ADCSPI Library Reference Manual

Revision: June 19, 2012

1300 NE Henley Court, Suite 3
Pullman, WA 99163
(509) 334 6306 Voice | (509) 334 6300 Fax

Introduction

The Digilent PmodAD1 contains two simultaneous A/D conversion channels. Digilent provides an Arduino driver library for this device that is able to access only one converter channel.
This document provides an overview of the operation of this driver library and describes the functions that define its programming interface.

Overview

The PmodAD1 device contains two separate channels having each an Analog Devices AD7476 12-Bit, 1 MSPS, Low-Power A/D Converter. Due to the physical pin layout of the SPI hardware interfaces, this library implements access to only one converter channel (corresponding to P3 pin of the J2 connector of PmodAD1).
For more information about the hardware interface of the PmodAD1, refer to the PmodAD1 Reference Manual available for download from the Digilent web site (www.digilentinc.com).

The converter is a serial device that is accessed (read) via an SPI interface. The library allows the use of any of the two SPI hardware interfaces.

The values read from the converter are on 12 bits (from 0 to 0x0FFF). This is considered "*IntegerValue*". The library also deals with "*PhysicalValue*", which is a floating point value that corresponds to the "*IntegerValue"* and to a reference value that defines the reference voltage (associated to the maximum integer value).

PhysicalValue = IntegerValue * (ReferenceValue / (2^12 – 1))

By default, the physical value is the voltage corresponding to the value read from converter and to the 3.3V maximum voltage.

Note: In order to use the "PhysicalValue" functionality you must have the following line placed at the top of the ADCSPI.h file.:

#define AD1_FLOATING_POINT

Library Operation

Library Interface

The header file ADCSPI.h defines the interfaces to the ADCSPI driver. The library is accessed via the methods and constants defined for the ADCSPI object class. In order to use this library, the user has to instantiate one library object.

SPI Initialization

In order to communicate with the PmodAD1, SPI must be initialized. Before making calls to any other library functions, the begin() function must be called. This function initializes the SPI used by the library.

AD Converter Functions

In order to read values from the AD converter, the library provides two functions: GetIntegerValue() and GetPhysicalValue() (see note below). Read Overview chapter for a description of these values.

Note: In order to use the "PhysicalValue" functionality you must have the following line placed at the top of the ADCSPI.h file.:

```
#define AD1_FLOATING_POINT
```

ADCSPI Library Functions

void begin(uint8_t bAccessType)

Parameters:

- uint8_t bAccessType – the SPI interface where the PmodAD1 is connected. It can be one of the parameters from the following list:

Table 1. List of values for bAccessType parameter of begin function

Value	Name	Connector on Cerebot MX4CK board
0	PAR_ACCESS_SPI0	JB
1	PAR_ACCESS_SPI1	J1

This function initializes the required SPI interface, setting the SPI frequency to 1 MHz.

uint16_t GetIntegerValue()

Return value

- uint16_t - the 12 bits value read from the AD converter

This function returns the 12 bits value read from the AD converter, obtained by reading 16 bits from SPI.
Read Overview chapter for a description of Integer Value.

float GetPhysicalValue(float dReference)

Parameters

- float dReference – the value corresponding to the maximum converter value. If this parameter is not provided, it has a default value of 3.3.

Return value

- float - the value corresponding to the value read from the AD converter and to the reference value

This function returns the value corresponding to the value read from the AD converter and to the reference value.
If the function argument is missing, 3.3 value is used as reference value.

Read Overview chapter for a description of Physical Value.

Note: In order to use the "PhysicalValue" functionality you must have the following line placed at the top of the ADCSPI.h file.:

#define AD1_FLOATING_POINT

Library usage

This section of the document describes the way the library is used:

- The PmodAD1 should be plugged in one of the SPI connectors.

Table 2. List of possible connections for PmodAD1

Connection	Connector on Cerebot MX4CK board
SPI0	JB
SPI1	J1

- Apply the analog value on P3 pin of the J2 connector of PmodAD1.
- Copy the library files according to the README.txt file.
- In the sketch, include the ADCSPI library header file
  ```
  #include <ADCSPI.h>
  ```
- In the sketch, include the DSPI library header file. It is needed in order to access the SPI functionality.
  ```
  #include <DSPI.h>
  ```
- In the sketch, instantiate one library object called, for example, myADCSPI
  ```
  ADCSPI myADCSPI;
  ```
- In the sketch, use library functions by calls such as:
  ```
  wValue = myADCSPI.GetIntegerValue();
  ```

Simple Demo

In order to run this demo, place the library and sketch and apply the analog value as explained in the Library usage section.
This demo performs the following operations:

- In the setup() function:
  - Initializes the ADCSPI library and Serial interface (in order to display information on the Serial terminal).
- In the loop() function:
  - Reads the AD integer and physical values. (For the demo, AD1_FLOATING_POINT is defined in ADCSPI.h)
  - Displays the two read values using the Serial terminal