

# MASTERS 2016

The Premier Technical Training Conference for Embedded Control Engineers



## Desenvolvendo aplicações com Bluetooth® Low Energy RN4020

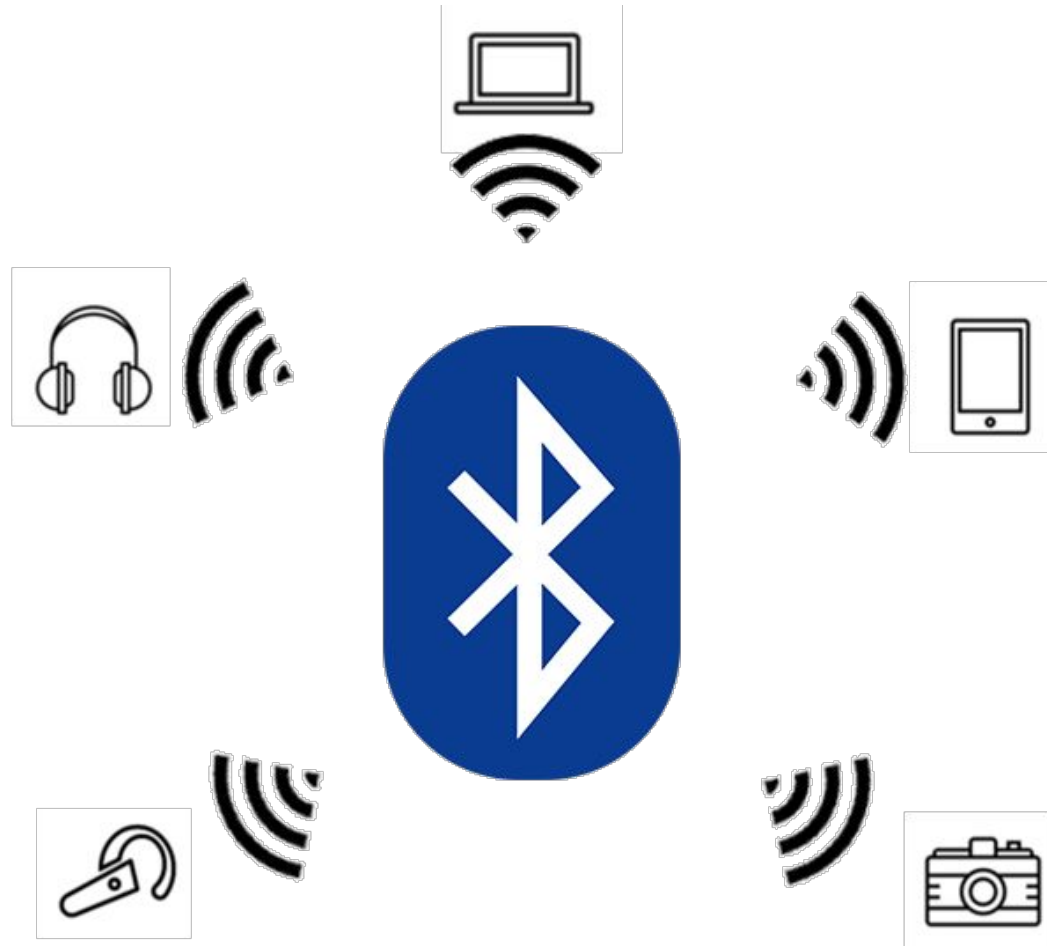
# Objetivos da Aula

- **Quando terminarmos esta aula, você poderá:**
- Explicar os fundamentos da tecnologia Bluetooth® Low Energy (BLE)
- Entender a capacidade do módulo RN4020 BLE e sua interface simples em comandos ASCII através da UART
- Usar o RN4020 para aplicações BLE

# Agenda

- **Fundamentos do Bluetooth® Low Energy**
- **Introdução do módulo Microchip RN4020 BLE**
- **Funcionalidades avançadas do RN4020**
- **Desenvolver aplicações BLE utilizando RN4020**

# O que é Bluetooth?



# Histórico do BT e BLE

- **Bluetooth clássico introduzido em 1994**
- **Crescimento explosivo depois da adoção em telefones celulares**
- **Bluetooth 2.x adicionou suporte à EDR**
- **Bluetooth 4.0 adicionou suporte à Bluetooth Low Energy**



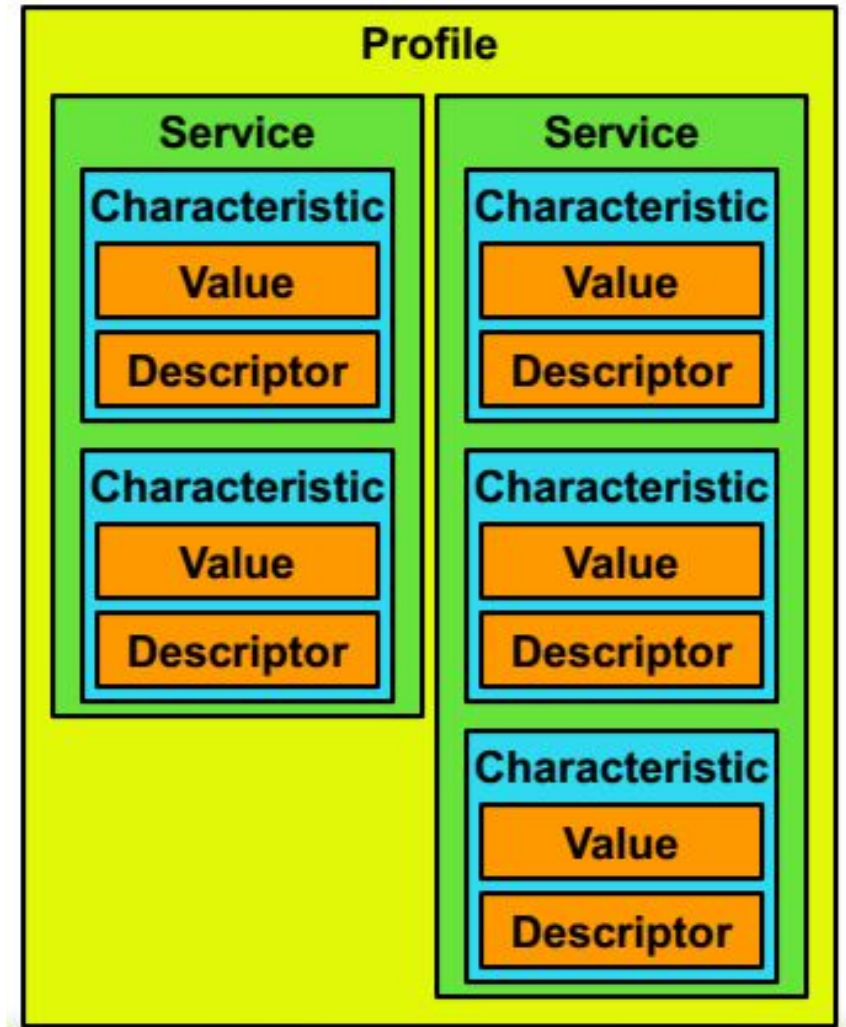
# BT Clássico x BLE

- **Buscam aplicações diferentes**
  - BT Classic: alta velocidade de comunicação e interoperabilidade
  - BLE: Baixo consumo de energia
- **Dois tipos de BLE:**
  - Bluetooth® Smart: Somente BLE
  - Bluetooth Smart Ready: Modo duplo
- **BLE é compatível com Apple.**



# Terminologias do BLE

- **GAP**
  - Central/Peripheral/  
Broadcaster/Observer
- **GATT**
  - Server / Client
- **Profiles**
- **Services**
- **Characteristics**
  - Read, Write, Notify
- **Public vs Private**





# Agenda

- Fundamentos do Bluetooth® Low Energy
- **Introdução do módulo Microchip RN4020 BLE**
- Funcionalidades avançadas do RN4020
- Desenvolver aplicações BLE utilizando RN4020



# Modulo ou Chip

- **Modulo:**

- Simples, facilidade de uso, Time-to-Market, Certificações, Sistema aberto, independente de recursos, facilidade de portabilidade

## **Chip:**

Baixo custo





# Módulo BLE MCHP



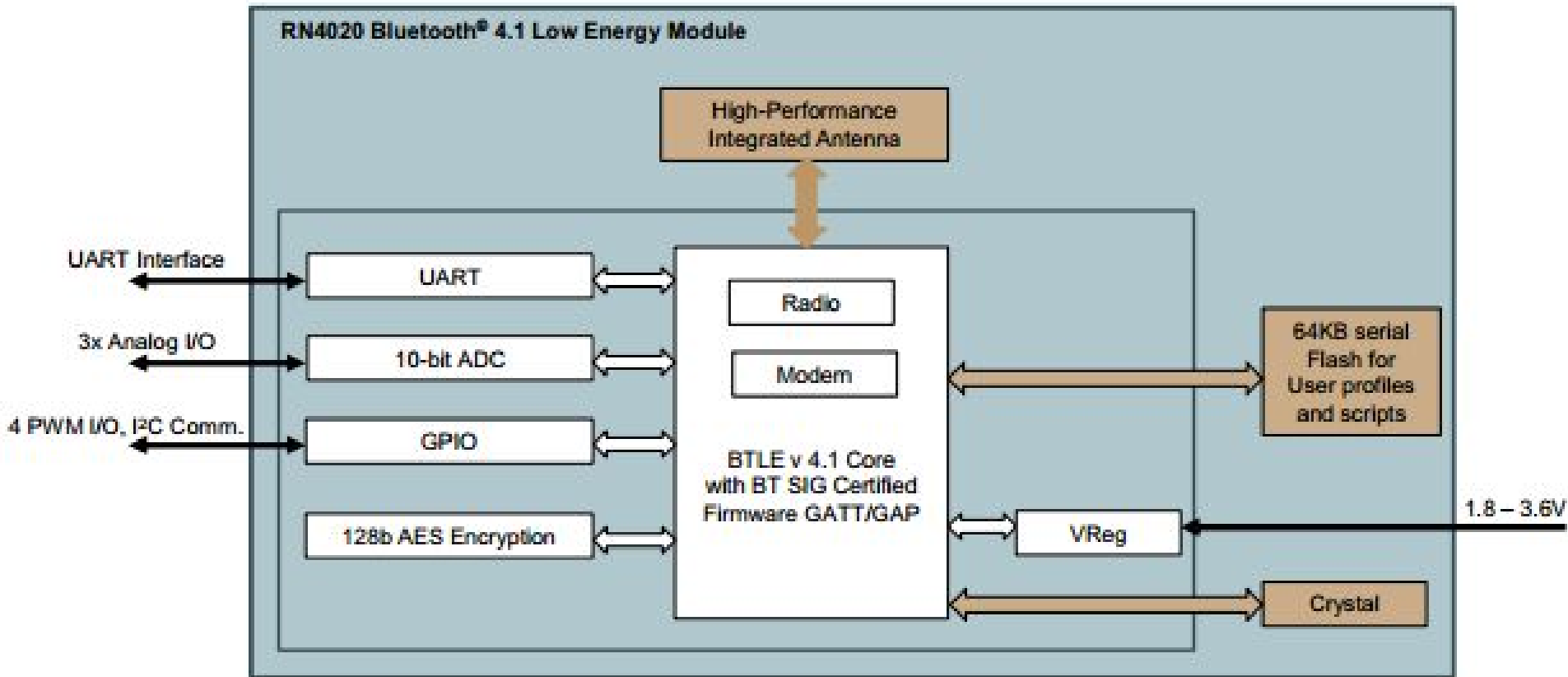
	RN41 / 41N	RN42 / RN42N	RN4020	BM70/71 RN4870/71	BM78 RN4678
Type	Class 1 Bluetooth 2.1	Class 2 Bluetooth 2.1	Bluetooth LE 4.1	Bluetooth LE 4.2	Bluetooth 4.2 Dual Mode (Classic & LE)
Interfaces	← UART / USB →		UART	UART	UART
Profiles/ Protocols	← SPP / HID / iAP / HCI →		GATT / Health, Fitness / Proximity, Etc. / Custom data	GAP / SDP / SPP / GATT	GAP / SDP / SPP / GATT/iAP2
Power	3.0-3.6V	3.0-3.6V	1.8-3.6V	1.9-3.6V	3.3-4.2V
Antenna	Ceramic chip Antenna Or no Antenna	PCB Antenna or no Antenna	PCB	Ceramic chip Antenna Or RF pad for external	Ceramic chip Antenna Or RF pad for external
Size (mm)	13.4 x 25.8 x 2.0	RN42: 13.4x25.8x2.4 RN42N: 13.4x20.5x2.4	11.5 x 19.5 x 2.5	BM70: Shield/Unshield 22 x 12x2.4/ 15x12x1.6 BM71:Shield/Unshield 11.5x9x2.1/6x8x1.5	Shielded: 22 x 12 x 2.4 Unshielded: 15x12x1.8
Certification	FCC / IC/ CE / AUS / JP	FCC / IC/ CE / AUS / JP / Korea / Taiwan	FCC / IC/ CE / AUS / JP / Korea / Taiwan	FCC, IC, CE, NCC, KCC, MIC(Japan)	FCC, IC, CE, NCC, KCC, MIC(Japan)

# RN4020

- **Funcionalidade de Hardware do RN4020**
  - RF Certificado (FCC, CE, IC)
  - UART com baud rate configurável até 921 Kbps
  - 3 I/O analógicas (10-bit)
  - Digital I/Os
  - 100 dBm Link
  - 700nA Modo Dormant, 16mA Tx/Rx
  - PCB antenna



# RN4020 - Hardware



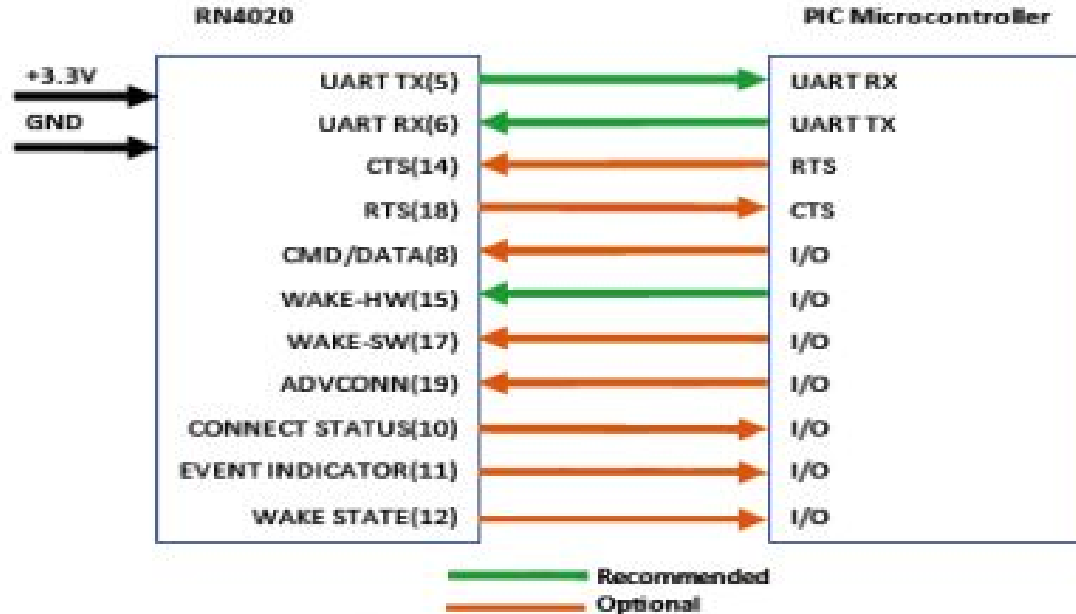
# RN4020

## ● Capacidades do Firmware

- Compatível com BT 4.1 Bluetooth® Smart
- Suporte imediato a 13 perfis e 17 serviços públicos.
- Central/Peripheral/Broadcaster/Observer (GAP)
- Server/Client (GATT)
- Suporte a serviços privados
- Perfil de dados Microchip LE
- Capacidade Hostless Scripting
- Comandos remotos
- Firmware Upgrade de device

# Interface de controle

- 3 entradas opcionais de controle
- 3 saídas opcionais de controle
- Interface de comandos ASCII via UART





# GATT (Service/Characteristics)

- **RN4020 Suporta 17 serviços públicos**
  - Healthcare
  - Sports & Fitness
  - Proximity Sense
  - Alerts
- **RN4020 Suporta serviço privado**
  - Até 10 características privadas auto definidas



# Serviço Privado

- **Usuário pode definir serviço/características no RN4020**
- **1 serviço privado e até 10 características privadas**
  - 1 serviço privado é suficiente?
- **UUIDs Públicos e Privados**
  - 128bit vs. 16bit
  - Propriedades das características
  - Tamanhos dos dados das características
  - Características se referem como “Handle over the air”

# Serviços

- **Características dos endereços de comandos ASCII por seus UUID ou “Handle”**

**SUW**

Access **S**erver Service by **U**UID to **W**rite  
Characteristic Value

Example: **SUW,2A19,64**

**CHR**

Access **C**lient Service by **H**andle to  
**R**ead Characteristic Value

Example: **CHR,001E**

# Agenda

- Fundamentos do Bluetooth® Low Energy
- Introdução do módulo Microchip RN4020 BLE
- **Funcionalidades avançadas do RN4020**
- Desenvolver aplicações BLE utilizando RN4020

# Funcionalidades avançadas do RN4020

- **Microchip Low Energy Data Protocol (MLDP)**
- **Capacidade de Scripting em Standalone**
- **Funcionalidade para comandos remotos**
- **Firmware Update do device**

# Microchip Low Energy Data Protocol (MLDP)

- BLE não possui suporte a “Data Stream” (SPP)
- Microchip define serviços privados que simulam o SPP (MLDP)
- Este serviço privado é transparente ao usuário no módulo RN4020

# Requerimentos do MLDP

- **H/W Flow Control para garantir integridade dos dados**
- **Two Control I/O Pins:**
  - CMD/DATA Pin: MLDP Mode
  - MLDP\_EV Pin: Indica mensagem de
  - CMD
- **Até 20Kbps**

# Capacidade de Scripting em Standalone

- **RN4020 pode rodar aplicações simples pelo “scripting”**
  - Reduzindo custos no MCU principal
  - Aumento da vida de bateria
  - Facilidade de aprendizado – Operações realizadas em comandos AT
  - Tempo de desenvolvimento reduzido



# RN4020 Scripting

- **Acionados por evento**
  - 11 Eventos: @PW\_ON, @CONN, @DISCON, @TMR1/2/3, @GPIO/H, @ALERTH/L/O
  - Sob cada evento, um ou mais comandos ASCII, atribuições, operações lógicas
  - RN4020 roda todas as linhas de um Script até o final do script ou outro evento
  - Uma porta I/O pode ser associada a um “Server Handle

# RN4020 Scripting

## Duas Variáveis

As Variáveis podem ter valores atribuídos

**\$VAR1 = @I,0**

Pode ser definido um Range

**\$VAR1 < “0200”**

Pode ser usada nos comando ASCII

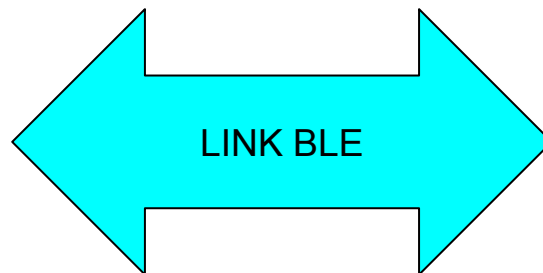
**SHW,001E,\$VAR1**

# Funcionalidade para comandos remotos

- **Construído sobre o MLDP**
- **Executa comandos AT em dispositivo remoto**
- **Resultados das execuções são enviados de volta para o device local**

# Funcionalidade para comandos remotos

- **Outro caminho para um periférico sem Host**
  - Típicamente a MCU host controla os módulos via UART
  - Comandos remotos podem controlar o dispositivo usando um link BLE
  - Toda aplicação lógica pode ser realizada num dispositivo local
  - Sem desenvolvimento no dispositivo remoto



# Firmware Update do device

- **O RN4020 pode se atualizar em campo**
- Pela UART
- Over The Air (OTA)
- **DFU: Detecção de erros e recuperação**

# Firmware Update do device

- **DFU pela UART**
  - Hardware Flow Control
  - 15 segundos (115Kbps UART)
- **OTA DFU**
  - Baseado no MLDP
  - Local Hardware Flow Control precisa estar habilitado
  - Mínimo de 30 segundos para Upgrade (depende muito da conexão)

# Agenda

- Fundamentos do Bluetooth® Low Energy
- Introdução do módulo Microchip RN4020 BLE
- Funcionalidades avançadas do RN4020
- **Desenvolver aplicações BLE utilizando RN4020 com a placa Curiosity**



# MASTERS 2016

The Premier Technical Training Conference for Embedded Control Engineers



## Laboratórios

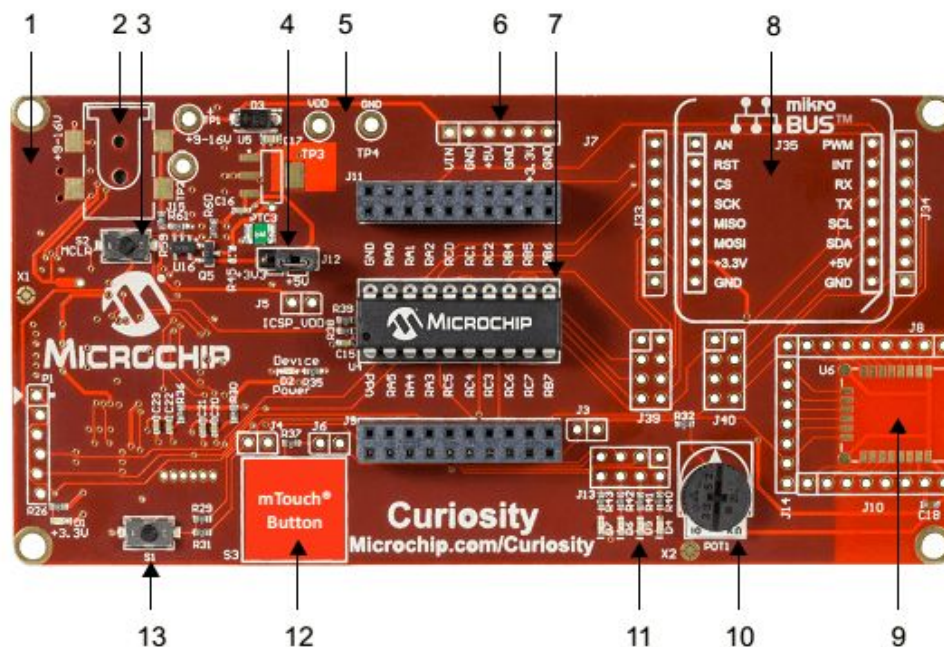
# Objetivos

- **Desenvolver aplicações com a placa Curiosity e a BLE2 Click**
- **Entender como fazer a interface do RN4020 com um microcontrolador**
- **Desenvolver aplicações para interface com smartphone**



# Curiosity

1. Conector USB mini-B (parte traseira);
2. Pads para conector para fonte 9V;
3. Botão para Master Clear Reset;
4. Jumper para seleção de fonte 3,3/5V p (J12);
5. Pads para fonte externa;
6. Conectores de expansão;
7. Soquete DIP para microcontrolador de 8, 14 e 20 pinos;
8. Conectores para padrão mikroBUS™;
9. Pads para módulo Bluetooth RN4020;
10. Potenciômetro;
11. LEDs;
12. Botão mTouch®;
13. Botão do tipo Push button.



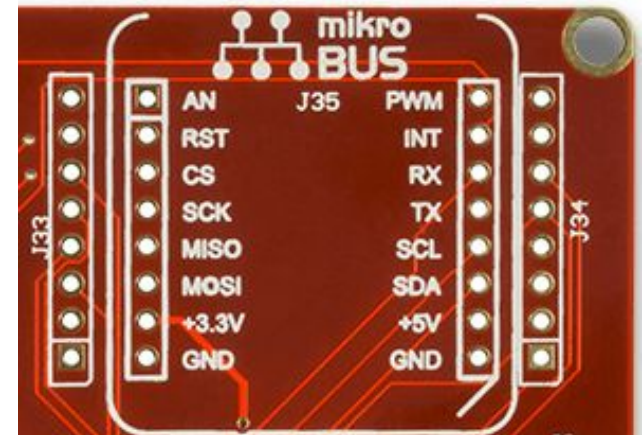
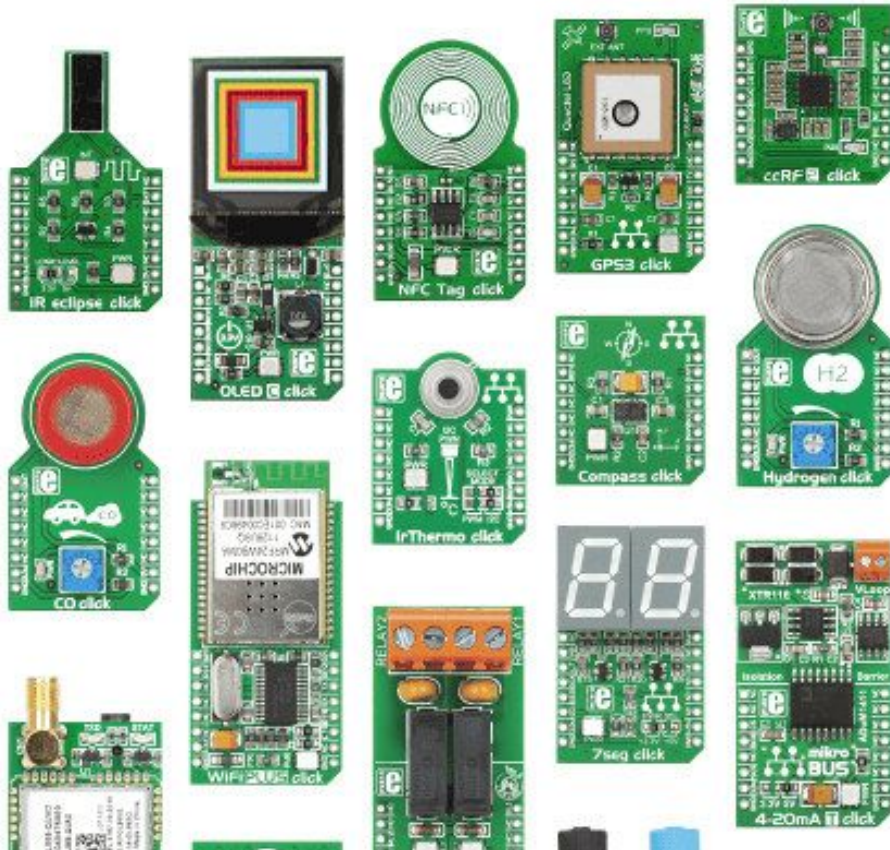
# PIC16F1619

Parameter Name	Value
Program Memory Type	Flash
Program Memory (KB)	14
CPU Speed (MIPS)	8
RAM Bytes	1,024
Digital Communication Peripherals	1-UART, 1-SPI, 1- I2C
Capture/Compare/PWM Peripherals	2 CCP
Timers	4 x 8-bit, 3 x 16- bit
Comparators	2
Temperature Range (C)	-40 to 125
Operating Voltage Range (V)	1.8 to 5.5
Pin Count	20
XLP	Yes



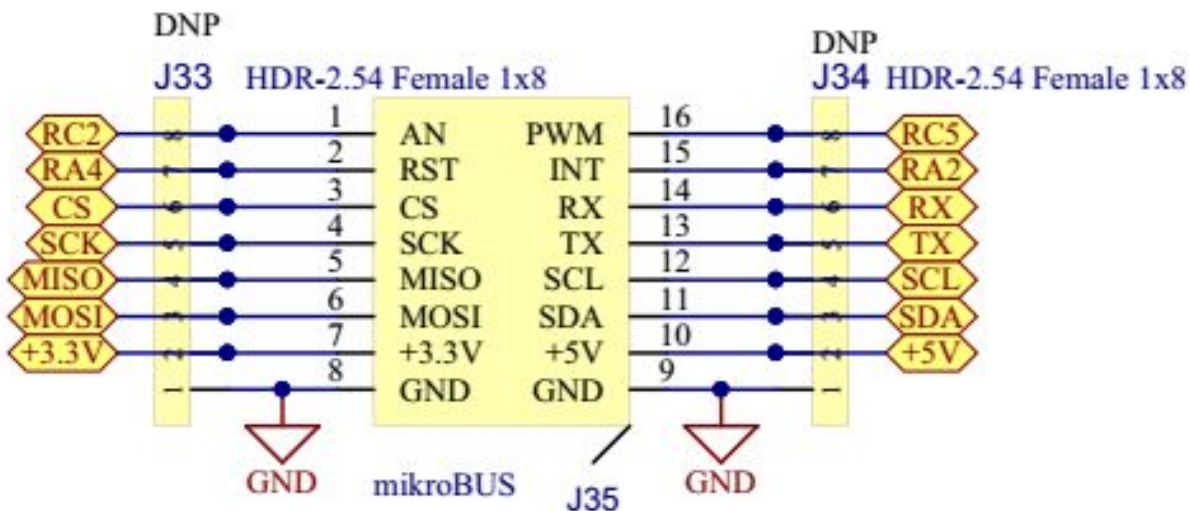


# Padrão mikroBUS™





# Padrão mikroBUS™

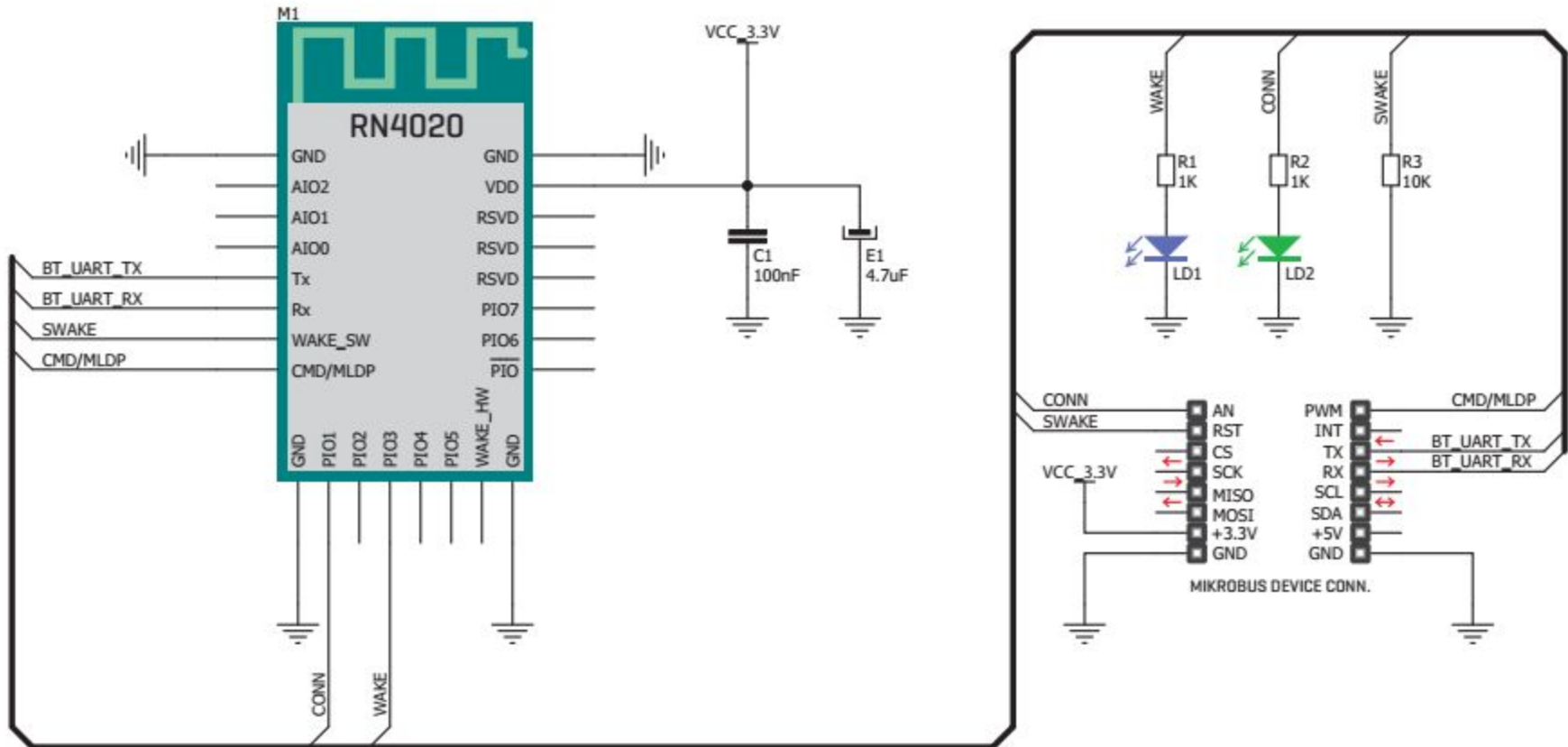








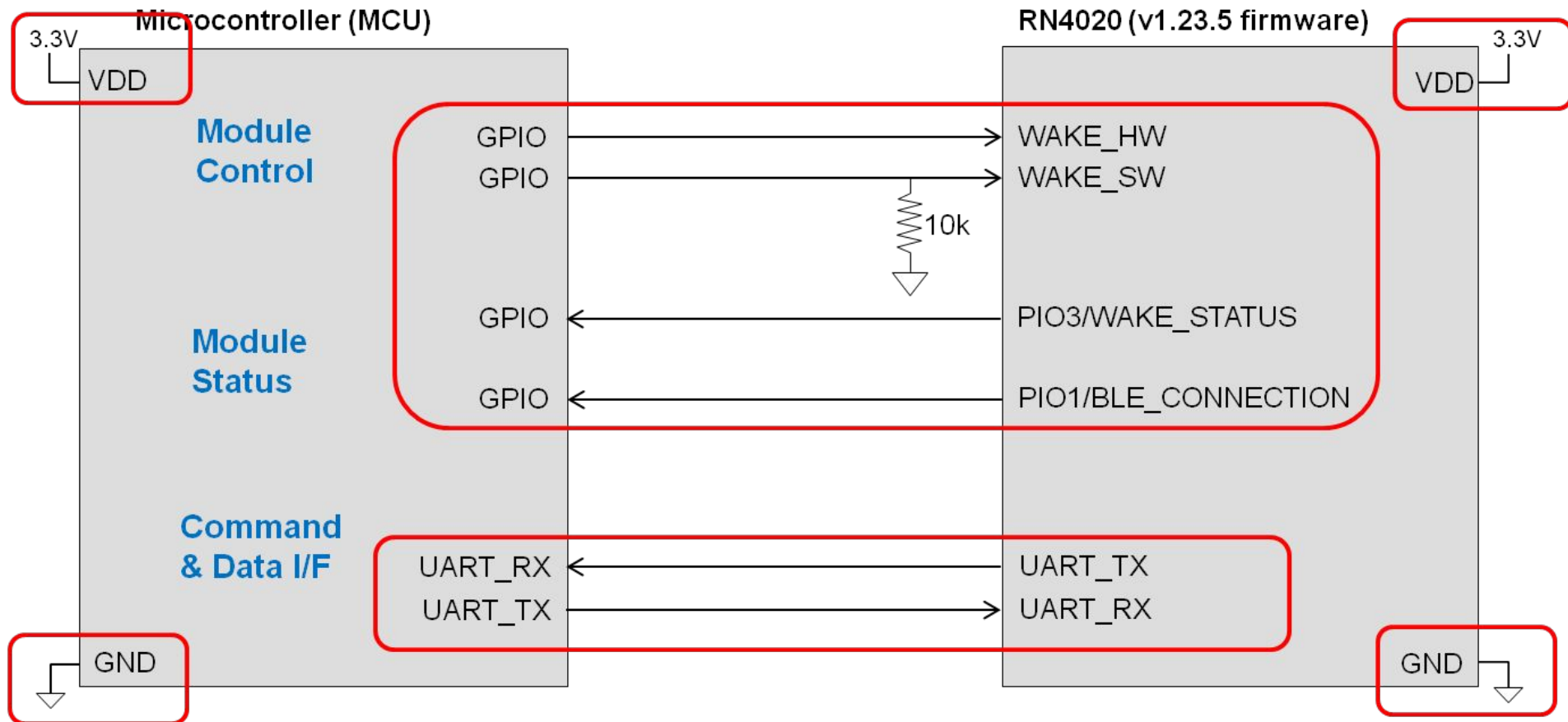
# BLE2 Click - Esquemático



# Ferramentas de software



# Interface Comando





# Pinos de Controle

7	WAKE_SW	Deep Sleep Wake; active-high to wake module from Deep Sleep.	Input; weak pull-down
8	CMD/MLDP	CMD – Command Mode – Module enters Command mode where UART commands and responses sent over UART are exchanged between the RN4020 command interpreter and the MCU host. MLDP Mode – Data Mode – Data through UART is sent over the Bluetooth Low Energy connection to the remote device using MLDP data service.	Input; active-high to enter Command
9	GND	Ground.	Ground
10	CONNECTION_LED SCK PIO[1]	Default state is output: Active-high indicates the module is connected to a remote device. Active-low indicates a disconnected state. Configurable as PIO[1] via software command. SCK for diagnostics and factory calibration if pin 17 is asserted.	<ul style="list-style-type: none"><li>• Green LED</li><li>• PIO[1]</li><li>• SCK</li></ul>

# Configuração da UART

Parameter	Value
Baud Rate	115200
Data Bits	8
Parity	None
Stop Bits	1
Flow Control	None

# **Lab 1:**

# **Criando estrutura do projeto**

# Lab 1 Objetivos

- **Desenvolver aplicação no MPLAB X**
- **Configurar o microcontrolador usando o MCC**
- **Prepar estrutura para as aplicações com RN4020**

# Lab 1 Resumo

- **Nesse Lab nós:**
  - Criamos um projeto com o MPLAB X para a placa Curiosity
  - Configuramos os periféricos com o MCC
  - Preparamos a estrutura para envio de comandos para o RN4020



# **Lab 2:**

# **Primeiros comandos para o**

# **RN4020**

# Lab 2 Objetivos

- **Entender como iniciar o RN4020**
- **Enviar comandos de configurações**
- **Fazer a leitura das informações do divice através do RN4020**

# Lab 2 Resumo

- **Nesse Lab nós:**
  - Enviamos comandos de configuração para o RN4020
  - Habilitamos o serviço de informações do dispositivo
  - Fizemos a leitura através de aplicativo no celular

# Lab 3: Battery Service

# Lab 3 Objetivos

- **habilitar o Battery Service**
- **Enviar status da bateria para o smartphone**
- **Fazer a leitura dos valores no smartphone**

# Lab 3 Resumo

- **Nesse Lab nós:**
  - Entendemos como habilitar o serviço de bateria
  - Configuramos o conversor AD usando o MCC
  - Simulamos o envio de valores
  - Fizemos a leitura através de aplicativo no celular

# Lab 4: Heart Rate Service

# Lab 4 Objetivos

- **Habilitar o Herat Rate Service**
- **Enviar status valores de Herat Rate para o smartphone**
- **Fazer a leitura dos valores no smartphone**



# Lab 3 Resumo

- **Nesse Lab nós:**
  - Entendemos como habilitar o serviço Herat Rate
  - Simulamos o envio de valores
  - Fizemos a leitura através de aplicativo no celular

# Lab 5: Health Thermometer Service

# Lab 5 Objetivos

- **Habilitar o Health Thermometer Service**
- **Enviar status valores de temperatura para o smartphone**
- **Fazer a leitura dos valores no smartphone**

# Lab 3 Resumo

- **Nesse Lab nós:**
  - Entendemos como habilitar o serviço Health Thermometer
  - Simulamos o envio de valores
  - Fizemos a leitura através de aplicativo no celular

# Resumo

- **BLE é uma tecnologia nova e promissora**
- **O RN4020 BLE OEM Module é simples e poderoso**
- **A interface com o microcontrolador é simples**
- **Rápido desenvolvimento de aplicações**

# Ferramentas usadas

- **DM164137 - Curiosity**
- **RN4020-V/RM123**
- **TMIK037 - BLE2 click**

# MASTERS 2016

The Premier Technical Training Conference for Embedded Control Engineers



## Obrigado!

## Contato

[fabio.souza@embarcados.com.br](mailto:fabio.souza@embarcados.com.br)

[www.embarcados.com.br](http://www.embarcados.com.br)