

リモートキーヤーキット for Arduino UNO R3

組立て及び初期設定手順書 Ver 1.0

高松オリーブハムクラブ

2025/03/09

★★★★★ ご注意 ★★★★★

- 本キットを完成させるためには、Arduino UNO を使った簡単なプログラミング及び電子工作の経験と、基本的な IP ネットワークの知識が必要となります。
- 本キットの運用には、高速かつ安定したネットワーク環境が必要です。インターネットを経由して運用する場合は、フレッツ光などの光回線の導入をお勧めします。また、ネットワーク環境によっては正常に動作しない場合があります。
- 本キットは、LAN や VPN などのセキュリティが確保されたネットワークに接続して運用することを前提としています。インターネットから直接アクセスされるようなネットワーク環境で運用することはセキュリティ上推奨しません。
- 本キットを運用して生じたいかなる結果に対しても、当クラブは一切の責任を負いません。
- 本キットにサイドトーンを発生させる機能はありません。

1 はじめに

この度は、リモートキーヤーキット for Arduino UNO をお求めいただき、ありがとうございます。本書は本キットの組立てや初期設定の方法を記載した資料です。

このキットは、電鍵やエレキーの接点情報を IP ネットワーク経由で遠隔地に伝送し、アマチュア無線のリモート運用において電鍵やパドルを使用した運用を可能とする物で、クライアントとサーバで構成されます。本キット 1 式で 2 台の無線機を切り替えてキーイングできます。

運用にはこのキットのほかに Arduino UNO R3 及びイーサネットシールドがそれぞれ 2 式必要です。

2 組立て方

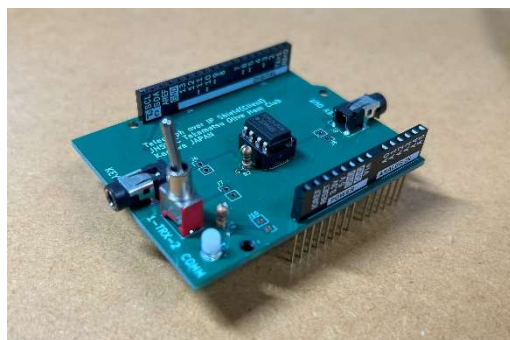
表 1 の上から順番に、各部品を基板にはんだ付けします。次ページの「組み立て上の注意点」をよく読んで作業してください。

表 1 部品表

ユニット	部品番号	部品名称	数量
クライアント	D1	LED Φ3.0mm 緑色	1
	J5, J9	3.5 ステレオミニジャック	2
	-	ICソケット 8ピン	1
	R1	カーボン抵抗 1/4W 300Ω (橙 黒 茶 金)	1
	R2	カーボン抵抗 1/4W 250Ω (赤 緑 茶 金)	1
	J3, J4	ARDUINO用ピンソケット 10P+8P	1
	J1, J2	ARDUINO用ピンソケット 8P+6P	1
	SW1	基板用小型3Pトグルスイッチ 1回路2接点	1
	U1	フォトカプラ TLP621-2	1
サーバ	D1, D2	LED Φ3.0mm 緑色	2
	J7, J9	3.5 ステレオミニジャック	2
	-	ICソケット 8ピン	1
	R1, R2	カーボン抵抗 1/4W 250Ω (赤 緑 茶 金)	2
	R3, R4	カーボン抵抗 1/4W 300Ω (橙 黒 茶 金)	2
	J3, J4	ARDUINO用ピンソケット 10P+8P	1
	J1, J2	ARDUINO用ピンソケット 8P+6P	1
	U1	フォトカプラ TLP621-2	1

★組み立て上の注意点

- ・ はんだ付けに使用するはんだごては熱くなりますので取り扱いには十分注意してください。
- ・ LED は、基板の丸いランドにアノード側（足が長い方）を挿入してください。
- ・ ARDUINO 用ピンソケットは、基板に対して垂直になるよう取り付けてください。取り付け時は、ピンの先にはんだが付かないよう注意してください。取り付け後は、ピンをカットしないでください。
- ・ スイッチやジャックなど、機械的な力が加わる部品は基板に密着させて取り付けてください。
- ・ IC ソケットは、切り欠きの向きを基板表面のシルク印刷と合わせて取り付けてください。
- ・ フォトカプラは IC ソケットに挿入してください。J1、J2 を手前側に見て、IC ソケットの左上に 1 番ピン（フォトカプラ表面に「●」の印刷があります。）が来る向きに挿入してください。
- ・ LED の光がまぶしい場合は、同梱の光拡散キャップを LED に取り付けてください。
- ・ 完成図は図 2-1 を参照してください。



クライアント



サーバ

図 2-1 完成図

3 検査

組立て後は、イモはんだやはんだブリッジ等のはんだ不良がないか、部品の取り付けミスがないか、目視により検査してください。

4 初期設定

初期設定は、(1)ソースプログラムのダウンロード、(2)ネットワークに関する設定、(3)ソースプログラムの書き込みの流れで行います。

(1) ソースプログラムのダウンロード

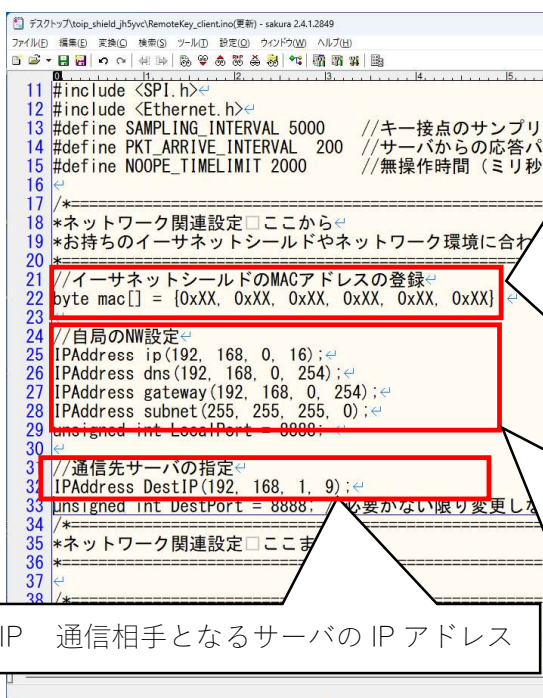
ア 次の URL にアクセスし、本キットを使用するためのソースプログラムをダウンロードします。

http://takamatsu-olive-ham.club/toip_shield_jh5yvc.zip

イ アでダウンロードしたファイルを解凍します。解凍すると RemoteKey_client.ino 及び RemoteKey_server.ino の 2 つのファイルが得られます。RemoteKey_client.ino がクライアント用、RemoteKey_server.ino がサーバ用のソースプログラムです。

(2) ネットワークに関する設定

ア エディタで RemoteKey_client.ino を開き、図 4-1 の説明に従ってクライアントのネットワーク設定を行います。文字はすべて半角で入力してください。設定が終わったら、ファイルを保存してエディタを閉じます。



```
11 #include <SPI.h>
12 #include <Ethernet.h>
13 #define SAMPLING_INTERVAL 5000 //キー接点のサンプリング時間
14 #define PKT_ARRIVE_INTERVAL 200 //サーバからの応答待ち時間
15 #define NOOPE_TIMELIMIT 2000 //無操作時間 (ミリ秒)
16
17 /*
18 *ネットワーク関連設定□ここから
19 *お持ちのイーサネットシールドやネットワーク環境に合わせて
20 */
21 //イーサネットシールドのMACアドレスの登録
22 byte mac[] = {0xXX, 0xXX, 0xXX, 0xXX, 0xXX, 0xXX};
23
24 //自局のNW設定
25 IPAddress ip(192, 168, 0, 16);
26 IPAddress dns(192, 168, 0, 254);
27 IPAddress gateway(192, 168, 0, 254);
28 IPAddress subnet(255, 255, 255, 0);
29 unsigned int LocalPort = 8080;
30
31 //通信先サーバの指定
32 IPAddress DestIP(192, 168, 1, 9);
33 unsigned int DestPort = 8888;
34
35 *ネットワーク関連設定□ここから
36
37
38
```

DestIP 通信相手となるサーバの IP アドレス

[XX]の部分に、イーサネットシールドの MAC アドレスを登録します。MAC アドレスはイーサネットシールドの裏面に記載されています。



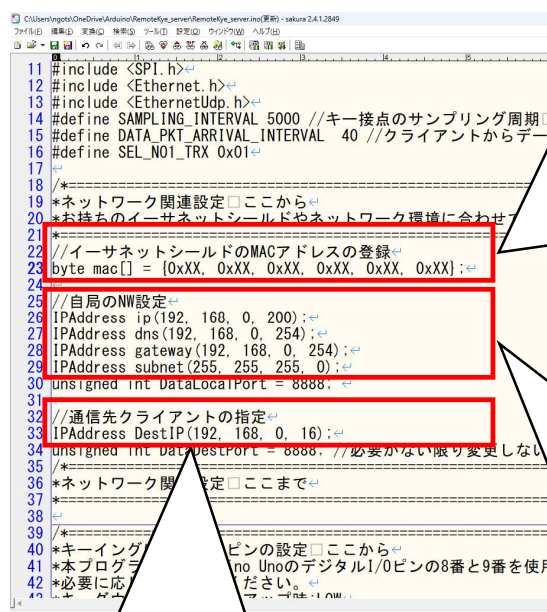
※記載がない場合は、イーサネットシールドの販売元にお問い合わせください。

各項目に必要な値を設定します。

ip	クライアントで使用する IP アドレス
dns	DNS サーバの IP アドレス (一般的にはルータの IP アドレスと同じです。)
gateway	ルータの IP アドレス
subnet	サブネットマスク (一般的には変更の必要なし)

図 4-1 クライアントのネットワーク設定

イ エディタで RemoteKey_server.ino を開き、図 4-2 の説明に従ってサーバのネットワーク設定を行います。文字はすべて半角で入力してください。設定が終わったら、ファイルを保存してエディタを閉じます。



[XX]の部分に、イーサネットシールドの MAC アドレスを登録します。MAC アドレスはイーサネットシールドの裏面に記載されています。

※記載がない場合は、イーサネットシールドの販売元にお問い合わせください。



各項目に必要な値を設定します。

ip サーバで使用する IP アドレス
dns DNS サーバの IP アドレス（一般的にはルータの IP アドレスと同じです。）
gateway ルータの IP アドレス
subnet サブネットマスク（一般的には変更の必要なし）

DestIP 通信相手となるクライアントの IP アドレス

図 4-2 サーバのネットワーク設定

(3) ソースプログラムの書き込み

(2)で編集したソースプログラムを、クライアント側、サーバ側それぞれの Arduino に書き込みます。ソースプログラムの書き込みを行う際は、Arduino UNO に USB ケーブル以外の物を接続しないでください。書き込みに失敗する場合があります。

書き込み方法は、インターネット等でお調べください。

★参考となるウェブページへのリンク

・ [Arduino IDE プログラミングの基礎とシリアルモニターの使い方 – Indoor Corgi](#)



(4) よくあるエラー

(2)で編集したソースプログラムを Arduino に書き込む際、次のようなエラーが発生する場合があります。

```
Compilation error: MsTimer2.h: No such file or directory
Compilation error: TimerOne.h: No such file or directory
```

この場合、次の手順で対処してください。

- ① Arduino IDE のメニューから「スケッチ」 - 「ライブラリをインクルード」 - 「ライブラリを管理」をクリックする。
- ② ライブラリマネージャの検索ボックスに「MsTimer2」と入力する。
- ③ MsTimer2 が見つかったら、「インストール」をクリックする。
- ④ ②と同様に、検索ボックスに「TimerOne」を入力する。
- ⑤ TimerOne が見つかったら、「インストール」をクリックする。
- ⑥ Arduino IDE を再起動する。

5 Arduino との接続

図 3 を参照し、Arduino UNO にイーサネットシールドと本キットを接続してください。接続方法は、クライアント、サーバ共に同じです。

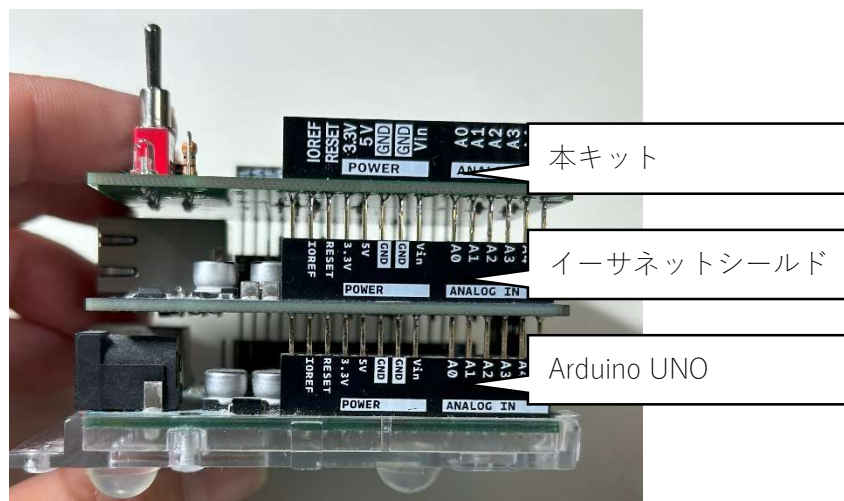


図 5-1 本キットと Arduino を接続した状態

6 動作確認

(1) LAN への接続

リモートキーヤーのクライアント、サーバをそれぞれ LAN に接続し、電源を投入します。

(3) ping コマンドによる疎通確認

パソコンのコマンドプロンプトを起動し、ping コマンドによりクライアント、サーバそれぞれに設定した IP アドレスから応答があることを確認します（図 6-1）。応答がない場合は、リモートキーヤーが正しく LAN に接続されているか、初期設定が正しく行われているか確認してください。

```
C:\Users¥xxxx>ping 192.168.0.16

192.168.0.16 に ping を送信しています 32 バイトのデータ:
192.168.0.16 からの応答: バイト数 =32 時間 <1ms TTL=128
192.168.0.16 からの応答: バイト数 =32 時間 <1ms TTL=128
192.168.0.16 からの応答: バイト数 =32 時間 <1ms TTL=128
192.168.0.16 からの応答: バイト数 =32 時間 <1ms TTL=128

192.168.0.16 の ping 統計:
    パケット数: 送信 = 4、受信 = 4、損失 = 0 (0% の損失)、
ラウンド トリップの概算時間 (ミリ秒):
    最小 = 0ms、最大 = 0ms、平均 = 0ms
```

図 6-1 ping コマンドによる疎通確認

(4) クライアントとサーバ間の疎通確認

- ① クライアントの「KEY IN」ジャックに電鍵またはエレキーの出力を接続します。
- ② クライアントのトグルスイッチを「1」側に倒し、キーイングします。キーイングに合わせてサーバの「TRX-1」LED が点滅することを確認します。
- ③ クライアントのトグルスイッチを「2」側に倒し、キーイングします。キーイングに合わせてサーバの「TRX-2」LED が点滅することを確認します。

- ④ 以上で動作確認は終了です。正しく動作しない場合は、本キットの組み立てに問題はないか、初期設定は正しく行われているか、正しく Arduino UNO に接続されているかを確認するとともに、クライアントとサーバ間の通信に到達性があるか、ネットワーク環境を確認してください。

付録

1 各部の名称及び機能

(1) クライアント

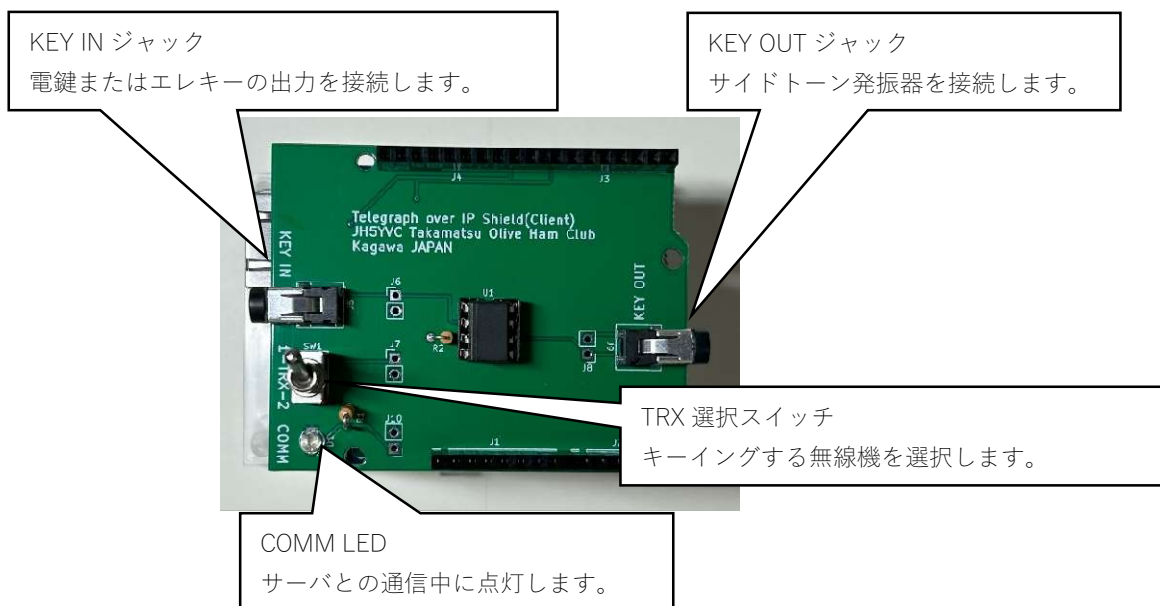


図 1-1 クライアント

(2) サーバ

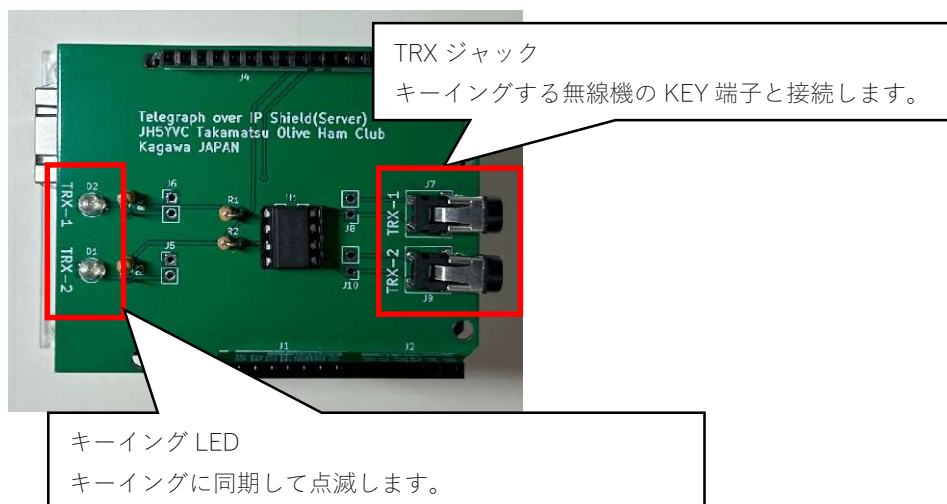


図 1-2 サーバ

2 回路図

(1) クライアント

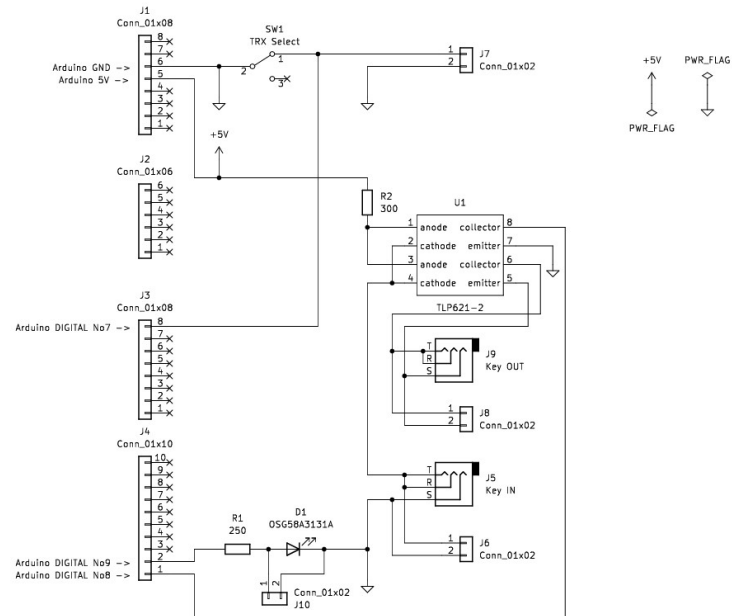


図 2-1 クライアント

(2) サーバ

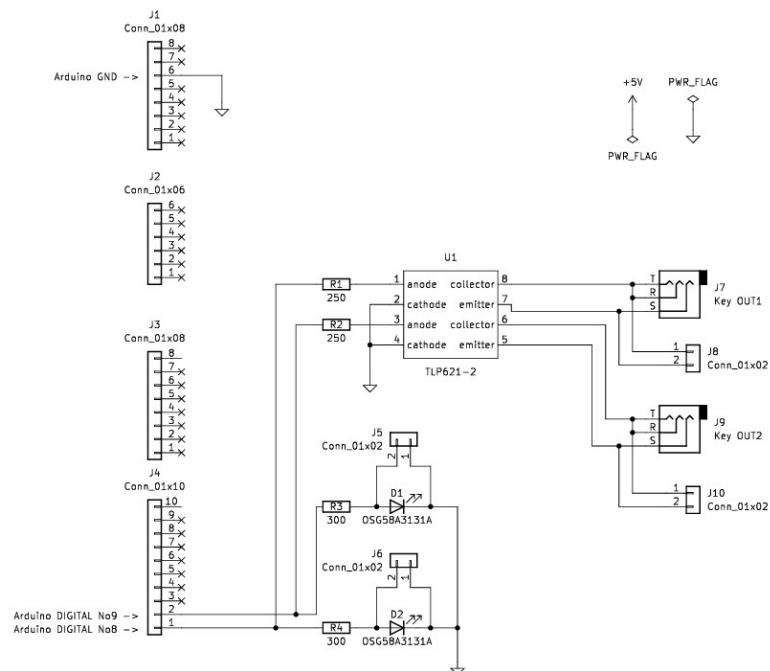


図 2-2 サーバ