This thread has been locked.
If you have a related question, please click the "Ask a related question" button in the top right corner. The newly created question will be automatically linked to this question.

CC2650: SimpleLink MCU

sadasivam arumu... Intellectual 940 points
Community Member

Part Number: CC2650
Other Parts Discussed in Thread: MSP430FR5969, SYSBIOS, LM35, CC2640R2F, CC2640

Hi Sir,

The below post I have made earlier has no reply. (I think of correct asking with apt expert will guide me). So, I am posting in SimpleLik MCU

I am using ***CC2650 launchxl*** for interfacing Temperature sensor(DS600).

I used TI driver example(***adcbufcontinuous_CC2650_LAUNCHXL_TI_CC2650F128***) for getting sensor values from analog pin -Board_DIO23_ANALOG.

In ADCCC26XX Hardware attributes, Board_DIO23_ANALOG is configured to index-0. Below are my configurations:

ADCCC26XX_HWAttrs structure{

```
.adcDIO                = Board_DIO23_ANALOG,
.adcCompBInput          = ADC_COMPB_IN_AUXIO7,
.refSource             = ADCCC26XX_FIXED_REFERENCE,
.samplingDuration       = ADCCC26XX_SAMPLING_DURATION_682_US,
.inputScalingEnabled   = true,
.triggerSource          = ADCCC26XX_TRIGGER_MANUAL
```

}

I didnt able to get the correct ADC value. Then only I can able to change for temperature value. Is the configuration correct.

I have searched for any thread, got the idea on Sensor controller. fyi: **RTOS/CC2650: ADC taking upto 50ms to get a valid result**

But, I Haven't checked these because I have not tried with sensor controller examples. So, it is difficult to handle in it.

over 5 years ago

YiKai Chen *over 5 years ago* Guru 735585 points

Do you try ADC example at

http://dev.ti.com/tirex/explore/node?node=AK4vHbPlXyzwff1C5hUswQ__Drd3Vmn__LATEST

?

sadasivam arumugam *over 5 years ago in reply to YiKai Chen* Intellectual 940 points

Thanks sir. Yes, I have used the same code for interfacing temperature sensor.

I am able to get the data from the channel. But, I am getting incorrect values. I have tried the same IC with another controller. It is working good as expected. Readings are obtained. I also verifyied by continuous collecteion of raw data in that testing. But unable to get from CC2650. is my configurations correct?

YiKai Chen *over 5 years ago in reply to sadasivam arumugam* Guru 735585 points

I am confused by your descriptions. Can you elaborate?

sadasivam arumugam *over 5 years ago in reply to YiKai Chen* Intellectual 940 points

Yes, Sorry.

I used another MCU(MSP430FR5969) for Temp. sensor interface-> I get correct digital values from a particular channel.

But When I am using with CC2650, I couldn't able to get correct values. So, is my config OK?

I am interfacing in the CC2650_ADC example.

What are the design constraints needed to done while interfacing ADC with Sensor.

YiKai Chen *over 5 years ago in reply to sadasivam arumugam* Guru 735585 points

What is your temperature sensor?

sadasivam arumugam *over 5 years ago in reply to YiKai Chen* Intellectual 940 points

Temp. Sensor - DS600.

YiKai Chen *over 5 years ago in reply to sadasivam arumugam* Guru 735585 points

According to DS600 datasheet, CC2650 ADC should be able to read DS600 voltage without problem. Can you specify what your problem is when doinf ADC reading from DS600?

sadasivam arumugam *over 5 years ago in reply to YiKai Chen* — Intellectual 940 points

I will specify this in mesage. Can you guide me in this. Because important queries should be posted in thread.

YiKai Chen *over 5 years ago in reply to sadasivam arumugam* — Guru 735585 points

I couldn't understand your descriptions.

sadasivam arumugam *over 5 years ago in reply to YiKai Chen* — Intellectual 940 points

Hi Sir, Please find my code below and raw data obtained from DS600.

```
* ======== adcsinglechannel.c ========
*/
/* XDCtools Header files */

#include <xdc/std.h>
#include <xdc/runtime/System.h>

/* BIOS Header files */
#include <ti/sysbios/BIOS.h>
#include <ti/sysbios/knl/Task.h>

/* Driver Header files */
#include <ti/drivers/ADC.h>
#if defined(CC2650DK_7ID) || defined(CC1310DK_7XD)
#include <ti/drivers/PIN.h>
#endif

/* Example/Board Header files */
#include "Board.h"

#define ADC_CONV_MIN_VAL 1200
#define ADC_CONV_MAX_VAL 4095
#define ROOM_TEMP_MIN 2500
#define ROOM_TEMP_MAX 3500
#define ADC_CELCIUS_CONV_CONST 509

/* ADC sample count */
#define ADC_SAMPLE_COUNT (10)

/*Task Specific defines */
#define TASKSTACKSIZE (768)

Task_Struct task0Struct;
Char task0Stack[TASKSTACKSIZE];

/* Pin driver handles */
static PIN_Handle buttonPinHandle;
/* Global memory storage for a PIN_Config table */
static PIN_State buttonPinState;

// Temperature sensor related variables declaration
/* ADC conversion result variables */
uint16_t adcValue1[50]={0};
const float adc_conv_volt_const = 0.293040;
const float adc_cel_conv_const = 6.45;
unsigned long volt = 0;
float cel = 0;
int g_Celsius[50]={0};
```

```
/*
* Application button pin configuration table:
* - Buttons interrupts are configured to trigger on falling edge.
*/
PIN_Config buttonPinTable[] = {Board_DIO21 | PIN_GPIO_OUTPUT_EN | PIN_GPIO_HIGH | PIN_PUSHPULL |
PIN_DRVSTR_MAX,PIN_TERMINATE};


/*
* ======== taskFxn1 ========
* Open a ADC handle and get a array of sampling results after
* calling several conversions.
*/
Void taskFxn0(void)
{
/*board initialization for ADC Pins*/
Board_initADC();
ADC_Handle adc;
ADC_Params params;
int_fast16_t res;
char currVal = 0;
char i = 0;
//clearing the buffer
for(i = 0; i<50; i++)
{
g_Celsius[i] = 0;
adcValue1[i] = 0;
}


PIN_setOutputValue(buttonPinHandle, Board_DIO21, currVal); // for temperature sensor active
ADC_Params_init(&params);
adc = ADC_open(Board_ADC0, &params);
//adc = ADC_open(CC2650_LAUNCHXL_ADC0, &params);

if (adc == NULL)
{
System_abort("Error initializing ADC channel 1\n");
}
else
{
System_printf("ADC channel 1 initialized\n");
}

for(i = 0; i<50; i++)
{
res = ADC_convert(adc, &adcValue1[i]);
volt = adcValue1[i] * ADC_CONV_MIN_VAL/ADC_CONV_MAX_VAL;
cel = ((volt - ADC_CELCIUS_CONV_CONST)/adc_cel_conv_const);
cel +=2; // 2 Calibration factor with LM35 measurement
cel *= 100; // Multiply by 100 to maintain 2 decimal points when casting to integer
g_Celsius[i] = (int)cel; // Cast down to integer to send over BLE
}
ADC_close(adc);
}

/*
* ======== main ========
```

```
*/
int main(void)
{
Task_Params taskParams;

/* Call board init functions */
Board_initGeneral();

buttonPinHandle = PIN_open(&buttonPinState, buttonPinTable);
if(!buttonPinHandle) {
System_abort("Error initializing button pins\n");
}

/* Create tasks */
Task_Params_init(&taskParams);
taskParams.stackSize = TASKSTACKSIZE;
taskParams.stack = &task0Stack;
Task_construct(&task0Struct, (Task_FuncPtr)taskFxn0, &taskParams, NULL);

/* SysMin will only print to the console when you call flush or exit */
System_flush();

BIOS_start();

return (0);
}
```

Raw data:

adcValue1 unsigned short[20]
[0] unsigned short 0
[1] unsigned short 487
[2] unsigned short 422
[3] unsigned short 600
[4] unsigned short 64
[5] unsigned short 0
[6] unsigned short 0
[7] unsigned short 579
[8] unsigned short 599
[9] unsigned short 417
[10] unsigned short 0
[11] unsigned short 0
[12] unsigned short 119
[13] unsigned short 7
[14] unsigned short 0
[15] unsigned short 0
[16] unsigned short 105
[17] unsigned short 245
[18] unsigned short 429
[19] unsigned short 578
[20] unsigned short 576

TER *over 5 years ago in reply to sadasivam arumugam* TI__Guru**** 317180 points

Do you get the expected result if you use a voltage source as input instead of the temp sensor?

sadasivam arumugam *over 5 years ago in reply to TER* Intellectual 940 points

I am getting the following values:

adcValue1 unsigned short[20]
[0] unsigned short 3126
[1] unsigned short 3123
[2] unsigned short 3125
[3] unsigned short 3124
[4] unsigned short 3125
[5] unsigned short 3125
[6] unsigned short 3126
[7] unsigned short 3125
[8] unsigned short 3124
[9] unsigned short 3126
[10] unsigned short 3126
[11] unsigned short 3125
[12] unsigned short 3125
[13] unsigned short 3124
[14] unsigned short 3123
[15] unsigned short 3125
[16] unsigned short 3127
[17] unsigned short 3125
[18] unsigned short 3123
[19] unsigned short 3125
[20] unsigned short 3125

I want to know why the interfacing with temp sensor having the errorred output.

TER *over 5 years ago in reply to sadasivam arumugam* — TI__Guru**** 317180 points

ok, and what was the input? (source and value)

sadasivam arumugam *over 5 years ago in reply to TER* — Intellectual 940 points

Its 3.3V, when I am using the Launchpad supply voltage to source the analog pin.

TER *over 5 years ago in reply to sadasivam arumugam* — TI__Guru**** 317180 points

Please confirm that you have run

/* Adjust raw adc values and convert them to microvolts */
ADCBuf_adjustRawValues(handle, completedADCBuffer, ADCBUFFERSIZE, completedChannel);
ADCBuf_convertAdjustedToMicroVolts(handle, completedChannel, completedADCBuffer, microVoltBuffer, ADCBUFFERSIZE);

as done in the example?

YiKai Chen *over 5 years ago in reply to sadasivam arumugam* — Guru 735585 points

I don't see you print the array value in your code. Please post your exact code in your test.

sadasivam arumugam *over 5 years ago in reply to TER* — Intellectual 940 points

Sir, I am using the example file:-> adcsinglechannel_CC2650_LAUNCHXL_TI_CC2650F128. I also executed that code by using the Supply voltage(CC2650_LAUNCHXL_ADCVDDS) as channel and Ground supply(CC2650_LAUNCHXL_ADCVSS)

I haven't used the above code. It is under :-> adcbufcontinuous_CC2650_LAUNCHXL_TI_CC2650F128 example file.

sadasivam arumugam *over 5 years ago in reply to YiKai Chen* — Intellectual 940 points

I have checked the array in Expressions for getting those result. @YiKai Chen

TER *over 5 years ago in reply to sadasivam arumugam* — TI__Guru**** 317180 points

You wrote in the first post that you use adcbufcontinuous, now you write that you use adcsinglechannel?

As YK wrote, please post the code you are using.

sadasivam arumugam *over 5 years ago in reply to TER* — Intellectual 940 points

Yes you are right. I have posted the reference to wrong example. But I am using the adcsinglechannel example code and changed that code which I have mentioned earlier. Sorry for not mentioning the wrong example in previous post.

YiKai Chen *over 5 years ago in reply to sadasivam arumugam* — Guru 735585 points

Can you post exact codes that you are testing?

sadasivam arumugam *over 5 years ago in reply to YiKai Chen* — Intellectual 940 points

The same code which I have mentioned earlier.

```
* ======== adcsinglechannel.c ========
*/
/* XDCtools Header files */

#include <xdc/std.h>
#include <xdc/runtime/System.h>

/* BIOS Header files */
#include <ti/sysbios/BIOS.h>
#include <ti/sysbios/knl/Task.h>

/* Driver Header files */
#include <ti/drivers/ADC.h>
#if defined(CC2650DK_7ID) || defined(CC1310DK_7XD)
#include <ti/drivers/PIN.h>
#endif

/* Example/Board Header files */
#include "Board.h"

#define ADC_CONV_MIN_VAL 1200
#define ADC_CONV_MAX_VAL 4095
#define ROOM_TEMP_MIN 2500
#define ROOM_TEMP_MAX 3500
#define ADC_CELCIUS_CONV_CONST 509

/* ADC sample count */
#define ADC_SAMPLE_COUNT (10)

/*Task Specific defines */
#define TASKSTACKSIZE (768)
```

```
Task_Struct task0Struct;
Char task0Stack[TASKSTACKSIZE];

/* Pin driver handles */
static PIN_Handle buttonPinHandle;
/* Global memory storage for a PIN_Config table */
static PIN_State buttonPinState;

// Temperature sensor related variables declaration
/* ADC conversion result variables */
uint16_t adcValue1[50]={0};
const float adc_conv_volt_const = 0.293040;
const float adc_cel_conv_const = 6.45;
unsigned long volt = 0;
float cel = 0;
int g_Celsius[50]={0};
/*
* Application button pin configuration table:
* - Buttons interrupts are configured to trigger on falling edge.
*/
PIN_Config buttonPinTable[] = {Board_DIO21 | PIN_GPIO_OUTPUT_EN | PIN_GPIO_HIGH | PIN_PUSHPULL |
PIN_DRVSTR_MAX,PIN_TERMINATE};


/*
* ======== taskFxn1 ========
* Open a ADC handle and get a array of sampling results after
* calling several conversions.
*/
Void taskFxn0(void)
{
/*board initialization for ADC Pins*/
Board_initADC();
ADC_Handle adc;
ADC_Params params;
int_fast16_t res;
char currVal = 0;
char i = 0;
//clearing the buffer
for(i = 0; i<50; i++)
{
g_Celsius[i] = 0;
adcValue1[i] = 0;
}


PIN_setOutputValue(buttonPinHandle, Board_DIO21, currVal); // for temperature sensor active
ADC_Params_init(&params);
adc = ADC_open(Board_ADC0, &params);
//adc = ADC_open(CC2650_LAUNCHXL_ADC0, &params);

if (adc == NULL)
{
System_abort("Error initializing ADC channel 1\n");
}
else
{
```

```
System_printf("ADC channel 1 initialized\n");
}

for(i = 0; i<50; i++)
{
res = ADC_convert(adc, &adcValue1[i]);
volt = adcValue1[i] * ADC_CONV_MIN_VAL/ADC_CONV_MAX_VAL;
cel = ((volt - ADC_CELCIUS_CONV_CONST)/adc_cel_conv_const);
cel +=2; // 2 Calibration factor with LM35 measurement
cel *= 100; // Multiply by 100 to maintain 2 decimal points when casting to integer
g_Celsius[i] = (int)cel; // Cast down to integer to send over BLE
}
ADC_close(adc);
}

/*
* ======== main ========
*/
int main(void)
{
Task_Params taskParams;

/* Call board init functions */
Board_initGeneral();

buttonPinHandle = PIN_open(&buttonPinState, buttonPinTable);
if(!buttonPinHandle) {
System_abort("Error initializing button pins\n");
}

/* Create tasks */
Task_Params_init(&taskParams);
taskParams.stackSize = TASKSTACKSIZE;
taskParams.stack = &task0Stack;
Task_construct(&task0Struct, (Task_FuncPtr)taskFxn0, &taskParams, NULL);

/* SysMin will only print to the console when you call flush or exit */
System_flush();

BIOS_start();

return (0);
}
```

TER *over 5 years ago in reply to sadasivam arumugam*

TI__Guru**** 317180 points

Try to add the functions after ADC_convert the newest version of this example does:

http://dev.ti.com/tirex/explore/node?a=VLyFKFf__3.6.2&node=ABO7T5VshSOTTVANhMb1NQ__eCfARaV__LATEST&r=VLyFKFf__LATEST

It looks like the example you have used does not compensate for offset/ gain errors in the ADC.

TER *over 5 years ago in reply to TER* — TI__Guru**** 317180 points

Btw, if you use CC2640R2F instead you will get a much more up to date software offering.

sadasivam arumugam *over 5 years ago in reply to TER* — Intellectual 940 points

Sir, refer to this link : https://e2e.ti.com/support/wireless-connectivity/bluetooth/f/538/p/404426/1433112#pi320995=1

This link uses driverlib for CC2640 MCU. Is there a driverlib support in CC2650?

YiKai Chen *over 5 years ago in reply to sadasivam arumugam* — Guru 735585 points

ADC of CC2640 and CC2650 are the same so you can use the same code.

sadasivam arumugam *over 5 years ago in reply to YiKai Chen* — Intellectual 940 points

Yes, I will try this code and will update you shortly.

Texas Instruments has been making progress possible for decades. We are a global semiconductor company that designs, manufactures, tests and sells analog and embedded processing chips. Our products help our customers efficiently manage power, accurately sense and transmit data and provide the core control or processing in their designs.

Previewing Staged Changes