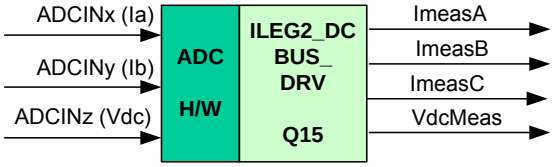
ILEG2_DCBBUS_DRV	*Two Line Currents/DC-Bus voltage Measurement ADC*

Description

This module allows 3-channel analog-to-digital conversion with programmable gains and offsets. The conversions are triggered on GP Timer 1 underflow (for 281x) or EPWM1 CNT_zero event (for 280x). The converted results represent load currents and DC-bus voltage in the inverter.

1. GP Timer 1 is the time base for symmetrical Pulse-Width Modulation (for 281x) or Time Base of EPWM1 CNT_zero event (for 280x);
2. For line current measurement, the analog inputs are the voltage across resistors placed between the sources or emitters of low-side power devices and low-side DC rail; and
3. For DC-bus voltage measurement, the analog input is the voltage across resistor by means of voltage divider concept.

.



Availability

This 16-bit module is available in one interface format:

1) The C interface version

Module Properties

Type: Target Dependent, Application Independent

Target Devices: x281x or x280x

C Version File Names: f281xileg_vdc.c, f281xileg_vdc.h (for x281x)
f280xileg_vdc.c, f280xileg_vdc.h (for x280x)

IQmath library files for C: N/A

Item	C version	Comments
Code Size□ (x281x/x280x)	147/155 words	
Data RAM	0 words•	
xDAIS ready	No	
XDAIS component	No	IALG layer not implemented
Multiple instances	Yes	
Reentrancy	Yes	

• Each pre-initialized ILEG2DCBUSMEAS structure consumes 15 words in the data memory

□ Code size mentioned here is the size of the ***init()*** and ***read()*** functions

C Interface

Object Definition

The structure of ILEG2DCBUSMEAS object is defined by following structure definition

```
typedef struct { int16 ImeasAGain;      // Parameter: gain for Ia (Q13)
                 int16 ImeasAOffset;    // Parameter: offset for Ia (Q15)
                 int16 ImeasA;          // Output: measured Ia (Q15)
                 int16 ImeasBGain;      // Parameter: gain for Ib (Q13)
                 int16 ImeasBOffset;    // Parameter: offset for Ib (Q15)
                 int16 ImeasB;          // Output: measured Ib (Q15)
                 int16 VdcMeasGain;     // Parameter: gain for Vdc (Q13)
                 int16 VdcMeasOffset;   // Parameter: offset for Vdc (Q15)
                 int16 VdcMeas;         // Output: measured Vdc (Q15)
                 int16 ImeasC;          // Output: computed Ic (Q15)
                 Uint16 ChSelect;       // Parameter: ADC channel selection
                 void (*init)();        // Pointer to the init function
                 void (*read)();        // Pointer to the read function
               } ILEG2DCBUSMEAS;

typedef ILEG2DCBUSMEAS *ILEG2DCBUSMEAS_handle;
```

Item	Name	Description	Format	Range(Hex)
Inputs	ADCINx, ADCINy, ADCINz	ADC pins in 281x device where x,y,z correspond to the channel numbers selected by ChSelect	N/A	0-3 V
Outputs	ImeasA	x^{th} channel digital representation for current I_a	Q15	8000-7FFF
	ImeasB	y^{th} channel digital representation for current I_b	Q15	8000-7FFF
	ImeasC	Computing current I_c	Q15	8000-7FFF
	VdcMeas	z^{th} channel digital representation for DC-bus voltage V_{dc}	Q15	8000-7FFF
ILEG2DCBUSMEAS parameter	ChSelect	16-bit ADC channel select format can be seen as: ChSelect = 0zyxh	Q0	x,y,z are in between 0h -> Fh
	ImeasAGain	Gain for x^{th} channel. Modify this if default gain is not used.	Q13	8000-7FFF
	ImeasBGain	Gain for y^{th} channel. Modify this if default gain is not used.	Q13	8000-7FFF
	VdcMeasGain	Gain for z^{th} channel. Modify this if default gain is not used.	Q13	8000-7FFF
	ImeasAOffset	Offset for x^{th} channel. Modify this if default offset is not used.	Q15	8000-7FFF
	ImeasBOffset	Offset for y^{th} channel. Modify this if default offset is not used.	Q15	8000-7FFF
	VdcMeasOffset	Offset for z^{th} channel. Modify this if default offset is not used.	Q15	8000-7FFF

Special Constants and Data types

ILEG2DCBUSMEAS

The module definition is created as a data type. This makes it convenient to instance an interface to the ILEG2DCBUSMEAS driver. To create multiple instances of the module simply declare variables of type ILEG2DCBUSMEAS.

ILEG2DCBUSMEAS_handle
User defined Data type of pointer to ILEG2DCBUSMEAS module

ILEG2DCBUSMEAS_DEFAULTS
Structure symbolic constant to initialize ILEG2DCBUSMEAS module. This provides the initial values to the terminal variables as well as method pointers.

Methods

void F281X_ileg2_dcbus_drv_init(ILEG2DCBUSMEAS *);
void F281X_ileg2_dcbus_drv_read(ILEG2DCBUSMEAS *);

void F280X_ileg2_dcbus_drv_init(ILEG2DCBUSMEAS *);
void F280X_ileg2_dcbus_drv_read(ILEG2DCBUSMEAS *);

This default definition of the object implements two methods – the initialization and the runtime read function for ILEG2DCBUSMEAS measurement. This is implemented by means of a function pointer, and the initializer sets this to F281X_ileg2_dcbus_drv_init and F281X_ileg2_dcbus_drv_read functions for x281x or F280X_ileg2_dcbus_drv_init and F280X_ileg2_dcbus_drv_read functions for x280x. The argument to this function is the address of the ILEG2DCBUSMEAS object.

Module Usage

Instantiation
The following example instances one ILEG2DCBUSMEAS object
ILEG2DCBUSMEAS ilg2_vdc1;

Initialization
To Instance pre-initialized objects
ILEG2DCBUSMEAS ilg2_vdc1 = ILEG2DCBUSMEAS_DEFAULTS;

Invoking the computation function
ilg2_vdc1.init(&ilg2_vdc1);
ilg2_vdc1.read(&ilg2_vdc1);

Example

The following pseudo code provides the information about the module usage.

```
main()
{

    ilg2_vdc1.init(&ilg2_vdc1);        // Call init function for ilg2_vdc1

}
```

```
void interrupt periodic_interrupt_isr()
{

        ilg2_vdc1.read(&ilg2_vdc1);                     // Call read function for ilg2_vdc1

        Ias = _IQ15toIQ((int32)ilg2_vdc1.ImeasA);           // Ias is in GLOBAL_Q
        Ibs = _IQ15toIQ((int32)ilg2_vdc1.ImeasB);           // Ibs is in GLOBAL_Q
        DC_bus = _IQ15toIQ((int32)ilg2_vdc1.VdcMeas);       // DC_bus GLOBAL_Q

}
```

Technical Background

The ADCIN pins accepts the analog input signals (I_a, I_b, and V_{dc}) in the range of 0-3 volts for x28xx based DSP with ground referenced to 0 volt (VREFLO = 0).

Consequently, before connecting these signals to ADCIN pins, the hardware adjustment by external op-amp circuits (for gain and offset adjustments) for these analog signals such that they represent according to the selected base quantities and the appropriate voltage range is required.

Four output variables of the module (ImeasA, ImeasB, ImeasC, and VdcMeas) are computed after three ADC analog input signals are digitized as seen below:

ImeasA = ImeasAGain*ADC_Ia_Q15 + ImeasAOffset
ImeasB = ImeasBGain*ADC_Ib_Q15 + ImeasBOffset
ImeasC = -(ImeasA + ImeasB)
VdcMeas = VdcMeasGain*ADC_Vdc_Q15 + VdcMeasOffset

Note that ADC_Ix_Q15 (x=a,b) and ADC_Vdc_Q15 are already converted to Q15 number.

Basically, the signals can be categorized into two main types: bipolar and unipolar signals. The AC currents (or AC voltages) are examples of bipolar signal and the DC-bus voltage is an example of unipolar signal.

The input AC currents (I_a, I_b) are typically sensed and re-scaled within the range of 0-3 volts for x28xx based DSP with the appropriate base current. Thus, the Q15-number conversion is necessary for the current measurements after they are digitized as seen in Figure 1.

It is emphasized that the ADC unit is 12-bit resolution with left justified in the 16-bit ADC result register. Thus, the ADC output range is in between 0000h and FFF0h.

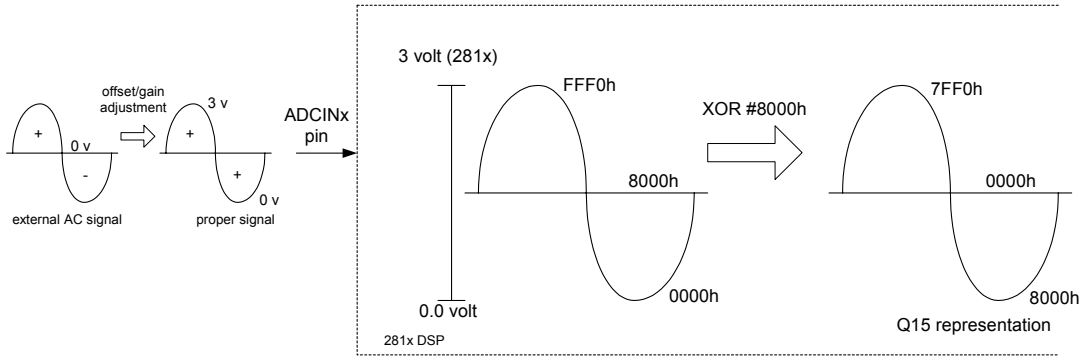


Figure 1: Q15-number conversion for current measurements (bipolar signal)

For DC-bus voltage (V_{dc}), practically the input signal is already represented in the positive range only, so its digitized variable has to be rescaled corresponding to the Q15 number.

For better understanding, Figure 2 illustrates the Q15-number conversion for the DC-bus voltage measurement.

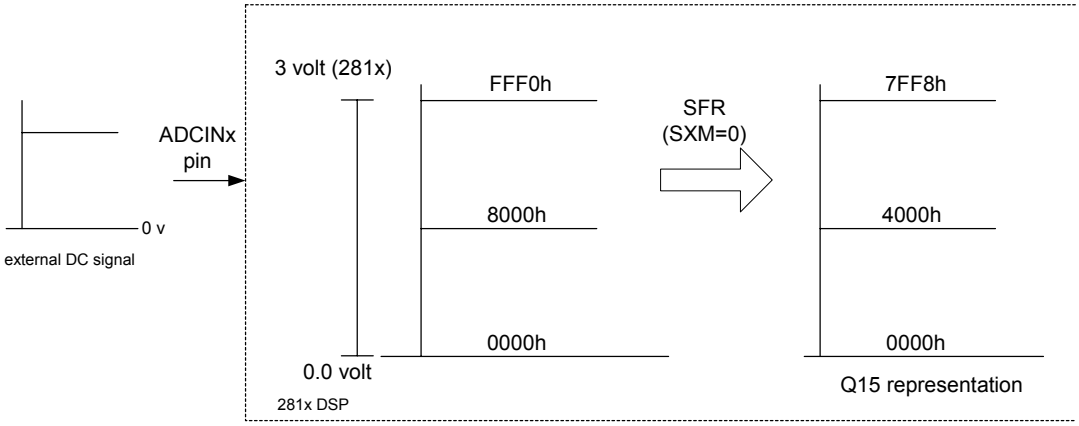


Figure 2: Q15-number conversion for DC-bus voltage measurement (unipolar signal)

In both cases of Q15-number conversion, the number is distorted a little bit about the maximum value (e.g., 7FF0h for bipolar and 7FF8h for unipolar at the maximum value of 7FFFh).