



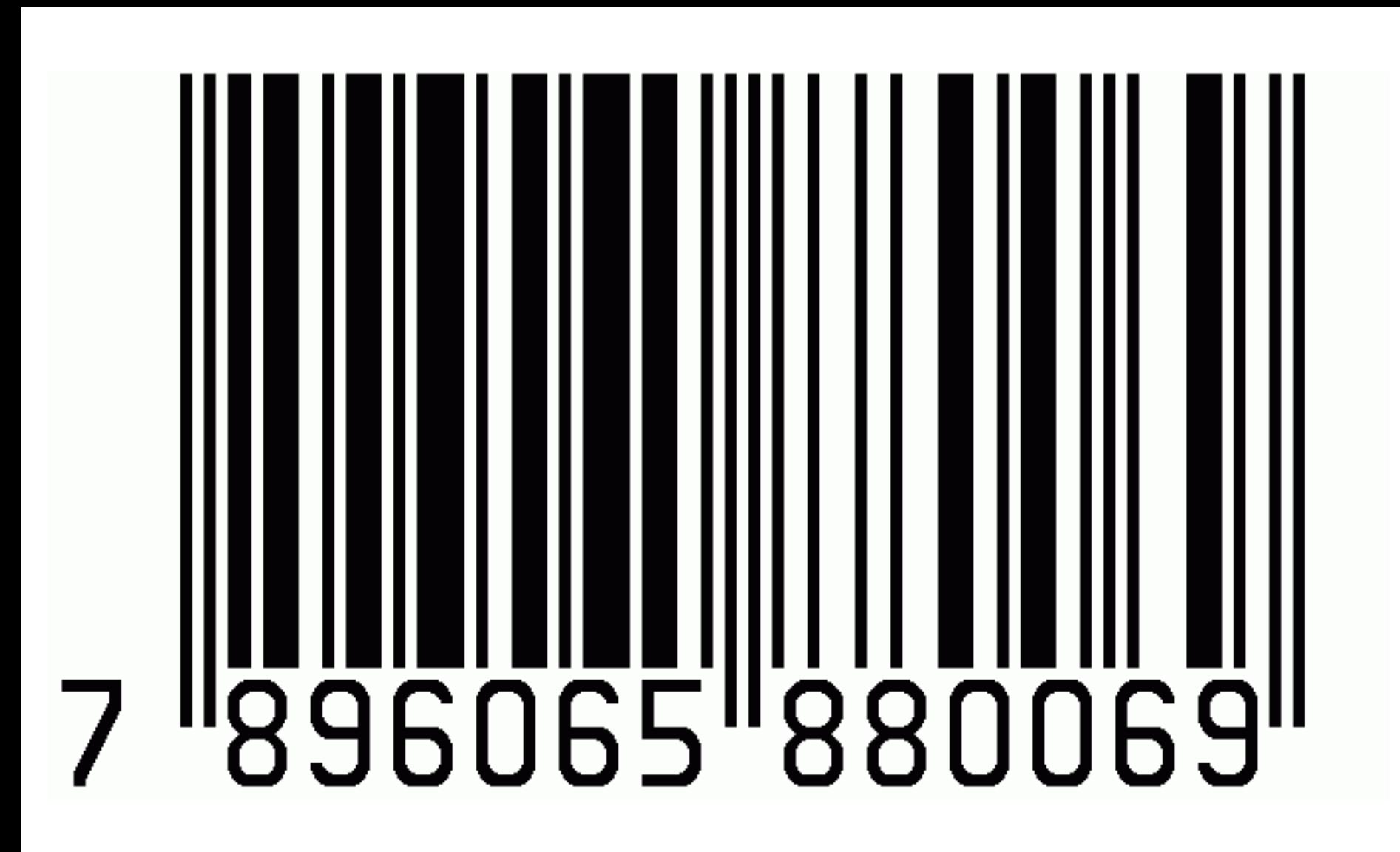
# Projeto 04

## Coisas à Venda – Teoria

Jan K. S. – [janks@puc-rio.br](mailto:janks@puc-rio.br)

ENG4051 – Projeto Internet das Coisas

Coisas



Código de Barras

**UPC-A**



**UPC-E**



**Code 128**



**EAN-13**



**EAN-8**



**Code 39**



**Interleaved 2 of 5**



**Codabar**



**PostNet**



Exemplos de Diferentes Tipos de Código de Barra

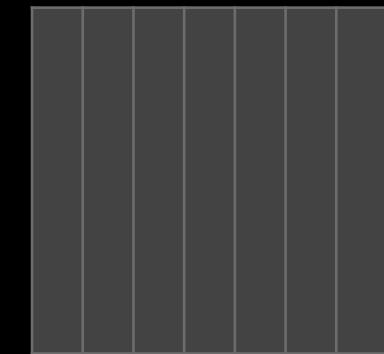
# 7896065880069

Primeiro dígito  
não é codificado

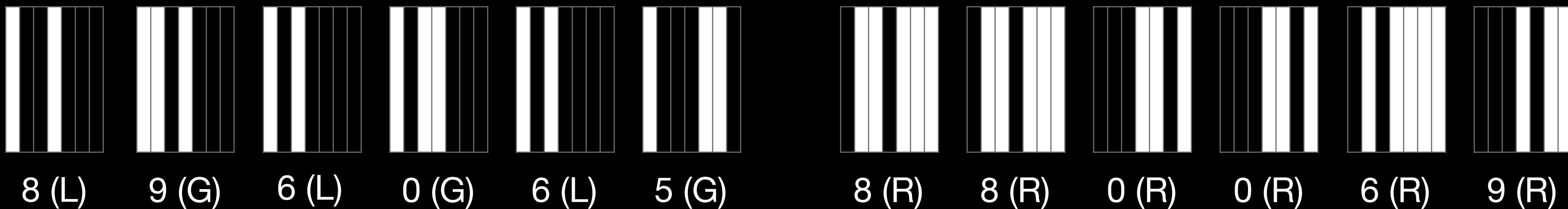
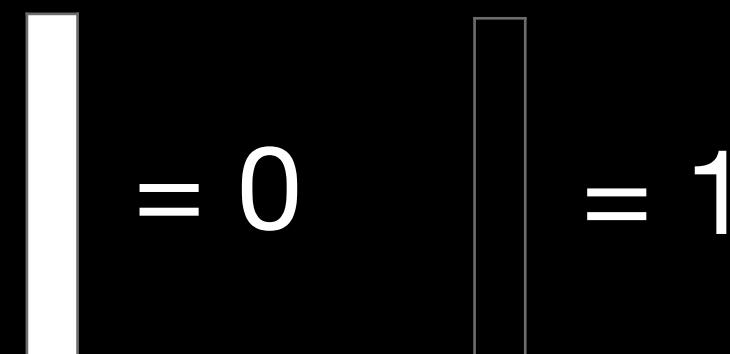


1º dígito	1º grupo de 6 dígitos	Último grupo de 6 dígitos
0	LLLLLL	RRRRRR
1	LLGLGG	RRRRRR
2	LLGGLG	RRRRRR
3	LLGGGL	RRRRRR
4	LGLLGG	RRRRRR
5	LGGLLG	RRRRRR
6	LGGGLL	RRRRRR
7	LGLGLG	RRRRRR
8	LGLGGL	RRRRRR
9	LGGLGL	RRRRRR

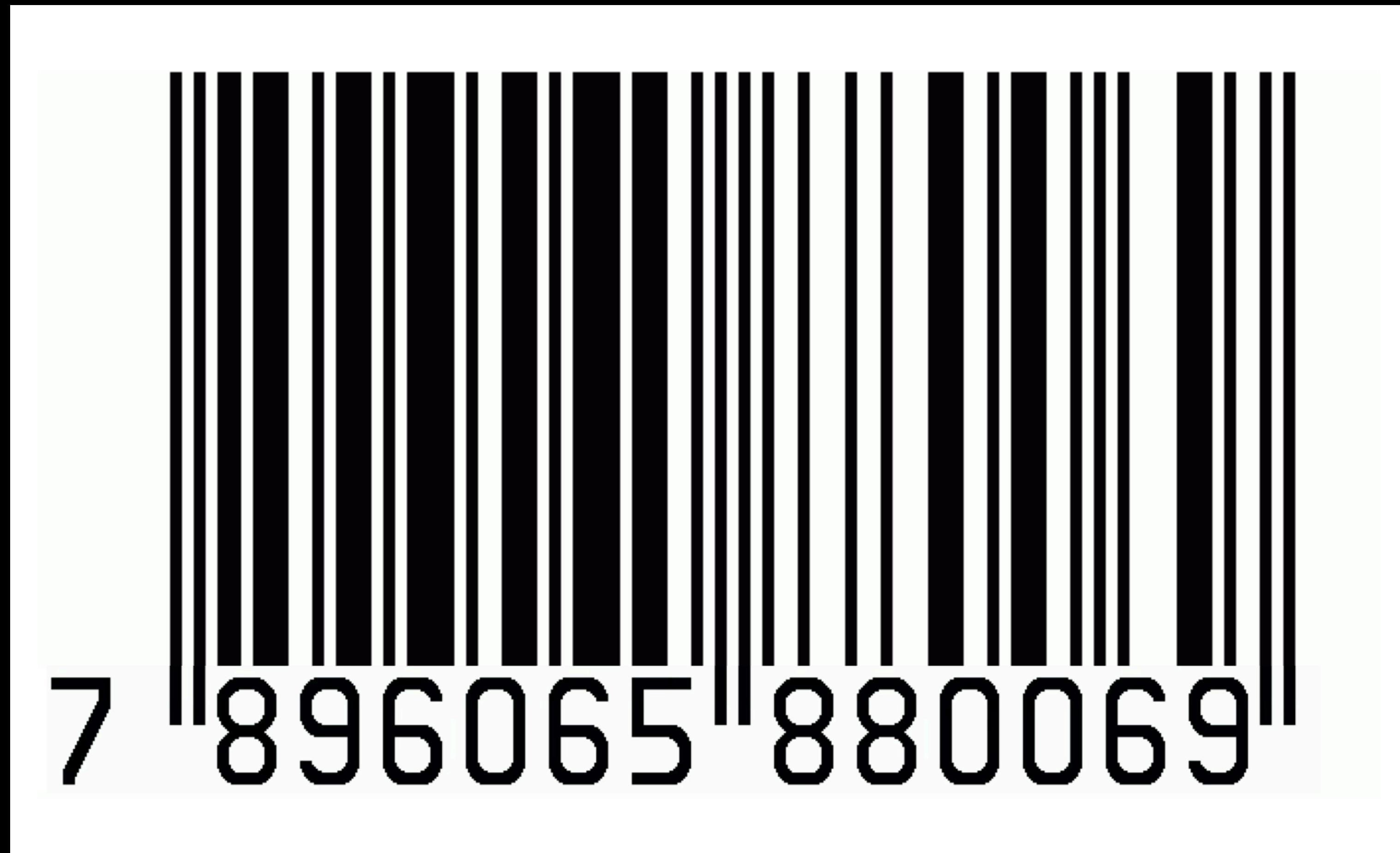
Dígito	L-code	G-code	R-code
0	0001101	0100111	1110010
1	0011001	0110011	1100110
2	0010011	0011011	1101100
3	0111101	0100001	1000010
4	0100011	0011101	1011100
5	0110001	0111001	1001110
6	0101111	0000101	1010000
7	0111011	0010001	1000100
8	0110111	0001001	1001000
9	0001011	0010111	1110100



7 bits por dígito



Codificação dos Dígitos em Barras no Padrão EAN-13



código do país  
(Brasil)

código do fabricante  
+ código do produto

dígito  
verificador

Resultado da Codificação

Não tem biblioteca  
para desenhar código  
de barras...

Vou ter que fazer uma  
biblioteca então...



The screenshot shows the Arduino IDE Library Manager window. The title bar reads "sketch\_apr2a | Arduino IDE 2.3.4". The toolbar includes icons for checkmark, arrow, and settings. The port dropdown shows "Arduino Mega or Mega 2...". The main area is titled "LIBRARY MANAGER" with a search bar containing "barcode". Filter options "Type: All" and "Topic: All" are present. The results list two items:

- AbleTP by Michael Hammel**: A thermal printer library for ArduinoBLE. Tested with PT-210 and ESP32. Prints custom text, ...  
Version: 1.0.1, **INSTALL** button
- SparkFun DE2120 2D Barcode Reader by...**: Serial Library for the DE2120 2D Barcode Reader An Arduino Library for the DE2120  
Version: 1.0.0, **INSTALL** button

The right panel shows the code for "sketch\_apr2a.ino":

```
1 void setup() {  
2     // put your setup code here,  
3 }  
4  
5 void loop() {  
6     // put your main code here, to  
7 }  
8  
9 }  
10
```

At the bottom, status text reads "Ln 10, Col 1 Arduino Mega or Mega 2560 on /dev/".

Biblioteca para Desenhar Código de Barras?

BarcodeGFX is an Arduino library for drawing barcode labels on displays. It is designed to be display-agnostic and works with any library that inherits from Adafruit\_GFX.



Example of barcodes drawn on different displays

Languages

- C++ 100.0%

sketch\_mar2a | Arduino IDE 2.3.4

Arduino Mega or Mega 2...

LIBRARY MANAGER

barcode

Type: All

Topic: All

BarcodeGFX by Jan K. S.  
<developer@janks.dev.br>  
1.0.1 installed  
Draw barcodes on displays compatible with Adafruit GFX. Supports EAN-13, EAN-8, UPC-A, and UPC-E barcodes....  
[More info](#)

void setup() {  
 // put your setup code here  
}  
  
void loop() {  
 // put your main code here  
}

Ln 1, Col 1 Arduino Mega or Mega 2560 on /dev/cu.usbserial-56B60028411 [not connected]

BarcodeGFX: Biblioteca para Desenhar Código de Barras

```

#include <GxEPD2_BW.h>
#include <BarcodeGFX.h>

GxEPD2_290_T94_V2 modeloTela(10, 14, 15, 16);
GxEPD2_BW<GxEPD2_290_T94_V2, GxEPD2_290_T94_V2::HEIGHT>
tela(modeloTela);
BarcodeGFX codigoBarras(tela);

void setup() {
    tela.init();
    tela.setRotation(3); // tela deitada (modo paisagem)
    tela.fillScreen(GxEPD_WHITE);
    tela.display(true);
}

                // em alguma função do código...
codigoBarras.setScale(2); // valor inteiro de 1 a 20
codigoBarras.draw("7896065880069", 0, 60, 65); // x, y, altura
tela.display(true);

```



Exemplo de Código de Barras no Display

```
// código como número em vez de String
long long codigo = 7896065880069; // ou int64_t
codigoBarras.draw(codigo, x, y, altura);

// suporte a outros formatos, como o EAN-8
codigoBarras.draw("42353720", x, y, altura);

// outras opções de personalização
codigoBarras.setScale(3)
    .setShowDigits(false)
    .setColors(TFT_YELLOW, TFT_BLUE); // shield TFT colorido

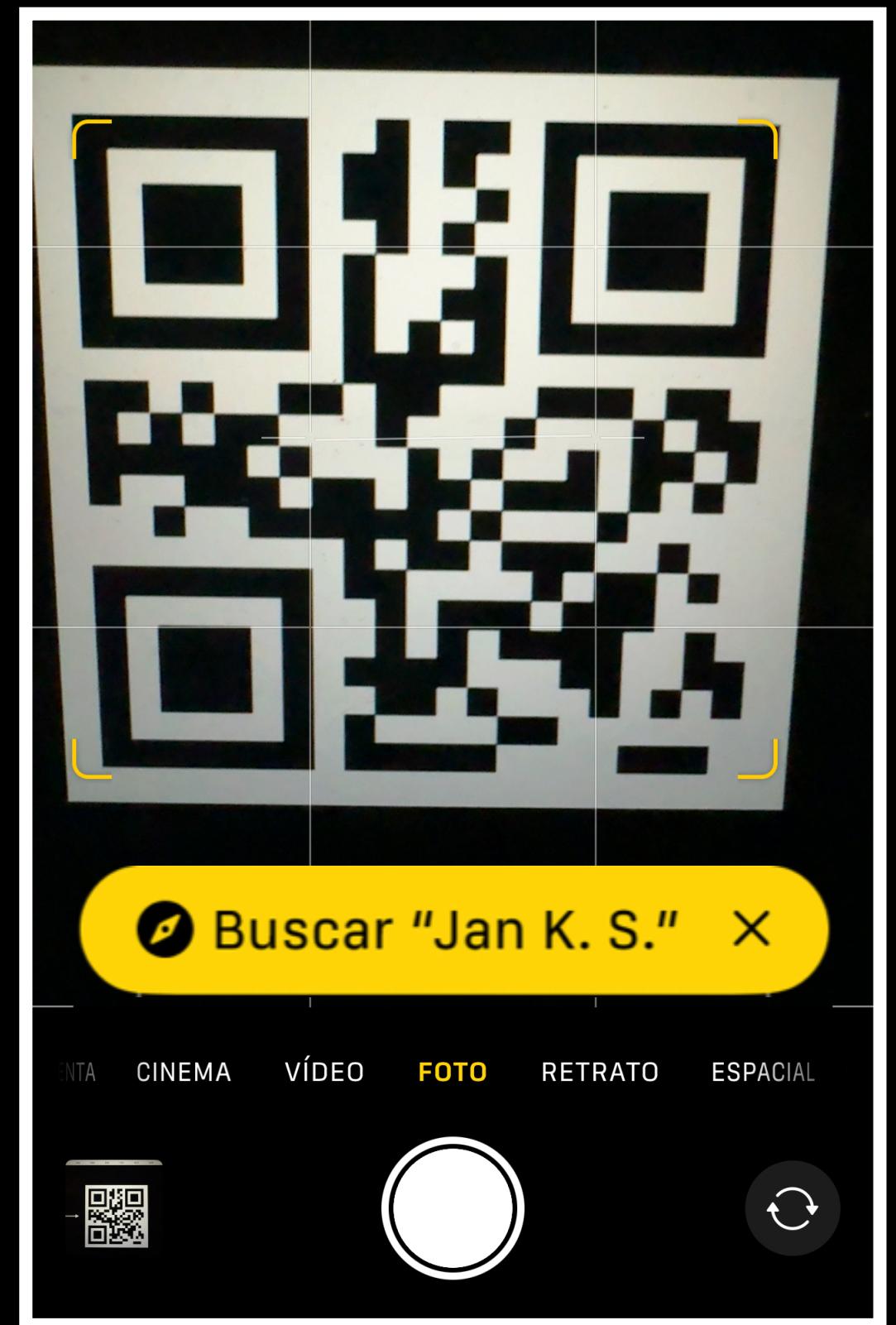
// centralizar na tela
String codigo = "0012345678905";
uint16_t comprimento = codigoBarras.getWidth(codigo);
int16_t xCentralizado = (tela.width() - comprimento) / 2;
codigoBarras.draw(codigo, xCentralizado, y, altura);

// validação do código de barras
bool sucesso = codigoBarras.draw("000000", x, y, altura);
if (!sucesso) {
    Serial.println("Código inválido! Nada foi desenhado!");
}
```



QR Code

Jan K. S. →



Codificação e Decodificação de Textos em QR Codes

## Mensagem

J → 01001010  
a → 01100001  
n → 01101110  
→ 00100000  
K → 01001011  
. → 00101110  
→ 00100000  
S → 01010011  
. → 00101110

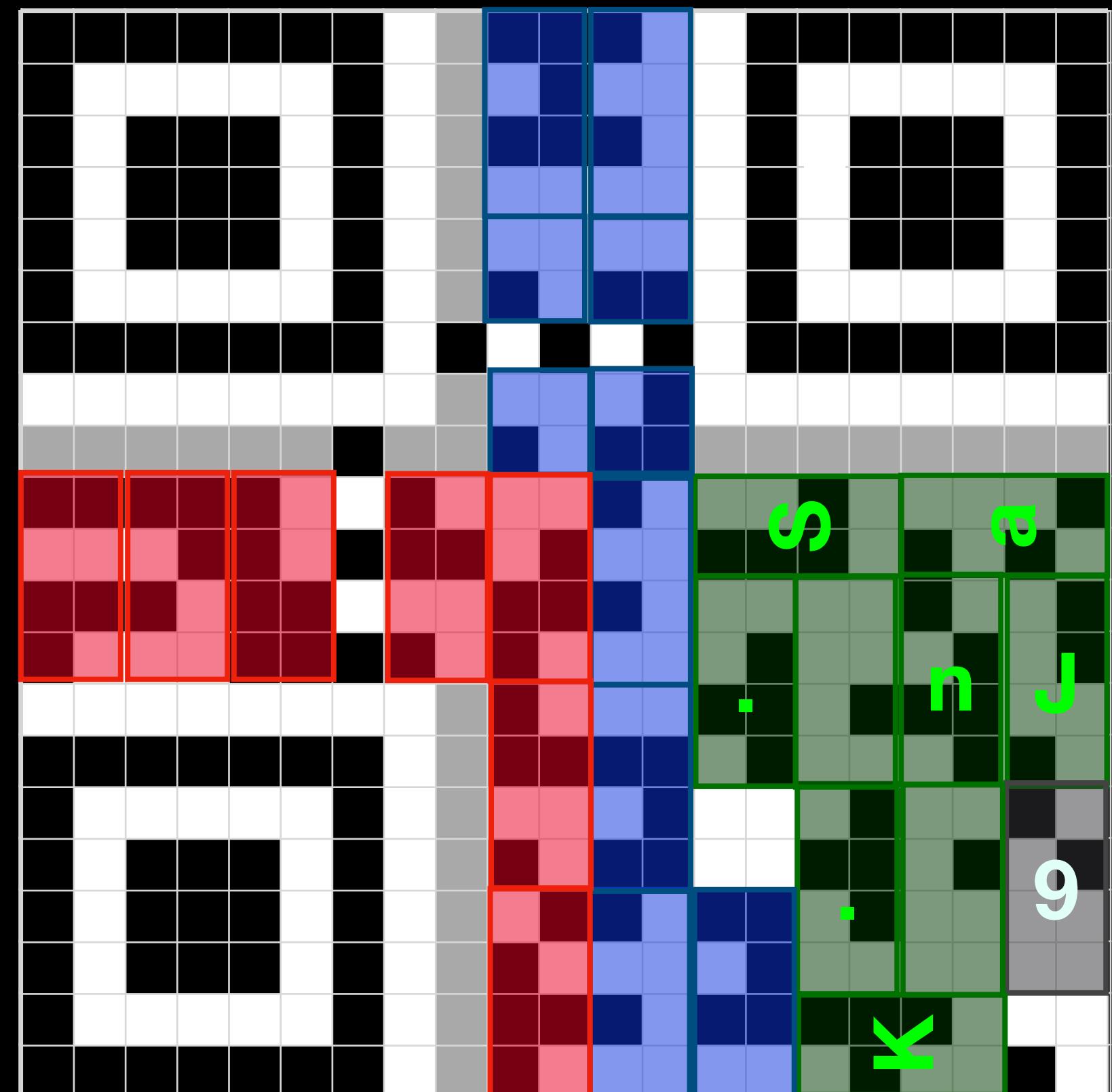
## Padrão de Preenchimento

11101100  
00010001  
11101100  
00010001  
11101100  
00010001  
11101100  
00010001  
11101100  
00010001

## Código de Correção de Erro (Nível Baixo)

00101101  
01110001  
10011101  
01001101  
01011111  
00011011  
11001101

■ bit 0 □ bit 1

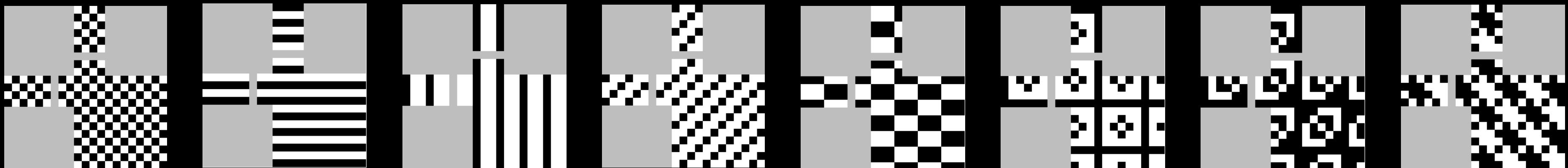
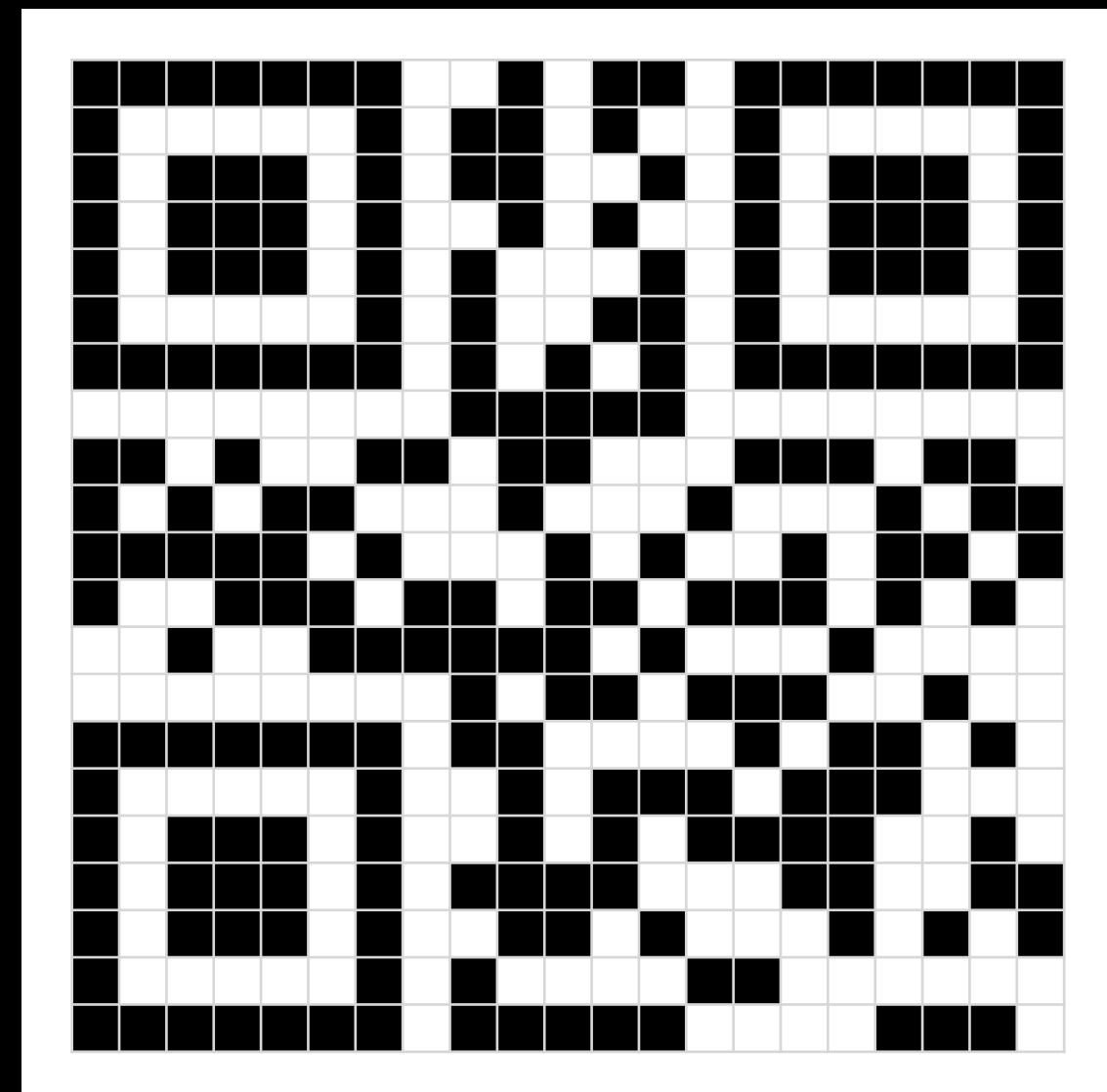
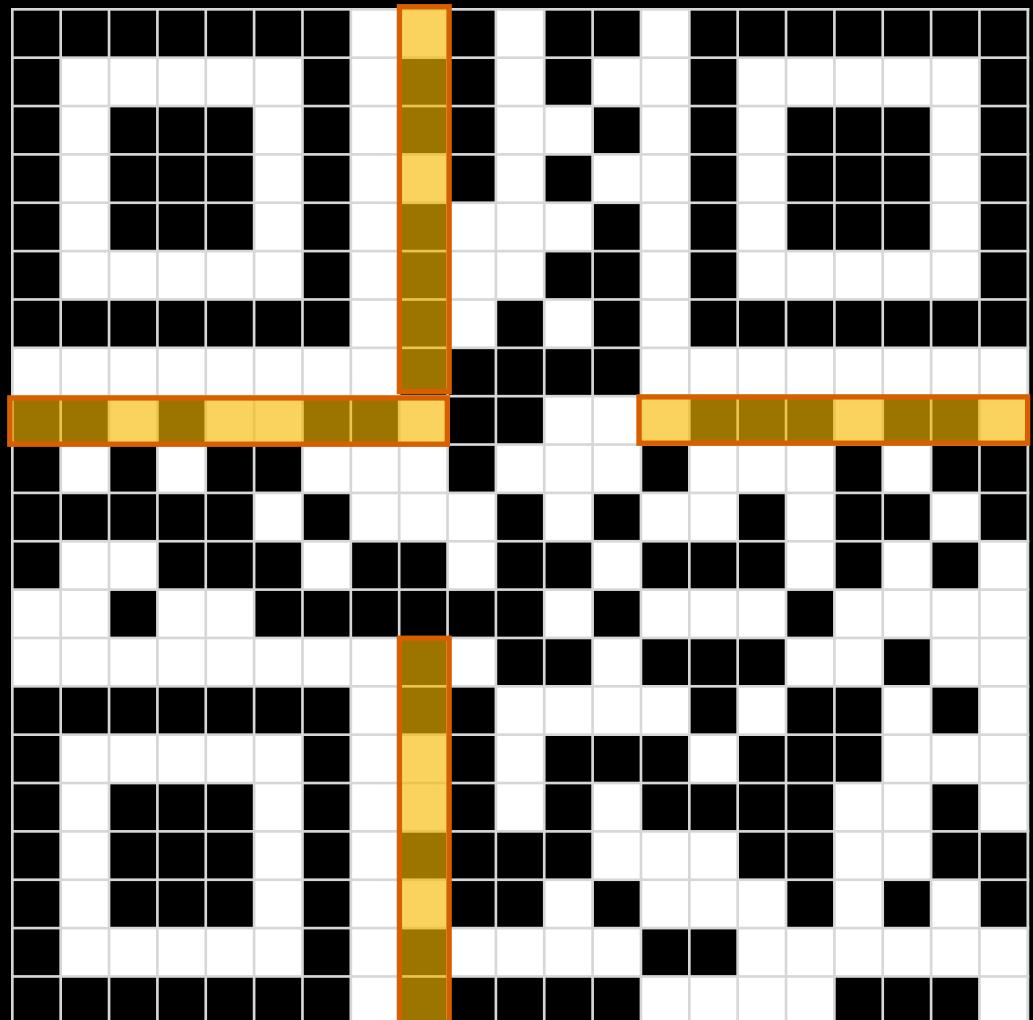
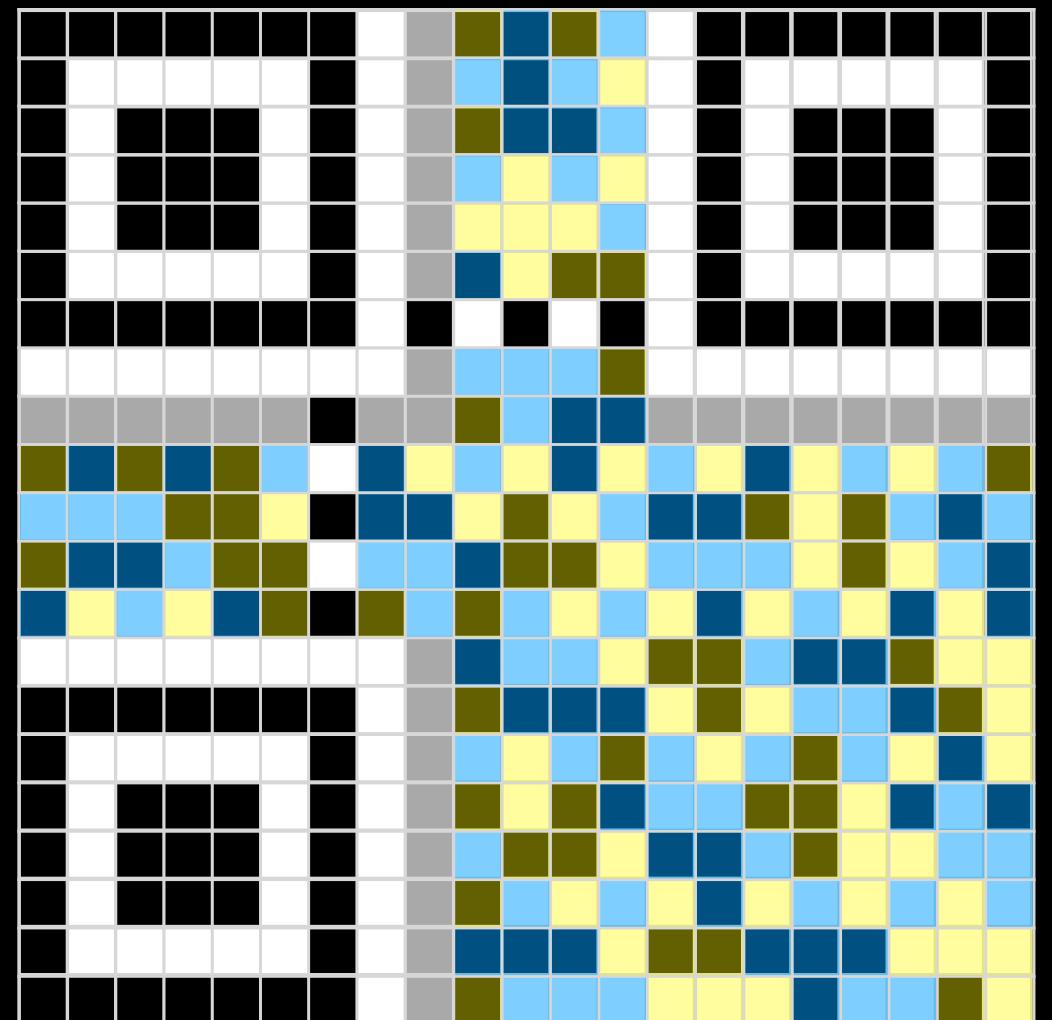


Codificação de Dados no QR Code

aplicação de  
máscara (XOR)

adição de bits de  
metadados

versão final com  
moldura branca

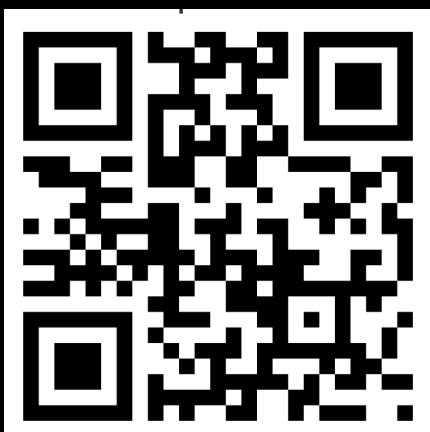


Aplicação da Máscara e Adição de Metadados

versão 1



versão 2



versão 3



versão 4



versão 5



...

correção de  
erro LOW



até 7%

correção de  
erro MEDIUM



até 15%

correção de  
erro QUARTILE



até 25%

correção de  
erro HIGH



até 30%

Versão (Tamanho) e Nível de Correção de Erro

Vou ter que fazer mais  
uma biblioteca...



sketch\_apr2a | Arduino IDE 2.3.4

Arduino Mega or Mega 2...

LIBRARY MANAGER

arcode

Type: All

Topic: All

**QRCode** by Richard Moore...  
A simple QR code generation library. A simple QR code generation library.  
[More info](#)

0.0.1

**ESP QRcode** by Jose Antonio Espinosa...  
ESP Generate QRCode for GDEH0213B72 eink display, SSD1306, SH1106 oled display...  
[More info](#)

sketch\_apr2a.ino

```
1 void setup() {  
2     // put your setup code here,  
3 }  
4  
5 void loop() {  
6     // put your main code here, t  
7 }  
8  
9 }  
10
```

Ln 10, Col 1 Arduino Mega or Mega 2560 on /dev/

Biblioteca para Desenhar QR Code?

github.com/wallysalami/QRCodeGFX

## QRCodeGFX

QRCodeGFX is an Arduino library for drawing QR codes on displays. It is designed to be display-agnostic and works with any library that inherits from Adafruit\_GFX.



Example of QR Codes drawn on different displays

### Languages

C 83.0% C++ 17.0%

LIBRARY MANAGER

qrcoodegfx

Type: All Topic: All

QRCodeGFX by Jan K. S.  
<developer@janks.dev.br>  
1.0.0 installed  
Draw QR codes on Adafruit\_GFX compatible displays. Use it with any library that subclasses Adafruit\_GFX, like GxEPD2...  
More info

1.0.0 REMOVE

sketch\_mar2a | Arduino IDE 2.3.4

Arduino Mega or Mega 2...

```
void setup() {  
    // put your setup code here  
}  
  
void loop() {  
    // put your main code here  
}
```

Ln 1, Col 1 Arduino Mega or Mega 2560 on /dev/cu.usbserial-56B60028411 [not connected]

QRCodeGFX: Biblioteca para Desenhar QR Code no Display

```
#include <GxEPD2_BW.h>
#include <QRCodeGFX.h>

GxEPD2_290_T94_V2 modeloTela(10, 14, 15, 16);
GxEPD2_BW<GxEPD2_290_T94_V2, GxEPD2_290_T94_V2::HEIGHT>
tela(modeloTela);
QRCodeGFX qrcode(tela);

void setup() {
    tela.init();
    tela.setRotation(3); // tela deitada (modo paisagem)
    tela.fillScreen(GxEPD_WHITE);
    tela.display(true);
}

        // em alguma função do código...

qrcode.setScale(2); // valor de escala inteiro ≥ 1
qrcode.draw("https://youtu.be/w5ebcowAJD8", 102, 5); // x, y

tela.display(true);
```

Exemplo de QR Code no Display



```
// Parâmetros da geração e de aparência do QR Code
qrcode.getGenerator()
    .setErrorCorrectionLevel(QRCodeECCLevel::High)
    .setVersion(5); // força versão específica

qrcode.setScale(3)
    .setColors(GxEPD_BLACK, GxEPD_WHITE) // cores invertidas
    .setRotation(QRCodeRotation::R90); // 0°, 90°, 180° ou 270°

// Validação e centralização
String link = "https://youtu.be/Vx5prDjKAcw";
bool sucesso = qrcode.generateData(link);
if (sucesso) {
    int16_t tamanho = qrcode.getSideLength();
    int16_t xCentralizado = (tela.width() - tamanho) / 2;
    int16_t yCentralizado = (tela.height() - tamanho) / 2;
    qrcode.draw(xCentralizado, yCentralizado);
}
else {
    Serial.println("Deu ruim!"); // talvez pouca memória RAM
}
```



QR Code para Pix

## Economia

# Com Pix, Brasil já é o segundo maior mercado de pagamentos instantâneos do mundo, mostra pesquisa

Alta das transações em tempo real no país foi de 78% no ano passado, quase o dobro do ritmo no mundo. Índia lidera ranking global, com quase metade das operações

Por Letycia Cardoso — Rio de Janeiro

30/04/2024 00h00 · Atualizado há 4 meses



Amplo Uso do Pix para Pagamentos



→ 000201 26360014BR.GOV.BCB.PIX011492958800000138  
52040000 5303986 540550.00 5802BR 5921SOS Rio Grande do Sul  
6012Porto Alegre 62070503\*\*\* 6304D0D9

Tipo de Dado	ID	Total de Caracteres	Dados			
			Tipo de Dado	ID	Total de Caracteres	Dado
versão do formato do código	00	02	01			
identificação do beneficiário	26	36	Tipo de Dado	ID	Total de Caracteres	Dado
			GUI	00	14	br.gov.bcb.pix
categoria do beneficiário	52	04	0000			
código da moeda (R\$, US\$, etc)	53	03	986			
valor da transferência	54	05	50.00			
código do país	58	02	BR			
nome do beneficiário	59	21	SOS Rio Grande do Sul			
cidade do beneficiário	60	12	Porto Alegre			
dados extras	62	07	Tipo de Dado	ID	Total de Caracteres	Dado
			id trans.	05	03	***
código verificador CRC	63	04	D0D9			

```
String inicio = "000201";
String prefixoPix = "0014BR.GOV.BCB.PIX";
String moeda = "5303986"; // R$
String pais = "5802BR"; // Brasil
String categoriaBeneficiario = "52040000"; // código MCC não definido
String dadosAdicionais = "62070503***"; // valor padrão

// AJUSTE OS DADOS ABAIXO PARA O SEU CASO!
// LEMBRE DE CONTAR E AJUSTAR O TOTAL DE CARACTERES (COR LILÁS)

String nomeBeneficiario = "5907JAN K S"; // máximo 25 letras
String cidadeBeneficiario = "6014RIO DE JANEIRO"; // máximo 15 letras
String chaveBeneficiario = "0114+5521910203040"; // ou email, CPF, etc

String chaveCompleta = prefixoPix + chaveBeneficiario;
String prefixoChave = "26" + String(chaveCompleta.length());

// código continua a seguir...
```

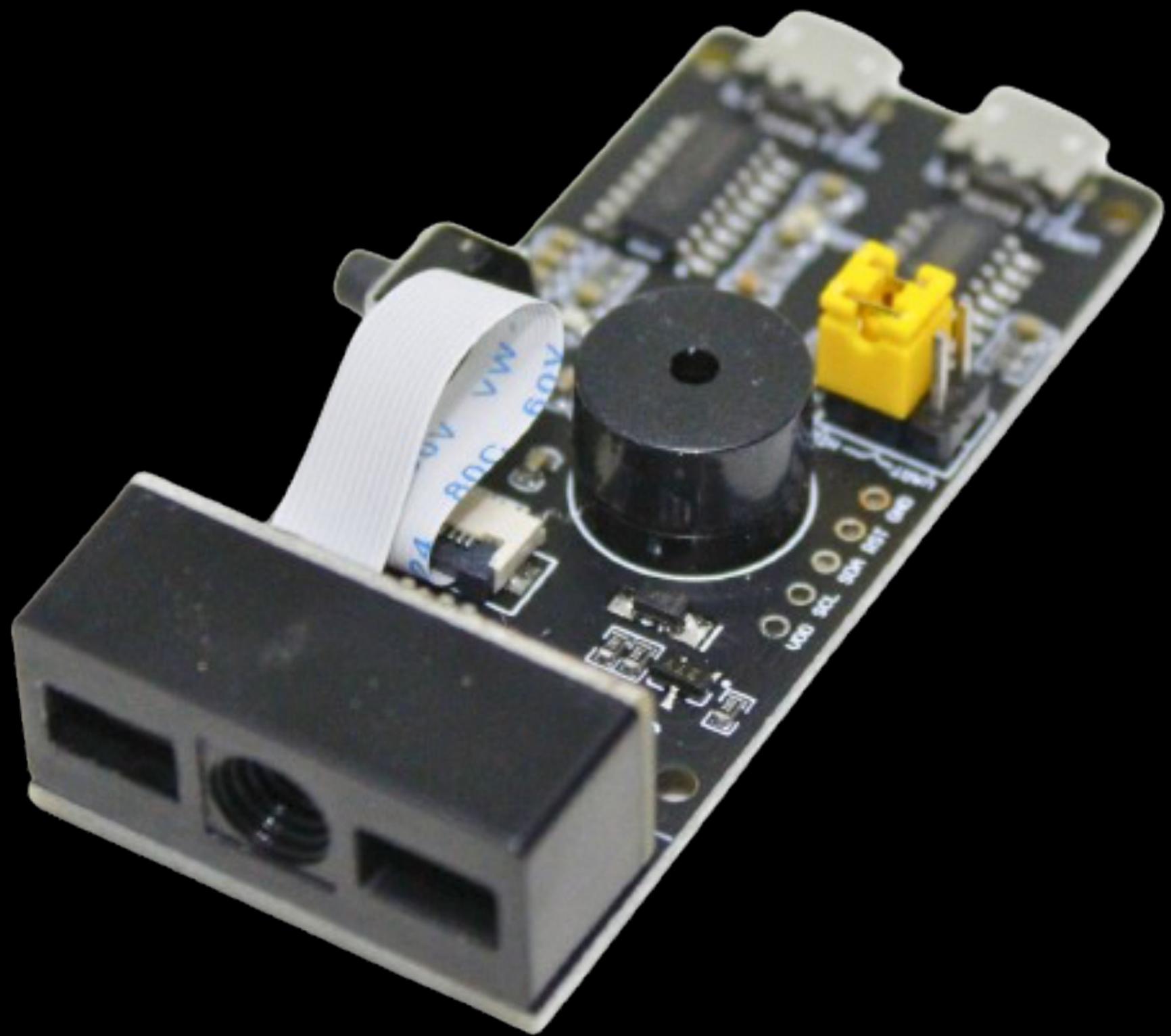
```
String adicionaCRC(String dados) {
    String dadosComPrefixoCRC = dados + "6304";
    const char *dadosChar = dadosComPrefixoCRC.c_str();
    uint16_t crc = 0xFFFF;
    while (*dadosChar) {
        crc ^= (uint8_t)(*dadosChar++) << 8;
        for (uint8_t i = 0; i < 8; i++) {
            if (crc & 0x8000){
                crc = (crc << 1) ^ 0x1021;
            }
            else{
                crc = crc << 1;
            }
        }
    }
    char codigoFinal[512];
    sprintf(
        codigoFinal, sizeof(codigoFinal),
        "%s%04X", dadosComPrefixoCRC.c_str(), crc
    );
    return String(codigoFinal);
}
// código continua a seguir...
```

Função para Adicionar Código Verificador de Erro (CRC)

```
// em alguma função do código...  
  
// ajuste o total de caracteres (COM 2 DÍGITOS) !  
String valor = "540512.00";  
  
String codigoPix = inicio + prefixoChave + chaveCompleta +  
categoriaBeneficiario + moeda + valor + pais +  
nomeBeneficiario + cidadeBeneficiario + dadosAdicionais;  
  
codigoPix = adicionaCRC(codigoPix);  
  
Serial.println("Código Pix: " + codigoPix);  
  
qrcode.setScale(3); // valor de escala inteiro  $\geq 1$   
qrcode.draw(codigoPix, 0, 0); // x, y  
tela.display(true);
```



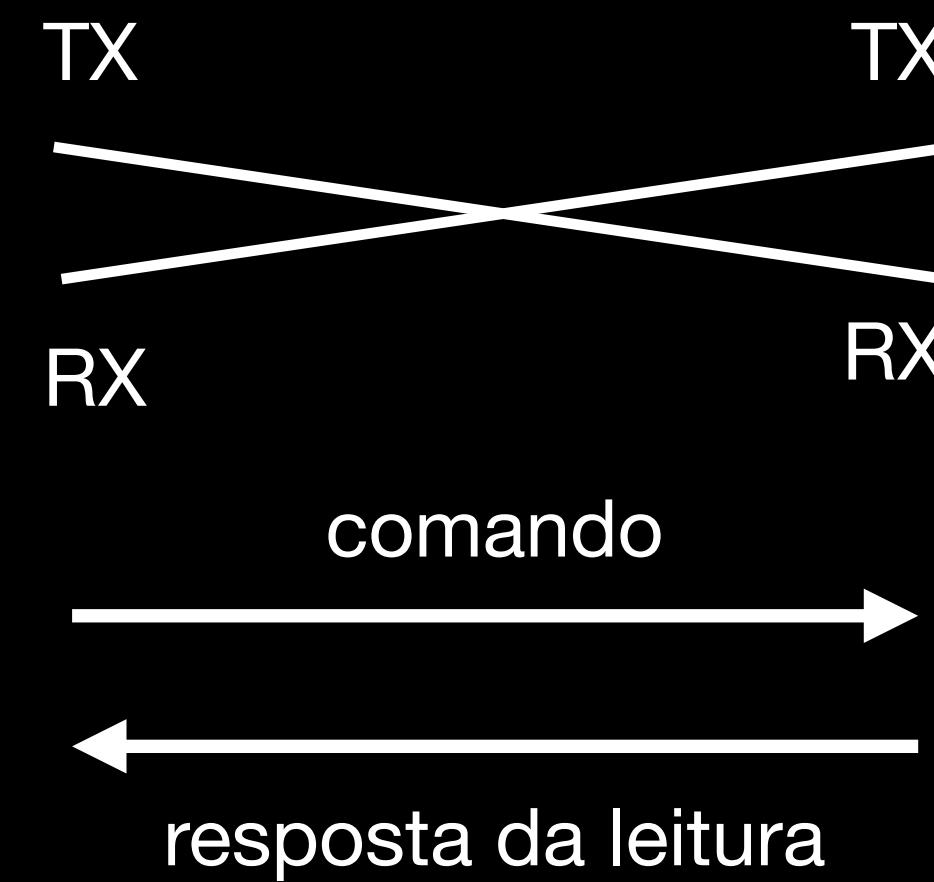
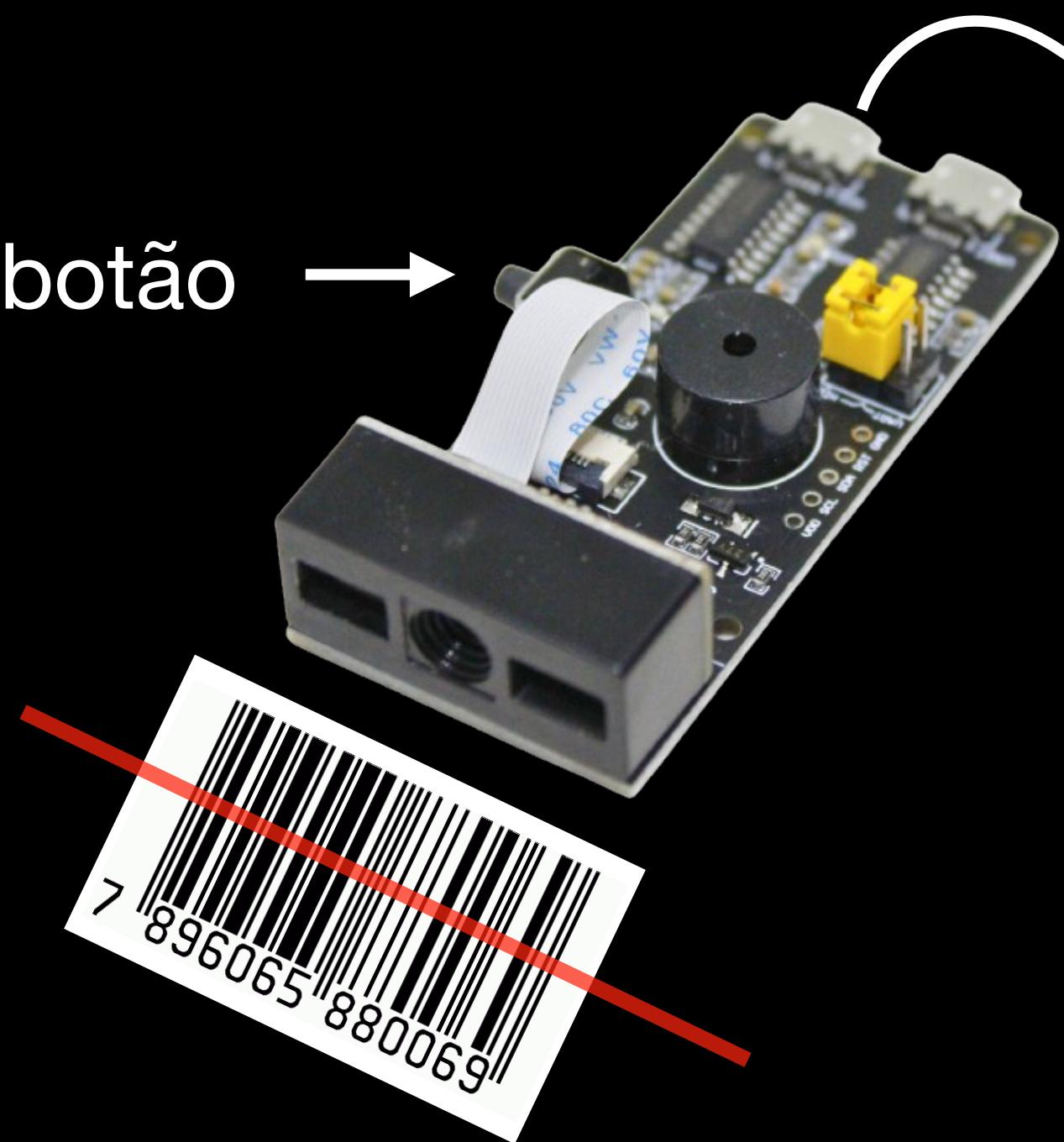
Geração do Código Pix



# Leitor de Código de Barra e QR Code (MH-ET Live Scanner 3.0)

aperta e segura o botão →

envia dígitos para  
o computador



Captura Manual x Automática de Código de Barras e QR Code



comando



MH-ET LIVE Scanner v3.0  
QR code barcode recognition module  
scan code module  
serial communication UART interface embedded

comando	ação
~M00910001.	habilitar comandos via serial (sempre chamar no começo!)
~M00210001.	escanear códigos continuamente
~M00210002.	escanear código uma vez até encontrar
~M00210003.	desligar escaner e aguardar pedido manual para escanear
~T.	pedir para escanear agora
~M00B0000A.	definir 1s de atraso para repetir leitura do mesmo código
~M00B00014.	definir 2s de atraso para repetir leitura do mesmo código
~M01030001.	ajustar brilho do flash para 25%
~M00FC0000.	desabilitar som ao receber comando de configuração
~M00FB0000.	desabilitar som ao ler código de barras / QR Code
consulte mais comandos em <a href="https://janks.link/manual-mh-et">https://janks.link/manual-mh-et</a>	

Exemplos de Comandos Serial para Configuração e para Leitura

```

void setup() {
    Serial.begin(115200); delay(500);
    Serial1.begin(9600, SERIAL_8N1, 47, 48); // pinos 47 e 48

    Serial1.println(~M0091001.); // habilita comandos serial
    delay(100);
    Serial1.println(~M0021001.); // escaneamento contínuo
    delay(100);
    Serial1.println(~M00B00014.); // atraso para ler mesmo produto
}

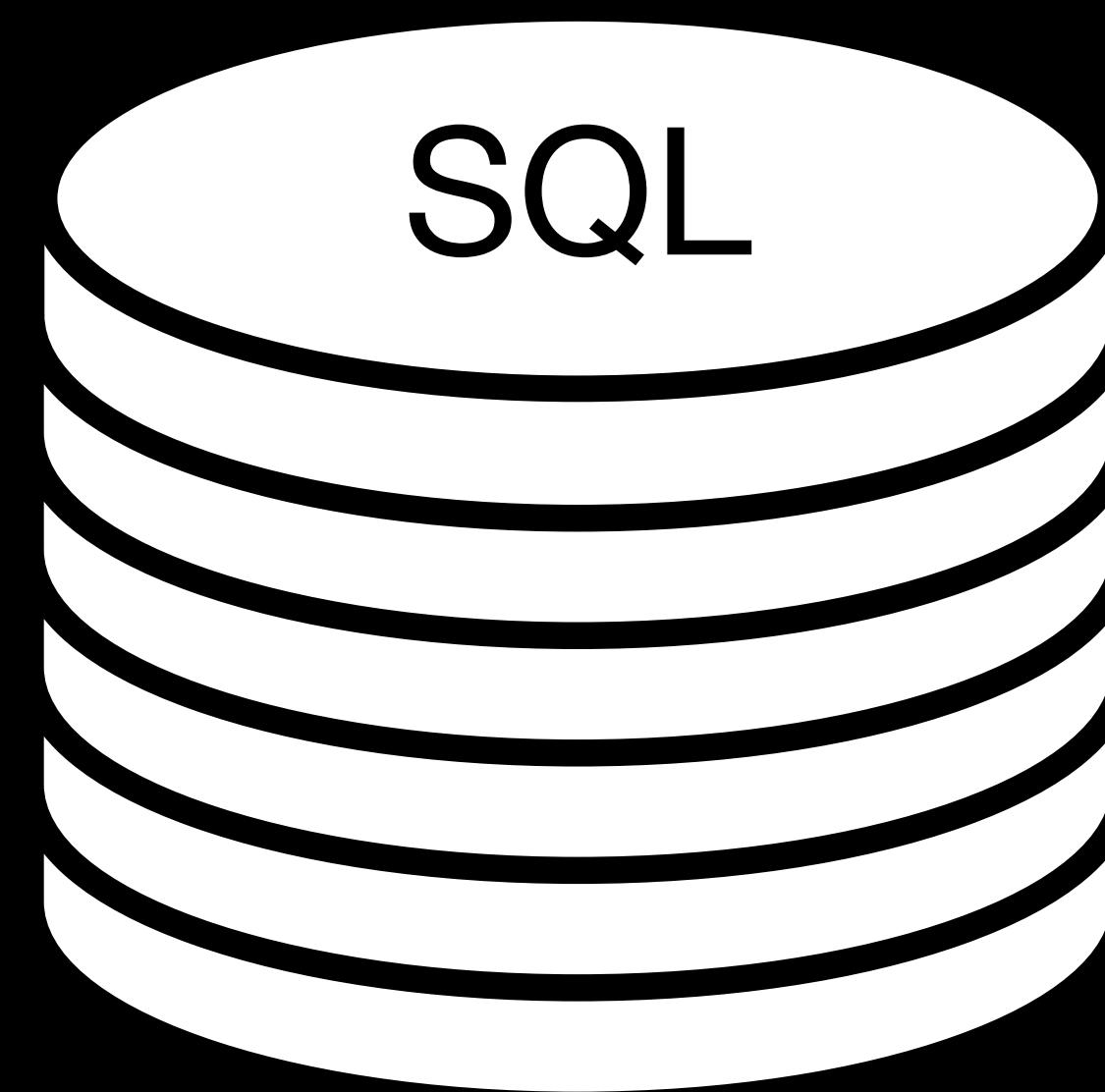
void loop() {
    if (Serial1.available() > 0) {
        String texto = Serial1.readStringUntil('\n');
        texto.trim();
        if (texto.length() > 5) { // ignora respostas estranhas
            Serial.println("Resposta do leitor: " + texto);
            long long codigo = strtoll(texto.c_str(), nullptr, 10);
        }
    }
}

```

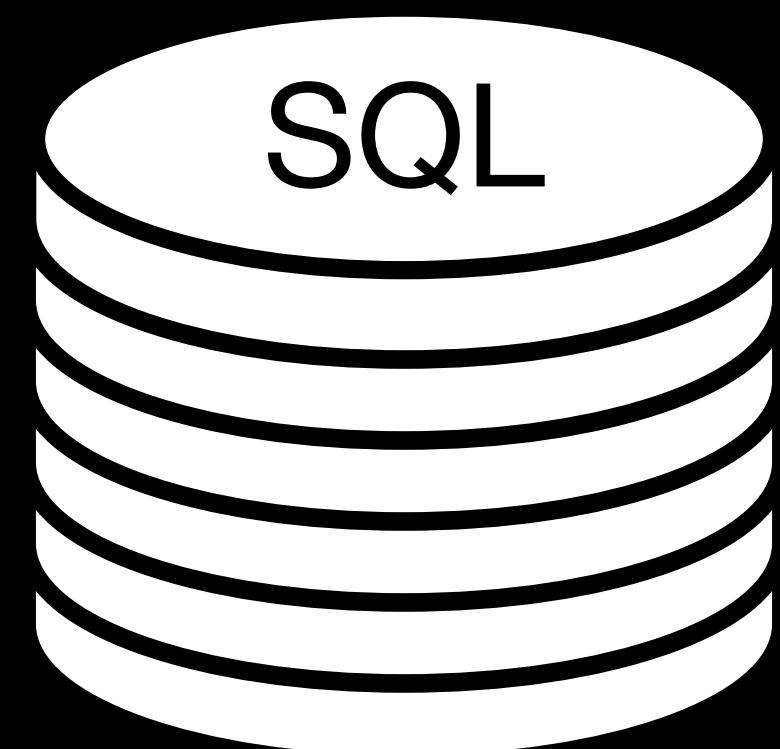
Resposta do leitor: 7896065880069  
 Resposta do leitor: <https://youtu.be/w5ebcowAJD8>

Exemplo de Código para Comunicação com Leitor de Código de Barras

# Internet



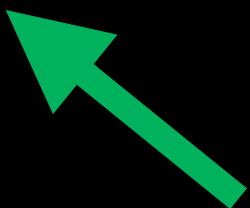
Banco de Dados SQL



Exemplos de Bancos de Dados SQL

## Provas

<b>id</b>	<b>nome</b>	<b>início</b>	<b>fim</b>
1	P1 de Programação	22/04/2024 às 17:00:00	22/04/2024 às 19:00:00
2	P2 de Programação	17/05/2024 às 17:00:00	17/05/2024 às 19:00:00



## Questões

<b>id</b>	<b>id_prova</b>	<b>numero</b>	<b>pontos</b>	<b>enunciado</b>	<b>gabarito</b>
1	1	1	6,0	Crie uma função que receba dois...	def funcao(x, y): ...
2	1	2	4,0	Faça um programa que solicite...	
3	2	1	3,0	Percorra uma lista contendo...	for elemento in lista: ...

```
CREATE TABLE provas (
    id INT NOT NULL AUTO_INCREMENT,
    nome VARCHAR(100),
    inicio DATETIME,
    fim DATETIME,
    PRIMARY KEY (id)
);
```

```
CREATE TABLE questoes (
    id INT NOT NULL AUTO_INCREMENT,
    prova_id INT,
    numero INT,
    pontos DECIMAL(3,1),
    enunciado TEXT,
    gabarito TEXT DEFAULT NULL,
    PRIMARY KEY (id),
    FOREIGN KEY (prova_id) REFERENCES prova(id)
);
```

Use sempre ASPAS SIMPLES  
nas strings do INSERT!



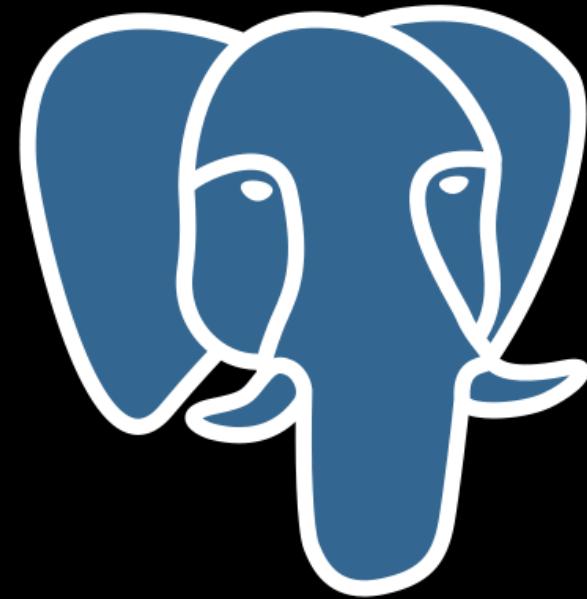
```
INSERT INTO provas (id, nome, inicio, fim) VALUES
(1, 'P1 de Programação', '2024-04-22 17:00:00', '2024-04-22 19:00:00'),
(2, 'P2 de Programação', '2024-05-17 17:00:00', '2024-05-17 19:00:00');
```

```
INSERT INTO questoes (id_prova, numero, pontos, enunciado) VALUES
(1, 1, 6.0, 'Crie uma função que receba dois...'),
(1, 2, 4.0, 'Faça um programa que solicite...'),
(2, 1, 4.0, 'Faça um programa que solicite...');
```

```
SELECT *
FROM questoes
WHERE pontos >= 2.0 AND id_prova = 1
ORDER BY id_prova ASC, numero ASC;
```



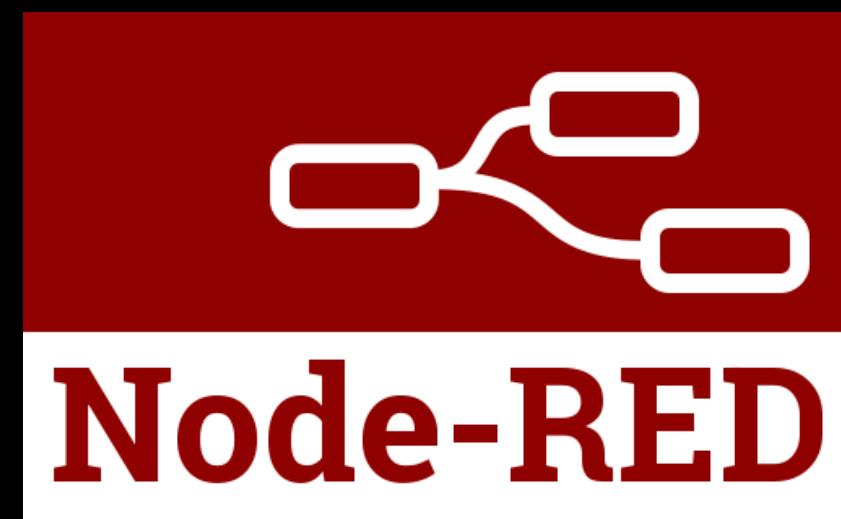
X



- Banco de dados local
- Bom para poucos dados



- Banco de dados no servidor
- Bom para muitos dados



Banco de Dados Local x Banco de Dados no Servidor

Essa biblioteca é boa,  
mas tem problemas...



sketch\_apr5a | Arduino IDE 2.3.4

Arduino Mega or Mega 2...

LIBRARY MANAGER

sqlite

Type: All

Topic: All

**SQLiteDatabaseConnection**  
by Jiatong Zhu...

Modern C++ interface for SQLite. Wraps SQLite C API into C++ OOP API with template support to simplify...

More info

1.0.1

INSTALL

sketch\_apr5a.ino

```
1 void setup() {  
2     // put your setup code here  
3 }  
4  
5 void loop() {  
6     // put your main code here  
7 }  
8  
9 }  
10
```

Ln 1, Col 1   Arduino Mega or Mega 2560 on /dev/

Biblioteca para SQLite?

The image shows two side-by-side screenshots. On the left is the GitHub repository page for `wallysalami/SQLiteManager`. It features a sidebar with links to `README` and `MIT license`. The main content area has a title **SQLiteManager** and a description: "Manage SQLite databases in ESP32 using the [Sqlite3Esp32 library](#) with a C++ API wrapper." Below this, it says "This is a fork of the [SQLiteDatabaseManager library](#) with a few improvements:" followed by a bulleted list: • Uses [ArduinoJson](#) instead of [Arduino\\_JSON](#) • Provides enhanced support for additional data types • Offers a more intuitive interface for database operations. At the bottom of the page are standard GitHub links: Terms, Privacy, Security, Status, Docs, Contact, and Manage cookies, along with the GitHub logo and the text "© 2025 GitHub, Inc.". On the right is a screenshot of the Arduino IDE Library Manager. The title bar says "sketch\_apr5a | Arduino IDE 2.3.4". The central panel shows the **LIBRARY MANAGER** with a search bar containing "sqlitemanager". Below the search bar are dropdown menus for **Type: All** and **Topic: All**. A detailed description of the **SQLiteManager** library is displayed, including its author "Jan K. S.", email "[developer@janks.dev.br](mailto:developer@janks.dev.br)", and a brief description: "Modern C++ interface for SQLite. Wraps SQLite C API into C++ OOP API with template support to simplify development." An [More info](#) link and a dropdown menu showing "1.0.0" are also present. To the right of the description is a code editor window showing the start of an Arduino sketch with `void setup()` and `void loop()` functions. The status bar at the bottom of the Arduino IDE shows "Ln 1, Col 1" and "ESP32 Dev Module [not connected]".

SQLiteManager: Biblioteca para Gerenciar SQLite no ESP32

```
#include <SQLiteManager.h>
#include <LittleFS.h>
#include <ArduinoJson.h>

SQLiteManager banco;

void setup() {
    Serial.begin(115200); delay(500);

    if (!LittleFS.begin(false)) {
        Serial.println("Falha ao montar o sistema de arquivos!");
        while (true); // trava programa aqui em caso de erro
    }

    try {
        banco.open("/littlefs/banco.db");
    }
    catch (const std::exception &e) {
        Serial.println(e.what());
        while (true); // trava programa aqui em caso de erro
    }
}

// código continua a seguir...
```

Inicialização do SQLite via LittleFS

```
#include <SQLiteManager.h>
#include <SD_MMC.h>
#include <ArduinoJson.h>

SQLiteManager banco;

void setup() {
    Serial.begin(115200); delay(500);

    SD_MMC.setPins(39, 38, 40);
    if (!SD_MMC.begin("/sdcard", true)) {
        Serial.println("Falha na inicialização do cartão SD!");
        while (true); // trava programa aqui em caso de erro
    }

    try {
        banco.open("/sdcard/banco.db");
    }
    catch (const std::exception &e) {
        Serial.println(e.what());
        while (true); // trava programa aqui em caso de erro
    }
}

// código continua a seguir...
```

Inicialização do SQLite via SD Card

```
// continuação do código anterior...

// em alguma função do código...

int id_prova = 1;
int numero = 3;
float pontos = 0.5;
String enunciado = "Crie uma função que receba dois...";

try {
    JsonDocument result = banco.execute(
        "INSERT INTO questoes (id_prova, numero, pontos, enunciado) "
        "VALUES(?, ?, ?, ?)", id_prova, numero, pontos, enunciado
    );
}
catch (const std::exception &e) {
    Serial.println(e.what());
}
```



```
// continuação do código anterior...

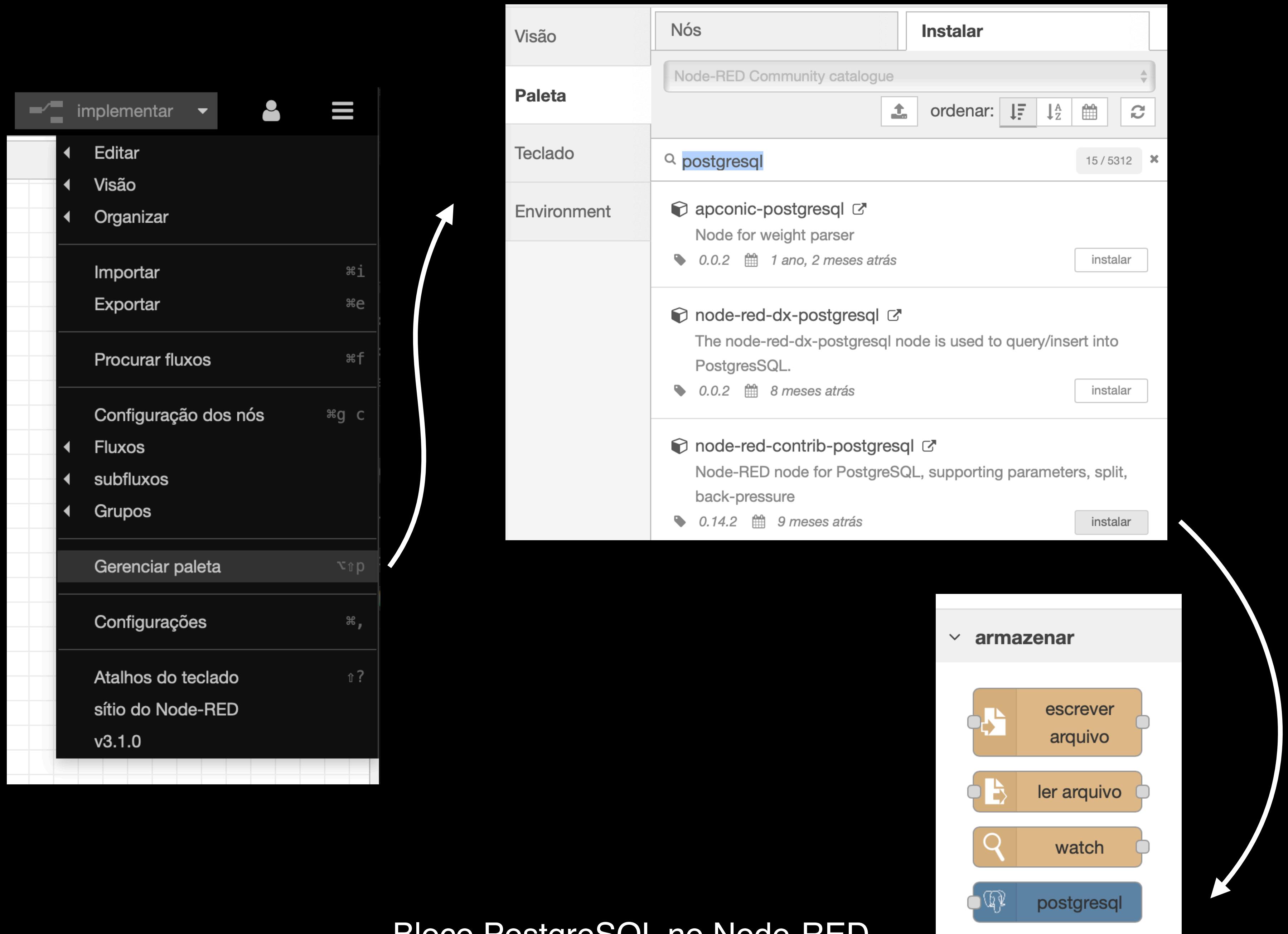
// em alguma função do código...

float pontos = 6.0;

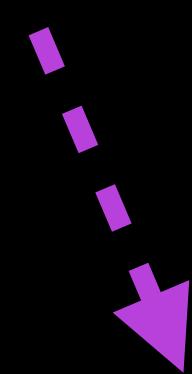
try {
    JsonDocument resultados = banco.execute(
        "SELECT * FROM questoes WHERE pontos = ?", pontos
    );
}

for (int i = 0; i < resultados.size(); i++) {
    JsonDocument questao = resultados[i];
    String enunciado = questao["enunciado"];
    int numeroDaQuestao = questao["numero"];
}
catch (const std::exception &e) {
    Serial.println(e.what());
}
```

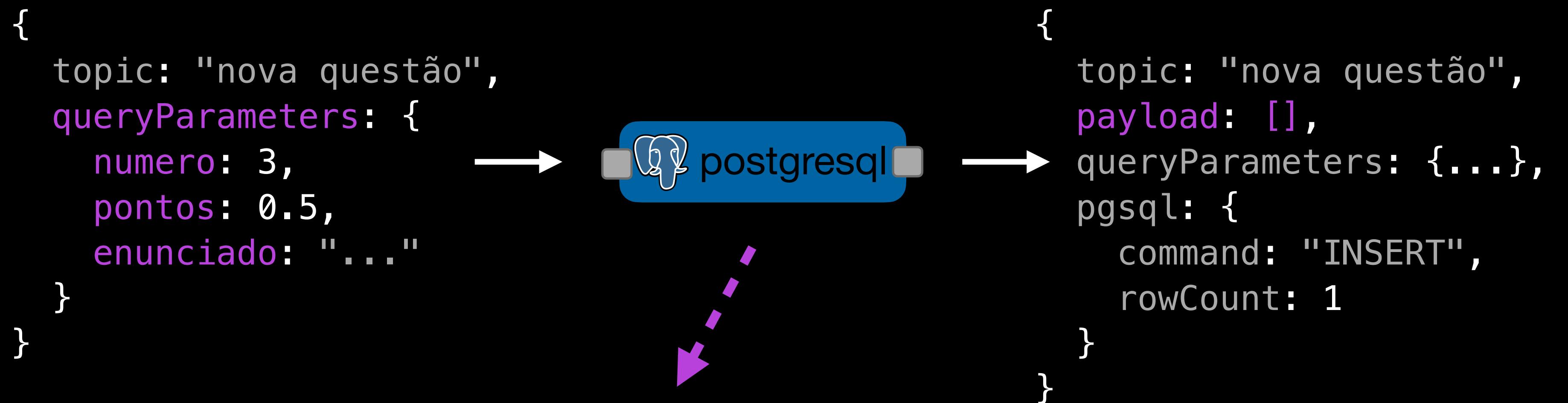




```
{  
topic: "nova questão",  
payload: {  
    numero: 3,  
    pontos: 0.5,  
    enunciado: "..."  
}  
}  
  
{  
topic: "nova questão",  
queryParameters: {  
    numero: 3,  
    pontos: 0.5,  
    enunciado: "..."  
}  
}
```



Criação do Campo QueryParameters para o PostgreSQL



A screenshot of a PostgreSQL query interface. The title bar says "Query". The query text is:

```
1  INSERT INTO questoes  
2  (id_prova, numero, pontos, enunciado) VALUES  
3  (1, $numero, $pontos, $enunciado);
```

Inserção de Dados no PostgreSQL via Node-RED

```
{  
  topic: "busca",  
  queryParameters: { → pgsql → {  
    minimo: 1.0  
  }  
}  
  topic: "busca",  
  payload: [  
    {id: 1, pontos: 6, ...},  
    {id: 2, pontos: 4, ...},  
    ...  
  ],  
  queryParameters: {...},  
  pgsql: {...}  
}
```

Query

```
1 SELECT *  
2 FROM questoes  
3 WHERE pontos >= $minimo;
```

Inserção de Dados no PostgreSQL via Node-RED



# Beekeeper Studio

postgresql.janks.dev.br - Beekeeper Studio

supermercado C + Busca Código Busca Nome Vendas produtos [all] Upgrade

Filter ENTITIES 527

public

- clientes
- produtos
- vendas

id nome codigo marca categoria preco

	id	nome	codigo	marca	categoria	preco
1	1	Barra De Cereais Super Mix S gl...	8898993006022	(NULL)	MERCEARIA	1500
2	2	Pao 7 Graos E Semente 300g H...	7896319000175	HUMMEL	PADARIA	1500
3	3	Sabonete Em Barra Anti Acne A...	650240012747	ASEPXIA	HIGIENE E BELEZA	1500
4	4	Creme Trat Dove Mascar Brilho...	7898422752347	DOVE	HIGIENE E BELEZA	1500
5	5	logurte Grego Sem Lactose Lat...	7898034922756	BELO VALE	FRIOS E LATICÍNIOS	1500
6	6	Multifibras Sb Morango 300g A...	7899468502613	APISNUTRI	SUPLEMENTOS	1500
7	7	Meia Trifil Europeu R 6308 Unid...	7891213397277	(NULL)	BAZAR	1500
8	8	Vinho Caballero De La Cepa M...	7790470000787	(NULL)	BEBIDAS	1500
9	9	Mirian D Or Sombra Iluminador...	7898918979029	(NULL)	HIGIENE E BELEZA	1500
10	10	Amendoim Torrado 100 Cacau ...	7898994524564	(NULL)	MERCEARIA	1500
11	11	Requeijao Lilitati Copo Plast	7897190403000	(NULL)	FRIOS E LATICÍNIOS	1500
12	12	Paellera Tramontina Aluminio 6...	7891112089518	ALUMINIO	BAZAR	1500
13	13	Faca Cozinha Simonaggio N5	7896211831167	(NULL)	BAZAR	1500
14	14	Polvilhos 40g Perrota	7898936578242	PERROTTA	MERCEARIA	1500

postgresql.janks.dev.br postgresql Structure # Unknown 3 days ago 1 > C +

Aplicativo Cliente de SQL