

12 - Mundo analógico \ Código

Rafael Corsi - rafael.corsi@insper.edu.br

Abril - 2017



Figure 1: Roland Kirk

Entrega final

Leitura de dois sinais analógicos : Temperatura do Chip e Potenciômetro (externo) e exibição dos seus valores no LCD (gráfico temporal), aplicação deve executar no RTOS.

Entrega parcial

Leitura do valor de temperatura e exibição do valor (numérico) no LCD já com uso do RTOS, inserido timer counter para realizar o trigger da conversão.

Exemplo

Introdução

O código exemplo *ADC.c* configura e demonstra a utilização do conversor analógico digital (AFEC) do microcontrolador SAME-70, esse periférico pode ser utilizado para transformar valores analógicos do mundo externo ao processador em valores digitais que podem ser processados pelo uC.

Em sua grande maioria os sensores elétricos são transdutores, ou seja, convertem grandezas diversas em tensão (sensor de luz, sensor de temperatura, gás, ...). Para podermos processar essas grandezas é necessário primeiramente digitalizarmos esses valores.

Periféricos utilizados : - Power Management Controller (PMC) - Analog Front-End Controller (AFEC) - Universal Synchronous Asynchronous Receiver Transceiver (USART)

Código Visão geral

```
/* **** */
/* Includes */
/* **** */
...

/* **** */
/* Defines */
/* **** */
...

/* **** */
/* Funcoes */
/* **** */

/**
 * \brief Configure UART console.
 * BaudRate : 115200
 * 8 bits
 * 1 stop bit
 * sem paridade
 */
static void configure_console(void)
{
    ...
}
```

```

/**
 * converte valor lido do ADC para temperatura em graus celsius
 * input : ADC reg value
 * output: Temperature in celsius
 */
static int32_t convert_adc_to_temp(int32_t ADC_value)
{
    ...
}

/*****
 * Call backs / Handler
 *****/

/**
 * \brief AFEC interrupt callback function.
 */
static void AFEC_Temp_callback(void){
    g_ul_value = afec_channel_get_value(AFEC0, AFEC_CHANNEL_TEMP_SENSOR);
    is_conversion_done = true;
}

/*****
 * Main
 *****/
int main(void)
{

    /* Initialize the SAM system. */
    sysclk_init();
    ioport_init();
    board_init();

    /* inicializa delay */
    delay_init(sysclk_get_cpu_hz());

    /* inicializa console (printf) */
    configure_console();

    /* Output example information. */
    puts(STRING_HEADER);

    /*****
     * Ativa e configura AFEC
     *****/
    ...

```

```

...

while (1) {
    if(is_conversion_done == true) {
        printf("Temp : %d \r\n", convert_adc_to_temp(g_ul_value));
        afec_start_software_conversion(AFEC0);
        delay_s(1);
    }
}
}

```

configure_console(void)

A função `configure_console()` inicializa o periférico **USART** para controlar os pinos (PA21, PB4) do microcontrolador, esses pinos estão conectados no EDBG (gravador embarcado no kit) que possibilitam abrirmos um terminal no computador e recebermos/enviarmos dados via uma comunicação serial, similar como é feito no arduino.

A função configura também o **stdio** para utilizar esse periférico como saída e entrada das funções **printf**, **getc**,

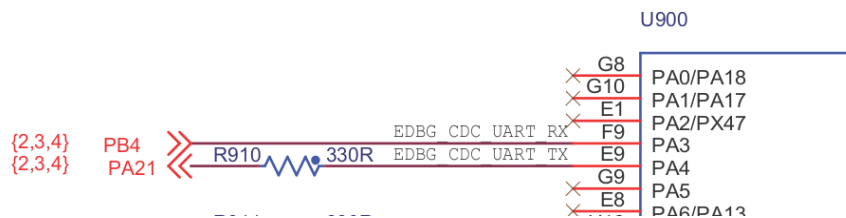


Figure 2: edbg comunicação UART - XPLD

static int32_t convert_adc_to_temp(int32_t ADC_value)

Essa é uma função que converte o valor lido (`ADC_value`) pelo conversor analógico digital e converte para temperatura em graus celsius. Sua implementação são duas simples transformações :

1. Regra de três para converter o valor em bits para Volts
2. Equacionamento linear para transformar Volts em graus Celsius

static void AFEC_Temp_callback(void)

É a função que é chamada quando a conversão analógica digital é concluída. Nessa função fazemos a leitura do valor da conversão analógica e salvamos o seu valor em uma variável global.

```
g_ul_value = afec_channel_get_value(AFEC0, AFEC_CHANNEL_TEMP_SENSOR);
```

Callbacks são interrupções já tratadas por software, no caso do AFEC não tratamos diretamente o HANDLER mas sim callbacks que são configuradas para cada canal do AFEC0:

```
main(){  
    ...  
  
    /* configura call back */  
    afec_set_callback(AFEC0, AFEC_INTERRUPT_EOC_11,    AFEC_Temp_callback, 1);  
    ...  
}
```

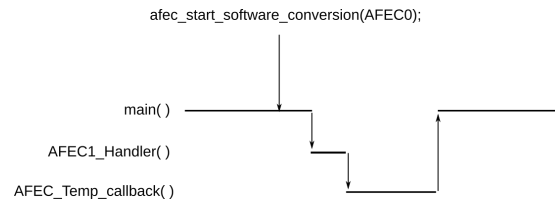


Figure 3: CallBack

Na função chamada no main, fazemos a associação do canal 11 que o sensor de temperatura está conectado a função de callback **AFEC_Temp_callback** nesse caso não trabalhamos com o Handler direto, mas sim com chamadas de funções internas a ele. O Handler do AFEC é gerenciado pelas funções da Atmel ASF.

AFEC

O Analog Front-End Controller (AFEC) nesse exemplo é configurado para operar lendo o sensor de temperatura interno do microcontrolador, que está conectado na entrada 11 do MUX.

Programando

A seguir uma sugestão de passos para implementar :

1. Copie o exemplo LCD-EXT2 para a pasta da aula 13
2. Faça um merge do main.c com o ADC.c
 - Dica: utilize o programa meld
3. No ASF-Wizard inclua :
 - AFEC : drivers do ADC
 - FREERTOS : biblioteca do RTOS
4. Execute o código, você deve obter na UART a temperatura do uC a cada segundo.
5. Porte o código para executar com tasks no RTOS
 - Aqui você pode ter uma task para inicializar o trigger, ou pode utilizar um TC (interrupção para isso).
 - Temperatura amostrada em 2Hz
6. Exiba o valor de temperatura no LCD
7. Adquira a resistência do potenciômetro e envie via UART
 - Resistência amostrada em 5Hz
 - Escolha uma nova entrada do mux do AFE0
 - (dica: utilize o pino 4 do EXT-1, AFE1_AD6)

Entrega Final

1. Exiba o valor da resistência no LCD
2. Crie um gráfico que exiba as duas grandezas no tempo (valor X tempo).