

# AD-FMCADC4-EBZ FMC Board

## Introduction

The [EVAL-AD-FMCADC4-EBZ](#) is a high speed four channel data acquisition board featuring two [AD9680](#) dual channel ADC at 1000 MSPS (1240 MSPS) and four ADA4961 [ADA4961](#) low distortion, 3.2 GHz, RF DGA driving each converter. The FMC form factor supports the JESD204B high speed serial interface. All clocking and channel synchronization is provisioned on-board using the AD9528 [AD9528](#) clock generator. This board meets most of the FMC specifications in terms of mechanical size, mounting hole locations etc., for further details, please refer to the FMC specification.

Although this board does meet most of the FMC specifications, it is not meant as a [commercial off the shelf](#) (COTS) board. If a commercial, ready to go integrate product is required, please refer to one of the many FMC manufacturers.

ADI also provides reference designs (HDL and software) for this board to work with commonly available Altera and Xilinx development boards.

## Hardware

The AD-FMCADC4-EBZ board's primary purpose is to demonstrate the capabilities of the devices on board quickly and easily by providing a seamless interface to an FMC carrier platform and running the reference design on the carrier FPGA. The board is designed to self power and self clock when connected to the FMC carrier. The analog signals (up to four) are connected to J301A, J301B, J301C and J301D. This rapid prototyping board can also be synchronized across channels.

## Devices

The FMC board includes the following products by Analog Devices:

- [AD9680](#) 14-bit dual channel ADC with sampling speeds of up to 1250 MSPS, with a [JESD204B](#) digital interface.
- [ADA4961](#) Low Distortion, 3.2 GHz, RF Digital Gain Amplifier.
- [AD9528](#) JESD204B Clock Generator with 14 LVDS Outputs
- [ADP2384](#) 20 V, 4 A, Synchronous, Step-Down DC-to-DC Regulator
- [ADP7104](#) is a 20V, 500mA, low noise, CMOS LDO
- [ADM7154](#) 600 mA, Ultra Low Noise, High PSRR, RF Linear Regulator
- [ADM7172](#) 6.5 V, 2 A, Ultralow Noise, High PSRR, Fast Transient Response CMOS LDO

- [ADP1741](#) is a 2A, low  $V_{in}$ , low dropout, CMOS linear regulator



Top View



Bottom View

## Clocking

The AD-FMCADC4-EBZ includes an on-board 80MHz reference oscillator from Crystek. This feature can be disconnected and an external reference can be applied through J901. When referencing the schematic make sure the proper component changes are made in order to directly route the input into the AD9528.

## Analog Front End

The AD-FMCADC4-EBZ uses a passive front end designed for very wide bandwidth. A single ended input needs to be provided to the analog inputs mentioned earlier. A 1:2 impedance ratio broadband balun then converts the input signal differentially to the ADA4961 inputs and has a 1.6GHz bandwidth at -3dB. Each channel amplifier can be adjusted independently in terms of gain.

## Revision A

The revision A board supports amplifier gain control via spi. After power-up, the gain of the amplifier defaults to an attenuated state. Use a low jitter, low noise signal source with a level at -20dBm to the analog inputs (J301-A/B/C/D). Apply a signal source no greater than -10dBm to achieve full-scale of the converter when maximum gain of the amplifier is applied.

# Running No-OS Application & Changing Sampling Rate to 1.24GHz

The HDL reference design is built around a processor as in an embedded system. You may use either Linux or No-OS software to demonstrate the design (details in the downloads section). In order to run the HDL with the No-OS application, first we need to build the HDL bit file and software elf file.

At the time of this writing, we are using the 'dev' branch for both. The [HDL user guide](#) contains the instructions to build the bit file. **Please make sure you use the 'dev' branch (checkout dev right after cloning).**

Once the bit file is ready, follow these instructions to build the elf file. This assumes you are following our directory structures. If you are not, just get the idea from here and port it to your environment. However you have to figure out things on your own.

1. Clone [No-OS](#) repository
2. Checkout the 'dev' branch (git checkout dev)
3. Change the directory to `ad-fmccadc4-ebz/zc706`.
4. Make the elf file by running `make HDF-FILE=<HDL-REPO>/projects/fmccadc4/zc706/fmccadc4\_zc706.sdk/system\_top.hdf`

The make will build the default 'hello-world', but we only need the bsp and I am no fan of eclipse, hence this method. If you are more comfortable with the GUI, import all the files (or folders) that the make uses.