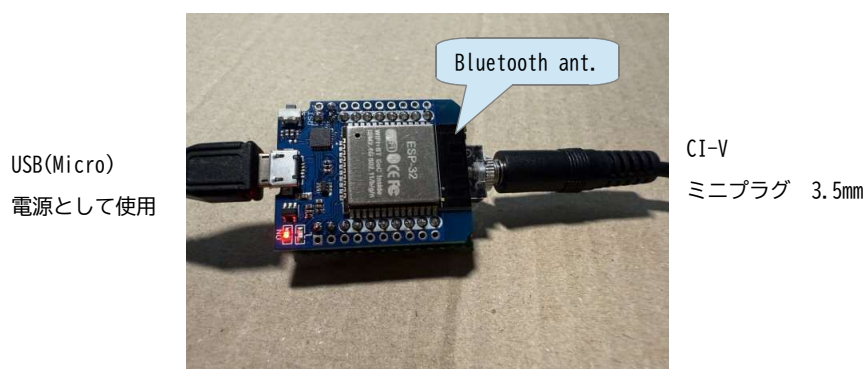


IC-705にはハードウェア的なCI-Vインターフェースがありませんが、Bluetoothを経由してサポートされています。
このインターフェースは、Bluetoothを経由してCI-Vインターフェースを提供するものです。

1. 外観と各部名称



2. IC-705の設定

インターフェースに電源を供給した状態で実施

- ① MENU ⇒ SET ⇒ 外部端子 ⇒ CI-V ⇒ CI-Vトランシーブ (ON)
- ② MENU ⇒ Bluetooth設定 ⇒ Bluetooth (ON)
- ③ MENU ⇒ Bluetooth設定 ⇒ Bluetoothユニット初期化
- ④ MENU ⇒ Bluetooth設定 ⇒ ペアリング/接続 ⇒ 機器検索 ⇒ データ端末検索

ここで " ic705_CI-V I/f" が検出されることを確認

⇒ ic705_CI-V i/f を選択 ⇒ 接続しますか? (はい) を選ぶ

⇒ ic705_CI-V I/fに次のパスキーが表示されている事を確認して下さい。
パスキー: ****

パスキーは無視して 「確認」 を押す

⇒ 再度 ic705_CI-V i/f を選択 ⇒ 接続しますか? (はい) を選択

ic705_CI-V I/f に接続しました。 と出ればOK

ここまでは最初の
一回だけ設定すればOK

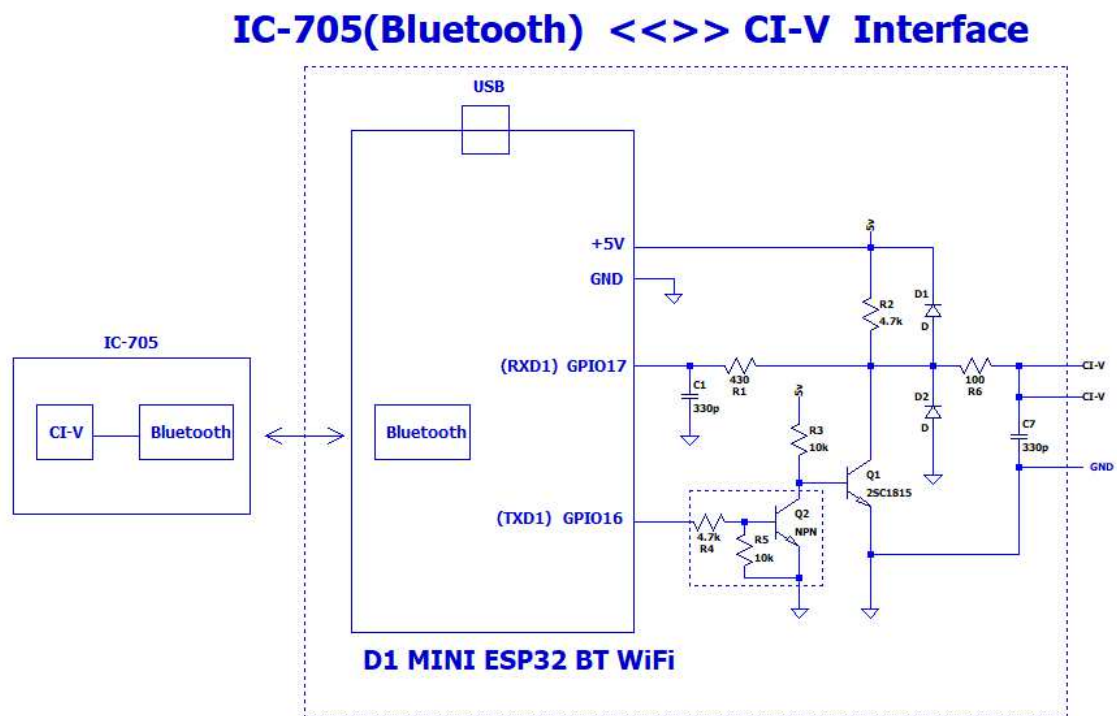
3. 接続機器の設定

接続機器のマニュアルに沿って設定

CI-Vの通信速度は、 9800bps に設定 (他の速度には対応していません)

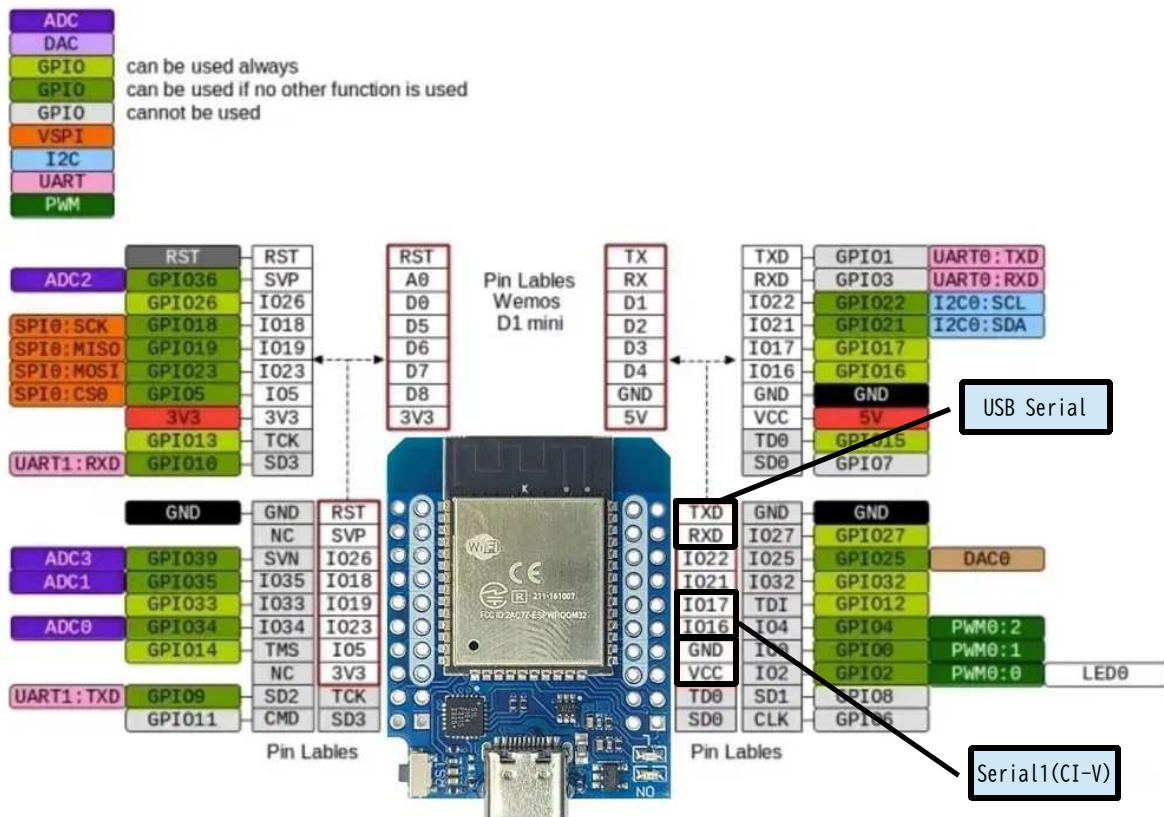
4. 資料

4-1. 回路図



4-2. ピンレイアウト

Pin Definitions



4-3. プログラム

```
/*
IC-705(Bluetooth) <-> C-V interface
20251107 JF3IPR/K.Ohmae

Board : D1 MINI ESP32 BT WiFi (ESP32 Dev Module)

*****
Port assignment

RST          TXD0    usb (programming/monitor/power supply)
SVP          RXD0    usb (programming/monitor/power supply)
26 IO26 ---      22 IO22 ---
18 IO18 ---      21 IO21 ---
19 IO19 ---      17 IO17 RXD(serial1)
23 IO23 ---      16 IO16 TXD(serial1)
5 IO5 ---        GND
3V3          VCC
TCK          TD0
SD3          SDO
*****
*/

#define PIN_TX1 16
#define PIN_RX1 17

#include <uTimerLib.h>
#include "BluetoothSerial.h"
BluetoothSerial SerialBT;

int RcvBT;          // Receive data from Bluetooth
int RcvCIVeb;       // Dummy read data when transmitted from Bluetooth to CI-V
int RcvCIV;         // Receive data from CI-V
int Frcv;           // Bluetooth > CI-V flag
int pass_f;         // 0.5ms interrupt flag

//*****
// Timer interrupt 0.5ms

void myTimer() {
  pass_f = 1;
}

//*****
void setup() {
  Serial.begin(9600);          //Serial communication begin to PC
  Serial1.begin(9600, SERIAL_8N1, PIN_RX1, PIN_TX1); //Serial1 communication begin to CI-V
  SerialBT.begin("ic705_CI-V i/f"); //Bluetooth communication begin
  TimerLib.setInterval_us(myTimer, 500); //0.5ms interrupt
  Serial.setRxBufferSize(1024);

  Frcv = 0;
  pass_f = 0;
}

//*****
void loop() {
  if (pass_f == 1) {          //pass_f 0.5ms flag
    pass_f = 0;

    //----- CI-V >> Bluetooth -----
    if (Frcv == 0){
      if(Serial1.available()){
        RcvCIV = Serial1.read();
        SerialBT.write(RcvCIV);
      }
      Serial.println(RcvCIV, HEX);          //monitor
    }

    //----- Bluetooth >> CI-V -----
    if( SerialBT.available()){
      Frcv = 1;
      RcvBT = SerialBT.read();          //Read from Bluetooth
      Serial1.write(RcvBT);             //Write to Serial1(CI-V)

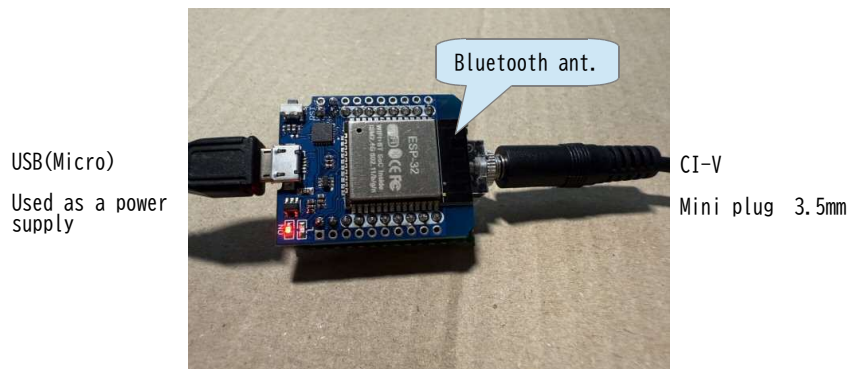
      while (Serial1.available()==0){
      }
      while (Serial1.available()){
        RcvCIVeb = Serial1.read();          //Dummy read from Serial1 buffer
      }
      Frcv = 0;
    }
  }
}
```

IC-705(Bluetooth) <> CI-V interface

2025/11/14
jf3ipr/K.Ohmae

The IC-705 lacks a physical CI-V interface, but it is supported through Bluetooth. This interface provides a CI-V connection via Bluetooth.

1. Appearance and Parts Names



2. IC-705 Settings

- ① MENU ⇒ SET ⇒ Connectors ⇒ CI-V ⇒ CI-V Transceive (ON)
- ② MENU ⇒ Bluetooth Set ⇒ Bluetooth (ON)
- ③ MENU ⇒ Bluetooth Set ⇒ Initialize Bluetooth Device
- ④ MENU ⇒ Bluetooth設定 ⇒ Pairing/Connect ⇒ Device Search ⇒ Search Data Device
Verify that 'ic705_CI-V I/f' appears here.
⇒ Select 'ic705_CI-V i/f' ⇒ Connect? (Yes)
⇒ Confirm the following passkey shown on ic705_CI-V i/f.
Passkey : *****
No need to enter the passkey and select 'Confirm'
⇒ Select 'ic705_CI-V i/f' again ⇒ Connect? (Yes)
Connected

These settings
only need to be
configured once
at the beginning.

3. Connected Device Setting

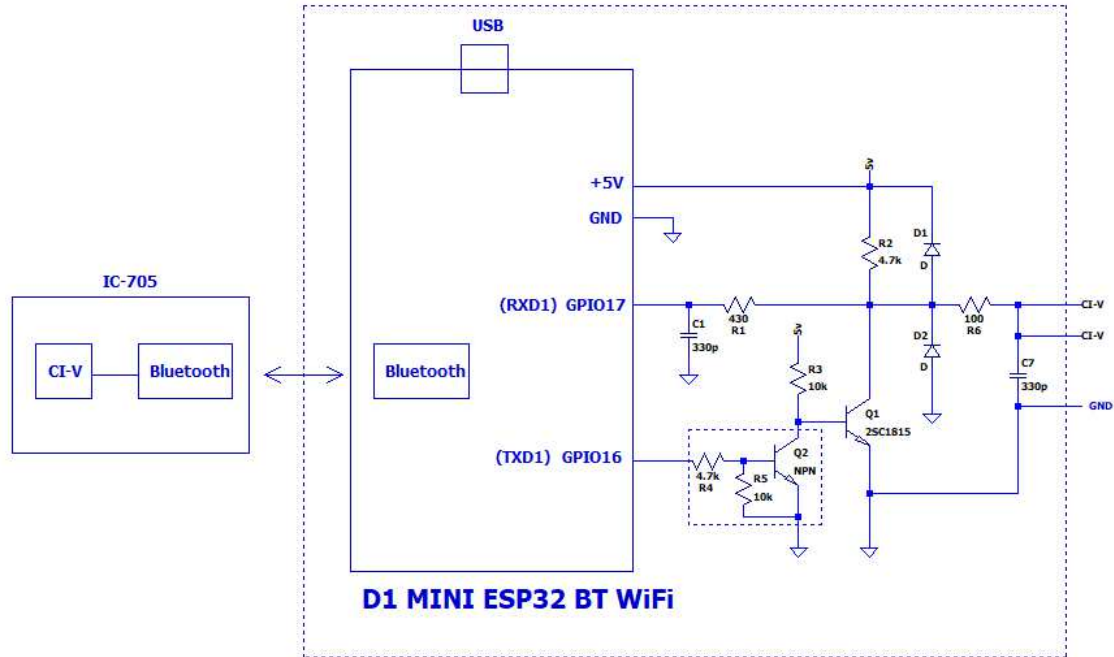
Configure according to the manual of the connected device.

Set the CI-V communication speed to 9800 bps (other speeds are not supported).

4. Appendix

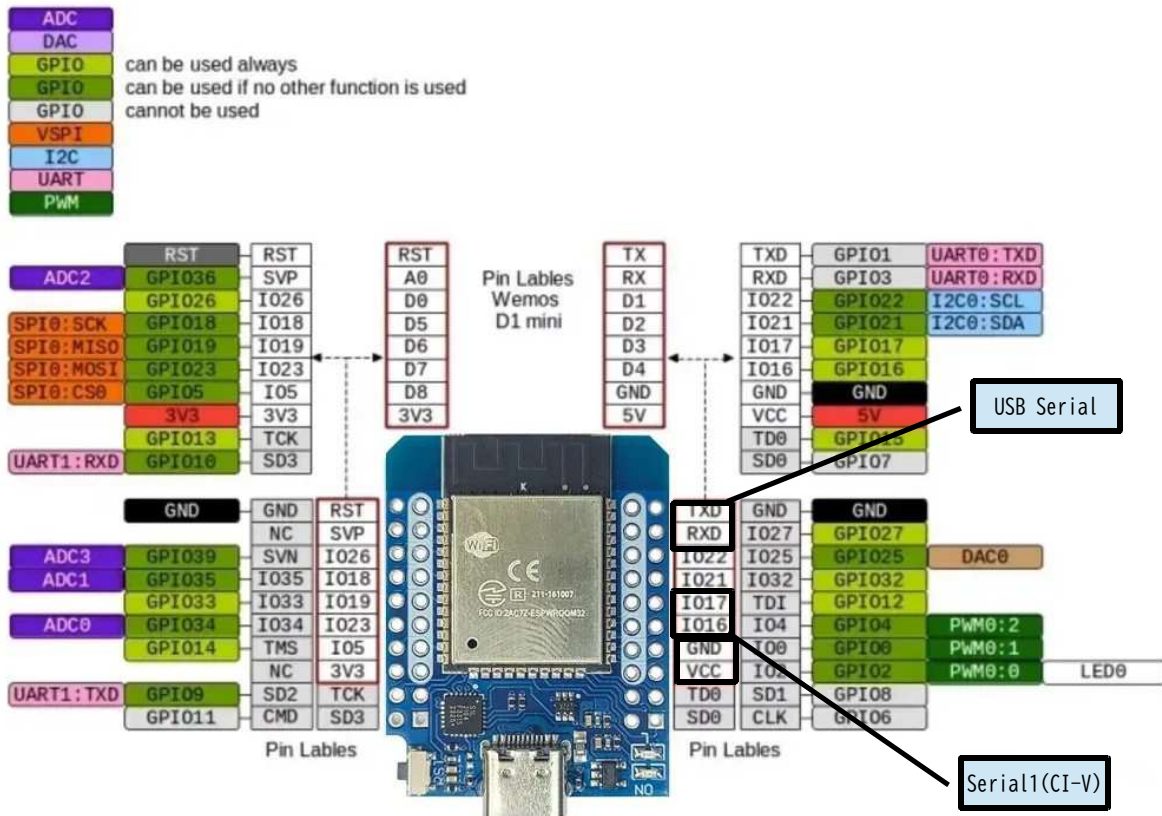
4-1. Circuit Diagram

IC-705(Bluetooth) <<>> CI-V Interface



4-2. Pin assignment

Pin Definitions



4-3. Program list

```

/*
IC-705(Bluetooth) <> C-V interface
20251107 JF3IPR/K.Ohmae

Board : D1 MINI ESP32 BT WiFi (ESP32 Dev Module)

*****
Port assignment

RST          TXD0    usb (programming/monitor/power supply)
SVP          RXD0    usb (programming/monitor/power supply)
26 IO26      ---     22 IO22      ---
18 IO18      ---     21 IO21      ---
19 IO19      ---     17 IO17      RXD(Serial1)
23 IO23      ---     16 IO16      TXD(Serial1)
5 IO5        ---     GND
3V3          ---     VCC
TCK          ---     TD0
SD3          ---     SD0
*****
*/

#define PIN_TX1 16
#define PIN_RX1 17

#include <uTimerLib.h>
#include "BluetoothSerial.h"
BluetoothSerial SerialBT;

int RcvBT;          // Receive data from Bluetooth
int RcvCIVeb;       // Dummy read data when transmitted from Bluetooth to CI-V
int RcvCIV;         // Receive data from CI-V
int Frcv;           // Bluetooth > CI-V flag
int pass_f;         // 0.5ms interrupt flag

//*****
//          Timer interrupt 0.5ms

void myTimer() {
    pass_f = 1;
}

//*****
void setup() {
    Serial.begin(9600);
    Serial1.begin(9600, SERIAL_8N1, PIN_RX1, PIN_TX1); //Serial communication begin to PC
    SerialBT.begin("ic705 CI-V i/f"); //Serial1 communication begin to CI-V
    TimerLib.setInterval_us(myTimer, 500); //Bluetooth communication begin
    Serial.setRxBufferSize(1024); //0.5ms interrupt

    Frcv = 0;
    pass_f = 0;
}

//*****
void loop() {
    if (pass_f == 1) { //pass_f 0.5ms flag
        pass_f = 0;

//----- CI-V >> Bluetooth -----
        if (Frcv == 0){
            if(Serial1.available()){
                RcvCIV = Serial1.read();
                SerialBT.write(RcvCIV);
            }
            Serial.println(RcvCIV, HEX); //monitor
        }

//----- Bluetooth >> CI-V -----
        if( SerialBT.available()){
            Frcv = 1;
            RcvBT = SerialBT.read(); //Read from Bluetooth
            Serial1.write(RcvBT); //Write to Serial1(CI-V)

            while (Serial1.available()==0){
            }
            while (Serial1.available()){
                RcvCIVeb = Serial1.read(); //Dummy read from Serial1 buffer
            }
            Frcv = 0;
        }
    }
}

```