

Drone2

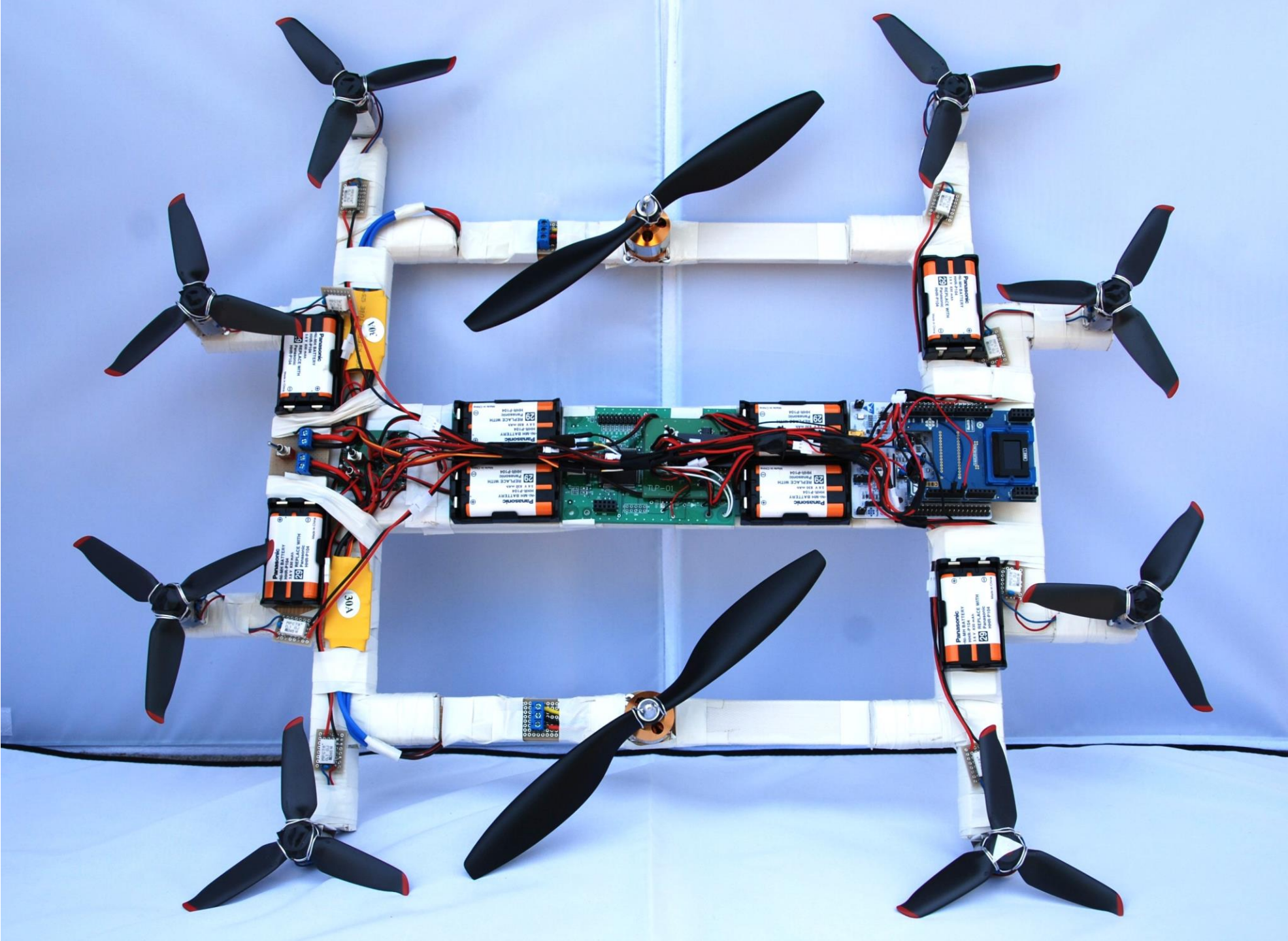
D-Slave9-8

Music Drone Project

製作開始 2023/02/22

アップョミュージックスタジオ事務所

2024/07/29 h.



目標

機体の軽量化

UNOを使わないでブラシレスモーター
を駆動する。(軽量化)

揚力を上げる

距離センサーを付ける

部品、材料

プロペラ	大	2	モーター	A2212/13T 1000KV
プロペラ	中	8	モーター	FA-130RA Mabuchi motor

LPC基盤

NUCLEO-F401RE

AE-Motor8830 x 8

TWELITE

電池ケース、 Ni-MH Battery

L型金具、 アルミ線

本体

木材、 布テープ、 ビニールテープ、 他

利用CPUとインターフェース

CPU基板、 NXP LPC1788 ,ARM ,Cortex-M3

開発環境 IAR Embedded Workbench for Arm

基板製造 Appyo Music Studio

利用 Interface

I2c0 BNO055

UART0 NUCLEO-F401RE との通信

UART1 TWELITE uart (Music Server との通信)

CPU基板、 NUCLEO-F401RE

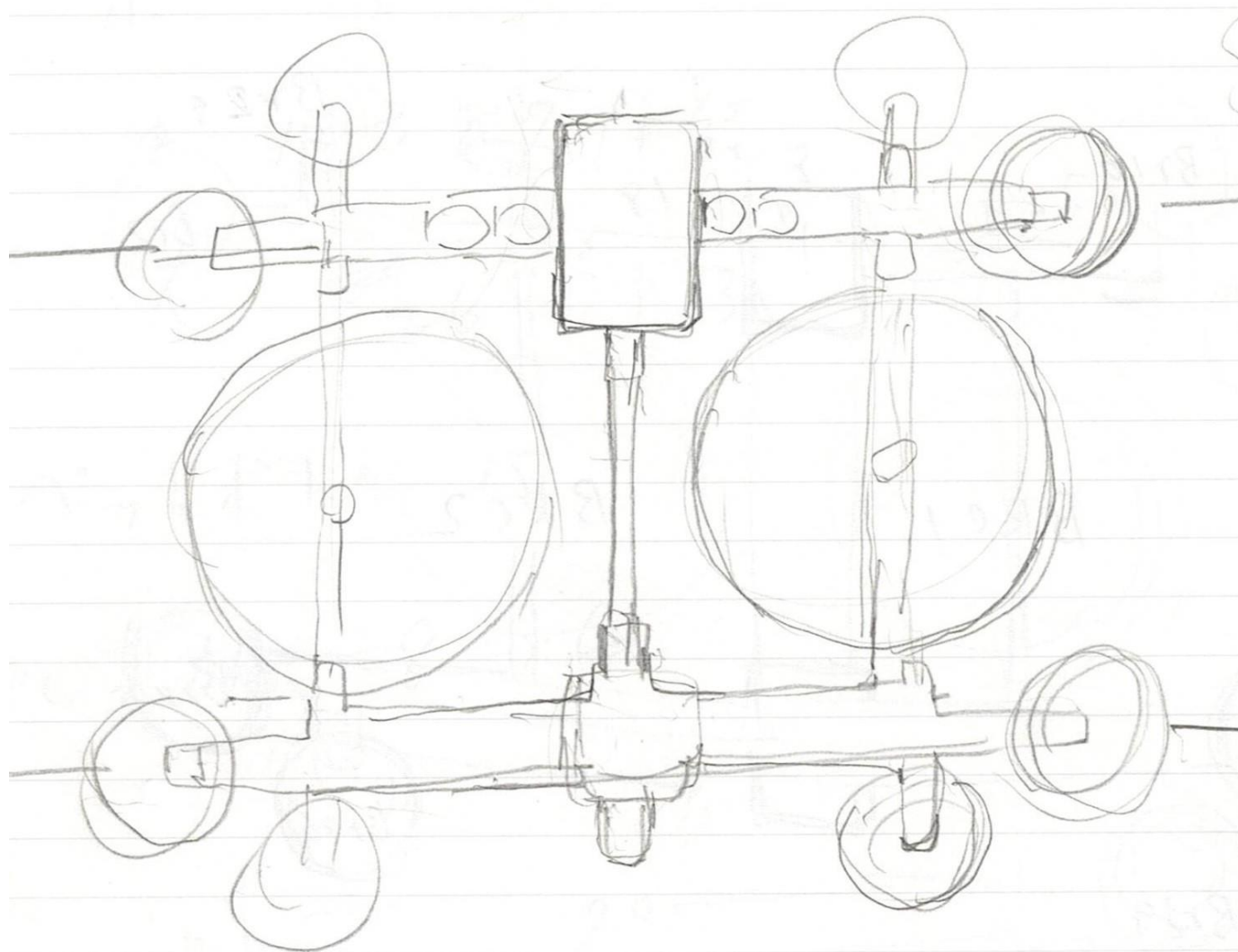
開発環境 STM32CubeMX , IAR Embedded Workbench for Arm

利用 Interface

I2C 距離センサー

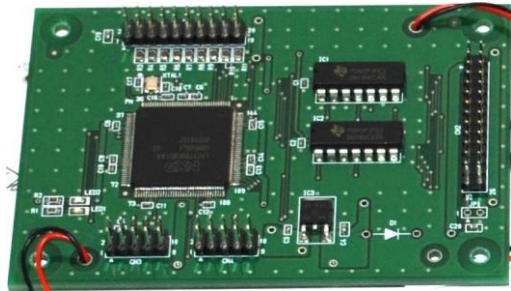
UART1 LPC1788 との通信

機体の軽量化



LPC1788でブラシレスモーターを駆動する。

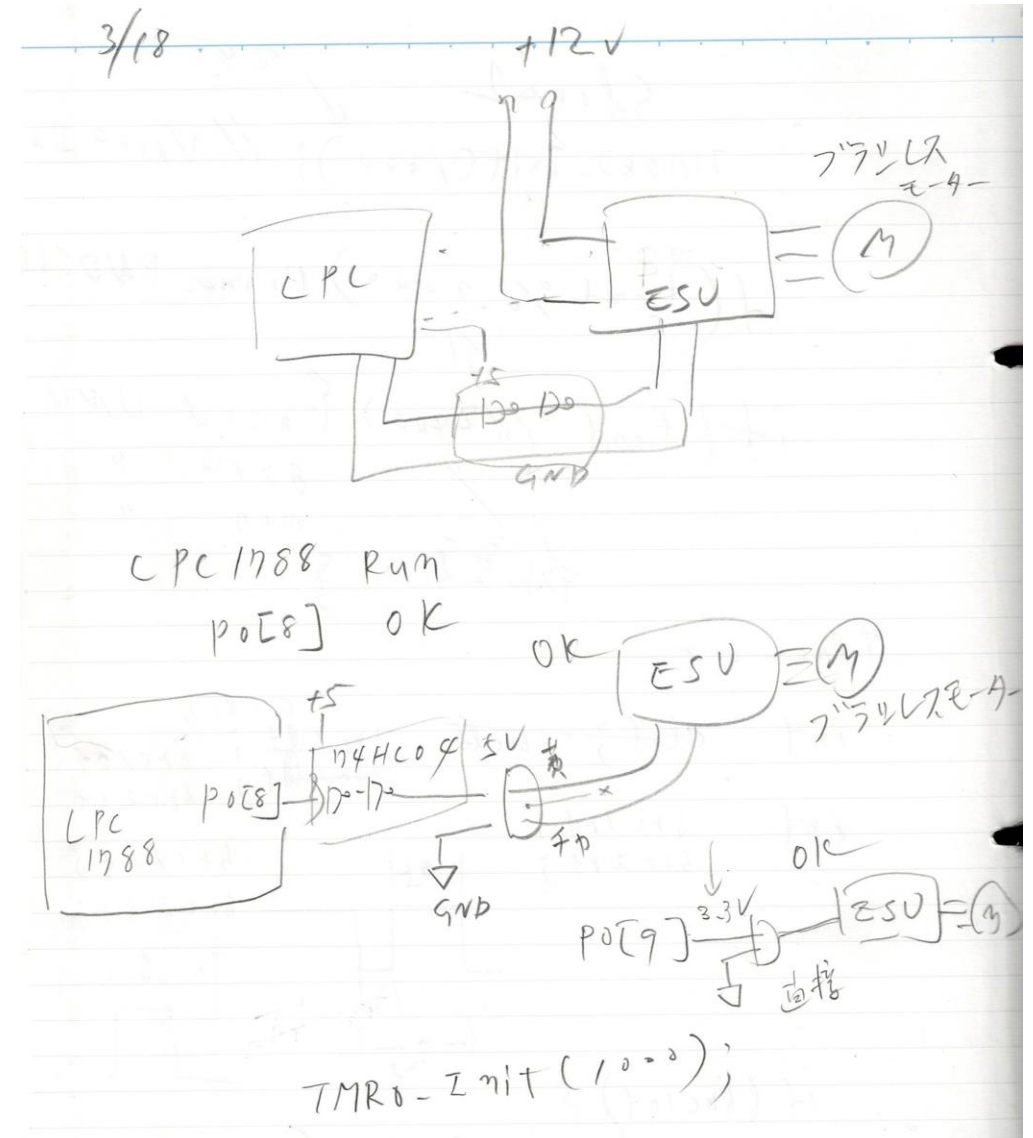
LPC1788

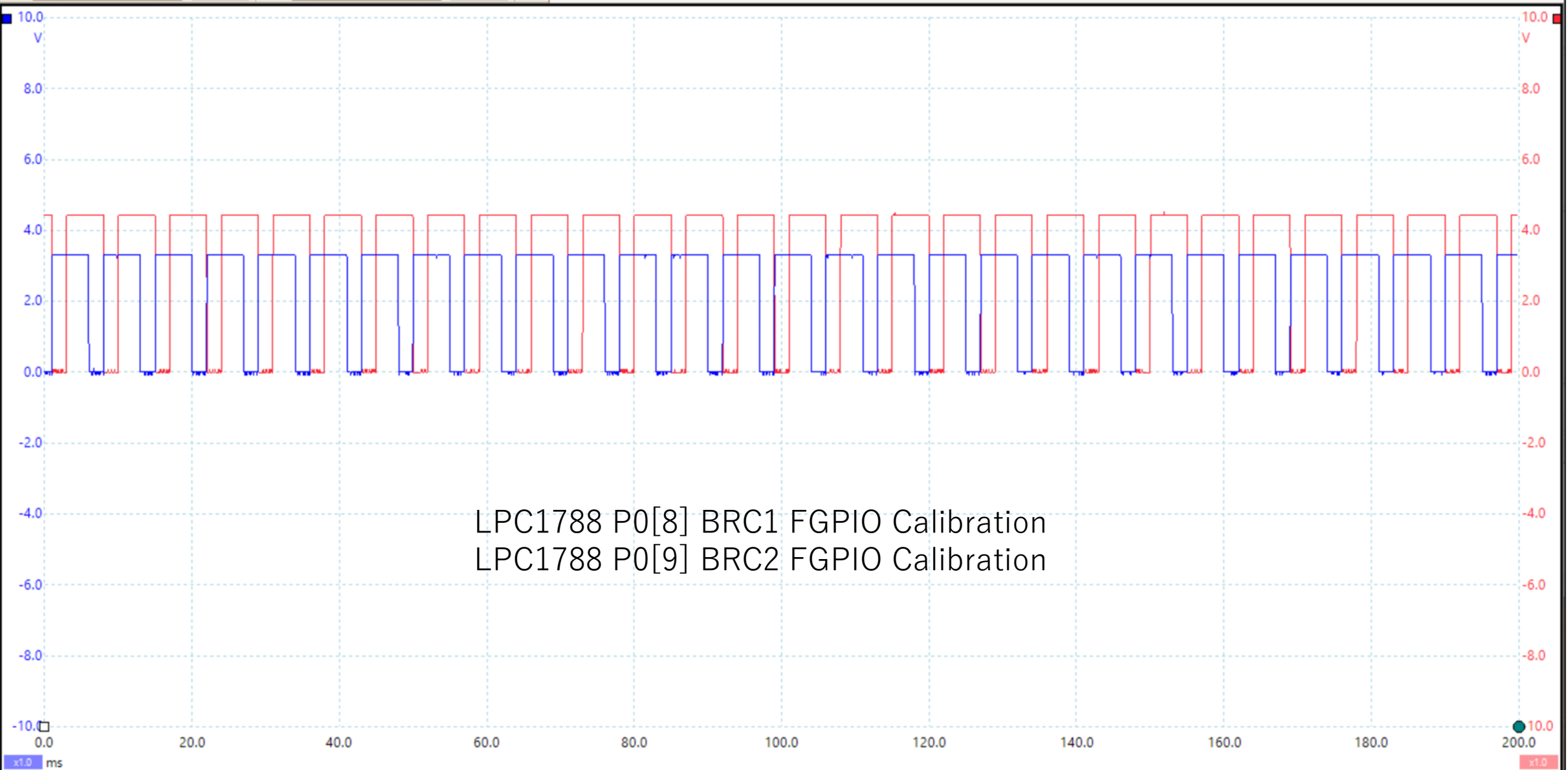


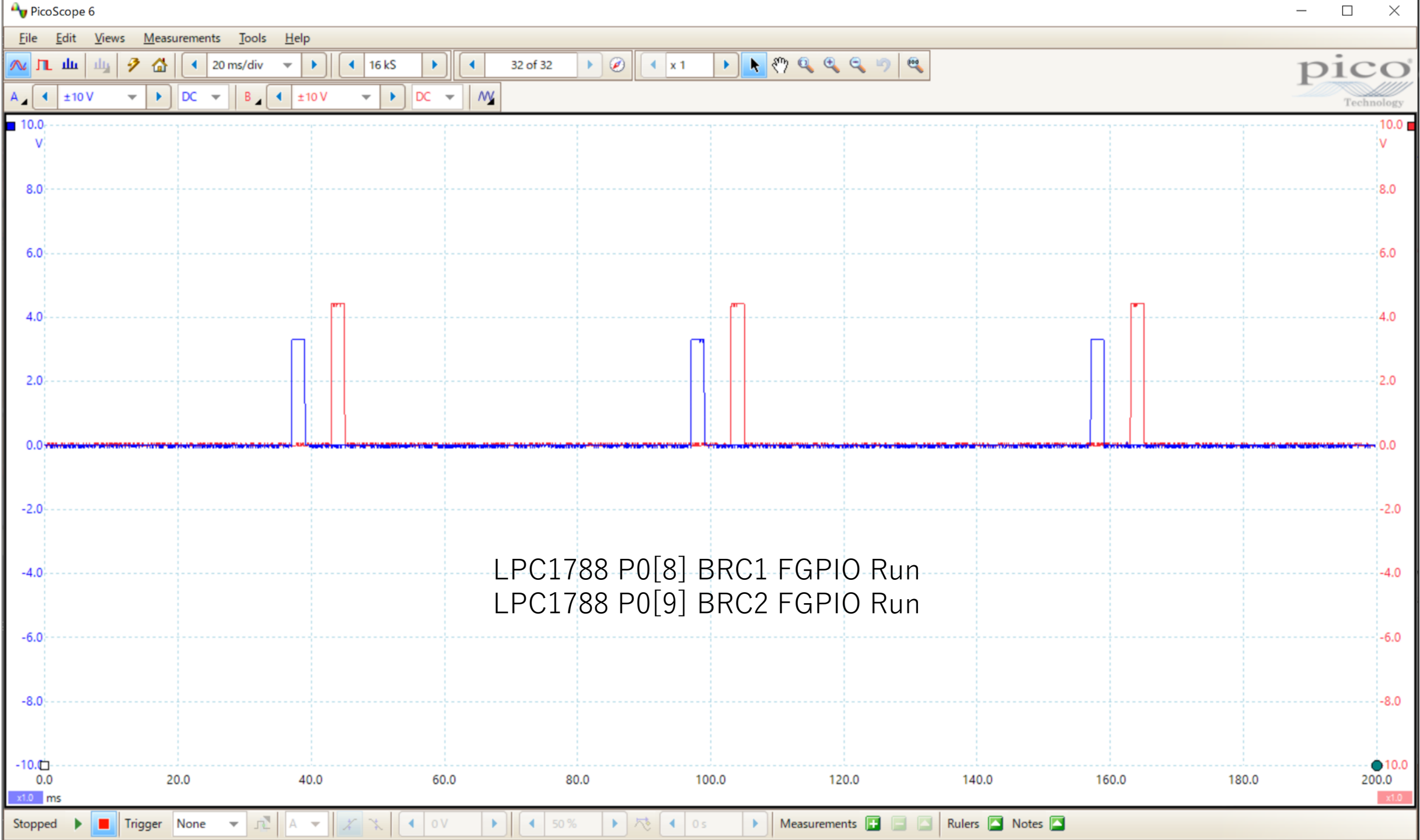
ブラシレスモーター



ESU







```

// GPIO IOCON
#define IOCON_PO_4      ((volatile unsigned int *) (0x4002C010)) // Gstop=1
#define IOCON_PO_5      ((volatile unsigned int *) (0x4002C014)) // Pose
#define IOCON_PO_6      ((volatile unsigned int *) (0x4002C018)) // Gstop=4
#define IOCON_PO_7      ((volatile unsigned int *) (0x4002C01C)) // Stop
#define IOCON_PO_8      ((volatile unsigned int *) (0x4002C020))
#define IOCON_PO_9      ((volatile unsigned int *) (0x4002C024))
#define IOCON_PO_14     ((volatile unsigned int *) (0x4002C038)) // GPIO
#define IOCON_P1_18     ((volatile unsigned int *) (0x4002C0C8)) // LED2

```

P0[8] P0[9] FGPIO

```

*IOCON_PO_8 = 0x00; // Set P0[8] -> GPIO BRC1
*IOCON_PO_9 = 0x00; // Set P0[9] -> GPIO BRC2
*IOCON_PO_10 = 0x00; // Tx2 D-Slave8 Tbr7
*IOCON_PO_14 = 0x00; // Set P0[14] -> GPIO
*IOCON_PO_19 = 0x30; // SDA1 -> GPIO Tbr5
*IOCON_PO_20 = 0x30; // SCL1 -> GPIO Tbr6
*IOCON_PO_22 = 0x30; // Tx4 -> GPIO test NG
//*IOCON_PO_25 = 0x30; // Tx3 -> GPIO Thr8

```

Main()

```

//////////////// BRC controle //////////////////
brc1cf = 0;
GPIO_PO_8_High(); // test
GPIO_PO_9_High(); // test
//GPIO_PO_14_High(); // Br SW ON
brcnt1 = 0;
brcnt2 = 0;
//brc1cf = 1; // BRC1 calibration ON
//for( i=0; i < 200000000; i++); //delay
//brc1cf = 0; // BRC1 calibration OFF
brc1rf = 2; // BRC1 Run
for( i=0; i < 60000; i++); //delay
brc2rf = 2; // BRC2 Run
Tbr_All_tst_5(); // Tbr All_5 test ok
for( i=0; i < 100000000; i++); //delay
brc1rf = 0; // BRC1 Run off
brc2rf = 0; // BRC2 Run off
//GPIO_PO_14_Low(); // Br SW OFF
GPIO_PO_8_Low();
GPIO_PO_9_Low();

```

```

/*****
* Function Name: TMRO_IRQHandler
* Parameters: none
*
* Return: none
*
* Description: Timer 0 interrupt handler
*
*****/
void TMRO_IRQHandler (void)
{
    if(brclcf) {
        if(brcnt1 >= 2) GPIO_P0_8_High();
        if(brcnt1 >= 7) {
            GPIO_P0_8_Low();
            brcnt1 = 0;
        }
        brcnt1++;
    }

    if(brclrf) {
        if(brcnt1 >= 58) GPIO_P0_8_High();
        //if(brcnt1 >= (60 - brclrf)) GPIO_P0_8_High();
        if(brcnt1 >= 60) {
            GPIO_P0_8_Low();
            brcnt1 = 0;
        }
        brcnt1++;
    }

    if(brc2cf) {
        if(brcnt2 >= 2) GPIO_P0_9_High();
        if(brcnt2 >= 7) {
            GPIO_P0_9_Low();
            brcnt2 = 0;
        }
        brcnt2++;
    }

    if(brc2rf) {
        if(brcnt2 >= 58) GPIO_P0_9_High();
        //if(brcnt2 >= (60 - brc2rf)) GPIO_P0_9_High();
        if(brcnt2 >= 60) {
            GPIO_P0_9_Low();
            brcnt2 = 0;
        }
        brcnt2++;
    }
}

```

Timer IRQ

揚力を上げる

DCモーターをRE-140RA から FA-130RA に変更（回転数を上げる）
DCモーターを追加 FA-130RA を 8個にする。

テスト1

SSRを使ってモーターを直接駆動してみる。

テスト1

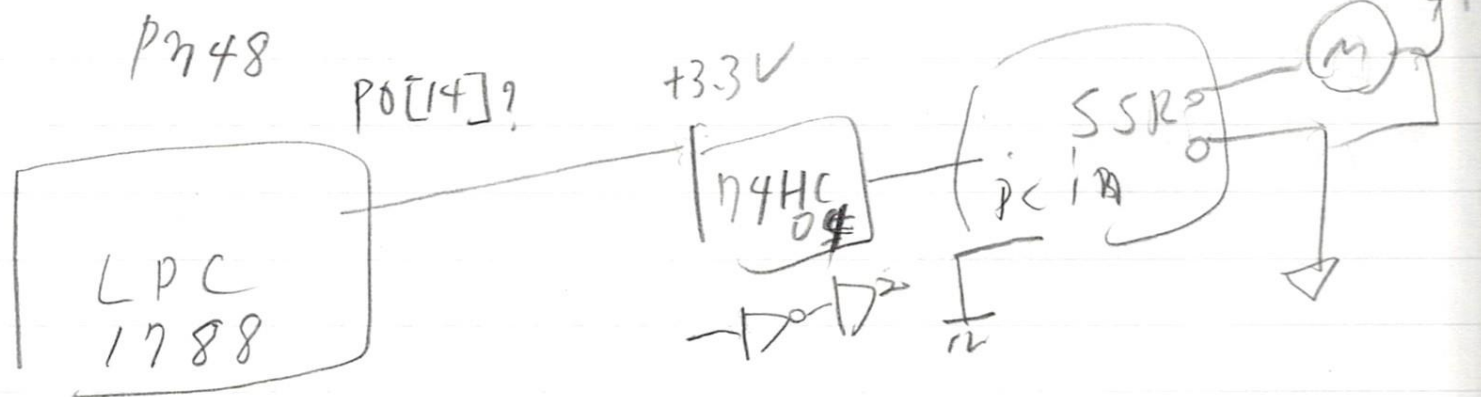
SSRキットを使ってモーターを直接駆動してみる。

3/9

x4

SSRキット 4回付 2個作成

DC 3.6V

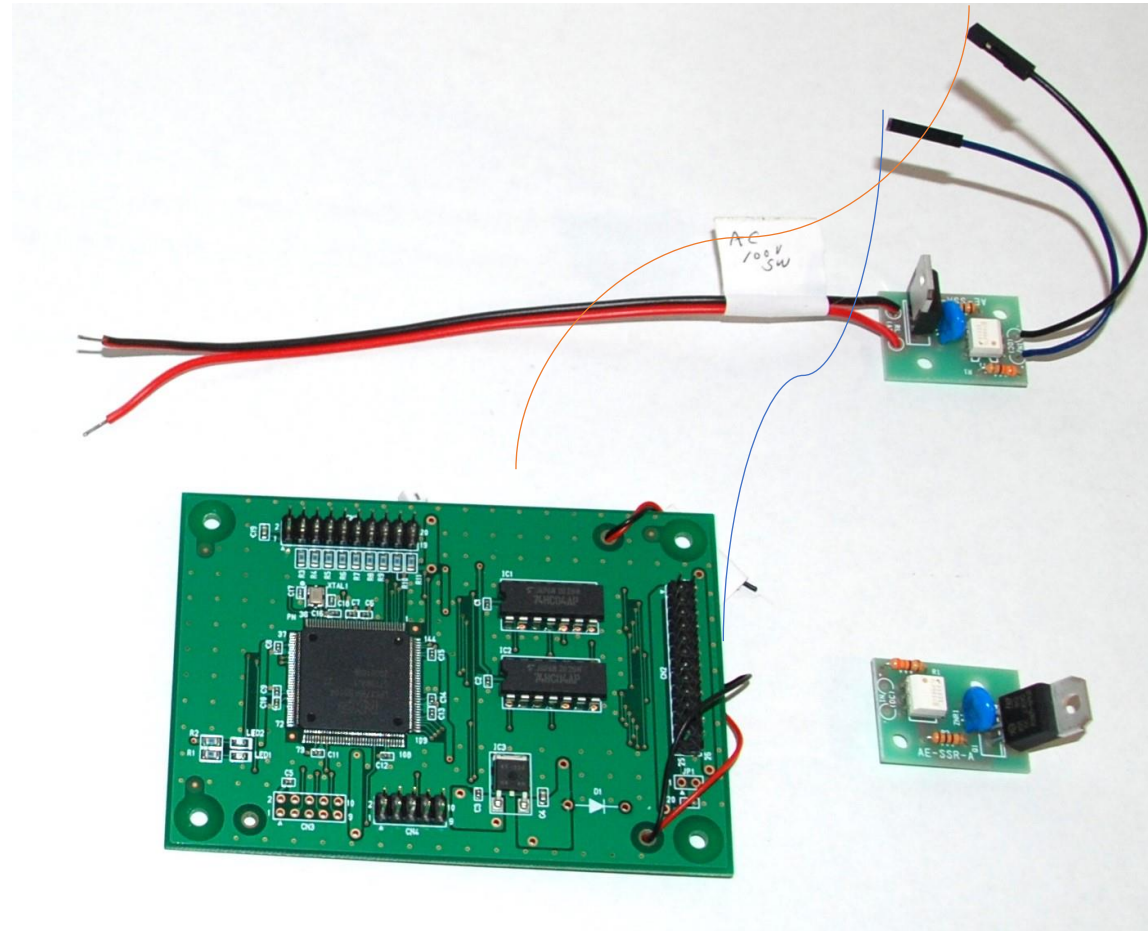


AC100/25A ゼロクロスSSR

テスト1

SSRキットを使ってモーターを直接駆動してみる。

AC 100V 接続



テスト1

SSRキットを使ってモーターを直接駆動してみる。

(今回のキットTLP561J利用では、DCでの利用は不可、AC接続はOK)

No. _____

Date 2023. 3/13 slave 2 SSRテスト BTA24-600CW

SSR DCでの利用不可 AC利用

SSRの411A DCはONのみ ON → OK OFF → NG

TLP 241A 100V ¥140 利用?

東芝フォトリレー 11-6111L

⇒ +5V NG 0V0.5447?

揚力を上げる

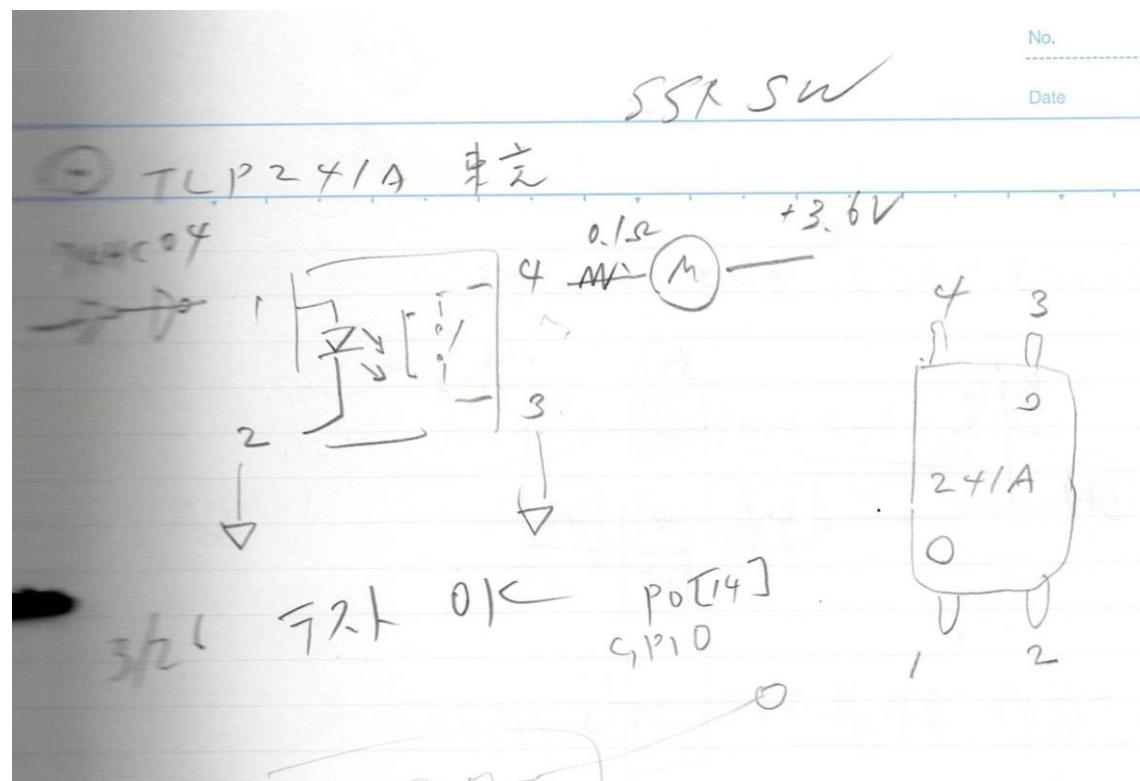
DCモーターをRE-140RA から FA-130RA に変更（回転数を上げる）

DCモーターを追加 FA-130RA を 8個にする。

テスト2

SSRの代わりに東芝TLP241Aフォトリレーを使ってモーターを直接駆動してみる。

LPC1788 GPIO は、P0[14]を利用する。



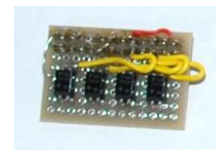
揚力を上げる

テスト2

SSRの代わりに東芝TLP241Aフォトリレーを使ってモーターを直接駆動してみる。

LPC1788 GPIO は、P0[14]を利用する。

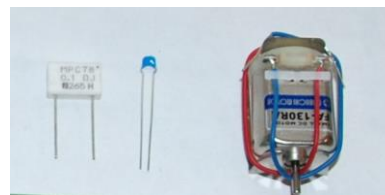
（回転数はパルス制御でコントロール）



テスト結果

Mini-motorは、DC3.6V接続でOK(かなり高速で回転する)

FA-130REは、3.6V接続で抵抗0.1Ω入れないとICが破壊する場合があります。また、ノイズ除去のため0.1μFのコンデンサーを入れる。少し熱を出すのがOKとする。



テスト

LPC1788 PN48(P0[14]) GPIO を利用

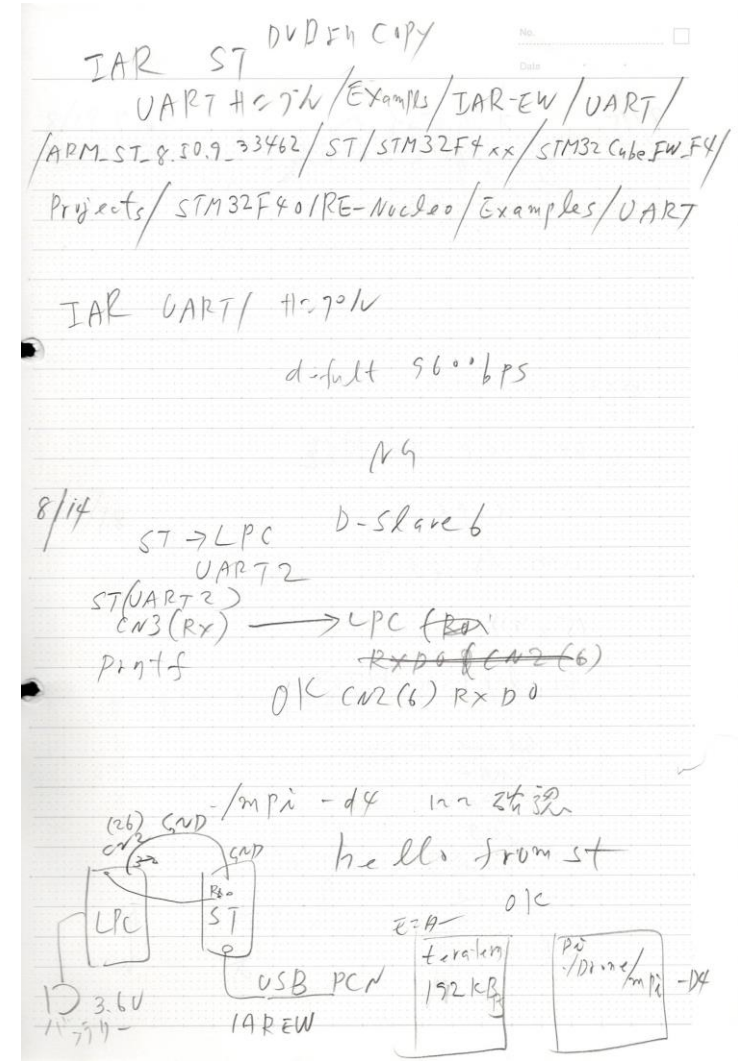
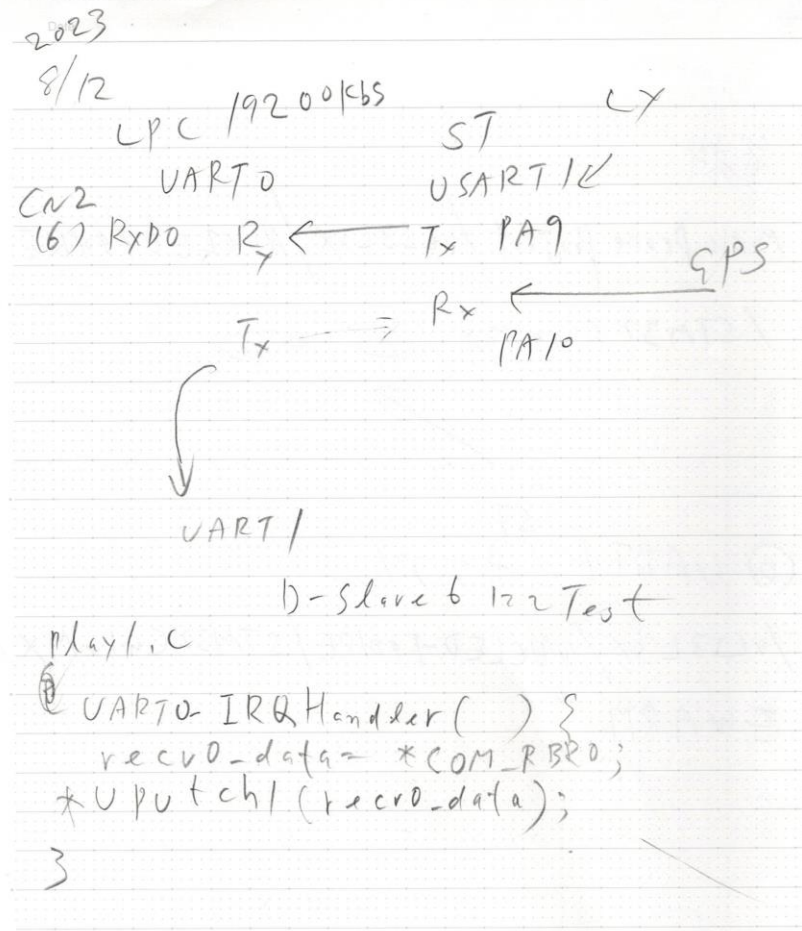
```
*IOCON_P0_14 = 0x00;           // Set P0[14] -> GPIO

*FGPIO_FIODIR = 0x00004300;    // GPIO P0[8] P0[9] P0[14] OutPut
// *FGPIO_FIODIR = 0x00000100;  // GPIO P0[8] OutPut
// *FGPIO_FIODIR = 0x00000200;  // GPIO P0[9] OutPut
// *FGPIO_FIOMASK = 0x00004000;  // P0[14] Mask test

LED_OFF(); // test
// LED_ON();
GPIO_P0_8_High(); // test
GPIO_P0_9_High(); // test
GPIO_P0_14_High();
for ( i=0; i < 30000000; i++); //delay
//GPIO_P0_8_Low(); // test
GPIO_P0_14_Low();
```


ST UART2 をLPC1788との通信に利用し、ST UART1はパソコンとのUSB接続(STLINK)で利用しようとテストをしました。
 結果、ST UART1の通信が思うように動作しないため
 ST UART2のみでUSBと切り替えてLPC1788 uart0との通信をします。

*ST(JP5) のジャンパー切り替えが必要



ST EW Program 拔粹

```
/**
*****
* File Name      : main.c
* Description    : Main program body
*****
*
* COPYRIGHT(c) 2020 STMicroelectronics
*
* Redistribution and use in source and binary forms, with or without modificatio
* are permitted provided that the following conditions are met:
* 1. Redistributions of source code must retain the above copyright notice,
*    this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright notice,
*    this list of conditions and the following disclaimer in the documentation
*    and/or other materials provided with the distribution.
* 3. Neither the name of STMicroelectronics nor the names of its contributors
*    may be used to endorse or promote products derived from this software
*    without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
* SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
* CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
* OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*
*****
*/
/* Includes -----*/
#include "stm32xxx_hal.h"

/* USER CODE BEGIN Includes */
#include "main.h"
#include "vl53lx_api.h"
#include "53L3A2.h"
/* USER CODE END Includes */

/* Private variables -----*/
/* USER CODE BEGIN PV */
/* Private variables -----*/
I2C_HandleTypeDef hi2c1;
UART_HandleTypeDef huart1;
UART_HandleTypeDef huart2;
VL53LX_Dev_t dev;
VL53LX_DEV Dev = &dev;
int status;
volatile int IntCount;
#define isInterrupt 1 /* If isInterrupt = 1 then device working in hardware inte
/* USER CODE END PV */

/* Private function prototypes -----*/
/* USER CODE BEGIN PFP */
/* Private function prototypes -----*/
void SystemClock_Config(void);
void Error_Handler(void);
static void MX_GPIO_Init(void);
static void MX_USART1_UART_Init(void);
static void MX_USART2_UART_Init(void);
static void MX_I2C1_Init(void);
void RangingLoop(void); /* */
/* USER CODE END PFP */

/* USER CODE BEGIN 0 */
#define PUTCHAR_PROTOTYPE int fputc(int ch, FILE *f)

PUTCHAR_PROTOTYPE
{
    HAL_UART_Transmit(&huart2, (uint8_t*)&ch, 1, 0xFFFF);
    return ch;
}
```

```
int main(void)
{
    /* USER CODE BEGIN 1 */

    uint8_t byteData;
    uint16_t wordData;
    uint8_t ToFSensor = 1; // 0=Left, 1=Center (default), 2=Right

    /* MCU Configuration -----

    /* Reset of all peripherals, Initializes the Flash interface and the Systick.
    HAL_Init();

    /* Configure the system clock */
    SystemClock_Config();

    /* Initialize all configured peripherals */
    MX_GPIO_Init();

    MX_USART1_UART_Init();
    MX_USART2_UART_Init();

    MX_I2C1_Init();

    XNUCLE053L3A2_Init();

    /* USER CODE END 1 */

    /* USER CODE BEGIN 2 */

    printf("VL53L1X Examples...\n");
    Dev->I2cHandle = &hi2c1;
    Dev->I2cDevAddr = 0x52;

    /* Allow to select the sensor to be used, multi-sensor is not managed in this
    Only when use the Left ToF in interrupt mode, solder the U7 on the X-Nucl
    Only when use the Right ToF in interrupt mode, solder the U7 on the X-Nucleo-E
    See "Solder drop configurations" in the X-Nucleo-53L3A2 User Manual for more c
    ToFSensor = 1; // Select ToFSensor: 0=Left, 1=Center, 2=Right
    status = XNUCLE053L3A2_ResetId(ToFSensor, 0); // Reset ToF sensor
    HAL_Delay(2);
    status = XNUCLE053L3A2_ResetId(ToFSensor, 1); // Reset ToF sensor
    HAL_Delay(2);

    VL53LX_RdByte(Dev, 0x010F, &byteData);
    printf("VL53LX_Model_ID: %02X\n", byteData);
    VL53LX_RdByte(Dev, 0x0110, &byteData);
    printf("VL53LX_Module_Type: %02X\n", byteData);
    VL53LX_RdWord(Dev, 0x010F, &wordData);
    printf("VL53LX: %02X\n", wordData);
    printf("====Appyo h. 2023====\n");

    /*
    // UART2 test NG
    rf = 0;
    //HAL_UART_Transmit_IT(&huart2, "uart2 tx text\r\n", 16);
    while(1) {
        HAL_UART_RxCpltCallback(&huart2);
        //HAL_UART_Receive_IT(&huart2, rx_buff, 10);
        //while(!rf);
        HAL_Delay(4);
        printf("////LPC1788: %s\n", rx_buff);

        rf = 0;
    }
    */

    RangingLoop();

    /* USER CODE END 2 */

    /* Infinite loop */
    /* USER CODE BEGIN WHILE */

    /*
    while (1)
    {
        /* USER CODE END WHILE
        if (HAL_UART_Receive(&huart1, rx_buff, 10, 1000)==HAL_OK) //if transfer is su
        {
            //__NOP(); //You need to toggle a breakpoint on this line!
            printf("////LPC1788: %s\n", rx_buff);
            printf("////STM32F401////UART1 recive test OK\n");
        } else {
            __NOP();
            printf("////LPC1788: %s\n", rx_buff);
            printf("////STM32F401////UART1 recive test NG\n");
        }
        /* USER CODE BEGIN 3
    }
    */

    /* USER CODE END 3 */
}
```

ST EW Program 拔粹

```
/* USART2 init function */
static void MX_USART2_UART_Init(void)
{
    huart2.Instance = USART2;
    //huart2.Init.BaudRate = 115200;
    huart2.Init.BaudRate = 19200;
    huart2.Init.WordLength = UART_WORDLENGTH_8B;
    huart2.Init.StopBits = UART_STOPBITS_1;
    huart2.Init.Parity = UART_PARITY_NONE;
    huart2.Init.Mode = UART_MODE_TX_RX;
    huart2.Init.HwFlowCtl = UART_HWCONTROL_NONE;
    huart2.Init.OverSampling = UART_OVERSAMPLING_16;
    if (HAL_UART_Init(&huart2) != HAL_OK)
    {
        Error_Handler();
    }
}
```

```
do{ // polling mode
    printf ("Hello from St¥r¥n");
    status = VL53LX_GetMeasurementDataReady (Dev, &NewDataReady);
    HAL_Delay(100); // 1 ms polling period, could be longer.

    if ((!status)&&(NewDataReady!=0)) {
        status = VL53LX_GetMultiRangingData (Dev, pMultiRangingData);
        no_of_object_found=pMultiRangingData->NumberOfObjectsFound;
        //printf("Count=%5d, ", pMultiRangingData->StreamCount);
        printf("#Objs=%1d ", no_of_object_found);
        for (j=0;j<no_of_object_found;j++) {
            if (j!=0) printf("¥r¥n");
            //printf("status=%d¥r¥n D=%5dmm Signal=%2.2f Mcps Ambient=%2.2f",
            printf(" D=%5dmm Signal=%2.2f Mcps Ambient=%2.2f",
                //pMultiRangingData->RangeData[j].RangeStatus,
                pMultiRangingData->RangeData[j].RangeMilliMeter,
                pMultiRangingData->RangeData[j].SignalRateRtnMegaCps/65536,
                pMultiRangingData->RangeData[j].AmbientRateRtnMegaCps/65536);
        }
        printf ("¥n");
        if (status==0) {
            status = VL53LX_ClearInterruptAndStartMeasurement (Dev);
        }
    }
} while (1);
//}
```

Tera Term設定とモニター表示
メッセージは中央のセンサでレー
ザ保護カバーを付けています。

The screenshot shows the "Tera Term: 新しい接続" (Tera Term: New Connection) dialog box. It has two main sections. The top section is for network connections, with "TCP/IP" selected. Fields include "Host (H): 192.168.1.14", a checked "History (O)" checkbox, "Service (S): Telnet", "TCP Port # (P): 22", "SSH Version (V): SSH2", and "Protocol (C): UNSPEC". The bottom section is for serial connections, with "Serial (E)" selected. The "Port (R):" field is set to "COM5: STMicroelectronics STLink Virtual...". At the bottom are three buttons: "OK", "Cancel", and "Help (H)".

Tera Term: 新しい接続

☐ TCP/IP ホスト(H): 192.168.1.14
☒ ヒストリ(O)
サービス(S): ☐ Telnet TCPポート#(P): 22
☒ SSH SSHバージョン(V): SSH2
☐ その他 プロトコル(C): UNSPEC

☒ シリアル(E) ポート(R): COM5: STMicroelectronics STLink Virtua

OK キャンセル ヘルプ(H)

Tera Term: シリアルポート 設定

ポート(P):	COM5	OK
スピード(E):	19200	キャンセル
データ(D):	8 bit	
パリティ(A):	none	
ストップビット(S):	1 bit	
フロー制御(F):	none	ヘルプ(H)

送信遅延

0	ミリ秒/字(C)	0	ミリ秒/行(L)
---	----------	---	----------

Tera Term - [未接続] VT

ファイル(F) 編集(E) 設定(S) コントロール(O) ウィンドウ(W) ヘルプ(H)

Hello from St

#Objs=1 D= 1657mm Signal=0.38 Mcps Ambient=0.42

Hello from St

#Objs=1 D= 1654mm Signal=0.47 Mcps Ambient=0.38

Hello from St

#Objs=1 D= 1651mm Signal=0.41 Mcps Ambient=0.41

Hello from St

#Objs=2 D= 88mm Signal=0.17 Mcps Ambient=0.38

D= 1642mm Signal=0.45 Mcps Ambient=0.38

Hello from St

#Objs=1 D= 1642mm Signal=0.45 Mcps Ambient=0.40

Hello from St

#Objs=1 D= 1659mm Signal=0.43 Mcps Ambient=0.40

Hello from St

#Objs=1 D= 1652mm Signal=0.44 Mcps Ambient=0.41

Hello from St

#Objs=2 D= 48mm Signal=0.16 Mcps Ambient=0.40

D= 1657mm Signal=0.43 Mcps Ambient=0.40

Hello from St

#Objs=1 D= 1664mm Signal=0.41 Mcps Ambient=0.41

Hello from St

#Objs=2 D= 68mm Signal=0.16 Mcps Ambient=0.38

D= 1656mm Signal=0.46 Mcps Ambient=0.38

Hello from St

#Objs=2 D= 89mm Signal=0.15 Mcps Ambient=0.38

D= 1652mm Signal=0.43 Mcps Ambient=0.38

Hello from St

#Objs=2 D= 73mm Signal=0.20 Mcps Ambient=0.38

D= 1645mm Signal=0.46 Mcps Ambient=0.38

Hello from St

#Objs=2 D= 70mm Signal=0.16 Mcps Ambient=0.38

D= 1654mm Signal=0.42 Mcps Ambient=0.38

Hello from St

#Objs=2 D= 68mm Signal=0.20 Mcps Ambient=0.38

D= 1641mm Signal=0.46 Mcps Ambient=0.38

Hello from St

#Objs=2 D= 66mm Signal=0.16 Mcps Ambient=0.38

D= 1653mm Signal=0.42 Mcps Ambient=0.38

Hello from St

#Objs=2 D= 83mm Signal=0.20 Mcps Ambient=0.38

D= 1642mm Signal=0.46 Mcps Ambient=0.38

Hello from St

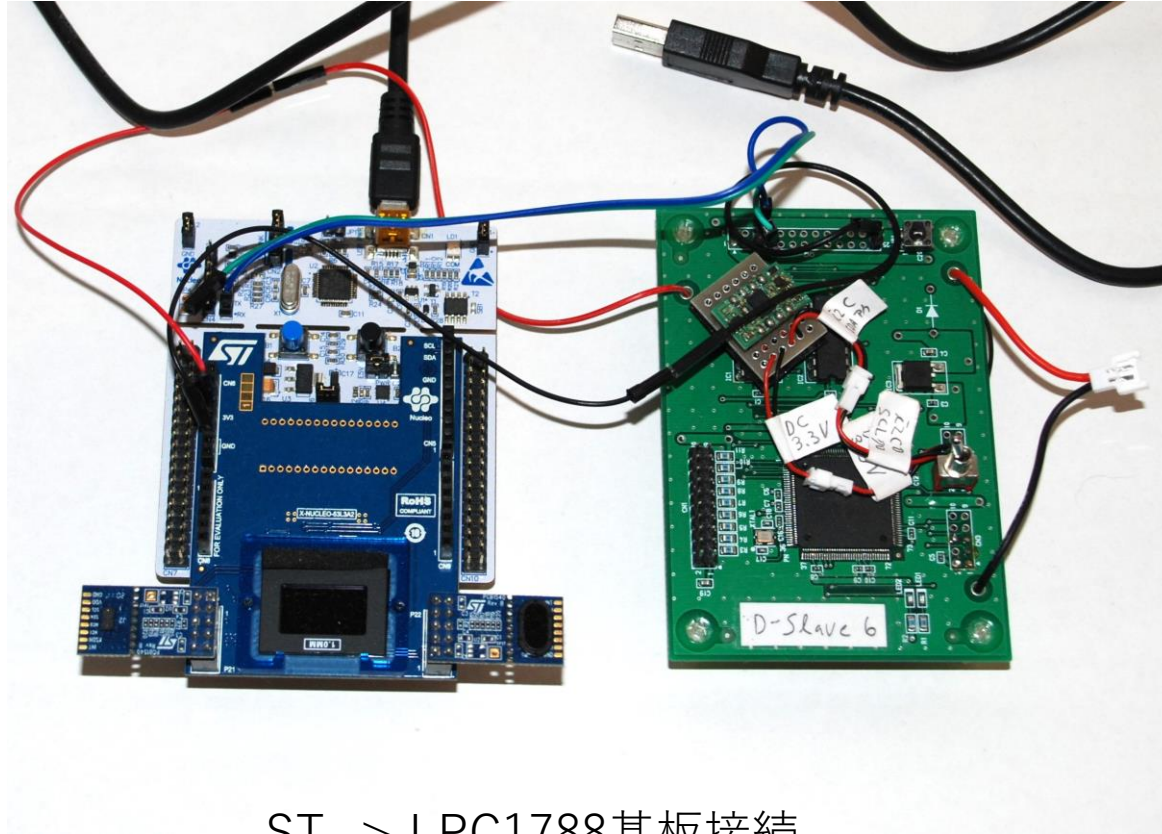
#Objs=2 D= 66mm Signal=0.16 Mcps Ambient=0.38

D= 1651mm Signal=0.43 Mcps Ambient=0.38

Hello from St

#Objs=2 D= 104mm Signal=0.23 Mcps Am

ST -> LPC基板 -> TWELITE -> M-Server 転送 TEST
D-Slave6 プログラム

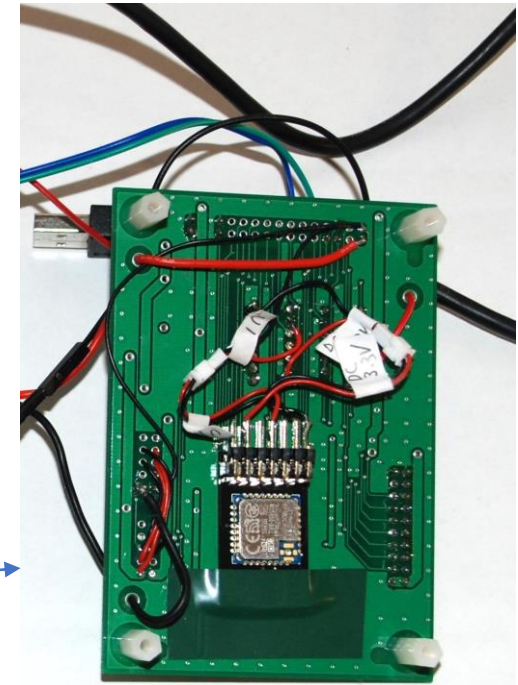


ST -> LPC1788基板接続

無線
TWELITE

M-Server へ接続

LPC1788基板裏側 TWELITE



D-Slave9-8 lpc1788 EW プログラム抜粋

```
//U0_48M_9600();          // 7 Seg LED
U0_48M_19200();
//U0_48M_38400();

//U1_48M_9600_int();      // TWELITE
U1_48M_19200_int();       // TWELITE
//U1_48M_38400_int();     // Master Server TWLITE // Test
```

```
// CPU Clock 96 MHz , PCLK 48MHz , Baud Rate 19200 bps
void U0_48M_19200(void)
{
    int Div;
    int AddDiv;
    int Mul;

    // *POW_PCONP = *POW_PCONP & (1 << 3); // bit3 -> 0 UART0 Default ON
    // PCONP_bit.PCUART0 = 1; // Reset Value ON

    Div = 104;
    AddDiv = 1;
    Mul = 2;          // MuVal

    *COM_LCR0 = 0x80;    // Enable access to devider Latches
    *COM_DLL0 = Div & 0xFF;
    *COM_DLM0 = (Div >> 8) & 0xFF;
    *COM_FDR0 = AddDiv + (Mul << 4);
    UOLCR_bit.DLAB = 0;  // Disable access to devider Latches
    *COM_LCR0 = 0x03;    // No Party , 1 Stop bit / 8 bits
    *COM_IER0 = 1;       // int use

    *IOCON_PO_2 = 0x31;  // Type_D func:001 U0_Txd Mode:Pull-UP HYS:1
    *IOCON_PO_3 = 0x31;  // RXD Type_D func:001_TXD2 Mode:Pull-UP HYS:1

    NVIC_IntEnable(NVIC_UART0);
    recv0_flg = 0;      // Uart0 Recive flag Clear
}
```

```
// CPU Clock 96 MHz , PCLK 48MHz , Baud Rate 19200 bps
void U1_48M_19200_int(void)
{
    int Div;
    int AddDiv;
    int Mul;
    //int Tmp;

    // *POW_PCONP = *POW_PCONP & (1 << 4); // bit4 -> 1 UART1 ON
    PCONP_bit.PCUART1 = 1; // Power ON

    Div = 104;
    AddDiv = 1;
    Mul = 2;          // MuVal

    *COM_LCR1 = 0x80;    // Enable access to devider Latches
    *COM_DLL1 = Div & 0xFF;
    *COM_DLM1 = (Div >> 8) & 0xFF;
    *COM_FDR1 = AddDiv + (Mul << 4);
    U1LCR_bit.DLAB = 0;  // Disable access to devider Latches
    *COM_LCR1 = 0x03;    // No Party , 1 Stop bit / 8 bits
    *COM_IER1 = 1;       // int use

    *IOCON_PO_15 = 0x31; // TXD Type_D func:001_TXD2 Mode:Pull-UP HYS:1
    *IOCON_PO_16 = 0x31; // RXD Type_D func:001_TXD2 Mode:Pull-UP HYS:1

    // *ISER0 = *ISER0 | 0x40; // UART1 interrupt enable
    // Transmit enable
    // U1TER_bit.TXEN = 1;
    // Tmp = U1IER; // Clear pending interrupts
    // U1IER = 0x01; // RBR interrupt Enable
    // Enable NVIC UART1 Interrupt
    NVIC_IntEnable(NVIC_UART1);

    recv1_flg = 0;      // Uart1 Recive flag Clear
}
```

ST -> LPC1788(UART0) ->
TW(UART1) ~~~ TW(M Server)

右のログはPCから Tera Term でServer
にログインして
/home/pi/Drone> ./mpi -d4
にて表示したログデータになります。
#Objs=2 で障害物の検出結果が表示さ
れます。

```
192.168.1.8 - Tera Term VT
ファイル(F) 編集(E) 設定(S) コントロール(O) ウィンドウ(W) ヘルプ(H)
RASPBERYPYI /home/pi> cd Drone
RASPBERYPYI /home/pi/Drone> ./mpi -d4

/// DroneRv_ckh() Drone serialData check ///

TW 0ent=0.33

TW Hello from St

TW #Objs=2 D= 51mm Signal=0.17 Mcps Ambient=0.33

TW Hello from St

TW #Objs=2 D= 59mm Signal=0.16 Mcps Ambient=0.33

TW Hello from St

TW #Objs=2 D= 44mm Signal=0.18 Mcps Ambient=0.33

TW Hello from St

TW Bno-Heading: x7e x16 xd1 xff xa4 xff

H EUL Heading Roll Pitch
hx=5758 hy=-47 hz=-92
Heading=359.875000 Roll=-47.000000 Pitch=-92.000000

TW ##Objs=2 D= 37mm Signal=0.17 Mcps Ambient=0.33

TW D= 1629mm Signal=0.47 Mcps Ambient=0.33

TW Hello from St

TW #Objs=2 D= 50mm Signal=0.16 Mcps Ambient=0.33

TW D= 1622mm Signal=0.46 Mcps Ambient=0.33

TW Hello from St

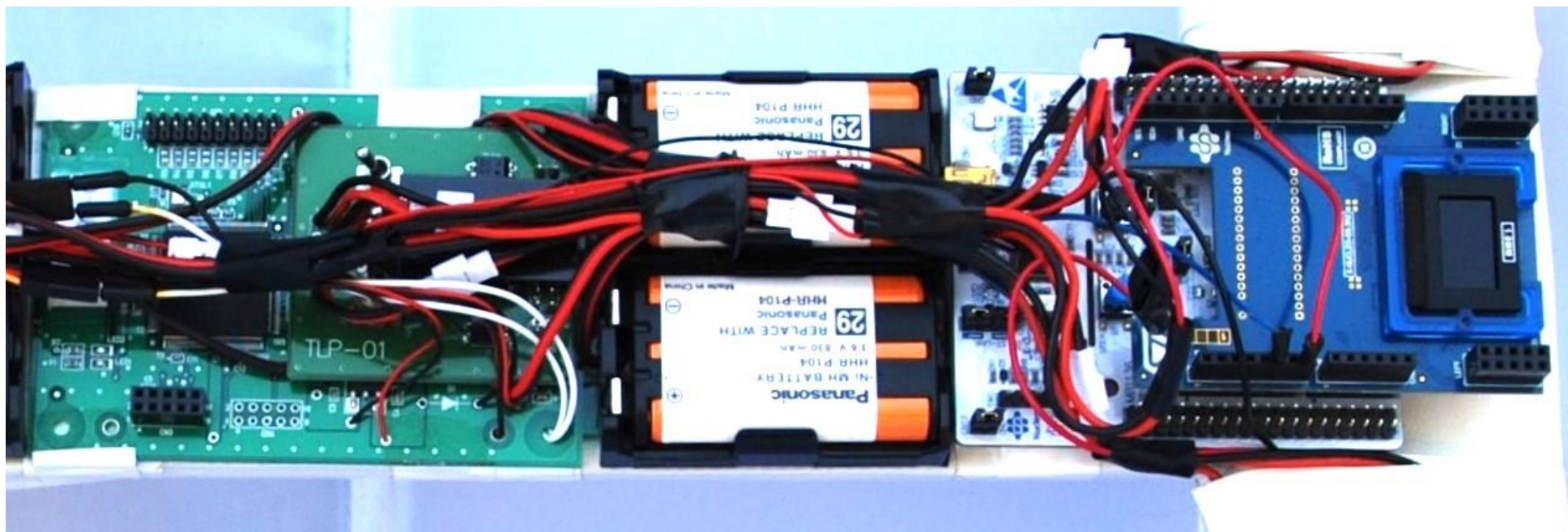
TW Hello from St

TW #Objs=2 D= 21mm Signal=0.16 Mcps Ambient=0.33

TW D= 1625mm Signal=0.46 Mcps Ambient=0.33

TW Hello from St
```

距離センサーを付ける NUCLEO-F401RE vl53l3cx-to-lpc1788 test



スペック

サイズ

フレーム 前後 550mm 左右 430mm
高さ 30mm

プロペラを含む 前後 660mm 左右 580mm
高さ 90mm

重量

バッテリーなし
1300g

バッテリーあり
1665g

結果

重量は下がらない。

原因

モーターが重い、L型金具が重い、
プロペラ固定用アルミ線が重い

FA-130RA 回転数が少ないので揚力が上がらない。