

MAX32600 ADC Simple Capture Demonstration

April 3, 2015

1 Requirements

- MAX32600B EvKit
- Sample code for this application located in `Firmware/Applications/ADCSimpleDemo`
- Olimex JTAG adapter and USB cable to connect the JTAG adapter to the PC
- GNU Compiler Collection (GCC) for ARM with newlib libc
- GNU Debugger (GDB) for ARM
- USB Full-size A to B cable for UART over USB
- PC or Workstation with USB and terminal emulator software

2 Setup

- Load the compiled `max32600.elf` file onto the MAX32600 EvKit. The green LED should be on.
- Connect an input voltage or signal; defaults for the sample applications:
 - Input on EvKit SMA `AIN2+` or pin `AIN2+` and ground
 - Reference voltage at $1.5V$
 - Sampling frequency of $10KHz$
 - 2048 samples per trigger
- Connect PC USB-serial terminal to EvKit full-size USB-B connector; BAUD => 115200; No parity; No flow control;

3 Observation

Press the `SW1 TEST` button and see a string of 2048 decimal, comma separated ascii numbers on the USB serial port output representative of the input voltage from $0 - 1.5V$ converted to decimal $0 - 65535$ (16bit values). The Yellow LED on the EvKit should be on during the capture and shut off when all samples are reported.

4 Source Code Overview

4.1 Drivers In Use

- Instruction Cache
- Clock Manager
- Power Manager
- IO Manager
- GPIO
- SysTick
- UART
- PMU
- AFE
- ADC

4.2 Interrupts Enabled

- PMU
- UART0

4.3 Code Operation

- Enable Instruction Cache
- Setup Clocks; trim ring oscillator, set ADC clock
- Set UART pin mapping
- Set UART configuration (baud rate and serial controls)
- Setup GPIO “SW TEST” button for ADC trigger
- Set ADC reference voltage
- Set ADC buffers
- Set ADC sample rate and mode
- Set ADC input mux
- Enable and power-up ADC
- Wait for user input via button