

## Arduino K3NG CW キーヤー 組み立て説明書

これは K3NG OM が配布している Arduino による CW キーヤー・スケッチに対応した CW キーヤー・シールドです。K3NG CW キーヤーが実現している機能はとても幅広く、このうちこのシールド基板が対応している機能は次の様になります。

このキーヤーで出来る事：

- アイアンビック A/B モードの切り替え
- Ultimatic モード (内容が良く分かりません(-\_-;))
- サイドトーン (モニター音) 周波数の調整
- Bug モードのサポート
- 送信イネーブル/ディセーブルの切り替え
- 単点・長点の比率調整
- パドル反転可能
- サイドトーンのオン/オフ
- メモリー機能 (4つまで)
- チューニングモード
- スピード調整の有効・無効切り替え
- 自動スペースのオン/オフ
- 送信せずにメモリー再生
- 送信機の切り替え機能 (TX1 / TX2)
- スピード調整用 VR

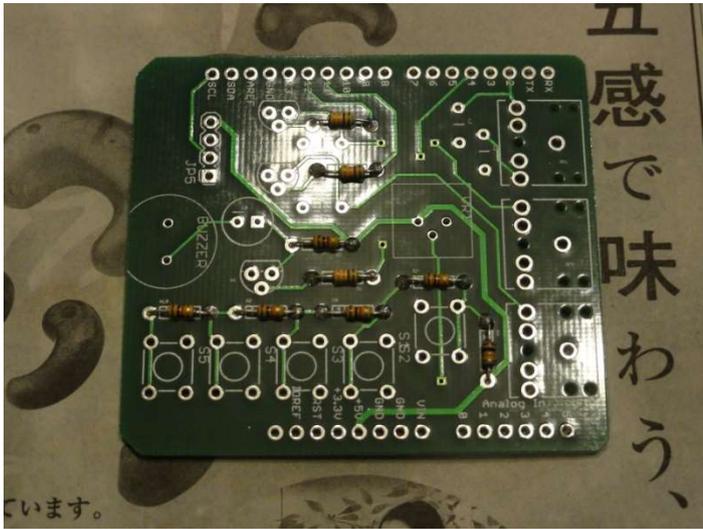
基板が完成したら Arduino に接続し、プログラムの定義ファイルを編集して、コンパイル、ダウンロードすれば直ぐに CW キーヤーが動作し始めます。

また、I2C 接続の LCD ディスプレイを接続できる端子を設けているので、外部に LCD ディスプレイを接続する事もできます。

## 1. 組立手順

### ① 抵抗の取り付け

図 1



基板のシルクに書かれている通りに  $1k\Omega$  4本、 $330\Omega$  3本、 $100\Omega$  1本、 $10k\Omega$  1本、計9本の抵抗を半田付けして行きます。部品表ではR5が欠番になっています。

### ② セラミックコンデンサの取り付け

図の赤丸の位置に  $0.1\mu$  の積層セラミックコンデンサを半田付けします。

図 2

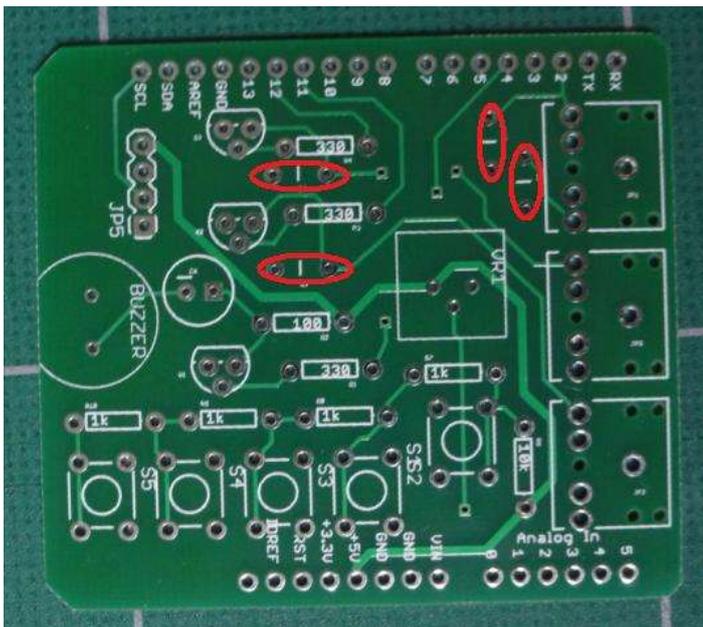
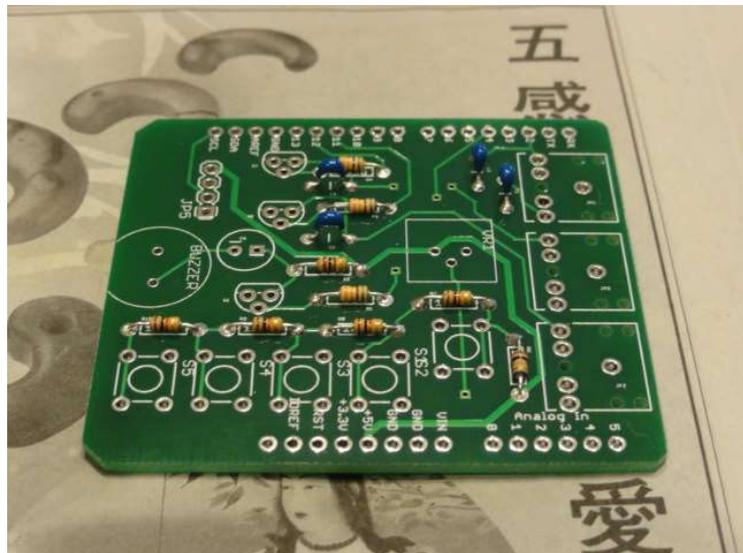
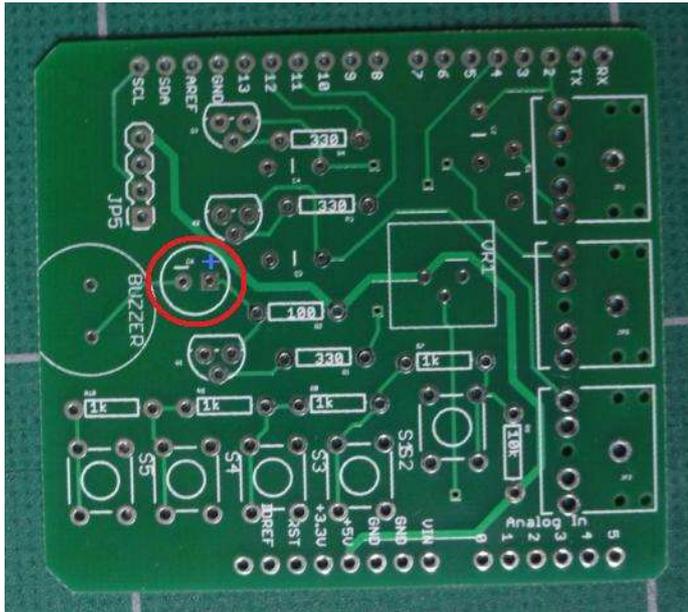


図 3



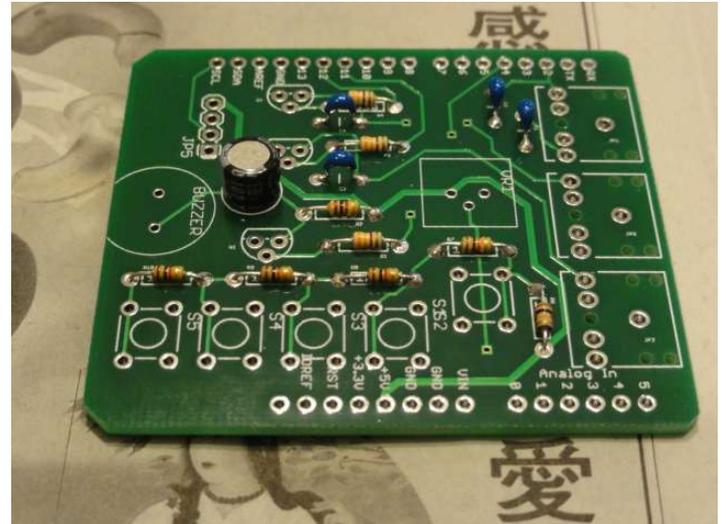
### ③ 電解コンデンサの取り付け

図 4



図の赤丸の位置に  $100\mu/16V$  の電解コンデンサを半田付けします。電解コンデンサは極性がありますので、図4で青く+と書かれた Pad(基板上では四角い Pad) に足の長い方が入るようにします。

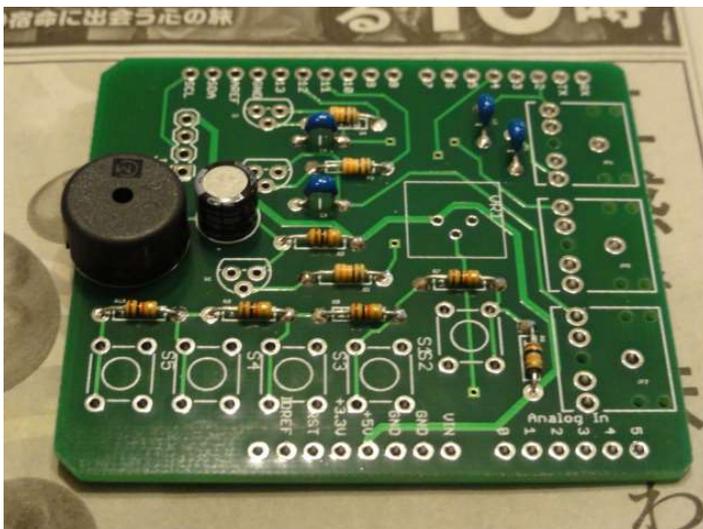
図 5



### ④ 圧電スピーカーの取り付け

図 6

圧電スピーカーには極性がありませんので、そのまま半田付けします。



⑤ 3.5mm ミニジャックの取り付け

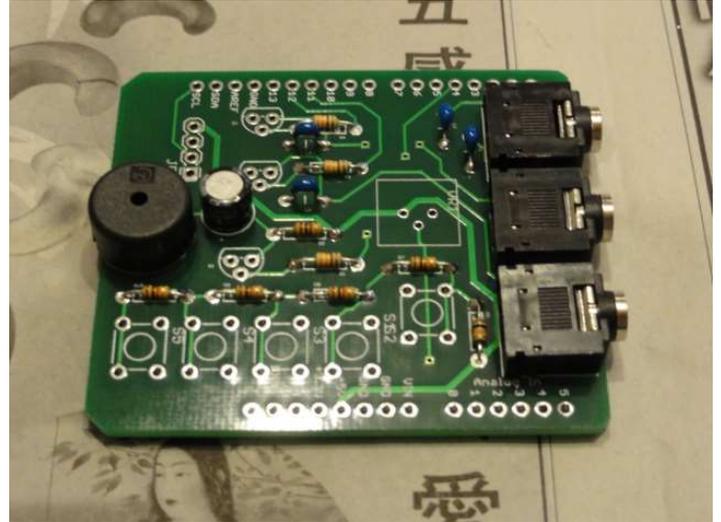
図 7



パドルと送信機のキー入力をつなぐ 3.5mm 用ミニジャック 3 個を半田付けします。

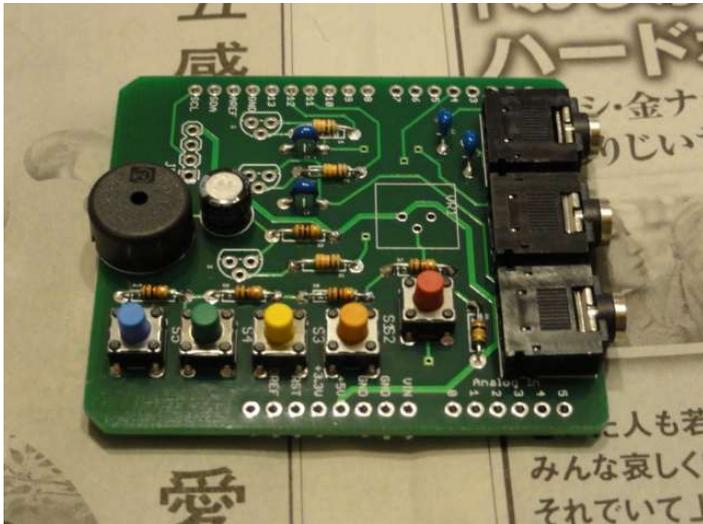
図 7 のような手を離すとばね性で部品を押さえられるようなピンセットを使うと便利です。

図 8



⑥ タクトスイッチの取り付け

図 9

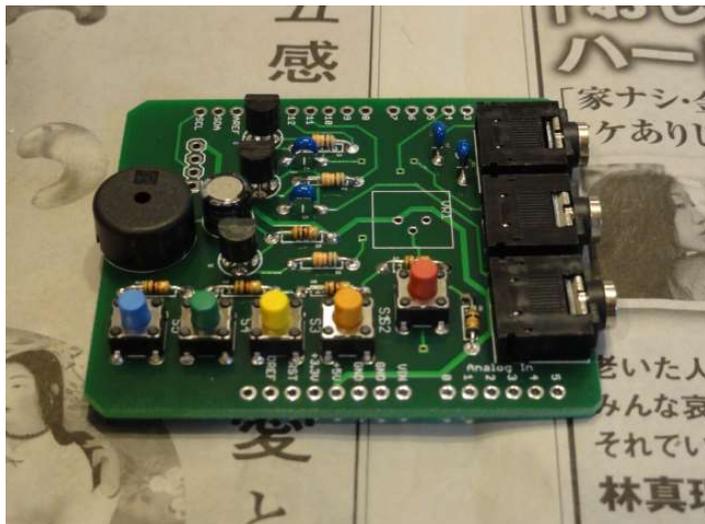


タクトスイッチを基板に挿入します。タクトスイッチの端子は爪状になっており、基板の Pad に合わせて挿入すると自分で基板に固定されます。この時、向きを間違とうまく入りません。端子の爪がきちんと Pad に合っている事を確認してから押し込んでください。タクトスイッチを基板と平行になるまで押しこんだら裏から半田付けします。

梱包されているタクトスイッチの色が異なっているかもしれませんが。各自のアイデアで配色してください。ここで一つだけ列からずれている赤色のタクトスイッチはコマンド・キーのスイッチです。

⑦ トランジスタの取り付け

図 10



トランジスタは極性がありますので、取り付け向きに注意して基板に挿入します。向きは、トランジスタの平らになった面を基板のシルクと合わせます。その後、半田付けします。

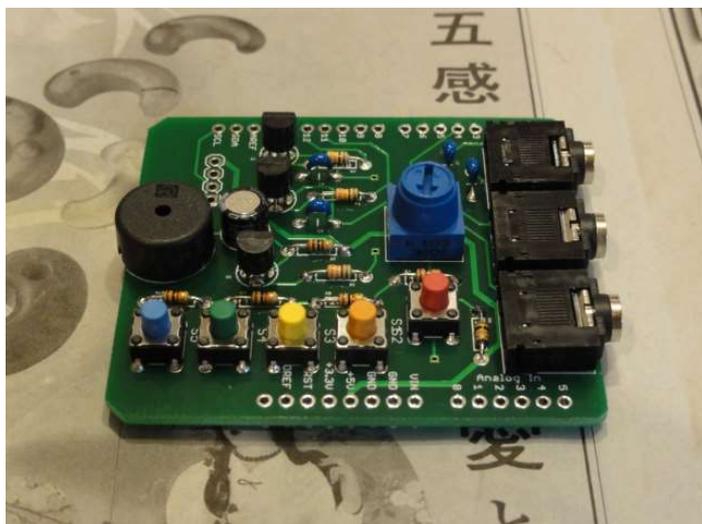
注) トランジスタは全品 hFE を測定して正常なものを出荷しています。

図 11



⑧ VR の取り付け

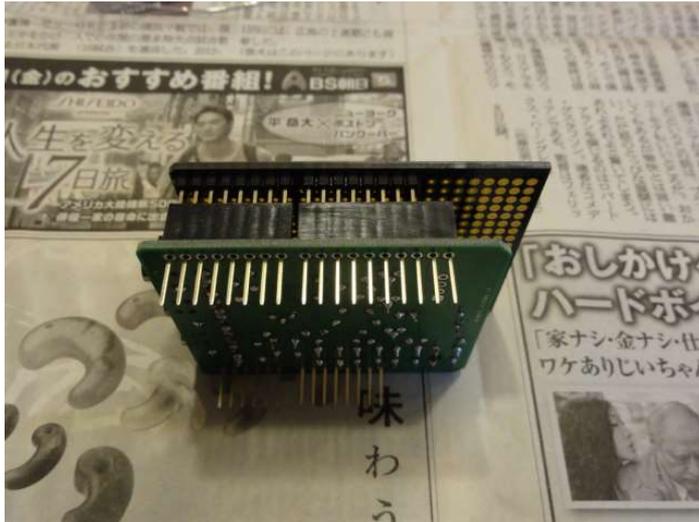
図 12



基板にスピード調整用の半固定 VR を取り付けます。

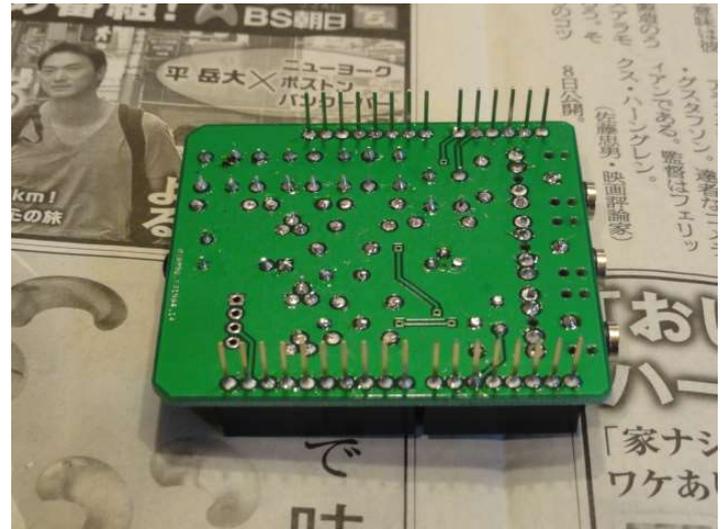
⑨ B2B コネクタの取り付け

図 13



Arduino の基板と接続するために基板と基板を接続するコネクタ 4 個を取り付けます。それぞれピン数が異なるので、ピン数を合わせて基板に挿入し、半田付けします。この時、コネクタが基板にちゃんと垂直になる様に気をつけます。簡単には写真のような拡張基板を治具代わりに使うと便利です。

図 14 半田付け後の様子



⑩ 完成写真

図 15 単体

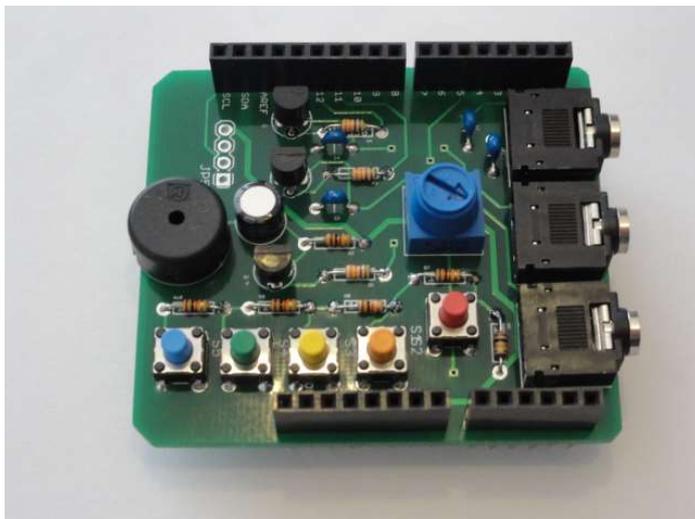


図 16 Arduino 互換基板と組み合わせた例



## 2. K3NG キーヤー・スケッチの修正箇所

K3NG OM のサイトからソース管理のサイトに移って最新版のソース・ファイルをダウンロードします。ダウンロードしたら、次の箇所を修正してコンパイルします。問題なくコンパイル出来たら Arduino 基板にダウンロードして、キーヤーを動かしてみます。

setting change log

### (1) Enable Command Botton, Memory feature and Speed controll potentiometer

直すファイル：**keyer\_features\_and\_options.h**

```
#define FEATURE_COMMAND_BUTTONS // do not enable unless you have the corresponding resistors  
connected to the analog_buttons_pin
```

```
#define FEATURE_MEMORIES
```

```
#define FEATURE_POTENTIOMETER // do not enable unless you have a potentiometer connected,  
otherwise noise will falsely trigger wpm changes
```

コマンド・ボタンの使用とメモリ機能、スピード調整 VR の使用を設定しています。

### (2) Enable Command Mode LED

直すファイル：**keyer\_pin\_settings.h**

```
#define command_mode_active_led 13
```

この指定により、コマンド・モードにある時は Arduino 基板上の LED が点灯します。

### (3) Initial setting of WPM and Sidetone Frequency, Number of Memories and WPM setting range

直すファイル：**keyer\_settings.h**

```
#define initial_speed_wpm 13 // "factory default" keyer speed setting
```

```
#define initial_sidetone_freq 1600 // "factory default" sidetone frequency setting
```

```
#define analog_buttons_number_of_buttons 5
```

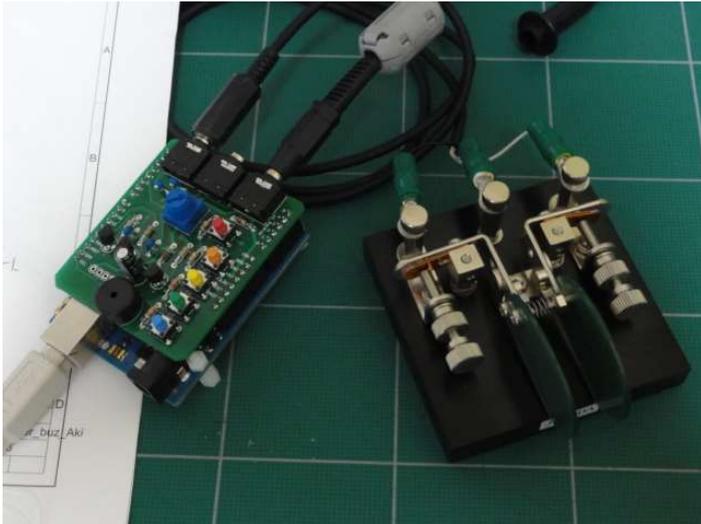
```
#define initial_pot_wpm_low_value 5 // Potentiometer WPM fully CCW
```

```
#define initial_pot_wpm_high_value 30 // Potentiometer WPM fully CW
```

ここで初期状態のスピードとスピードの可変範囲、サイドトーンの周波数を設定しています。

### 3. 動作確認

図 17

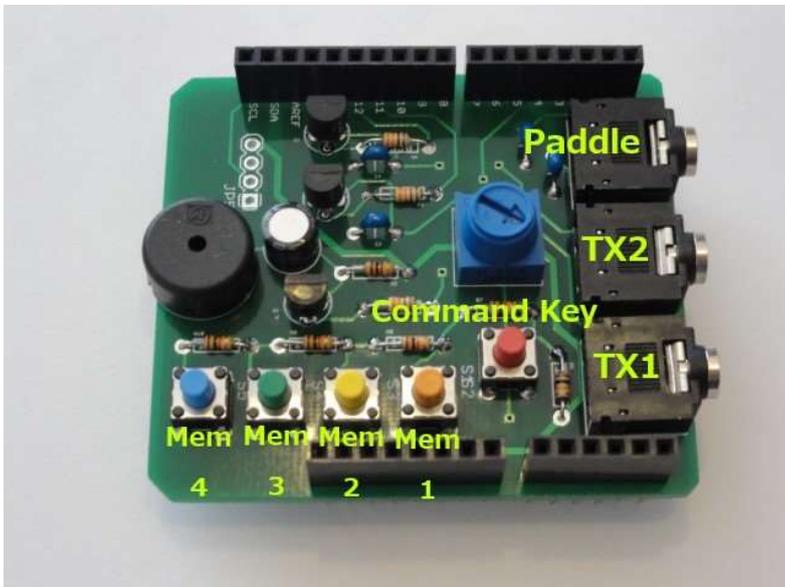


プログラムがダウンロードされ、起動されるとブザーから ..... と起動音が聞こえます。ここでパドルをつないで操作するとモニターできる事を確認します。次に、送信機（トランシーバ）をつないでキーイングできる事を確認して見てください。

キーヤーの動作を変えるコマンドは K3NG OM のサイトに説明が出ています。

このキーヤーでは4つのメモリーが使えます。

図 18



パドルは図の Paddle と書かれているジャックに接続します。

送信機は2台まで接続でき、それぞれ Mem 1 ボタンと Mem 2 ボタンを長押しする事で送信機を切り替える事ができます。

Mem 1 から Mem 4 までのボタンを押すと、それぞれプログラムされた内容が送信されます。

#### 4. コマンド一覧

コマンドを入力するには、コマンド・ボタンを押して、次のコードをパドルから入力します。コマンド・モードの間は Arduino 基板の LED が点灯しています。

- A - Switch to Iambic A mode
- B - Switch to Iambic B mode
- D - Switch to Ultimatic mode
- E - Announce the speed in WPM
- F - Adjust sidetone frequency
- G - Switch to bug mode
- I - TX enable / disable
- J - Dah to dit ratio adjust
- N - Toggle paddle reverse
- O - Toggle sidetone on / off
- P# - Program a memory (#には数字をパドルから入力します)
- S - Alphabet Send Practice
- T - Tune mode
- V - Toggle potentiometer active / inactive
- W - Change speed
- X - Exit command mode (you can also press the command button (button0) to exit)
- Z - Autospace On/Off
- # - Play a memory without transmitting

コマンドの実行にはコマンド SW を押して、パドルからコマンドの文字を打ちます。コマンドが受け付けられると、ピープ音が ”ピッ” と鳴り、次にそれぞれのコマンドに応じた動作をします。

メモリー機能を使うには、コマンド・ボタンを押し、P1 と打つと ”ピッ” と音がするので、登録するキー・ストロークを打ちます。終わったらコマンド・ボタンを押すと、登録した内容が再生されます。再度、コマンド・ボタンを押してコマンド・モードから抜けます。メモリーした内容を再生するには、それぞれのボタンをチョンと押すと、再生されます。

コマンド・モードの状態は Arduino の D13 に接続された LED で表示されており、コマンド・モードでは点灯状態になります。この LED は、Arduino の基板上で L というラベルが付いた LED です。

このシールドでは送信機を 2 台接続して、切り替えながら使えるようになっています。それぞれ該当するメモリーのボタンを長押しして離すと、ボタンに応じて TX1 とか、TX2 とか、切り替えた結果を返してきます。これで、一つのパドルで 2 台の送信機(トランシーバ)を切り替えて使えます。