

NanoC6 Zigbee Arduino

NanoC6 Zigbee Arduino に関連するサンプルプログラム。

準備作業

コンパイル要件

- M5Stack ボードマネージャーバージョン >= 3.2.5
- 開発ボードオプション = M5NanoC6

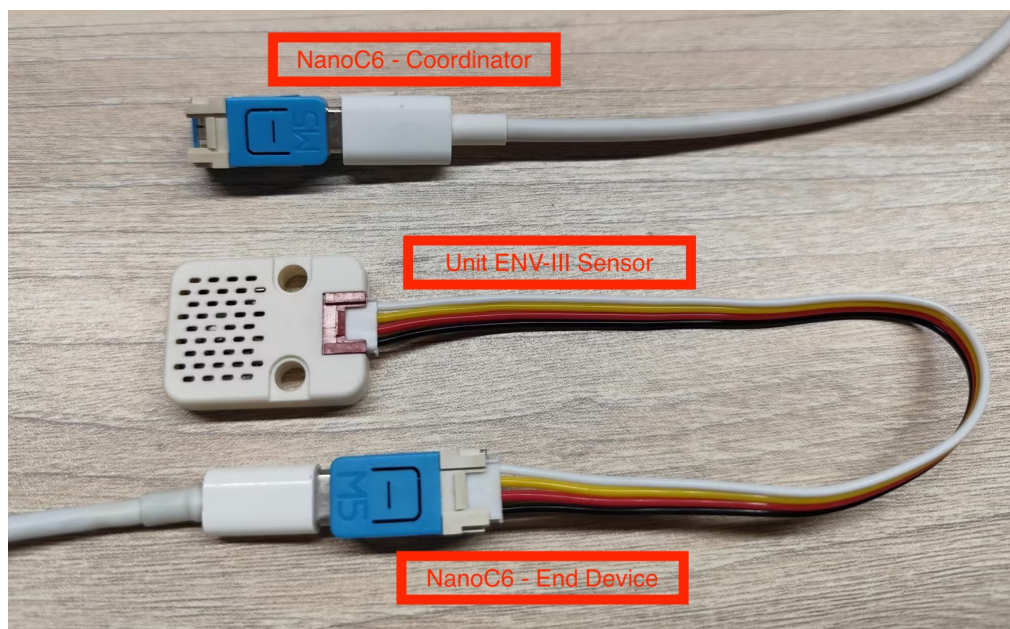
Zigbee サーマスタット (コーディネータ)

このサンプルは、Zigbee コーディネータ (Coordinator) を設定し、サーモスタットとして使用して以下の機能を実現する方法を示します。

- Zigbee コーディネータとして動作
- 温度センサーデータを受信
- 温度センサーのレポート間隔を設定
- シリアルに温度および設定情報を表示

ハードウェア要件

- NanoC6 ×1 (Zigbee コーディネータ、サーモスタット例を実行)
- NanoC6 ×1 (Zigbee エンドデバイス、温度センサー例を実行)
- Unit ENV-III ×1 (環境温度測定用)



設定説明

Arduino IDE ツールメニュー設定:

- 正しい開発ボードを選択: **Tools -> Board: M5NanoC6**
- USB シリアル起動を有効化: **Tools -> USB CDC On Boot: Enabled**
- 消去を有効化: **Tools -> Erase All Flash Before Sketch Upload: Enabled** (無効の場合、接続失敗の可能性あり)
- flash サイズを選択: **Tools -> Flash Size: 4MB**

- Zigbee パーティションスキームを選択: Tools -> Partition Scheme: Zigbee ZCZR 4MB with spiffs
- コーディネータモードを選択: Tools -> Zigbee mode: Zigbee ZCZR (coordinator/router)
- 正しいシリアルポートを選択: Tools -> Port

Board: "M5NanoC6"	>
Port: "/dev/cu.usbmodem141101"	>
Reload Board Data	
Get Board Info	
USB CDC On Boot: "Enabled"	>
CPU Frequency: "160MHz (WiFi)"	>
Core Debug Level: "Verbose"	>
Erase All Flash Before Sketch Upload: "Enabled"	>
Flash Frequency: "80MHz"	>
Flash Mode: "QIO"	>
Flash Size: "4MB (32Mb)"	>
JTAG Adapter: "Disabled"	>
Partition Scheme: "Zigbee ZCZR 4MB with spiffs"	>
Upload Speed: "921600"	>
Zigbee Mode: "Zigbee ZCZR (coordinator/router)"	>

サンプルプログラム

```
1  #ifndef ZIGBEE_MODE_ZCZR
2  #error "Zigbee coordinator mode is not selected in Tools->Zigbee mode"
3  #endif
4
5  #include "Zigbee.h"
6
7  // 定義エンドポイント番号
8  #define THERMOSTAT_ENDPOINT_NUMBER 5
9
10 // Zigbee サーモスタットオブジェクトを作成 (温度データ受信)
11 ZigbeeThermostat zbThermostat = ZigbeeThermostat(THERMOSTAT_ENDPOINT_NUMBER);
12
13 // 温度データ変数
14 float sensor_temp = 0.0;
15 float sensor_max_temp = 120.0;
16 float sensor_min_temp = -40.0;
17 float sensor_tolerance = 1.0;
18
19 // 温度受信コールバック関数
20 void receiveSensorTemp(float temperature) {
21     Serial.printf("Temperature received: %.2f°C\n", temperature);
22     sensor_temp = temperature;
23 }
24
25 // センサー設定受信コールバック関数
26 void receiveSensorConfig(float min_temp, float max_temp, float tolerance) {
27     Serial.printf("Sensor config: min=%.2f°C, max=%.2f°C, tolerance=%.2f°C\n",
28         min_temp, max_temp, tolerance);
29     sensor_min_temp = min_temp;
30     sensor_max_temp = max_temp;
31     sensor_tolerance = tolerance;
32 }
33
34 void setup() {
35     Serial.begin(115200);
36     delay(1000);
37     Serial.println("=== NanoC6 Zigbee Coordinator ===");
38
39     // コールバック関数を設定
40     zbThermostat.onTempReceive(receiveSensorTemp);
41     zbThermostat.onConfigReceive(receiveSensorConfig);
42
43     // デバイス情報を設定
44     zbThermostat.setManufacturerAndModel("Espressif", "ZigbeeThermostat");
45
46     // エンドポイントを追加
47     Zigbee.addEndpoint(&zbThermostat);
48
49     // ネットワークを 180 秒間オープン
50     Zigbee.setRebootOpenNetwork(180);
```

```

51
52 // コーディネータを起動
53 Serial.println("Starting Zigbee Coordinator...");
54 if (!Zigbee.begin(ZIGBEE_COORDINATOR)) {
55     Serial.println("Zigbee failed to start!");
56     Serial.println("Rebooting...");
57     ESP.restart();
58 }
59
60 Serial.println("Zigbee Coordinator started");
61 Serial.println("Network is open for 180 seconds");
62 Serial.println("Waiting for temperature sensor to bind...");
63
64 // End Device のバインドを待機
65 while (!zbThermostat.bound()) {
66     Serial.print(".");
67     delay(500);
68 }
69
70 Serial.println("\nTemperature sensor bound successfully!");
71
72 // レポート間隔を一度だけ設定
73 zbThermostat.setTemperatureReporting(0, 10, 2);
74 Serial.println("Temperature reporting configured");
75
76 // センサー設定を取得
77 zbThermostat.getSensorSettings();
78 }
79
80 void loop() {
81     // 定期的に温度データを表示
82     static uint32_t last_print = 0;
83     if (millis() - last_print > 10000) {
84         last_print = millis();
85
86         int temp_percent = (int)((sensor_temp - sensor_min_temp) /
87                                 (sensor_max_temp - sensor_min_temp) * 100);
88         Serial.printf("Current temperature: %.2f°C (%d%%)\n",
89                       sensor_temp, temp_percent);
90     }
91
92     delay(100);
93 }

```

使用手順

1. サーマスタットコードをコーディネータ (Coordinator) デバイスに書き込む
2. 温度センサーコードをエンドデバイス (End Device) に書き込む
3. コーディネータ起動後、自動的にネットワークを作成しデバイス参加を待機、10 秒ごとに現在の温度情報を表示
4. シリアルモニタで温度データを確認し、温度変化時の自動レポートおよび設定情報の受信を確認

OutputSerial Monitor ×

Message (Enter to send message to 'M5NanoC6' on '/dev/cu.usbmodem141101')

New Line115200 baud

[108922] [V] [ZigbeeHandlers.cpp:104] zb_attribute_reporting_handler(): Received report from address(0x2da0) src endpoint(10) to dst endpoint(5) cluster(0x40
Temperature received: 30.11°C
Current temperature: 30.11°C (43%)
[118949] [V] [ZigbeeHandlers.cpp:104] zb_attribute_reporting_handler(): Received report from address(0x2da0) src endpoint(10) to dst endpoint(5) cluster(0x40
Temperature received: 30.03°C
Current temperature: 30.03°C (43%)
[128976] [V] [ZigbeeHandlers.cpp:104] zb_attribute_reporting_handler(): Received report from address(0x2da0) src endpoint(10) to dst endpoint(5) cluster(0x40
Temperature received: 29.85°C
Current temperature: 29.85°C (43%)
[139004] [V] [ZigbeeHandlers.cpp:104] zb_attribute_reporting_handler(): Received report from address(0x2da0) src endpoint(10) to dst endpoint(5) cluster(0x40
Temperature received: 29.76°C
Current temperature: 29.76°C (43%)
[149031] [V] [ZigbeeHandlers.cpp:104] zb_attribute_reporting_handler(): Received report from address(0x2da0) src endpoint(10) to dst endpoint(5) cluster(0x40
Temperature received: 29.63°C
Current temperature: 29.63°C (43%)

Zigbee 温度センサー

このサンプルは、Zigbee エンドデバイス（End Device）を設定し、家庭用オートメーション（HA）温度センサーとして使用方法を示します。

- Zigbee エンドデバイスとして動作
- チップ温度データを読み取り
- 定期的にコーディネータへ温度をレポート
- 要求に応じた温度レポートをサポート

設定説明

Arduino IDE ツールメニュー設定:

- 正しい開発ボードを選択: **Tools -> Board: M5NanoC6**
- USB シリアル起動を有効化: **Tools -> USB CDC On Boot: Enabled**
- 消去を有効化: **Tools -> Erase All Flash Before Sketch Upload: Enabled**（無効の場合、接続失敗の可能性あり）
- flash サイズを選択: **Tools -> Flash Size: 4MB**
- Zigbee パーティションスキームを選択: **Tools -> Partition Scheme: Zigbee ZCZR 4MB with spiffs**
- エンドデバイスモードを選択: **Tools -> Zigbee mode: Zigbee ED (end device)**
- 正しいシリアルポートを選択: **Tools -> Port**

Board: "M5NanoC6"	>
Port: "/dev/cu.usbmodem14201"	>
Reload Board Data	
Get Board Info	
USB CDC On Boot: "Enabled"	>
CPU Frequency: "160MHz (WiFi)"	>
Core Debug Level: "Verbose"	>
Erase All Flash Before Sketch Upload: "Enabled"	>
Flash Frequency: "80MHz"	>
Flash Mode: "QIO"	>
Flash Size: "4MB (32Mb)"	>
JTAG Adapter: "Disabled"	>
Partition Scheme: "Zigbee ZCZR 4MB with spiffs"	>
Upload Speed: "921600"	>
Zigbee Mode: "Zigbee ED (end device)"	>

サンプルプログラム

```
1  #ifndef ZIGBEE_MODE_ED
2  #error "Zigbee end device mode is not selected in Tools->Zigbee mode"
3  #endif
4
5  #include "Zigbee.h"
6  #include <Wire.h>
7
8  // 定義エンドポイント番号
9  #define TEMP_SENSOR_ENDPOINT_NUMBER 10
10
11 // 定義 SHT30 I2C アドレス
12 #define SHT30_I2C_ADDR 0x44
13
14 // NanoH2 Grove インターフェースの I2C ピン
15 #define I2C_SDA 2    // G2 = GPIO2 (黄色線)
16 #define I2C_SCL 1    // G1 = GPIO1 (白色線)
17
18 // Zigbee 温度センサーオブジェクトを作成
19 ZigbeeTempSensor zbTempSensor = ZigbeeTempSensor(TEMP_SENSOR_ENDPOINT_NUMBER);
20
21 // 温度データ変数
22 float temperature = 0.0;
23
24 // タスク同期フラグ
25 volatile bool temp_updated = false;
26
27 /***** SHT30 温度読み取り関数 (温度のみ) *****/
28 bool readSHT30Temperature(float &temp) {
29     // 測定コマンド送信 (高再現性測定)
30     Wire.beginTransmission(SHT30_I2C_ADDR);
31     Wire.write(0x2C);
32     Wire.write(0x06);
33
34     if (Wire.endTransmission() != 0) {
35         Serial.println("SHT30 communication failed!");
36         return false;
37     }
38
39     delay(20);
40
41     Wire.requestFrom(SHT30_I2C_ADDR, 6);
42
43     if (Wire.available() < 6) {
44         Serial.println("SHT30 data not available!");
45         return false;
46     }
47
48     uint8_t tempData[3];
49     for (int i = 0; i < 3; i++) {
50         tempData[i] = Wire.read();
```

```

51     }
52
53     for (int i = 0; i < 3; i++) {
54         Wire.read();
55     }
56
57     uint16_t rawTemp = (tempData[0] << 8) | tempData[1];
58     temp = -45.0 + 175.0 * ((float)rawTemp / 65535.0);
59
60     if (temp < -40.0 || temp > 125.0) {
61         Serial.printf("Invalid temperature: %.2f°C\n", temp);
62         return false;
63     }
64
65     return true;
66 }
67
68 /***** 温度読み取りタスク *****/
69 static void temp_sensor_value_update(void *arg) {
70     for (;;) {
71         if (readSHT30Temperature(temperature)) {
72             Serial.printf("Temperature: %.2f°C\n", temperature);
73             zbTempSensor.setTemperature(temperature);
74             temp_updated = true;
75         } else {
76             Serial.println("Failed to read SHT30");
77         }
78
79         delay(10000);
80     }
81 }
82
83 void setup() {
84     Serial.begin(115200);
85     delay(1000);
86     Serial.println("=== NanoC6 End Device + Unit ENV III (Temperature Only) ===");
87
88     Wire.begin(I2C_SDA, I2C_SCL, 100000);
89     Serial.printf("I2C initialized (SDA=GPIO%d, SCL=GPIO%d, 100kHz)\n",
90                 I2C_SDA, I2C_SCL);
91
92     delay(100);
93
94     Serial.println("Scanning I2C bus...");
95     Wire.beginTransmission(SHT30_I2C_ADDR);
96     uint8_t error = Wire.endTransmission();
97
98     if (error == 0) {
99         Serial.println("SHT30 detected at address 0x44");
100     } else {
101         Serial.printf("SHT30 not found! Error code: %d\n", error);
102         Serial.println("\n    Expected wiring:");
103         Serial.println("    - Black (GND) → GND");

```

```

104     Serial.println("    - Red (5V) → 5V");
105     Serial.println("    - Yellow (SDA) → G2 (GPIO2)");
106     Serial.println("    - White (SCL) → G1 (GPIO1)");
107     while (1) delay(1000);
108 }
109
110 zbTempSensor.setManufacturerAndModel("Espressif", "TempSensor");
111 zbTempSensor.setMinMaxValue(-40, 120);
112 zbTempSensor.setTolerance(1);
113
114 Zigbee.addEndpoint(&zbTempSensor);
115
116 Serial.println("Starting Zigbee End Device...");
117 if (!Zigbee.begin()) {
118     Serial.println("Zigbee failed to start!");
119     Serial.println("Rebooting in 5 seconds...");
120     delay(5000);
121     ESP.restart();
122 }
123
124 Serial.println("Zigbee End Device started");
125 Serial.println("Connecting to network...");
126
127 while (!Zigbee.connected()) {
128     Serial.print(".");
129     delay(1000);
130 }
131
132 Serial.println("\nConnected to Zigbee network!");
133
134 xTaskCreate(temp_sensor_value_update, "temp_sensor_update", 4096, NULL, 10, NULL);
135
136 zbTempSensor.setReporting(1, 0, 1);
137 Serial.println("Temperature reporting configured");
138 }
139
140 void loop() {
141     if (temp_updated) {
142         temp_updated = false;
143         zbTempSensor.reportTemperature();
144         Serial.println("Temperature reported to Coordinator");
145     }
146
147     delay(100);
148 }

```

使用手順

1. コーディネータが起動してネットワークを作成していることを確認し、温度センサーコードをエンドデバイスに書き込む
2. デバイス起動後、自動的にネットワークを検索して参加し、10 秒ごとに温度を読み取り、温度変化が 0.01℃ を超えると自動的にレポート

OutputSerial Monitor X

Message (Enter to send message to 'M5NanoC6' on '/dev/cu.usbmodem14201')

New Line115200 baud

Temperature: 30.11°C
[112420] [V] [ZigbeeTempSensor.cpp:79] setTemperature(): Updating temperature sensor value...
[112420] [D] [ZigbeeTempSensor.cpp:81] setTemperature(): Setting temperature to 3011
[112438] [V] [ZigbeeHandlers.cpp:397] zb_cmd_default_resp_handler(): Received default response: from address(0x0), src_endpoint(5) to dst_endpoint(10), clust
[112470] [V] [ZigbeeTempSensor.cpp:111] reportTemperature(): Temperature report sent
Temperature reported to Coordinator
[112486] [V] [ZigbeeHandlers.cpp:397] zb_cmd_default_resp_handler(): Received default response: from address(0x0), src_endpoint(5) to dst_endpoint(10), clust
[122420] [V] [esp32-hal-i2c-ng.c:262] i2cWrite(): i2c_master_transmit: bus=0 addr=0x44 handle=0x40823414 size=2
[122441] [V] [esp32-hal-i2c-ng.c:308] i2cRead(): i2c_master_receive: bus=0 addr=0x44 handle=0x40823414 size=6
Temperature: 30.03°C
[122443] [V] [ZigbeeTempSensor.cpp:79] setTemperature(): Updating temperature sensor value...
[122443] [D] [ZigbeeTempSensor.cpp:81] setTemperature(): Setting temperature to 3003
[122497] [V] [ZigbeeTempSensor.cpp:111] reportTemperature(): Temperature report sent
Temperature reported to Coordinator
[122514] [V] [ZigbeeHandlers.cpp:397] zb_cmd_default_resp_handler(): Received default response: from address(0x0), src_endpoint(5) to dst_endpoint(10), clust
[132443] [V] [esp32-hal-i2c-ng.c:262] i2cWrite(): i2c_master_transmit: bus=0 addr=0x44 handle=0x40823414 size=2
[132464] [V] [esp32-hal-i2c-ng.c:308] i2cRead(): i2c_master_receive: bus=0 addr=0x44 handle=0x40823414 size=6
Temperature: 29.86°C
[132466] [V] [ZigbeeTempSensor.cpp:79] setTemperature(): Updating temperature sensor value...
[132466] [D] [ZigbeeTempSensor.cpp:81] setTemperature(): Setting temperature to 2985
[132524] [V] [ZigbeeTempSensor.cpp:111] reportTemperature(): Temperature report sent
Temperature reported to Coordinator

Zigbee ネットワークスキャン

このサンプルは、周囲の Zigbee ネットワークをスキャンし、ネットワーク情報をシリアルに表示します。

1. 周囲にアクティブな Zigbee ネットワークがあることを確認し、スキャンコードをデバイスに書き込む
2. デバイス起動後、自動的にスキャンを開始し、各スキャン完了時に結果を表示して次のスキャンを開始

サンプルプログラム

```
1  #if !defined(ZIGBEE_MODE_ED) && !defined(ZIGBEE_MODE_ZCZR)
2  #error "Zigbee device mode is not selected in Tools->Zigbee mode"
3  #endif
4
5  #include "Zigbee.h"
6
7  #ifdef ZIGBEE_MODE_ZCZR
8  zigbee_role_t role = ZIGBEE_ROUTER;
9  #else
10 zigbee_role_t role = ZIGBEE_END_DEVICE;
11 #endif
12
13 void printScannedNetworks(uint16_t networksFound) {
14     if (networksFound == 0) {
15         Serial.println("No networks found");
16     } else {
17         zigbee_scan_result_t *scan_result = Zigbee.getScanResult();
18         Serial.println("\nScan done");
19         Serial.print(networksFound);
20         Serial.println(" networks found:");
21         Serial.println("Nr | PAN ID | CH | Permit Joining | Router Capacity | End Device Capacity |");
22         for (int i = 0; i < networksFound; ++i) {
23             Serial.printf("%2d", i + 1);
24             Serial.print(" | ");
25             Serial.printf("0x%04hx", scan_result[i].short_pan_id);
26             Serial.print(" | ");
27             Serial.printf("%2d", scan_result[i].logic_channel);
28             Serial.print(" | ");
29             Serial.printf("%-14.14s", scan_result[i].permit_joining ? "Yes" : "No");
30             Serial.print(" | ");
31             Serial.printf("%-15.15s", scan_result[i].router_capacity ? "Yes" : "No");
32             Serial.print(" | ");
33             Serial.printf("%-19.19s", scan_result[i].end_device_capacity ? "Yes" : "No");
34             Serial.print(" | ");
35             Serial.printf(
36                 "%02x:%02x:%02x:%02x:%02x:%02x:%02x", scan_result[i].extended_pan_id[7], scan_result[i].extended_pan_id[6],
37                 scan_result[i].extended_pan_id[4], scan_result[i].extended_pan_id[3], scan_result[i].extended_pan_id[2],
38                 scan_result[i].extended_pan_id[0]
39             );
40             Serial.println();
41             delay(10);
42         }
43         Serial.println("");
44         Zigbee.scanDelete();
45     }
46 }
47
48 void setup() {
49     Serial.begin(115200);
50 }
```



```

51     if (!Zigbee.begin(role)) {
52         Serial.println("Zigbee failed to start!");
53         Serial.println("Rebooting...");
54         ESP.restart();
55     }
56
57     Serial.println("Setup done, starting Zigbee network scan...");
58     Zigbee.scanNetworks();
59 }
60
61 void loop() {
62     int16_t ZigbeeScanStatus = Zigbee.scanComplete();
63     if (ZigbeeScanStatus < 0) {
64         if (ZigbeeScanStatus == ZB_SCAN_FAILED) {
65             Serial.println("Zigbee scan has failed. Starting again.");
66             delay(1000);
67             Zigbee.scanNetworks();
68         }
69         delay(100);
70     } else {
71         printScannedNetworks(ZigbeeScanStatus);
72         delay(1000);
73         Zigbee.scanNetworks();
74     }
75 }

```

使用手順

1. 周囲にアクティブな Zigbee ネットワークがあることを確認し、スキャンコードをデバイスに書き込む
2. デバイス起動後、自動的にスキャンを開始し、各スキャン完了時に結果を表示して次のスキャンを開始

OutputSerial Monitor X

Message (Enter to send message to 'M5NanoC6' on '/dev/cu.usbmodem141301')

New Line115200 baud

```

Scan done
1 networks found:
Nr | PAN ID | CH | Permit Joining | Router Capacity | End Device Capacity | Extended PAN ID
1 | 0xda1b | 19 | No | Yes | Yes | 40:4c:ca:ff:fe:5a:ef:58

[ 16658] [V] [ZigbeeCore.cpp:515] scanNetworks(): Scanning Zigbee networks
[ 24772] [V] [ZigbeeCore.cpp:484] scanCompleteCallback(): Zigbee network scan complete
[ 24772] [V] [ZigbeeCore.cpp:486] scanCompleteCallback(): Found 2 networks
[ 24772] [V] [ZigbeeCore.cpp:489] scanCompleteCallback(): Network 0: PAN ID: 0xda1b, Permit Joining: No, Extended PAN ID: 40:4c:ca:ff:fe:5a:ef:58, Channel: 1
[ 24773] [V] [ZigbeeCore.cpp:489] scanCompleteCallback(): Network 1: PAN ID: 0x9f02, Permit Joining: No, Extended PAN ID: 05:b6:db:e8:c5:d6:da:6d, Channel: 1

Scan done
2 networks found:
Nr | PAN ID | CH | Permit Joining | Router Capacity | End Device Capacity | Extended PAN ID
1 | 0xda1b | 19 | No | Yes | Yes | 40:4c:ca:ff:fe:5a:ef:58
2 | 0x9f02 | 15 | No | Yes | Yes | 05:b6:db:e8:c5:d6:da:6d

[ 25803] [V] [ZigbeeCore.cpp:515] scanNetworks(): Scanning Zigbee networks
[ 33917] [V] [ZigbeeCore.cpp:484] scanCompleteCallback(): Zigbee network scan complete
[ 33917] [V] [ZigbeeCore.cpp:486] scanCompleteCallback(): Found 2 networks
[ 33917] [V] [ZigbeeCore.cpp:489] scanCompleteCallback(): Network 0: PAN ID: 0xda1b, Permit Joining: No, Extended PAN ID: 40:4c:ca:ff:fe:5a:ef:58, Channel: 1
[ 33918] [V] [ZigbeeCore.cpp:489] scanCompleteCallback(): Network 1: PAN ID: 0x9f02, Permit Joining: No, Extended PAN ID: 05:b6:db:e8:c5:d6:da:6d, Channel: 1

```

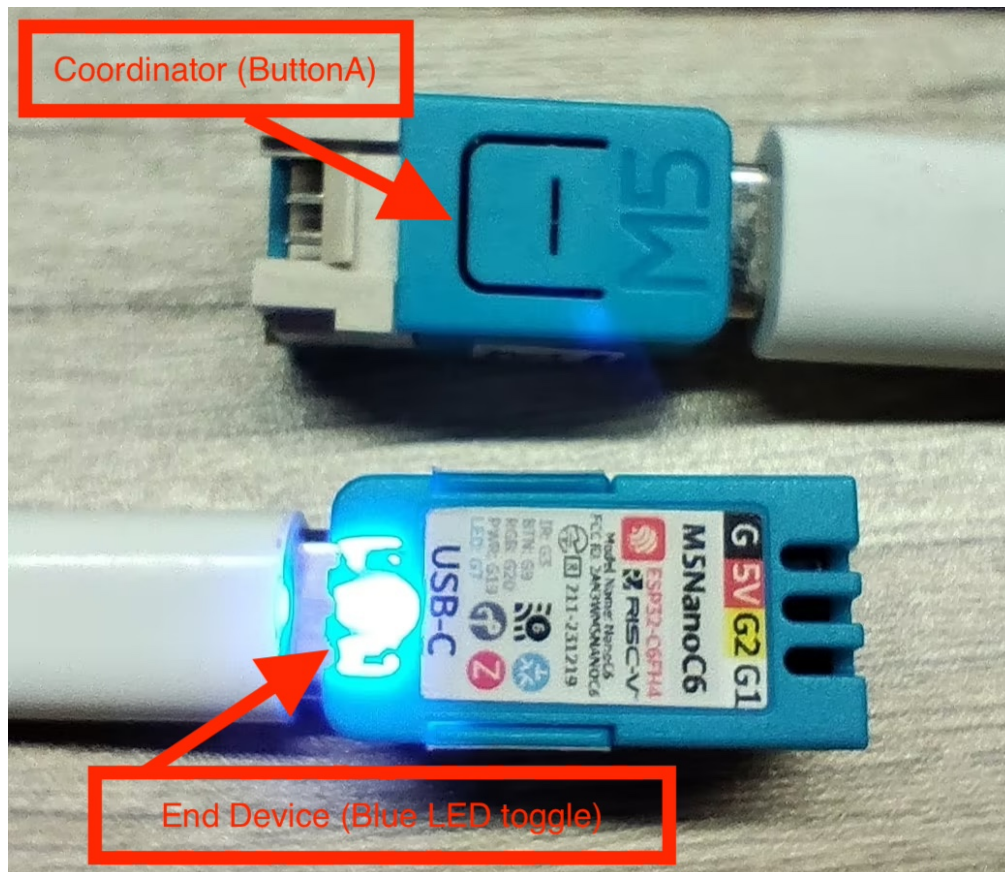
Zigbee Switch（コーディネータ）

このサンプルは、Zigbee コーディネータを設定し、ButtonA を検出してエンドデバイスの青色 LED を制御します。

- Zigbee コーディネータとして動作
- ButtonA の押下を検出
- エンドデバイスの青色 LED の ON/OFF を制御
- 指令情報をシリアルに表示

ハードウェア要件

- NanoC6 ×1 (Zigbee コーディネータ / スイッチ)
- NanoC6 ×1 (Zigbee エンドデバイス / 青色 LED 制御)



設定説明

Arduino IDE ツールメニュー設定:

- 正しい開発ボードを選択: **Tools -> Board: M5NanoC6**
- USB シリアル起動を有効化: **Tools -> USB CDC On Boot: Enabled**
- 消去を有効化: **Tools -> Erase All Flash Before Sketch Upload: Enabled**
- flash サイズを選択: **Tools -> Flash Size: 4MB**
- Zigbee パーティションスキーム: **Tools -> Partition Scheme: Zigbee ZCZR 4MB with spiffs**
- コーディネータモード: **Tools -> Zigbee mode: Zigbee ZCZR (coordinator/router)**
- 正しいシリアルポート: **Tools -> Port**

Board: "M5NanoC6"	>
Port: "/dev/cu.usbmodem141101"	>
Reload Board Data	
Get Board Info	
USB CDC On Boot: "Enabled"	>
CPU Frequency: "160MHz (WiFi)"	>
Core Debug Level: "Verbose"	>
Erase All Flash Before Sketch Upload: "Enabled"	>
Flash Frequency: "80MHz"	>
Flash Mode: "QIO"	>
Flash Size: "4MB (32Mb)"	>
JTAG Adapter: "Disabled"	>
Partition Scheme: "Zigbee ZCZR 4MB with spiffs"	>
Upload Speed: "921600"	>
Zigbee Mode: "Zigbee ZCZR (coordinator/router)"	>

サンプルプログラム

```
1  #ifndef ZIGBEE_MODE_ZCZR
2  #error "Zigbee coordinator mode is not selected in Tools->Zigbee mode"
3  #endif
4
5  #include "Zigbee.h"
6
7  #define SWITCH_ENDPOINT_NUMBER 5
8
9  #define BUTTON_A_PIN 9
10
11 ZigbeeSwitch zbSwitch = ZigbeeSwitch(SWITCH_ENDPOINT_NUMBER);
12
13 bool led_state = false;
14
15 unsigned long last_button_press = 0;
16 const unsigned long debounce_delay = 200;
17
18 void setup() {
19     Serial.begin(115200);
20     delay(1000);
21     Serial.println("=== NanoC6 Zigbee Coordinator (ButtonA Controller) ===");
22
23     pinMode(BUTTON_A_PIN, INPUT_PULLUP);
24     Serial.println("ButtonA (G9/GPIO9) initialized");
25
26     zbSwitch.setManufacturerAndModel("Espressif", "ZigbeeSwitch");
27
28     Zigbee.addEndpoint(&zbSwitch);
29
30     Zigbee.setRebootOpenNetwork(180);
31
32     Serial.println("Starting Zigbee Coordinator...");
33     if (!Zigbee.begin(ZIGBEE_COORDINATOR)) {
34         Serial.println("Zigbee failed to start!");
35         ESP.restart();
36     }
37
38     Serial.println("Zigbee Coordinator started");
39     Serial.println("Waiting for LED device to bind...");
40
41     while (!zbSwitch.bound()) {
42         Serial.print(".");
43         delay(500);
44     }
45
46     Serial.println("\nLED device bound successfully!");
47     Serial.println("\nPress ButtonA to toggle LED");
48 }
49
50 void loop() {
```

```

51   if (digitalRead(BUTTON_A_PIN) == LOW) {
52       unsigned long current_time = millis();
53       if (current_time - last_button_press > debounce_delay) {
54           last_button_press = current_time;
55           led_state = !led_state;
56           if (led_state) {
57               Serial.println("Sending ON command...");
58               zbSwitch.lightOn();
59           } else {
60               Serial.println("Sending OFF command...");
61               zbSwitch.lightOff();
62           }
63           while (digitalRead(BUTTON_A_PIN) == LOW) {
64               delay(10);
65           }
66       }
67   }
68   delay(10);
69 }

```



Zigbee コマンド受信による青色 LED 制御

このサンプルは、Zigbee エンドデバイスを設定し、コーディネータからのコマンドで青色 LED を制御します。

- Zigbee エンドデバイスとして動作
- コーディネータからの指令を受信
- 受信データをシリアルに表示
- NanoC6 上の青色 LED を ON/OFF

設定説明

Arduino IDE ツールメニュー設定:

- 正しい開発ボードを選択: **Tools -> Board: M5NanoC6**
- USB シリアル起動を有効化: **Tools -> USB CDC On Boot: Enabled**
- 消去を有効化: **Tools -> Erase All Flash Before Sketch Upload: Enabled**
- flash サイズを選択: **Tools -> Flash Size: 4MB**
- Zigbee パーティションスキーム: **Tools -> Partition Scheme: Zigbee ZCZR 4MB with spiffs**
- エンドデバイスモード: **Tools -> Zigbee mode: Zigbee ED (end device)**

◦ 正しいシリアルポート: Tools -> Port

Board: "M5NanoC6"	>
Port: "/dev/cu.usbmodem14201"	>
Reload Board Data	
Get Board Info	
USB CDC On Boot: "Enabled"	>
CPU Frequency: "160MHz (WiFi)"	>
Core Debug Level: "Verbose"	>
Erase All Flash Before Sketch Upload: "Enabled"	>
Flash Frequency: "80MHz"	>
Flash Mode: "QIO"	>
Flash Size: "4MB (32Mb)"	>
JTAG Adapter: "Disabled"	>
Partition Scheme: "Zigbee ZCZR 4MB with spiffs"	>
Upload Speed: "921600"	>
Zigbee Mode: "Zigbee ED (end device)"	>

サンプルプログラム

```
1  #ifndef ZIGBEE_MODE_ED
2  #error "Zigbee end device mode is not selected in Tools->Zigbee mode"
3  #endif
4
5  #include "Zigbee.h"
6
7  #define LIGHT_ENDPOINT_NUMBER 10
8
9  #define BLUE_LED_PIN 7
10
11  ZigbeeLight zbLight = ZigbeeLight(LIGHT_ENDPOINT_NUMBER);
12
13  void setLED(bool state) {
14      digitalWrite(BLUE_LED_PIN, state ? HIGH : LOW);
15  }
16
17  void setup() {
18      Serial.begin(115200);
19      delay(1000);
20      Serial.println("=== NanoC6 Zigbee End Device (LED) ===");
21
22      pinMode(BLUE_LED_PIN, OUTPUT);
23      digitalWrite(BLUE_LED_PIN, HIGH);
24      Serial.println("Blue LED (G4/GPIO4) initialized");
25
26      zbLight.setManufacturerAndModel("Espressif", "ZigbeeLight");
27      zbLight.onLightChange(setLED);
28      Zigbee.addEndpoint(&zbLight);
29
30      Serial.println("Starting Zigbee End Device...");
31      if (!Zigbee.begin()) {
32          Serial.println("Zigbee failed to start!");
33          delay(5000);
34          ESP.restart();
35      }
36
37      Serial.println("Zigbee End Device started");
38      Serial.println("Searching for Zigbee network...");
39
40      while (!Zigbee.connected()) {
41          Serial.print(".");
42          delay(1000);
43      }
44
45      Serial.println("\nConnected to Zigbee network!");
46      Serial.println("Waiting for commands from Coordinator...");
47  }
48
49  void loop() {
50
```

Output Serial Monitor x

Message (Enter to send message to 'M5NanoC6' on '/dev/cu.usbmodem14201')

New Line 115200 baud

===== After Setup End =====
[116971] [V] [ZigbeeHandlers.cpp:77] zb_attribute_set_handler(): Received message: endpoint(10), cluster(0x6), attribute(0x0), data size(1)
[171720] [V] [ZigbeeHandlers.cpp:77] zb_attribute_set_handler(): Received message: endpoint(10), cluster(0x6), attribute(0x0), data size(1)
[287266] [V] [ZigbeeHandlers.cpp:77] zb_attribute_set_handler(): Received message: endpoint(10), cluster(0x6), attribute(0x0), data size(1)
[287566] [V] [ZigbeeHandlers.cpp:77] zb_attribute_set_handler(): Received message: endpoint(10), cluster(0x6), attribute(0x0), data size(1)
[287776] [V] [ZigbeeHandlers.cpp:77] zb_attribute_set_handler(): Received message: endpoint(10), cluster(0x6), attribute(0x0), data size(1)
[287982] [V] [ZigbeeHandlers.cpp:77] zb_attribute_set_handler(): Received message: endpoint(10), cluster(0x6), attribute(0x0), data size(1)
[288192] [V] [ZigbeeHandlers.cpp:77] zb_attribute_set_handler(): Received message: endpoint(10), cluster(0x6), attribute(0x0), data size(1)
[288482] [V] [ZigbeeHandlers.cpp:77] zb_attribute_set_handler(): Received message: endpoint(10), cluster(0x6), attribute(0x0), data size(1)
[288692] [V] [ZigbeeHandlers.cpp:77] zb_attribute_set_handler(): Received message: endpoint(10), cluster(0x6), attribute(0x0), data size(1)
[288992] [V] [ZigbeeHandlers.cpp:77] zb_attribute_set_handler(): Received message: endpoint(10), cluster(0x6), attribute(0x0), data size(1)
[289202] [V] [ZigbeeHandlers.cpp:77] zb_attribute_set_handler(): Received message: endpoint(10), cluster(0x6), attribute(0x0), data size(1)