# Tutorial - Comunicação UART entre Raspberry e Arduíno

Programação de Periféricos – CC – FACIN – PUCRS

### Introdução

Nesta etapa será exercitada a comunicação entre a Raspberry e o microcontrolador Arduíno empregando um protocolo padrão de comunicação. Será explorado o barramento UART comunicação serial), o qual é amplamente empregado na industria. O principal objetivo deste tutorial é permitir compreender como controlar periféricos através de barramentos de comunicação.

Os passos para a elaboração do código a ser compilado, transferido e executado no processador arduíno foram explorados na última aula. Para a execução destes passos consulte o tutorial realizado na última aula (i.e. Tutorial Raspberry & Arduíno).

# Configuração da GertBoard

Para este tutorial, mantenha a configuração para gravação do microcontrolador e a ligação do Pino PB5 ao BUF1 conforme exemplo do LED piscante do tutorial sobre o Arduíno. Além desta configuração básica, será necessário fazer a ligação da UART da Raspberry com a UART do microcontrolador. Para isso, basta ligar dois jumpers no pinos GP15 e GP14 conforme figura abaixo.



Figura 1- Ligação da UART da Raspberry com a UART do microcontrolador

#### Programando a Serial no Arduíno

O Arduíno permite fácil utilização da UART. O exemplo abaixo, lê e reenvia cada byte recebido da serial (echo server). Ainda, a recepção e reenvio do byte é sinalizado visualmente através de um LED conectado ao pino PB5 do microcontrolador. Note que a UART está sendo configurada para operar a 9600 bytes/s.

```
- Arduino Serial Echo Server -
Cada byte recebido e reenvidado pela serial. Ainda durante a recepço
de um byte o LED ligado ao pino 13 pisca.
*/
void setup() {
  /* Inicializa a porta serial em 9600 bytes/s */
  Serial.begin(9600);
  /* Configura o pino de IO para piscar o led durante a recepção. */
 pinMode(13, OUTPUT);
}
void loop() {
  int inByte;
  /* Apaga LED */
  digitalWrite(13, LOW);
  /* Aquarda o recebimento de dados */
  if (Serial.available()) {
    /* liga o LED */
   digitalWrite(13, HIGH);
    /* Le byte da serial */
    inByte = Serial.read();
    /* Responde o byte pela serial */
    Serial.write(inByte);
    /* Desliga o LED */
    digitalWrite(13, LOW);
  }
```

Antes de utilizar o IDE do Arduíno é necessário configurar o ambiente para a frequência do cristal utilizado na GuertBoard, que é 12MHz. Caso isso não seja feito, a UART não fucionará corretamente. Para alterar essa configuração, abra o arquivo:

# \$ sudo vi /usr/share/arduino/hardware/arduino/boards.txt

Localize a propriedade "uno.build.f\_cpu=16000000L" e altere para "uno.build.f\_cpu=12000000L".

Salve o arquivo e vá para o IDE do Arduíno. Use o IDE para compilar o exemplo acima e graveo no microcontrolador conforme aprendido na última aula. Depois de gravado, o LED deve ficar apagado e piscar rapidamente somente quando um caracter for recebido pela UART.

# Utilizando a Serial na RaspBerry

Para um teste rápido utilizaremos um software para comunicação serial denominado Minicom. Este software permite abrir uma serial, configurar os devidos parâmetros e enviar caracteres digitados pelo usuário ou mesmo arquivos seguindo um determinado protocolo. Para utilizar o Minicom execute na Raspberry:

1) Instalar o Minicom.

\$ sudo apt-get install minicom

2) Execute o Minicom abrindo a serial desejada. O Linux nomeia a serial como ttyAMA0. Portanto, deve-se executar o seguinte comando:

### \$ minicom -b 9600 -D /dev/ttyAMA0

O parâmetro -b indica o baudrate (velocidade) da serial. Observe que ambos, microcontrolador e Raspberry, devem estar configurados para funcionar na mesma velocidade. Todos os caracteres digitados no terminal do Minicom apareceram de volta na console. Ainda, para ter certeza que está funcionando, envie um caractere e observe o LED piscar.

Para sair do Minicom pressione:

### CTRL + A, em seguida pressione q.

### Implementar um aplicativo em C para comunicação com o Microcontrolador.

Agora, implemente um aplicativo para comunicação serial em linguagem C a ser executado no Arduíno o qual interface com o aplicativo Minicom em execução no Raspberry pi. O aplicativo a ser implementado permitir ligar o LED2 sempre que o caracter 'L' for recebido no arduíno. O mesmo LED2 deve ser desligado sempre que o caracter 'D' for recebido via interface serial do arduíno. Utilize o código fonte (serial.c) disponível no site da disciplina.