signal displays the reconstructed sine wave from coherency calculation corresponding to 1 period of the input signal:


**Fig 15. SW_ADC_1_r02: “Acquisition” page, reorganized signal graph**

Press the “Autoscale” button to display the whole content.

### 3.3.5.3 Unreconstructed signal

The unreconstructed signal displays the unreconstructed sine wave corresponding to the whole number of period being acquired following the coherency rule:

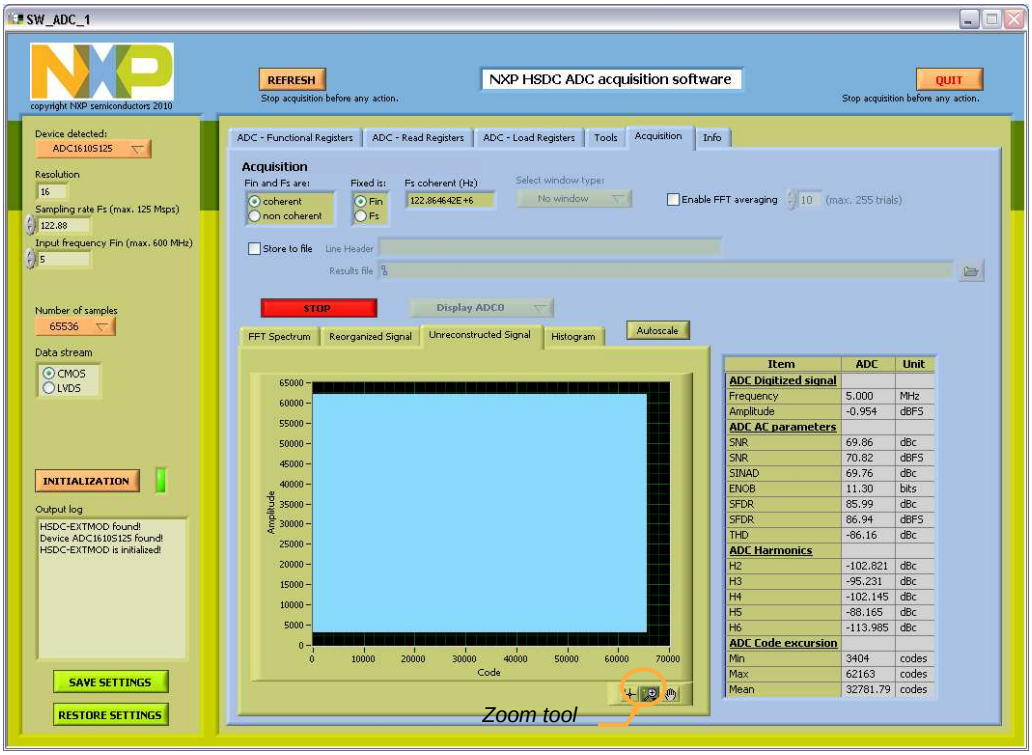



**Fig 16. SW_ADC_1_r02: “Acquisition” page, unreconstructed signal graph**

Press the “Autoscale” button to display the whole content.

Use the zoom tool to observe in more details all the captured data.

### 3.3.5.4 Histogram

The histogram graph shows the distribution of output codes. This graph shows which code is present and if there is any missing code in the conversion range:

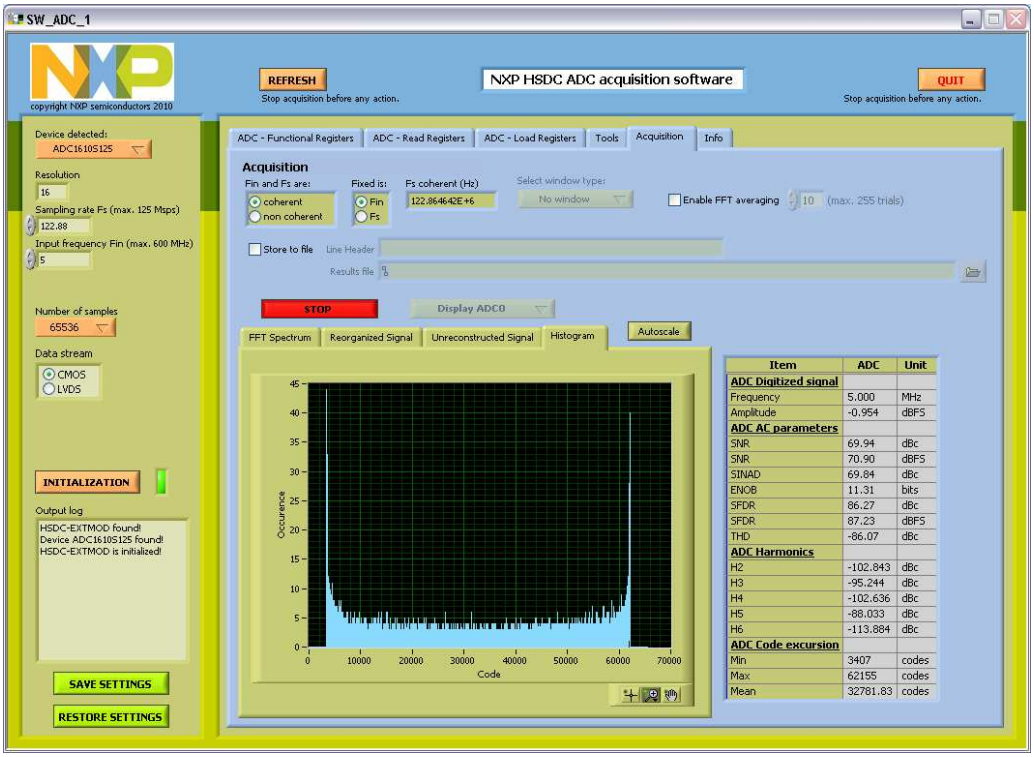



**Fig 17. SW_ADC_1_r02: “Acquisition” page, code histogram graph**

Press the “Autoscale” button to display the whole content.

The table shows the range of output codes.

### 3.3.6 Info page

This page will give practical information related to software and hardware settings:

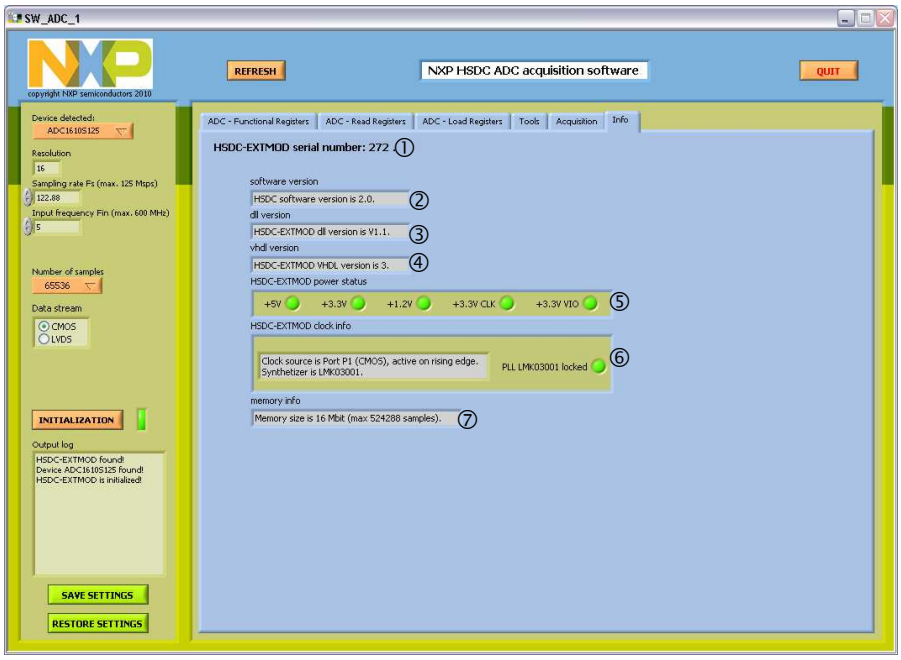



**Fig 18. SW_ADC_1_r02: “Info” page**

The information visible on this page is:

- board serial number ①;
- HSDC software release number ②;
- HSDC-EXTMOD dll version ③;
- HSDC-EXTMOD vhdl version ④;
- HSDC-EXTMOD supply status ⑤;
- HSDC-EXTMOD clock capability and status version ⑥;
- HSDC-EXTMOD memory capability⑦.

# 4. Appendix A.1: coherency calculation

The coherency relies on the fact that clock and analog input signal are synchronized and the first and last samples being captured are adjoining samples: it ensures a continuous digitized time process for the FFT processing.

To achieve this, one has to follow the equation:

$$\frac{F_{in}}{F_s} = \frac{M}{N}$$

Where M is an odd integer equal to the number of periods being acquired and N the number of samples acquired.

With Fin, Fs and N known, M has to be chosen such that it follows the equation above. To do this iterative calculation, one has to decide whether Fin or Fs is fixed.

To illustrate this process, let's consider our current example with Fin = 5 MHz, Fs = 122.88Msps and N = 65536 samples acquired:

- if Fin is fixed, this leads to M = 2667 periods of input signal to be acquired and a real sampling frequency to be Fs = 122.864642 MHz;
- If Fs is fixed, this leads to M = 2667 periods of input signal to be acquired and a real input frequency to be Fin = 5.000625 MHz.

Those values needs to be programmed in the signal generator and clock generator before capture is done, otherwise the FFT calculation will lead to a non-coherent result as shown below:

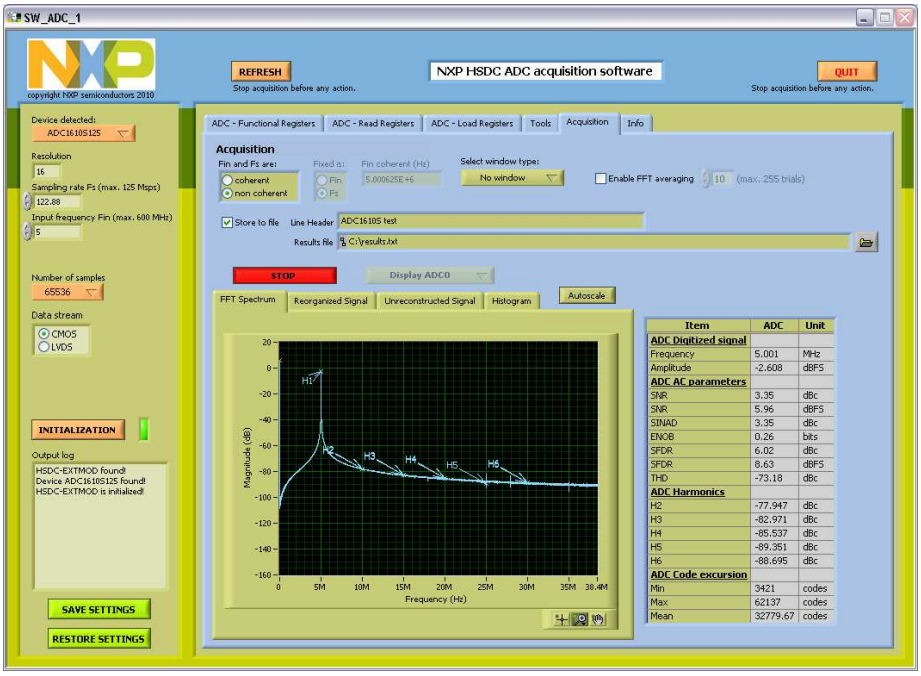



**Fig 19. SW_ADC_1_r02: "Acquisition" page, non-coherent capture example**

The numbers given for SNR, SFDR are completely wrong if coherency is not respected.

## 5. Notes

For any question, feel free to contact us at the following e-mail dataconverter-support@nxp.com.

# 6. Contents





For more information, please visit: http://www.nxp.com.
For sales office addresses, email to: salesaddresses@nxp.com.

**Date of release: January 2011**
**Document identifier: QS_ADC1610S_F1F2_5**