

# Introdução a Engenharia Elétrica - 323100

## Aula S8

---

## Módulo 1 – Conversores AD e DA

---

Escola Politécnica da Universidade de São Paulo

Departamentos da Engenharia Elétrica

PCS Computação e Sistemas Digitais

PEA Energia e Automação Elétricas

PSI Sistemas Eletrônicos

PTC Telecomunicações e Controle



V1.1

Agosto de 2014



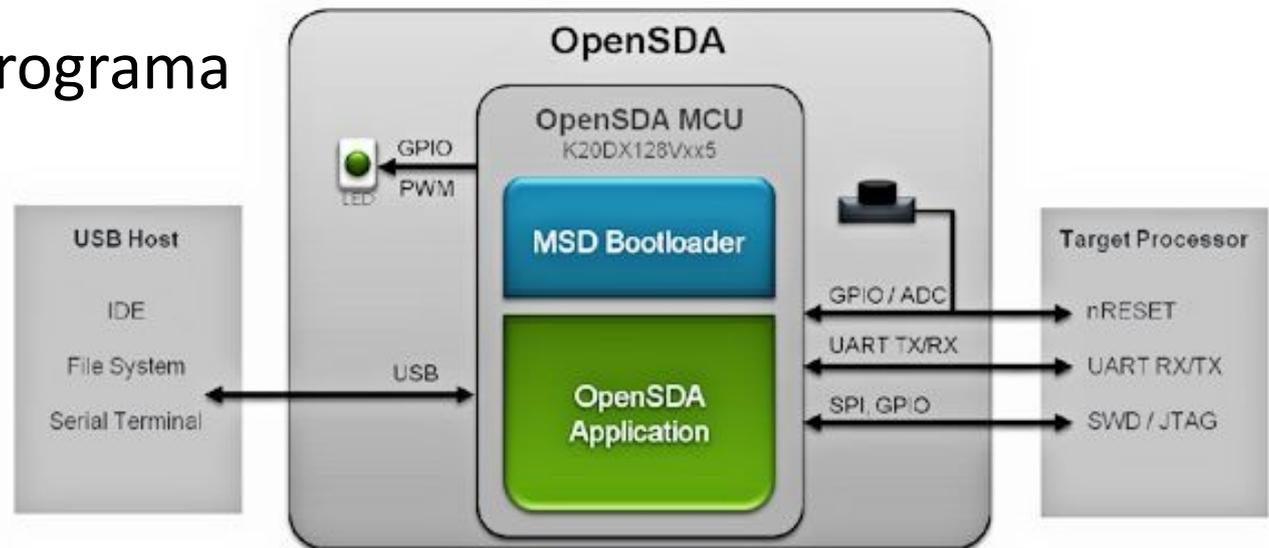
# Sumário

- Porta serial de comunicação com o Kit
- Sinais analógicos e digitais
- Conversores Analógico - Digitais
- Conversor Digital - Analógico



# Porta serial virtual via USB

- Atualmente: sistema de arquivos
  - Para armazenar os executáveis produzidos pelos programadores.
- Nova função: serial *virtual* através da USB
  - Para comunicação com o Kit
  - Para debugar o programa





## Como se preparar para o uso...

- Instalar o driver da porta serial virtual

No windows, uso o arquivo **PEDrivers\_install.exe**

Mais detalhes, vide apresentação específica no moodle.

- Instalar um programa emulador de terminal

No windows, uso o arquivo **teraterm-4.84.exe**

Mais detalhes, vide apresentação específica no moodle.

Obs.: Há outros programas emuladores de terminal, tais como o Hyperterminal, Putty, etc.



Esses programas já devem estar instalados na sala de aula.



## Como utilizar no seu programa

- Em geral, utilize chamadas para a rotina printf da forma tradicional do ANSI C.

```
printf("Ola mundo!!!\r\n");
```

- O kit KL25Z envia os caracteres através da porta serial virtual até o seu computador.
- Seu computador devem estar executando o programa emulador de terminal Teraterm (ou similar), com a serial USB virtual selecionada como porta de comunicação.
- As definições padrão da comunicação são: velocidade de 9600 bps, 8 bits de dados, paridade nenhuma, 1 bit de parada.



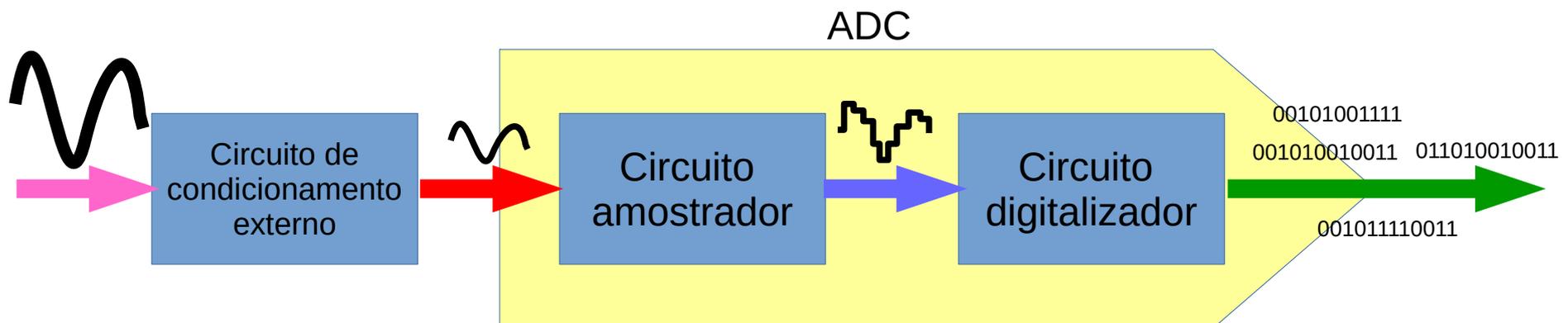
# Conversão analógico - digital

- Sinais analógicos – sinais análogos às grandezas físicas.
  - São contínuos no tempo.
  - Podem ser medidos ou aferidos em uma ampla gama de valores e unidades.
- Como um computador/sistema eletrônico pode usar esses sinais?
  1. Convertendo a grandeza física em uma tensão equivalente, ou fazendo sua transdução, usando um sensor por exemplo.
  2. Transformando o valor instantâneo do sinal de tensão, em uma representação numérica simbólica, com uma regra de formação conhecida.
- Esse dispositivo é denominado de ADC (analog to digital converter).



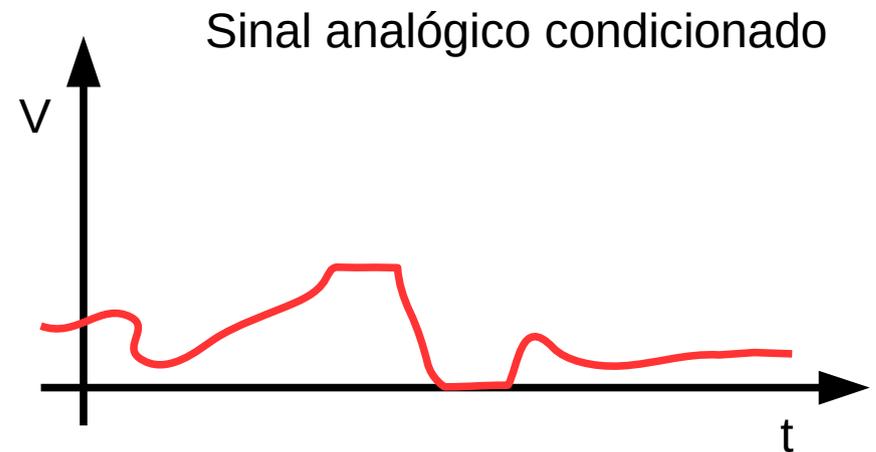
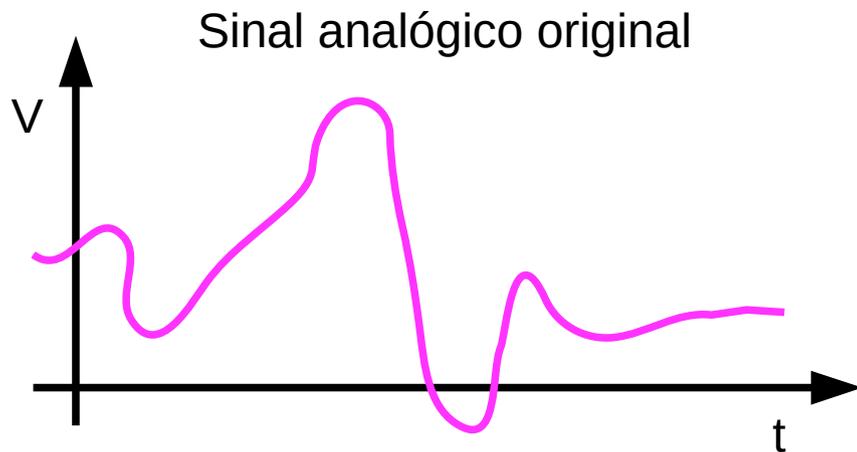
# Conversor Analógico Digital (ADC)

- Dispositivo eletrônico capaz de gerar uma representação digital de uma grandeza analógica, tipicamente uma tensão.
- Empregados na interface entre dispositivos digitais (microcontroladores, etc) e dispositivos analógicos, como sensores de temperatura, pressão, audio, vídeo, etc.





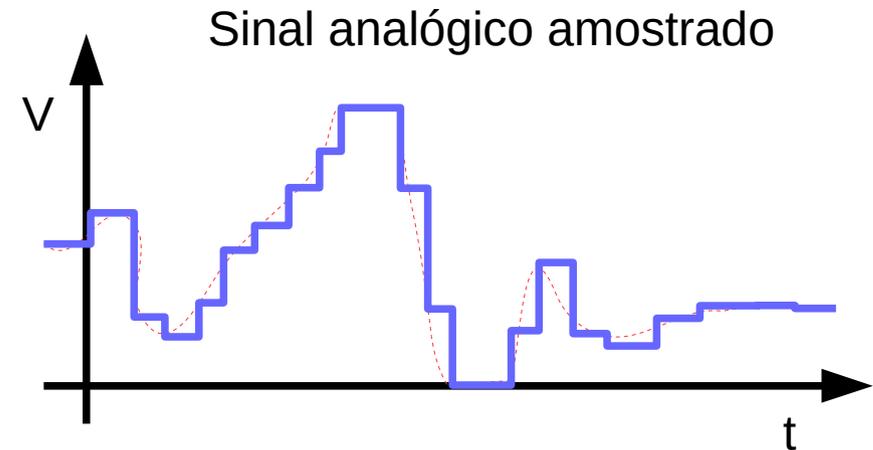
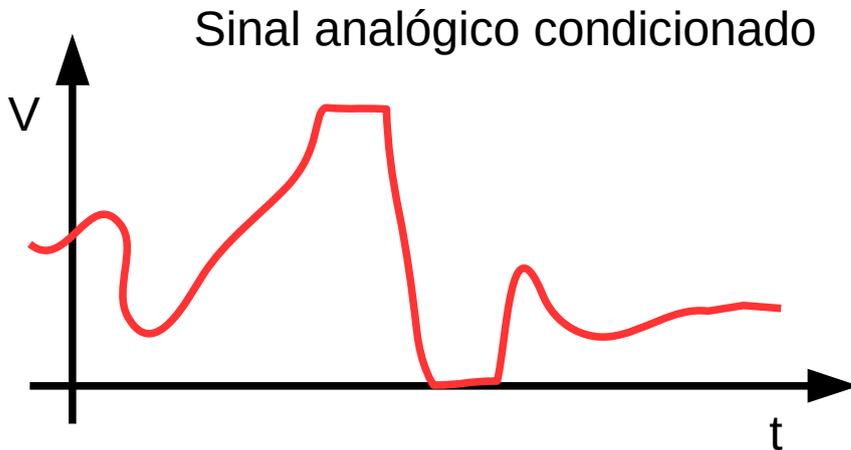
# Conversão analógico/digital - condicionamento



- Problema: Os sinais analógicos podem assumir valores dos mais diversos em magnitude. Os computadores só trabalham com tensões próximas de suas tensões de alimentação. Como compatibilizar esses níveis de tensão?
- Solução: O computador utiliza um circuito de condicionamento para tornar as medidas condizentes com os limites do dispositivo. As magnitudes são reescaladas e limitadas, de forma a proteger os demais circuitos do ADC do microcontrolador.



# Conversão analógico/digital - amostragem

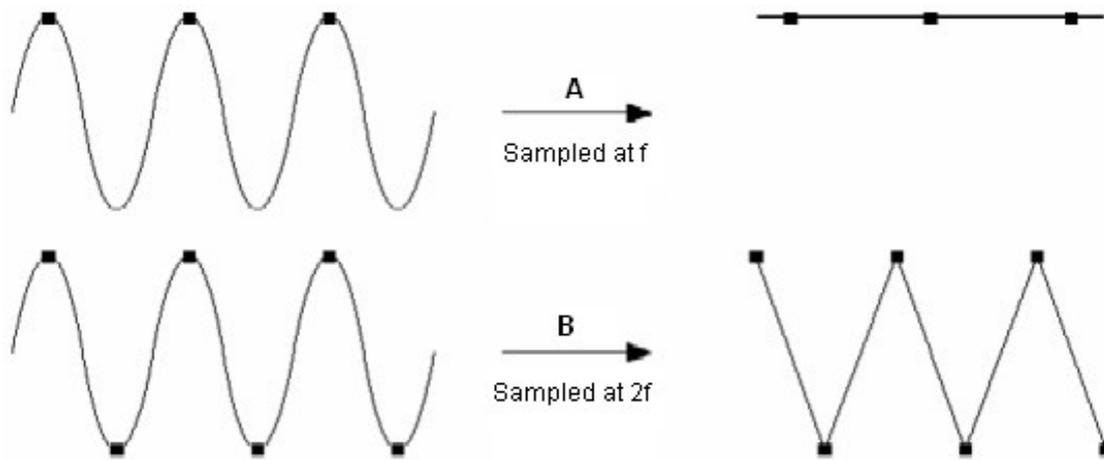


- Problema: Computadores executam operações de forma cíclica, conforme seu clock ou relógio interno, enquanto que um sinal analógico pode variar continuamente ao longo do tempo. Dessa forma, como deve-se registrar um sinal contínuo?
- Solução: O computador registra o valor dos sinais analógicos periodicamente, conforme uma dada frequência de aquisição, ajustada adequadamente pelo usuário. Um sinal contínuo torna-se uma série, uma sequência de valores.



# Amostragem no conversor analógico digital

- Processo de capturar o valor de um sinal durante um intervalo de tempo.
- Normalmente, os ADCs são usados para capturar amostras em intervalos regulares de tempo.
- A taxa ou frequência de captura (amostras/s) deve ser escolhida corretamente para que seja possível continuar a representar (observar) o sinal original, mesmo após essa amostragem.

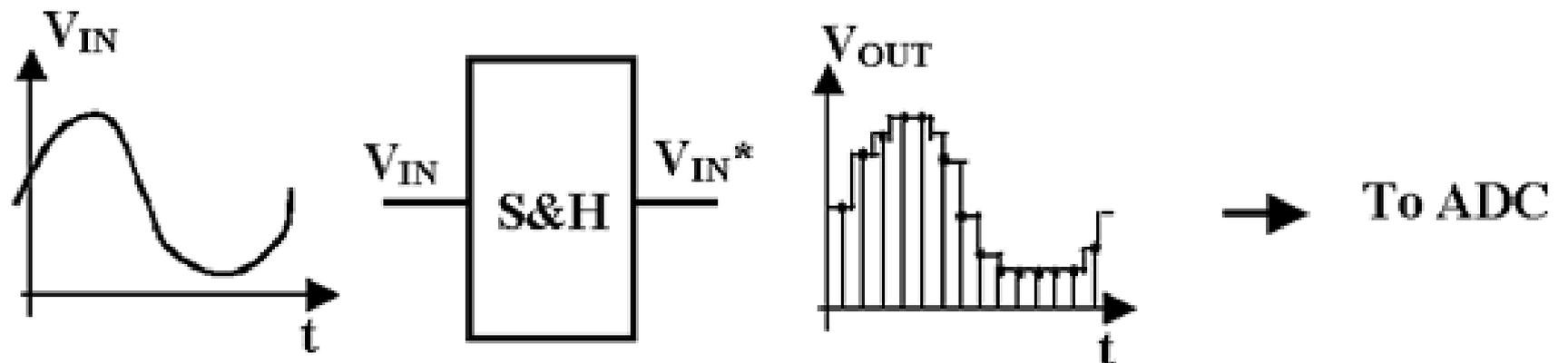
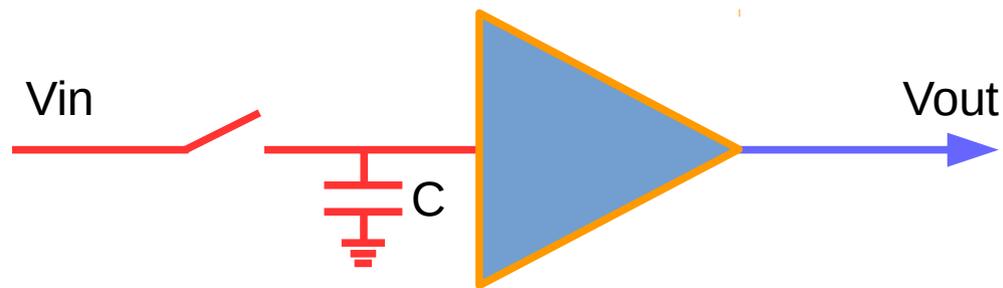


Exemplos de frequência de amostragem insuficiente para representar o sinal original analógico. Os pontos em preto são as amostras do sinal original.



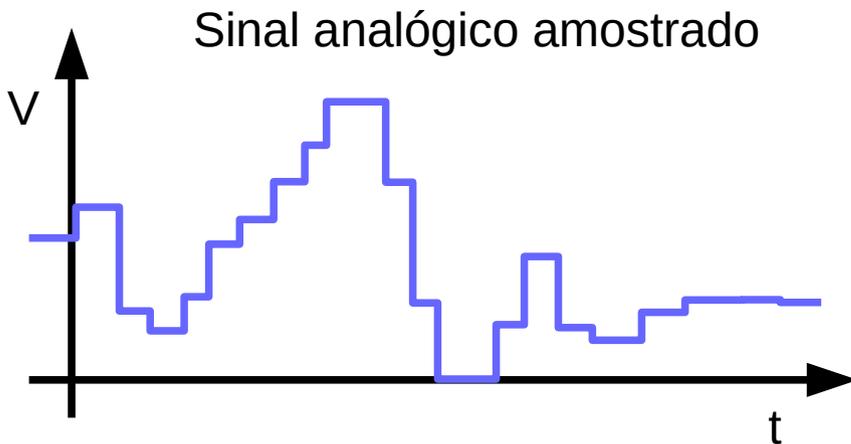
# Funcionamento da amostragem

- A amostragem é realizada por um circuito chamado *sample-and-hold* (S/H).
- Ele mantém constante o valor do sinal analógico de entrada enquanto o circuito seguinte de digitalização realiza a quantização do sinal.





# Conversão analógico/digital - digitalização



Série de valores digitalizados

100, 130, 50, 30, 55, 89, 105, 145, 160,  
255, 145, 56, 0, 0, 32, 78, 29, 25, 42, 48,  
49, 51, 50

- O computador pode classificar a tensão amostrada entre os níveis ou valores disponíveis em uma tabela.
- Os sinais que chamamos de digitais, utilizados até então no microcontrolador, foram digitalizados usando apenas dois níveis, 0 ou 1. Um conversor AD possui uma quantidade (resolução) muito maior de símbolos, valores ou patamares para representar esses sinal.
- Quanto melhor a resolução, melhor será o discernimento ou discretização dos valores assumidos pela tensão externa.



# Funcionamento da digitalização

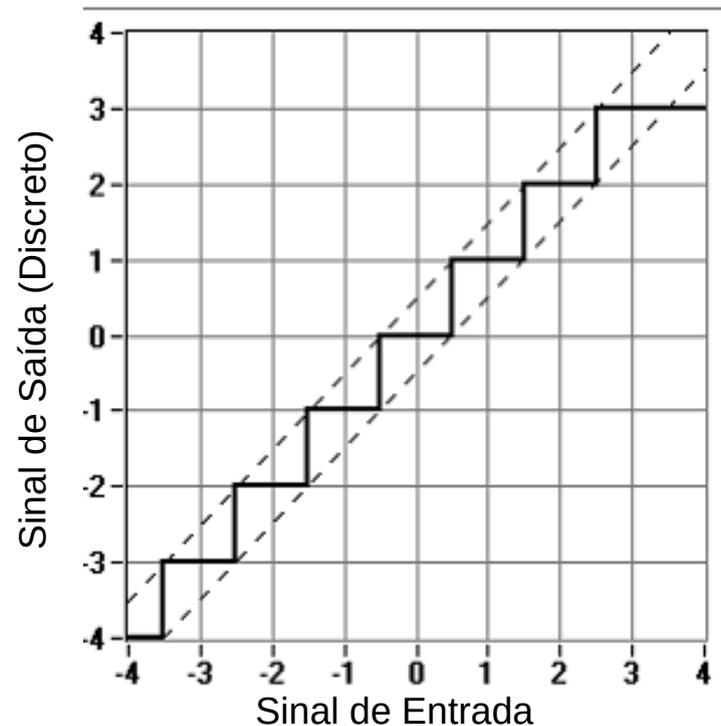
- Um ADC digitaliza um sinal amostrado utilizando um valor inteiro aproximado (quantizado), para representar cada amostra analógica, constituindo uma série de valores discretos (quantização).
- O valor escolhido é proveniente de uma lista pré-determinada.
  - O tamanho da lista é definida pelo número de bits do ADC (resolução).
  - Um conversor de N bits possui  $2^N$  valores para representar o sinal amostrado.
- Exemplo: supondo um conversor de 8 bits
  - $2^8 = 256$  valores distintos, por exemplo, de 0 a 255
  - Para uma tensão máxima na entrada de 3,3V (referência) → utiliza-se o valor máximo da tabela, ou seja, 255.
  - Para uma tensão mínima na entrada de 0,0 V → utiliza-se o menor valor da tabela, 0.
  - Todos os valores intermediários estão separados por níveis equivalentes a  $3,3/256 = 0,012$  [V/patamar]



# Conversor Analógico Digital (ADC)

- Quantização

- Normalmente, um ADC escolhe o valor mais próximo da amostra atual de uma lista uniformemente espaçada.
- Pode haver algum erro de aproximação (erro de quantização no processo).

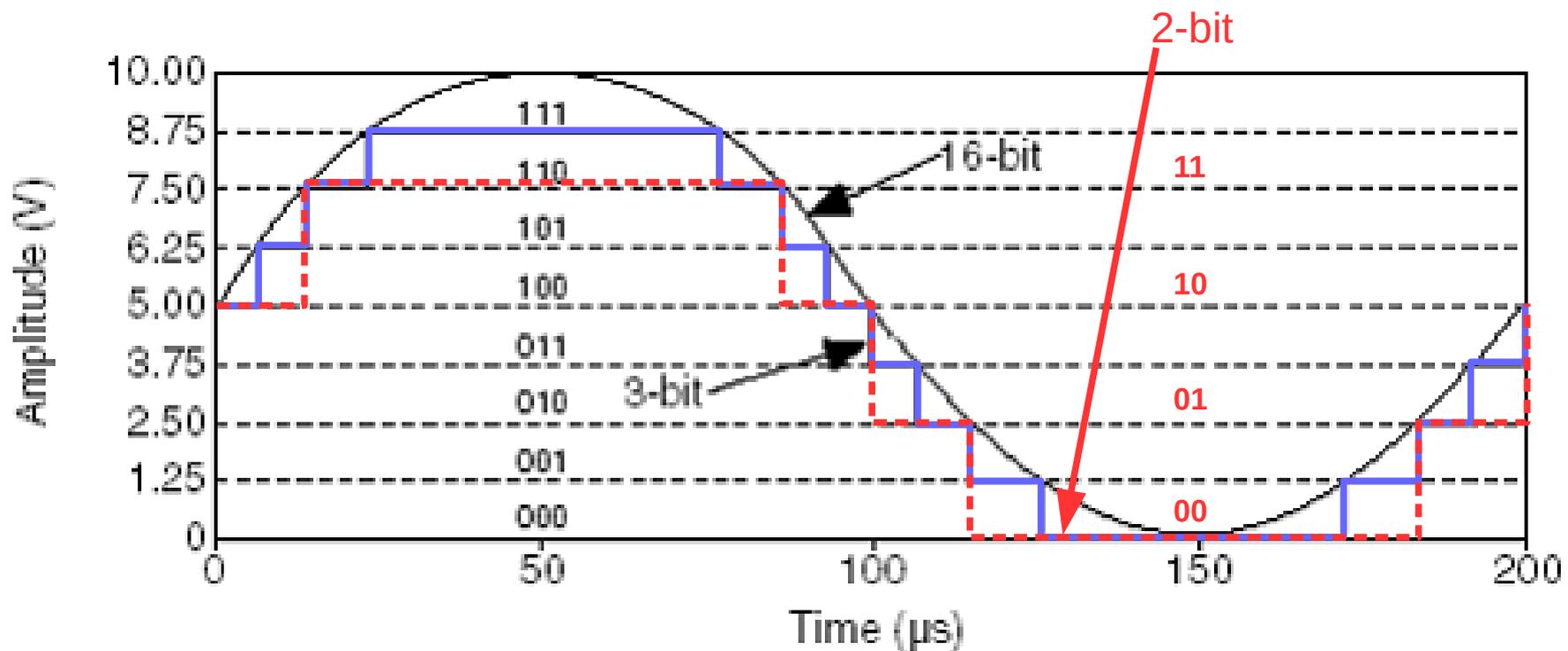




# Conversor Digital Analógico (ADC)

- Exemplo de Conversão

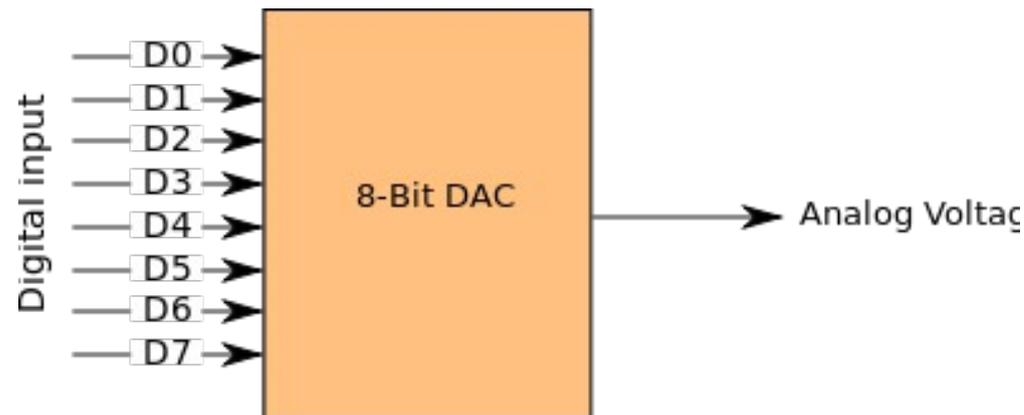
- Sinal Senoidal de entrada de 5kHz, oscilando entre 0 e 10 [V].
- Taxa de Amostragem de 100kHz (100.000 amostras/s).
- ADC de 2 Bits → Resolução de 2,5 [V]
- ADC de 3 Bits → Resolução  $10/(2^3)=1,25$  [V]
- ADC de 16 bits → Resolução de 0,000152 [V]





# Conversor Digital Analógico (DAC)

- Circuito eletrônico capaz de converter uma grandeza digital em uma grandeza analógica.
- Esse processo é denominado de síntese digital-analógica.
- Características Básicas:
  - Velocidade de conversão
  - Resolução ou quantidade de patamares para síntese do sinal analógico (número de bits).

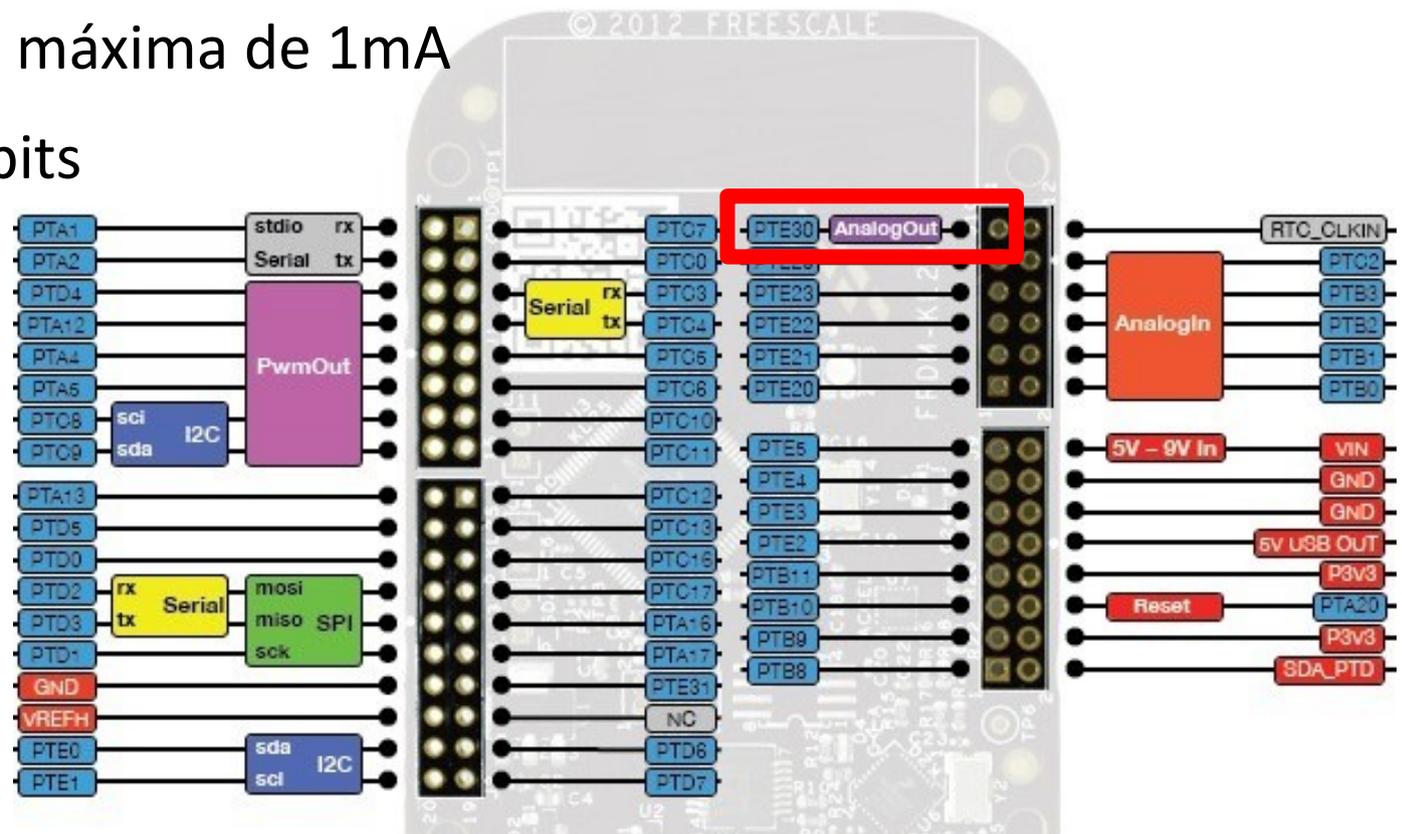






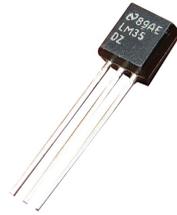
# Periféricos analógicos de saída do Kit KL25Z

- 1 canal de um DAC
  - Limites da tensão de saída de 0,0 a 3,3 [V]
  - Taxa de amostragem 5 a 30kHz
  - Corrente saída máxima de 1mA
  - Resolução 12-bits

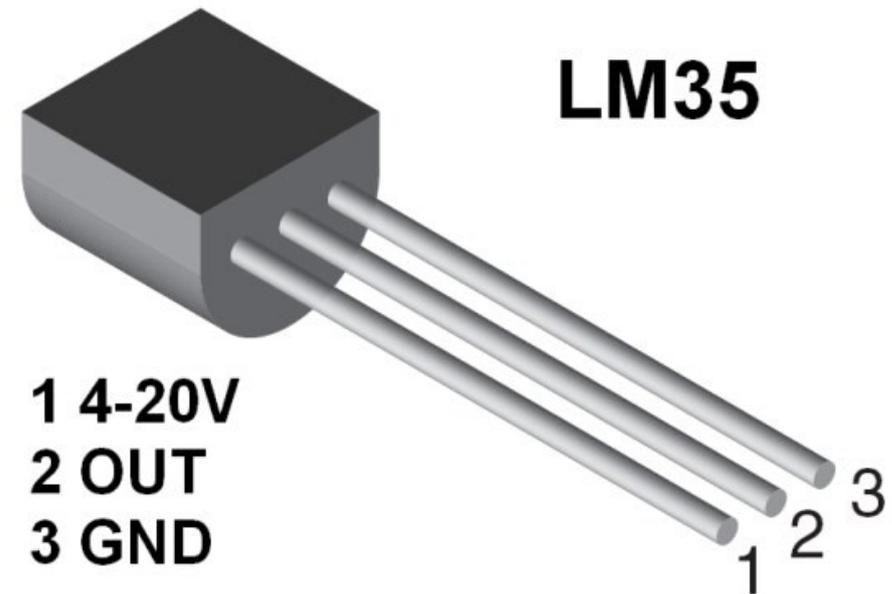




# Sensor LM-35



- **Transdutor Temperatura-Tensão**
- **Tensão de saída x Temperatura**
  - Relação linear
  - +10 mV / °C
- **Faixa de Operação**
  - -55°C até 150°C
- **Pinagem**
  - Pino 1 – Tensão de alimentação entre 4 a 20,0 [V]
  - Pino 2 – Saída do sinal analógico de tensão
  - Pino 3 – Terminal de terra ou negativo.



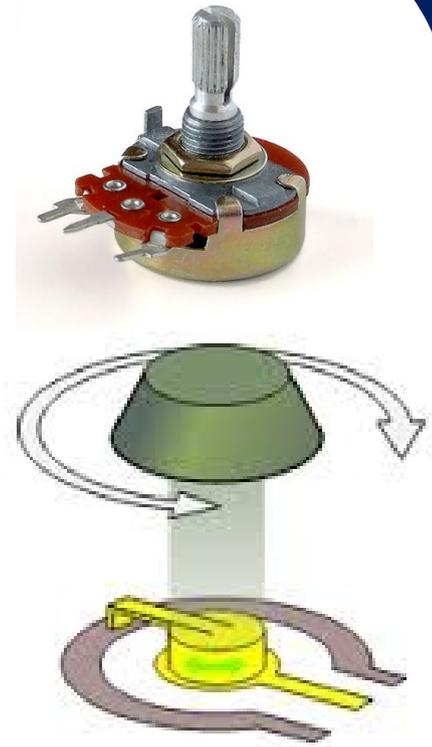
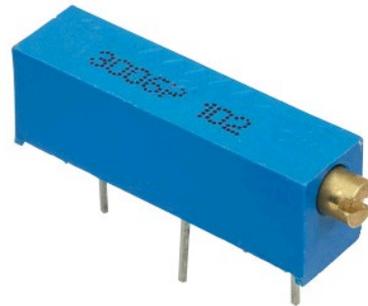
**Atenção: O LM-35 não pode ser ligado invertido, com as tensões erradas. O dispositivo pode queimar!!!!**





# Potenciômetro

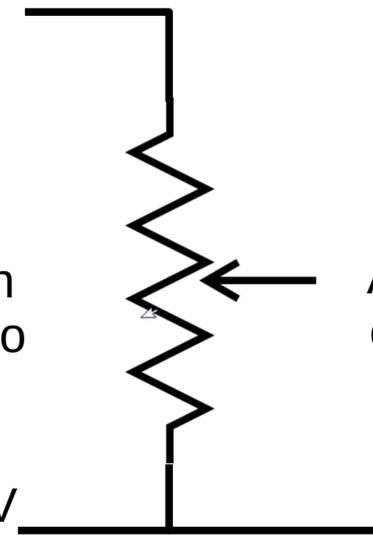
- Resistor variável.
- Possui um valor nominal de resistência e potência máximas.
- Possui três terminais.
- Permite implementar um divisor de tensão, para produzir uma tensão entre um valor máximo e mínimo.



+5V

Vin  
Fixo

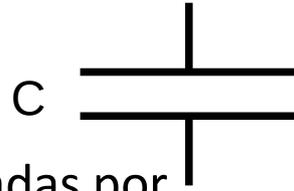
0V



Vout  
Ajustável  
entre 0 e  
5,0 V

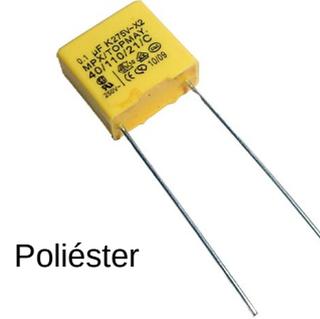


# Capacitores

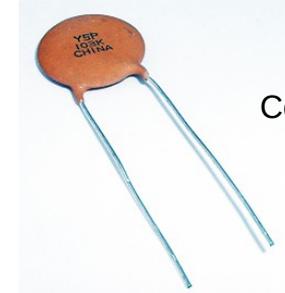


- Duas placas metálicas paralelas, isoladas por um material dielétrico (isolante).
- Várias tecnologias de materiais isolantes (cerâmicos, eletrolíticos, etc).
- Armazenam energia na forma do acúmulo de cargas elétricas entre suas placas.
- Utilizados tipicamente como acumuladores, ou em circuitos de filtragem.
- Alguns tipos são **polarizados**. Atente à forma de ligação do componente ao circuito.
- A relação entre a tensão e a corrente em um capacitor obedece a expressão:

$$v(t) = \frac{1}{C} \cdot \int_{t_0}^t i(t) \cdot dt + v(t_0)$$



Poliéster



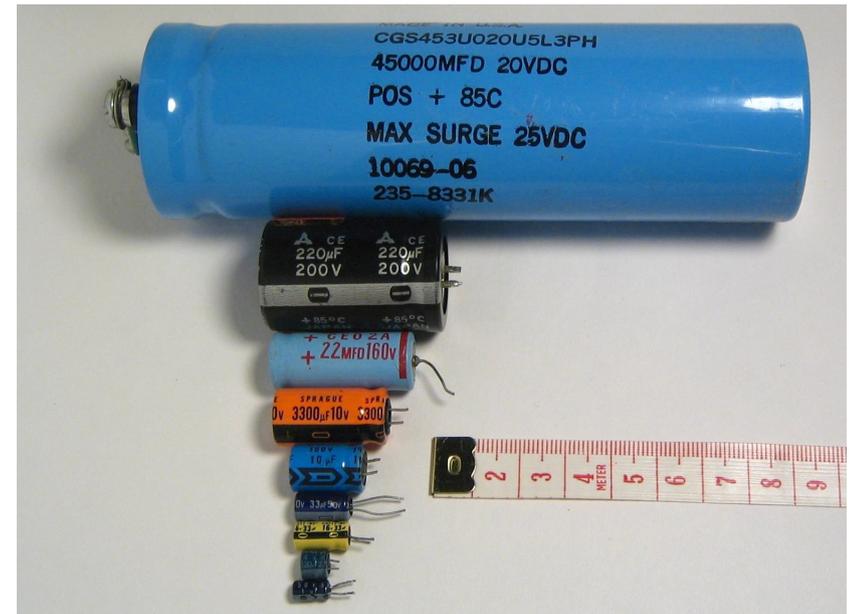
Cerâmico



Eletrolítico Polarizado

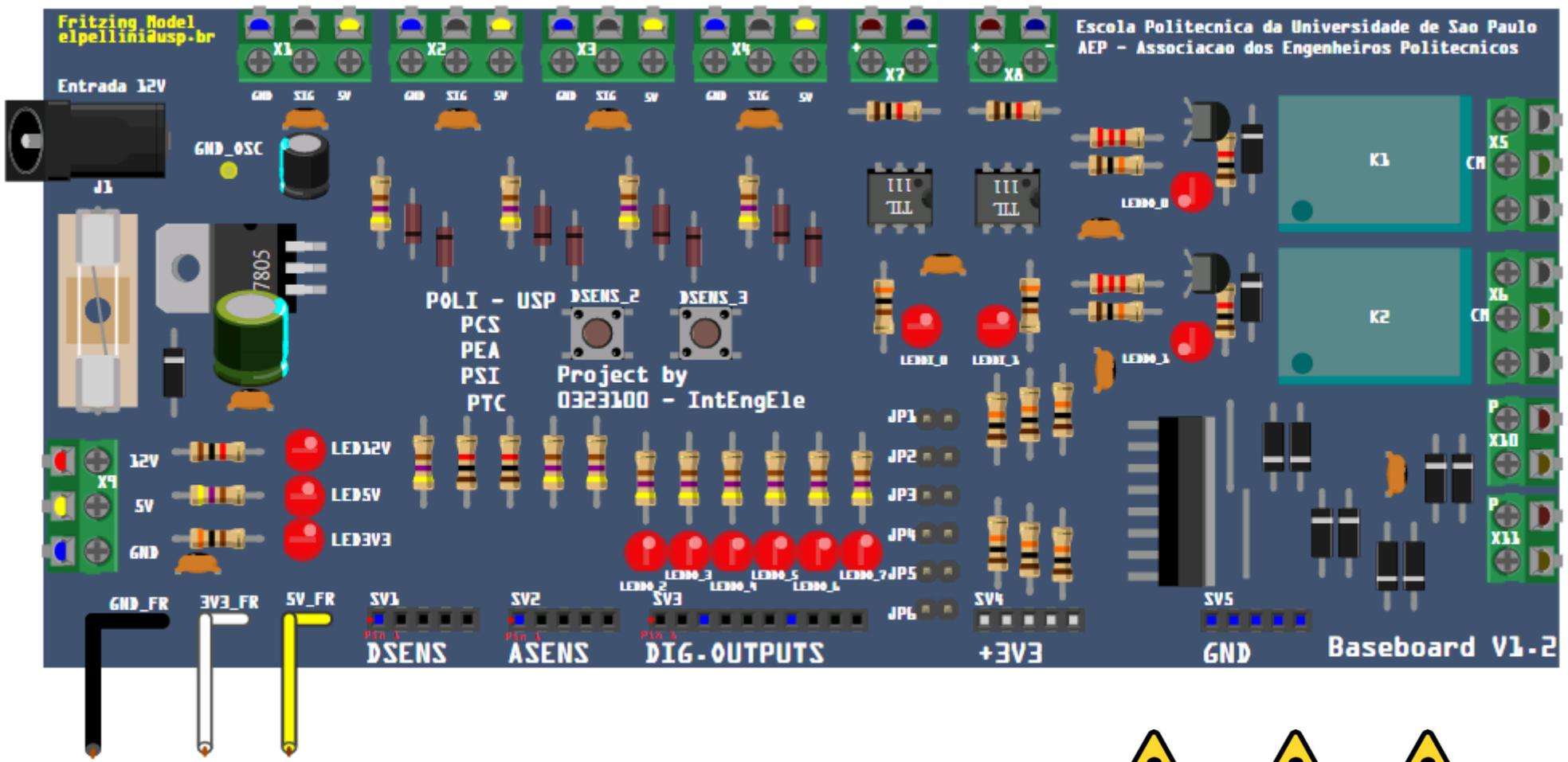


Tântalo - polarizado





# Baseboard da disciplina



Atenção à documentação e manual de uso.



- A placa só está pronta para uso se as três luzes LED12V, LED5V e LED3V3 estiverem ligadas!!!



# Exercício 1

## Exercício SerialComm

```
#include "mbed.h"

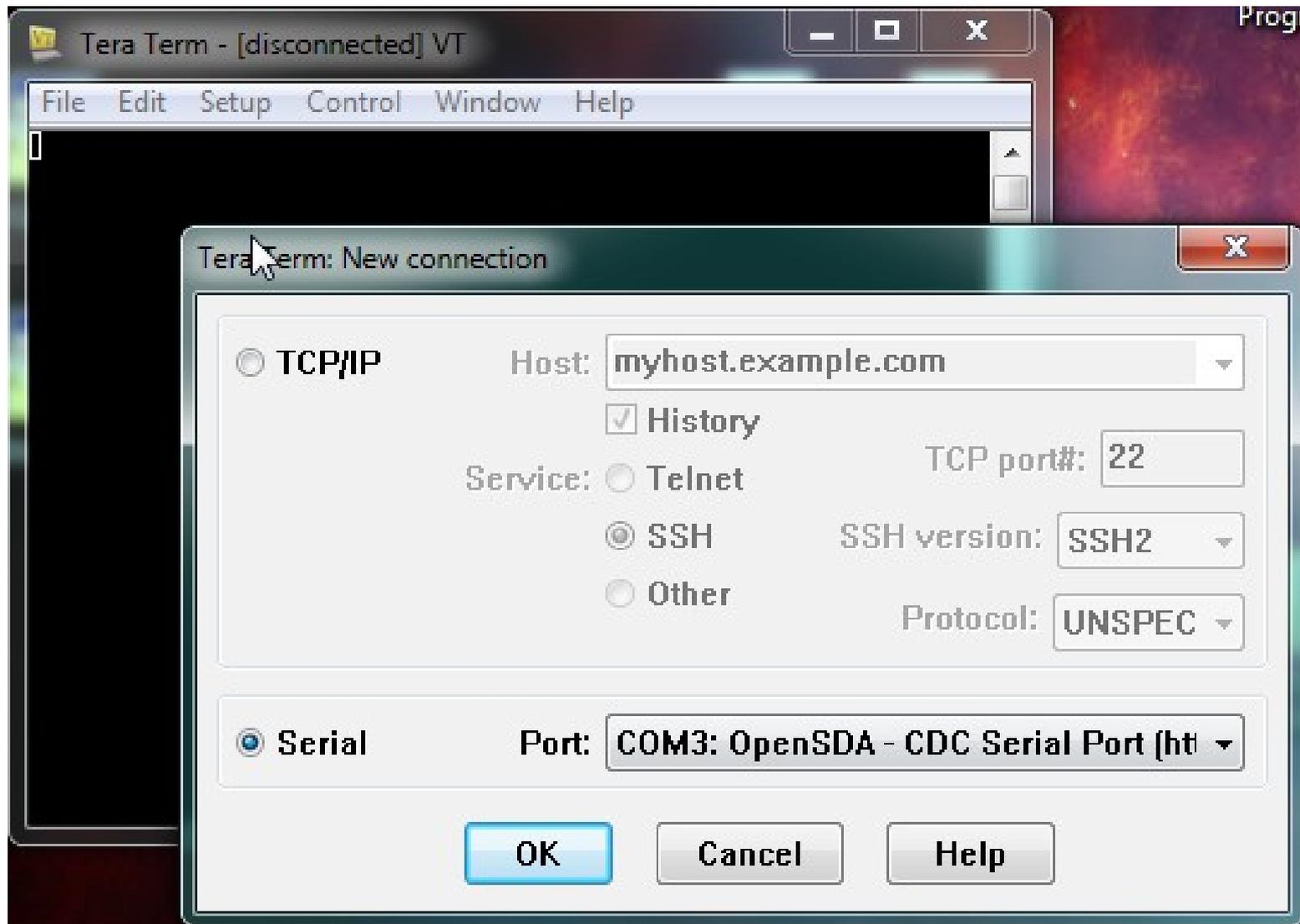
Serial PC(USBTX, USBRX);

int main() {
    int Contador;
    Contador=0;
    while(1) {
        PC.printf("Hello Word #%d\r\n",Contador);
        Contador++;
        wait(0.5); // 500 ms
    }
}
```



# Exercício 1

## Tela do teraterm para abertura da porta serial do kit





# Exercício 1

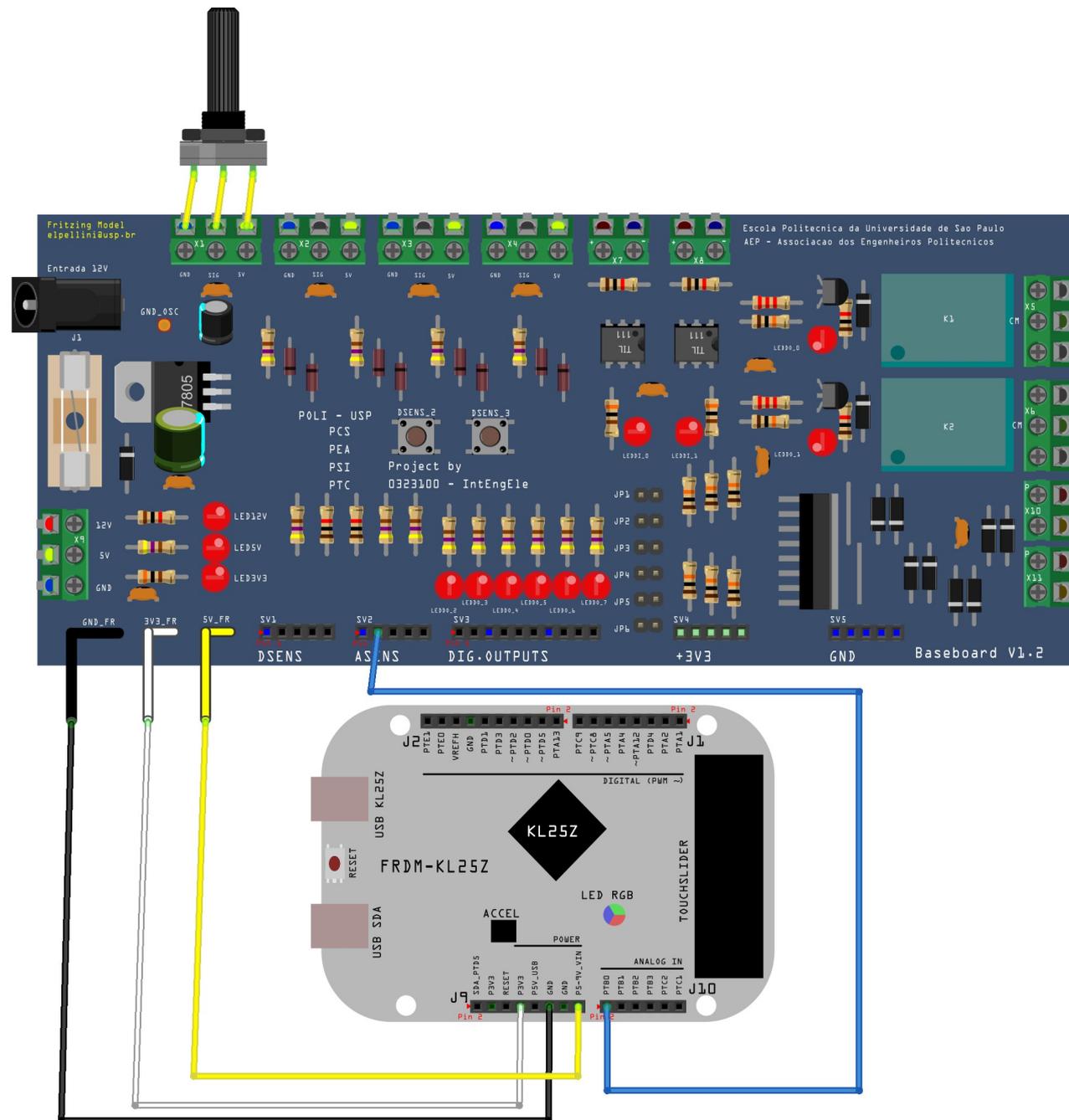
## Exercício SerialComm – Saída esperada

```
COM3:9600baud - Tera Term VT
File Edit Setup Control Window Help
Hello Word #158
Hello Word #159
Hello Word #160
Hello Word #161
Hello Word #162
Hello Word #163
Hello Word #164
Hello Word #165
Hello Word #166
Hello Word #167
Hello Word #168
Hello Word #169
Hello Word #170
Hello Word #171
Hello Word #172
Hello Word #173
Hello Word #174
Hello Word #175
Hello Word #176
Hello Word #177
Hello Word #178
Hello Word #179
Hello Word #180
```



# Exercício 2 - Montagem ADC\_Pot

Veja o arquivo PDF da montagem ou o arquivo do Fritzing para mais detalhes.



Não esqueça de ligar a fonte de 12,0 V que alimenta a baseboard.



## Exercício ADC\_Pot

```
#include "mbed.h"

#define MV(x) ((3.3*x)/0xFFFF)

AnalogIn Entrada_AN(PTB0);
Serial PC(USBTX, USBRX);

int main() {
    unsigned short Medida_Inteira;
    float Valor_Normalizado;
    while(1) {
        Valor_Normalizado = Entrada_AN;
        Medida_Inteira = Entrada_AN.read_u16();
        PC.printf("Tensao: %d(Inteiro) %f(Float) %f(Volts)\r\n"
                ,Medida_Inteira, Valor_Normalizado, MV(Medida_Inteira));
        wait(0.5); // 500 ms
    }
}
```



## Exercício ADC\_Pot – Saída esperada

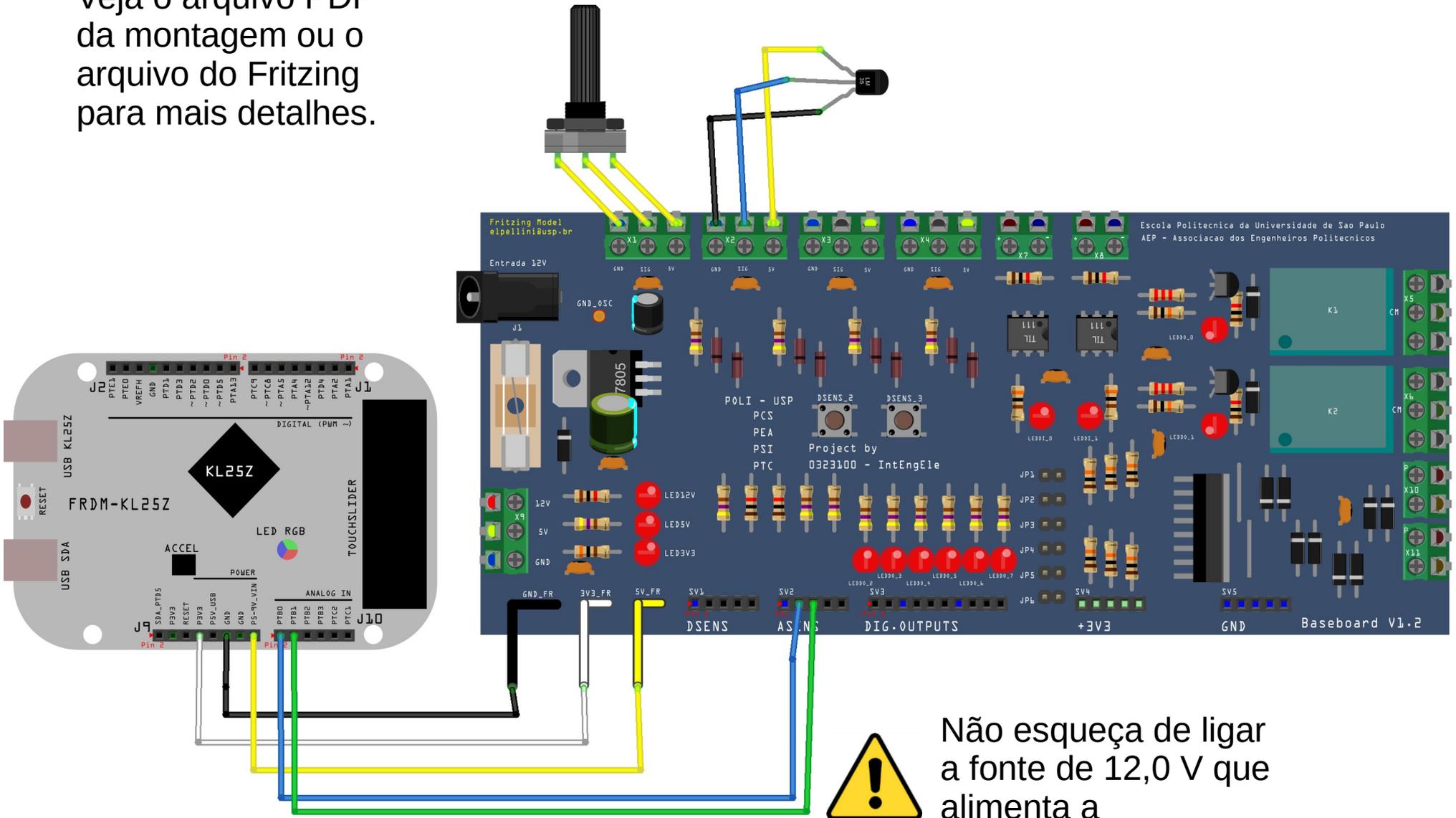
```
COM3:9600baud - Tera Term VT
File Edit Setup Control Window Help
Tensao: 65369(Inteiro) 0.999573(Float) 3.291641(Volts)
Tensao: 65471(Inteiro) 0.996979(Float) 3.296777(Volts)
Tensao: 65431(Inteiro) 0.999435(Float) 3.294763(Volts)
Tensao: 65468(Inteiro) 0.998184(Float) 3.296626(Volts)
Tensao: 65497(Inteiro) 0.999954(Float) 3.298087(Volts)
Tensao: 65535(Inteiro) 0.997223(Float) 3.300000(Volts)
Tensao: 65535(Inteiro) 0.997726(Float) 3.300000(Volts)
Tensao: 65535(Inteiro) 0.998657(Float) 3.300000(Volts)
Tensao: 65479(Inteiro) 1.000000(Float) 3.297180(Volts)
Tensao: 65535(Inteiro) 0.998047(Float) 3.300000(Volts)
Tensao: 65473(Inteiro) 1.000000(Float) 3.296878(Volts)
Tensao: 65535(Inteiro) 1.000000(Float) 3.300000(Volts)
Tensao: 65405(Inteiro) 1.000000(Float) 3.293454(Volts)
Tensao: 65535(Inteiro) 1.000000(Float) 3.300000(Volts)
Tensao: 65512(Inteiro) 1.000000(Float) 3.298842(Volts)
Tensao: 65535(Inteiro) 1.000000(Float) 3.300000(Volts)
Tensao: 65395(Inteiro) 0.999832(Float) 3.292950(Volts)
Tensao: 65535(Inteiro) 1.000000(Float) 3.300000(Volts)
Tensao: 65535(Inteiro) 0.998474(Float) 3.300000(Volts)
Tensao: 65324(Inteiro) 0.998398(Float) 3.289375(Volts)
Tensao: 65535(Inteiro) 0.999130(Float) 3.300000(Volts)
Tensao: 51640(Inteiro) 0.788251(Float) 2.600320(Volts)
Tensao: 38427(Inteiro) 0.587961(Float) 1.934983(Volts)
Tensao: 33065(Inteiro) 0.505913(Float) 1.664981(Volts)
Tensao: 33004(Inteiro) 0.505409(Float) 1.661909(Volts)
Tensao: 33021(Inteiro) 0.504234(Float) 1.662765(Volts)
```

Varie o Pot!!



# Exercício 3 - Montagem ADC\_Pot\_Temp

Veja o arquivo PDF da montagem ou o arquivo do Fritzing para mais detalhes.



Não esqueça de ligar a fonte de 12,0 V que alimenta a baseboard.



## Exercício ADC\_Pot\_Temp

```
#include "mbed.h"

AnalogIn Entrada_AN_Potenciometro(PTB0);
AnalogIn Entrada_AN_Sensor(PTB1);

Serial PC(USBTX, USBRX);

int main()
{
    float Medida_Potenciometro, Medida_Sensor, Temperatura;
    while(1) {
        Medida_Potenciometro=Entrada_AN_Potenciometro*3.3;
        Medida_Sensor=Entrada_AN_Sensor*3.3;
        Temperatura=Medida_Sensor/10E-3; //10mV/C
        PC.printf("Pot.: %3.4f [V] ", Medida_Potenciometro);
        PC.printf("LM35: %3.4f [V] ", Medida_Sensor);
        PC.printf("LM35: %3.1f [oC] \r\n", Temperatura);
        wait(0.5); // 500 ms
    }
}
```



# Exercício 3 - Montagem ADC\_Pot\_Temp

## Exercício ADC\_Pot\_Temp – Saída esperada



Observe a variação da temperatura!!!

Além do sinal esperado, existe um ruído substancial!!!

As medidas possuem uma incerteza de mais de 5 °C!!!

Esse problema do ruído é ainda mais expressivo se o LM35 estiver distante da placa Baseboard, com o uso de um conjunto de cabos de comprimento maior.

Como resolver isso?!?!

```
COM3:9600bau - Tera Term VT
File Edit Setup Control Window Help
Pot.: 1.4967 [V] LH35: 0.2600 [V] LH35: 26.0 [oC]
Pot.: 1.4416 [V] LH35: 0.2588 [V] LH35: 25.9 [oC]
Pot.: 1.4917 [V] LH35: 0.2538 [V] LH35: 25.4 [oC]
Pot.: 1.4883 [V] LH35: 0.2554 [V] LH35: 25.5 [oC]
Pot.: 1.4812 [V] LH35: 0.2485 [V] LH35: 24.9 [oC]
Pot.: 1.4486 [V] LH35: 0.2503 [V] LH35: 25.0 [oC]
Pot.: 1.4553 [V] LH35: 0.2533 [V] LH35: 25.3 [oC]
Pot.: 1.4520 [V] LH35: 0.2552 [V] LH35: 25.5 [oC]
Pot.: 1.4946 [V] LH35: 0.2583 [V] LH35: 25.8 [oC]
Pot.: 1.4503 [V] LH35: 0.2441 [V] LH35: 24.4 [oC]
Pot.: 1.4958 [V] LH35: 0.2561 [V] LH35: 25.6 [oC]
Pot.: 1.4500 [V] LH35: 0.2464 [V] LH35: 24.6 [oC]
Pot.: 1.4521 [V] LH35: 0.2475 [V] LH35: 24.7 [oC]
Pot.: 1.4786 [V] LH35: 0.2550 [V] LH35: 25.5 [oC]
Pot.: 1.5024 [V] LH35: 0.2521 [V] LH35: 25.2 [oC]
Pot.: 1.4967 [V] LH35: 0.2511 [V] LH35: 25.1 [oC]
Pot.: 1.4523 [V] LH35: 0.2517 [V] LH35: 25.2 [oC]
Pot.: 1.4963 [V] LH35: 0.2595 [V] LH35: 26.0 [oC]
Pot.: 1.4491 [V] LH35: 0.2577 [V] LH35: 25.8 [oC]
Pot.: 1.4403 [V] LH35: 0.2573 [V] LH35: 25.7 [oC]
Pot.: 1.4934 [V] LH35: 0.2525 [V] LH35: 25.3 [oC]
Pot.: 1.5021 [V] LH35: 0.2604 [V] LH35: 26.0 [oC]
Pot.: 1.4985 [V] LH35: 0.2513 [V] LH35: 25.1 [oC]
Pot.: 1.4954 [V] LH35: 0.2495 [V] LH35: 24.9 [oC]
Pot.: 1.4953 [V] LH35: 0.2534 [V] LH35: 25.3 [oC]
Pot.: 1.4908 [V] LH35: 0.2511 [V] LH35: 25.1 [oC]
```

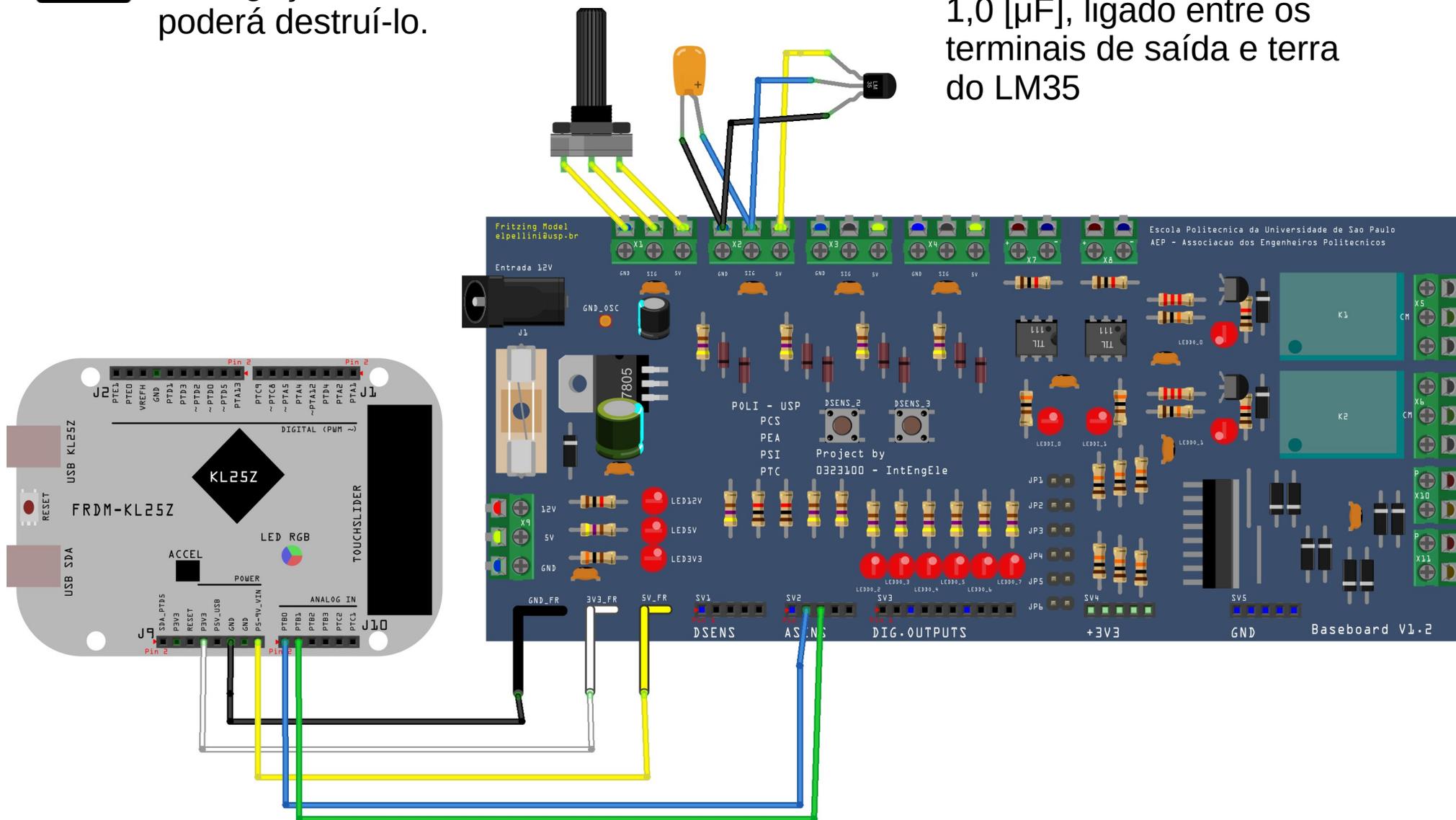


# Alteração na montagem – Sugestão 1



Cuidado com a polaridade do capacitor. Sua ligação invertida poderá destruí-lo.

Eliminação de ruído com capacitor de 470,0 nF a 1,0  $\mu\text{F}$ , ligado entre os terminais de saída e terra do LM35





## Exercício ADC\_Pot\_Temp – Uso de médias

```
#include "mbed.h"
AnalogIn Entrada_AN_Potenciometro(PTB0);
AnalogIn Entrada_AN_Sensor(PTB1);
Serial PC(USBTX, USBRX);

int main()
{
    float Medida_Potenciometro, Medida_Sensor, Temperatura, Media=0.0;
    while(1) {
        Medida_Potenciometro=Entrada_AN_Potenciometro*3.3;
        Medida_Sensor=Entrada_AN_Sensor*3.3;
        Temperatura=Medida_Sensor/10E-3; //10mV/C
        Media = (4.0*Media + Temperatura)/5.0;
        PC.printf("Pot.: %3.4f [V] ", Medida_Potenciometro);
        PC.printf("LM35: %3.4f [V] ", Medida_Sensor);
        PC.printf("LM35: %3.1f [oC] ", Temperatura);
        PC.printf("Med.: %3.1f [oC] \r\n", Media);
        wait(0.5); // 500 ms
    }
}
```



## Exercício DAC – Para verificar no osciloscópio

```
#include "mbed.h"
#define PI 3.14159

AnalogOut Saida_AN(PTE30);

int main()
{
    float Frequencia,dT,Angulo;
    dT=1E-3;
    Frequencia=60;
    Angulo=0;
    while(1) {
        Angulo=Angulo+2*PI*Frequencia*dT;
        Saida_AN=0.5*sin(Angulo)+0.5;
        wait(dT);
    }
}
```



## Exercício ADC-DAC – Para verificar no Osc.

```
#include "mbed.h"
#define PI 3.14159

AnalogOut Saida_AN(PTE30);
AnalogIn Entrada_AN_Potenciometro(PTB0);

int main()
{
    float Frequencia,dT,Angulo;
    dT=1E-3;
    Frequencia=60;
    Angulo=0;
    while(1) {
        Angulo=Angulo+2*PI*Frequencia*dT;
        Saida_AN=(Entrada_AN_Potenciometro/2)*sin(Angulo)+0.5;
        wait(dT);
    }
}
```