DIO482FMC
Product Specification

High Performance Simultaneous Data Acquisition

Table of Contents

1 Product Description

1. ***DIO482FMC*** is a 32 channel simultaneous digital input/output module.
2. Standard configuration: 32 channels up to 30MSPS/channel.
3. Byte-wide direction control.
4. Complies with *VITA57 FMC* standard, *LPC* version.
5. Single-ended I/O on VHDCI connector.
6. Immediate (asynchronous) or Clocked (synchronous) update modes.
7. Compliant with D-TACQ *ELF* sites.

1.1 Product Variants

- ***DIO482FMC*** : 32 channels, 32MSPS/channel, FMC compliant

1.2 Applications

- High speed control and diagnostics.

1.3 Overview

The *FMC* module standard adds user IO to carrier modules fitted with *FPGA* resource. D-TACQ recommends modules based on the *Xilinx ZYNQ* system on chip, combining *FPGA* resource with a dual-core ARM Cortex A9 and gigabit Ethernet. Compatible modules include

- D-TACQ ***ACQ1001*** : D-TACQ single slot *FMC* carrier, Z7020
- D-TACQ ***ACQ2106*** : D-TACQ 6 slot *FMC* carrier, Z7030

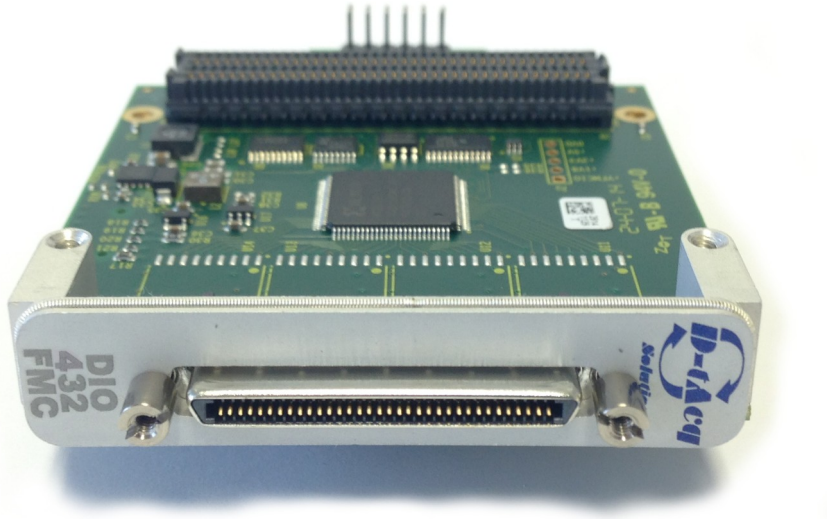
D-TACQ supplies a complete working Intelligent Digitizer appliance including programmable logic and microprocessor system running Linux.

1.4 Glossary

- *FMC*: VITA57 FPGA Mezzanine Card.
- Xilinx ZYNQ Soc
- *FPGA* : Field Programmable Gate Array.
- *LPC* : *FMC* Low pin count wiring standard.
- *ULPC*: *FMC* Ultra low pin count (D-TACQ).
- ULPC+ D-TACQ Ultra low pin count with LVDS
- Extended, ELF : *FMC* Extended size module (D-TACQ).

2 Physical

2.1 DIO482FMC Dimensions



2.2 Appearance

3 Interface Specification

2.3 Front Panel Connectors

- 68-way VHDCI connector (MOLEX 71430-0019).
- Pinout compatible with D-TACQ BNCPANEL-S1, SMAPANEL-S1, LEMOPANEL-S1, PTBPANEL-S1.
- Compatible DIN-RAIL module TERM04 available, providing screw terminals for easy connectivity.

2.4 Channel Pinout

Pin No.	Signal	Pin No.	Signal
1	0V / Clock Input (build option)*	35	0V
2	0V	36	0V
3	Digital I/O Channel 1	37	0V
4	Digital I/O Channel 2	38	0V
5	Digital I/O Channel 3	39	0V
6	Digital I/O Channel 4	40	0V
7	Digital I/O Channel 5	41	0V
8	Digital I/O Channel 6	42	0V
9	Digital I/O Channel 7	43	0V
10	Digital I/O Channel 8	44	0V
11	Digital I/O Channel 9	45	0V
12	Digital I/O Channel 10	46	0V
13	Digital I/O Channel 11	47	0V
14	Digital I/O Channel 12	48	0V
15	Digital I/O Channel 13	49	0V
16	Digital I/O Channel 14	50	0V
17	Digital I/O Channel 15	51	0V
18	Digital I/O Channel 16	52	0V
19	Digital I/O Channel 17	53	0V
20	Digital I/O Channel 18	54	0V
21	Digital I/O Channel 19	55	0V
22	Digital I/O Channel 20	56	0V
23	Digital I/O Channel 21	57	0V
24	Digital I/O Channel 22	58	0V
25	Digital I/O Channel 23	59	0V
26	Digital I/O Channel 24	60	0V
27	Digital I/O Channel 25	61	0V
28	Digital I/O Channel 26	62	0V
29	Digital I/O Channel 27	63	0V
30	Digital I/O Channel 28	64	0V
31	Digital I/O Channel 29	65	0V
32	Digital I/O Channel 30	66	0V
33	Digital I/O Channel 31	67	0V
34	Digital I/O Channel 32	68	0V

* Please contact D-TACQ for details.

3 DIO482FMC I/O Electrical Specification

The table below is for the 32 I/O signals

#	Parameter	Value
1	Number of Channels	32
2	Update Rate	32 I/Os up to 30MSPS* 16 I/Os up to 50MSPS*
3	I/O Voltage Range	5V TTL
4	High-level input voltage	> 3.5V
5	Low-level input voltage	< 1.5V
6	Input Voltage Withstand	-0.5 to 6.5V
7	Output Current High	-32mA, -100mA max per 8 I/Os**
8	Output Current Low	32mA, 100mA max per 8 I/Os**
10	Output Voltage High	> 3.8V
11	Output Voltage Low	< 0.55V

* Recommended maximum update is dependent on number of I/Os used, See description in Section 4.1

** I/Os are arranged in banks of 8 see Section 4 the current limitation refers to 8 I/Os on the same device

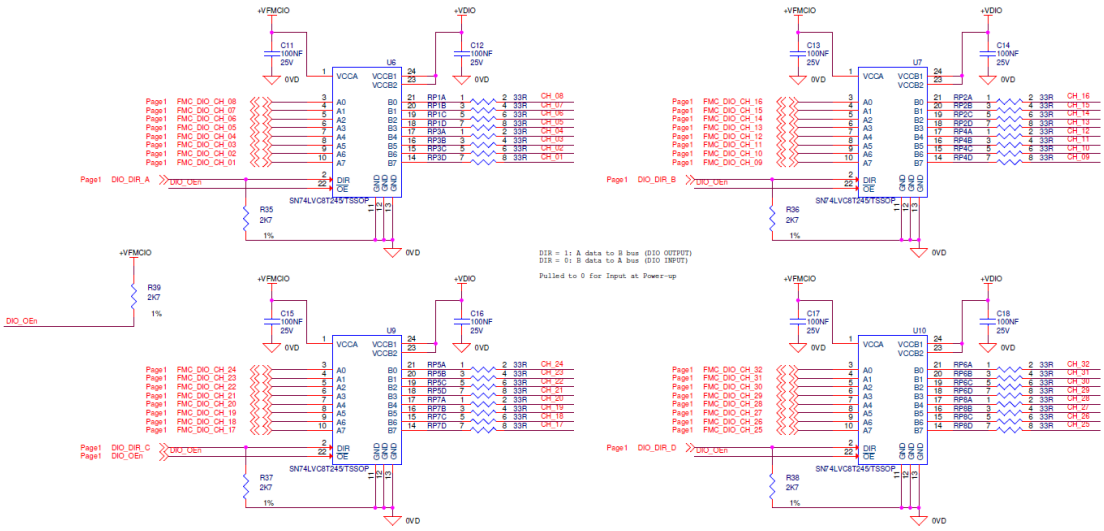
The optional dedicated input clock buffer has the same input voltage characteristics as the 32 I/Os

3.1 I/O Buffers

The I/O buffers are Octal (8 bit) devices, more information on I/O characteristics can be found in the buffer data sheet:

- Texas Instruments SN74LVC8T245

This is shown on the schematic fragment below



Additional details on the Clock Input buffer can be found in the component data sheet:

- Texas Instruments SN74LVC1T45

4 DIO482FMC FMC Specification

#	Parameter	Value
1	Form Factor	Standard FMC
2	Power source	External DC 12V, 200mA External DC 3.3V, 100 mA
3	I/O Voltage	LVCMOS 1.8V,2.5V,or 3.3V.
3	Environmental	0°C-50°C Operational -10°C-85°C Non-Operational
4	FMC Socket	Standard FMC, Low Pin Count LPC

The DIO482 32 I/Os are arranged in 4 banks of 8 bits. Input/Output direction is set at a bank level. All 8 bits in each bank have the same direction. This offers the following configurations

32 Outputs
24 Outputs and 8 Inputs
16 Outputs and 16 Inputs
8 Outputs and 24 Inputs
32 Inputs

The DIO482FMC 32 signal I/Os are also controlled by 4 direction controls bits and an overall output enable for the 4 I/O buffers

The FPGA designer must ensure that the I/O direction in the FPGA matches the I/O direction on the buffers.
The Output enable for the 4 buffers has a pull-up resistor to disable the outputs on the I/O Buffers during configuration. This signal should not be set low until the desired direction and signal values are on all the connections to the FPGA.

The 4 Banks of 8 bits are arranged as follows

Bank A is Digital I/Os 1-8, Direction control is DIO_DIR_A
Bank B is Digital I/Os 9-16, Direction control is DIO_DIR_B
Bank C is Digital I/Os 17-24, Direction control is DIO_DIR_C
Bank D is Digital I/Os 25-32, Direction control is DIO_DIR_D

Direction Control bit definition
0 = Input
1 = Output

Overall Output Enable control is DIO_Oen
0 = Outputs Enabled
1 = Outputs Tri-Stated

The dedicated Clock input is CLK_IN

4.1 FMC Pin Out

The DIO482FMC module uses a LPC FMC connector with the following pin out

FMC Pin	**Signal Name**	**Description**
p_FMC_CLK0_M2C_p	CLK_IN	System Clock from Front Panel
p_FMC_CLK0_M2C_n	DIO_OEn	Overall I/O Buffer Output Enable (pull-up by default
p_FMC_LA00_CC_p	DIO_CH01	I/O bit 1 (Bank A)
p_FMC_LA00_CC_n	DIO_CH018	I/O bit 18 (Bank C)
p_FMC_LA01_CC_p	DIO_CH03	I/O bit 3 (Bank A)
p_FMC_LA01_CC_n	DIO_CH020	I/O bit 20 (Bank C)
p_FMC_LA02_p	DIO_CH05	I/O bit 5 (Bank A)
p_FMC_LA02_n	DIO_CH22	I/O bit 22 (Bank C)
p_FMC_LA03_p	DIO_CH07	I/O bit 7 (Bank A)
p_FMC_LA03_n	DIO_CH24	I/O bit 24 (Bank C)
p_FMC_LA04_p	DIO_CH09	I/O bit 9 (Bank B)
p_FMC_LA04_n	DIO_CH26	I/O bit 26 (Bank D)
p_FMC_LA05_p	DIO_CH11	I/O bit 11 (Bank B)
p_FMC_LA05_n	DIO_CH28	I/O bit 28 (Bank D)
p_FMC_LA06_p	DIO_CH13	I/O bit 13 (Bank B)
p_FMC_LA06_n	DIO_CH30	I/O bit 30 (Bank D)
p_FMC_LA07_p	DIO_CH15	I/O bit 15 (Bank B)
p_FMC_LA07_n	DIO_CH32	I/O bit 32 (Bank D)
p_FMC_LA08_p	DIO_CH17	I/O bit 17 (Bank C)
p_FMC_LA08_n	DIO_DIR_A	Bank A Direction
p_FMC_LA09_p	DIO_CH19	I/O bit 19 (Bank C)
p_FMC_LA09_n	DIO_DIR_B	Bank B Direction
p_FMC_LA10_p	DIO_CH21	I/O bit 21 (Bank C)
p_FMC_LA10_n	DIO_DIR_C	Bank C Direction
p_FMC_LA11_p	DIO_CH23	I/O bit 23 (Bank C)
p_FMC_LA11_n	DIO_DIR_D	Bank D Direction
p_FMC_LA12_p	DIO_CH25	I/O bit 25 (Bank D)
p_FMC_LA13_p	DIO_CH27	I/O bit 27 (Bank D)
p_FMC_LA14_p	DIO_CH29	I/O bit 29 (Bank D)
p_FMC_LA15_p	DIO_CH31	I/O bit 31 (Bank D)
p_FMC_LA16_p	DIO_CH02	I/O bit 2 (Bank A)
p_FMC_LA17_CC_p	DIO_CH04	I/O bit 4 (Bank A)
p_FMC_LA18_CC_p	DIO_CH06	I/O bit 6 (Bank A)
p_FMC_LA19_p	DIO_CH08	I/O bit 8 (Bank B)
p_FMC_LA20_p	DIO_CH10	I/O bit 10 (Bank B)
p_FMC_LA21_p	DIO_CH12	I/O bit 12 (Bank B)
p_FMC_LA22_p	DIO_CH14	I/O bit 14 (Bank B)
p_FMC_LA23_p	DIO_CH16	I/O bit 16 (Bank C)

Note the pin out of the FMC connector allows for both LVCMOS and LVDS standards. The FMC naming convention with _p and _n signals is for LVDS pairing. However when using LVCMOS signals on these pins may result in possible cross-talk between the pairs. The DIO482FMC has been designed with the lower 16 I/O signals

only using one side of these pairs offering the potential for higher update rates when only 16 I/Os are required.

4.2 Example Top Level VHDL

The sample text below gives a prototype VHDL entity declaration for the FMC I/O pins of the DIO482FMC Module. The pin naming convention on the module is as per standard FMC specification pin names.

```
-------------------------------------------------------------------------------
--                    DIO482ELF Zynq Project Top Level of Module Connection
-------------------------------------------------------------------------------

-- Standard Libraries - numeric.std for all designs
library ieee;
use ieee.std_logic_1164.all;      -- Standard Logic Functions
use ieee.numeric_std.all;   -- Numeric Functions for Signed / Unsigned Arith-
metic

-- If using Xilinx primitives need the Xilinx library
library UNISIM;
use UNISIM.VComponents.all;       -- Xilinx Primitives


--!  Top Level of the DIO482 Board for the ACQ400 System
entity DIO482ELF_TOP is
port(
-----------------------
--   External I/O hooks   --
-----------------------
   p_FMC_CLK0_M2C_p    : inout std_logic;     --! CLK_IN
   p_FMC_CLK0_M2C_n    : inout std_logic;     --! DIO_OE_n
   p_FMC_LA00_CC_p     : inout std_logic;     --! CH01
   p_FMC_LA00_CC_n     : inout std_logic;     --! CH18
   p_FMC_LA01_CC_p     : inout std_logic;     --! CH03
   p_FMC_LA01_CC_n     : inout std_logic;     --! CH20
   p_FMC_LA02_p        : inout std_logic;     --! CH05
   p_FMC_LA02_n        : inout std_logic;     --! CH22
   p_FMC_LA03_p        : inout std_logic;     --! CH07
   p_FMC_LA03_n        : inout std_logic;     --! CH24
   p_FMC_LA04_p        : inout std_logic;     --! CH09
   p_FMC_LA04_n        : inout std_logic;     --! CH26
   p_FMC_LA05_p        : inout std_logic;     --! CH11
   p_FMC_LA05_n        : inout std_logic;     --! CH28
   p_FMC_LA06_p        : inout std_logic;     --! CH13
   p_FMC_LA06_n        : inout std_logic;     --! CH30
   p_FMC_LA07_p        : inout std_logic;     --! CH15
   p_FMC_LA07_n        : inout std_logic;     --! CH32
   p_FMC_LA08_p        : inout std_logic;     --! CH17
   p_FMC_LA08_n        : inout std_logic;     --! DIO_DIR_A
   p_FMC_LA09_p        : inout std_logic;     --! CH19
   p_FMC_LA09_n        : inout std_logic;     --! DIO_DIR_B
   p_FMC_LA10_p        : inout std_logic;     --! CH21
   p_FMC_LA10_n        : inout std_logic;     --! DIO_DIR_C
   p_FMC_LA11_p        : inout std_logic;     --! CH23
   p_FMC_LA11_n        : inout std_logic;     --! DIO_DIR_D
   p_FMC_LA12_p        : inout std_logic;     --! CH25
   p_FMC_LA13_p        : inout std_logic;     --! CH27
```

```
    p_FMC_LA14_p        : inout std_logic;     --! CH29
    p_FMC_LA15_p        : inout std_logic;     --! CH31
    p_FMC_LA16_p        : inout std_logic;     --! CH02
    p_FMC_LA17_CC_p     : inout std_logic;     --! CH04
    p_FMC_LA18_CC_p     : inout std_logic;     --! CH06
    p_FMC_LA19_p        : inout std_logic;     --! CH08
    p_FMC_LA20_p        : inout std_logic;     --! CH10
    p_FMC_LA21_p        : inout std_logic;     --! CH12
    p_FMC_LA22_p        : inout std_logic;     --! CH14
    p_FMC_LA23_p        : inout std_logic      --! CH16
    );
end DIO482ELF_TOP;
```

4.3 FPGA to DIO482 Direction Control

For Xilinx based FPGAs the following code fragment shows an interlock setup on the I/O signals to ensure the direction pins on the FPGA follow the direction pins on the DIO482FMC.

The signal `FMC_MODULE_ENABLE_n` is an overall FPGA internal signal used to enable the system

```
-- DIO_DIR_X = 0 - Input, DIO_DIR_X = 1 - Output
DIO_DIRA_BUF : IOBUF port map(IO => p_DIO_DIR_A, I => DIO_DIR_A, T =>
FMC_MODULE_ENABLE_n);
DIO_DIRB_BUF : IOBUF port map(IO => p_DIO_DIR_B, I => DIO_DIR_B, T =>
FMC_MODULE_ENABLE_n);
DIO_DIRC_BUF : IOBUF port map(IO => p_DIO_DIR_C, I => DIO_DIR_C, T =>
FMC_MODULE_ENABLE_n);
DIO_DIRD_BUF : IOBUF port map(IO => p_DIO_DIR_D, I => DIO_DIR_D, T =>
FMC_MODULE_ENABLE_n);

DIO_OE_BUF   :    IOBUF port map(IO => p_DIO_OE_n, I => '0', T =>
FMC_MODULE_ENABLE_n);

-- Instantiate DIO buffers, T0 = Output, T1 = Input
DIO_BUF1 : IOBUF port map(IO => p_FMC_LA00_CC_p,  I => DO_DATA(1),  T => not
DIO_DIR_A, O => DI_DATA(1));
DIO_BUF2 : IOBUF port map(IO => p_FMC_LA16_p,     I => DO_DATA(2),  T => not
DIO_DIR_A, O => DI_DATA(2));
DIO_BUF3 : IOBUF port map(IO => p_FMC_LA01_CC_p,  I => DO_DATA(3),  T => not
DIO_DIR_A, O => DI_DATA(3));
DIO_BUF4 : IOBUF port map(IO => p_FMC_LA17_CC_p,  I => DO_DATA(4),  T => not
DIO_DIR_A, O => DI_DATA(4));
DIO_BUF5 : IOBUF port map(IO => p_FMC_LA02_p,     I => DO_DATA(5),  T => not
DIO_DIR_A, O => DI_DATA(5));
DIO_BUF6 : IOBUF port map(IO => p_FMC_LA18_CC_p,  I => DO_DATA(6),  T => not
DIO_DIR_A, O => DI_DATA(6));
DIO_BUF7 : IOBUF port map(IO => p_FMC_LA03_p,     I => DO_DATA(7),  T => not
DIO_DIR_A, O => DI_DATA(7));
DIO_BUF8 : IOBUF port map(IO => p_FMC_LA19_p,     I => DO_DATA(8),  T => not
DIO_DIR_A, O => DI_DATA(8));
DIO_BUF9 : IOBUF port map(IO => p_FMC_LA04_p,     I => DO_DATA(9),  T => not
DIO_DIR_B, O => DI_DATA(9));
DIO_BUF10  IOBUF port map(IO => p_FMC_LA20_p,     I => DO_DATA(10), T => not
DIO_DIR_B, O => DI_DATA(10));
```

```
DIO_BUF11  IOBUF port map(IO => p_FMC_LA05_p,     I => DO_DATA(11), T => not
DIO_DIR_B, O => DI_DATA(11));
DIO_BUF12  IOBUF port map(IO => p_FMC_LA21_p,     I => DO_DATA(12), T => not
DIO_DIR_B, O => DI_DATA(12));
DIO_BUF13  IOBUF port map(IO => p_FMC_LA06_p,     I => DO_DATA(13), T => not
DIO_DIR_B, O => DI_DATA(13));
DIO_BUF14  IOBUF port map(IO => p_FMC_LA22_p,     I => DO_DATA(14), T => not
DIO_DIR_B, O => DI_DATA(14));
DIO_BUF15  IOBUF port map(IO => p_FMC_LA07_p,     I => DO_DATA(15), T => not
DIO_DIR_B, O => DI_DATA(15));
DIO_BUF16  IOBUF port map(IO => p_FMC_LA23_p,     I => DO_DATA(16), T => not
DIO_DIR_B, O => DI_DATA(16));
DIO_BUF17  IOBUF port map(IO => p_FMC_LA08_p,     I => DO_DATA(17), T => not
DIO_DIR_C, O => DI_DATA(17));
DIO_BUF18  IOBUF port map(IO => p_FMC_LA00_CC_n,  I => DO_DATA(18), T => not
DIO_DIR_C, O => DI_DATA(18));
DIO_BUF19  IOBUF port map(IO => p_FMC_LA09_p,     I => DO_DATA(19), T => not
DIO_DIR_C, O => DI_DATA(19));
DIO_BUF20  IOBUF port map(IO => p_FMC_LA01_CC_n,  I => DO_DATA(20), T => not
DIO_DIR_C, O => DI_DATA(20));
DIO_BUF21  IOBUF port map(IO => p_FMC_LA10_p,     I => DO_DATA(21), T => not
DIO_DIR_C, O => DI_DATA(21));
DIO_BUF22  IOBUF port map(IO => p_FMC_LA02_n,     I => DO_DATA(22), T => not
DIO_DIR_C, O => DI_DATA(22));
DIO_BUF23  IOBUF port map(IO => p_FMC_LA11_p,     I => DO_DATA(23), T => not
DIO_DIR_C, O => DI_DATA(23));
DIO_BUF24  IOBUF port map(IO => p_FMC_LA03_n,     I => DO_DATA(24), T => not
DIO_DIR_C, O => DI_DATA(24));
DIO_BUF25  IOBUF port map(IO => p_FMC_LA12_p,     I => DO_DATA(25), T => not
DIO_DIR_D, O => DI_DATA(25));
DIO_BUF26  IOBUF port map(IO => p_FMC_LA04_n,     I => DO_DATA(26), T => not
DIO_DIR_D, O => DI_DATA(26));
DIO_BUF27  IOBUF port map(IO => p_FMC_LA13_p,     I => DO_DATA(27), T => not
DIO_DIR_D, O => DI_DATA(27));
DIO_BUF28  IOBUF port map(IO => p_FMC_LA05_n,     I => DO_DATA(28), T => not
DIO_DIR_D, O => DI_DATA(28));
DIO_BUF29  IOBUF port map(IO => p_FMC_LA14_p,     I => DO_DATA(29), T => not
DIO_DIR_D, O => DI_DATA(29));
DIO_BUF30  IOBUF port map(IO => p_FMC_LA06_n,     I => DO_DATA(30), T => not
DIO_DIR_D, O => DI_DATA(30));
DIO_BUF31  IOBUF port map(IO => p_FMC_LA15_p,     I => DO_DATA(31), T => not
DIO_DIR_D, O => DI_DATA(31));
DIO_BUF32  IOBUF port map(IO => p_FMC_LA07_n,     I => DO_DATA(32), T => not
DIO_DIR_D, O => DI_DATA(32));


cmp_DIO_CLK: IBUF port map(I => p_CLK_IN, O => CLK_IN);
```