

JK1DVP

DVLogger 取り扱い説明書

2025年 改版 10 版

Eiichiro Araki

2025年 8月 19 日

目次

1 はじめに	7
2 2024 年版からの更新について	8
2.1 mini 版.....	8
2.2 Wide 版.....	8
3 本器で何ができるのか？	10
4 同梱品・別途用意品.....	12
5 接続・設定.....	15
5.1 全体端子構成.....	15
5.1.1 mini 版.....	15
5.1.2 Wide 版.....	16
5.2 リグの接続	17
5.2.1 IC-705, (IC-905)	17
5.2.2 IC-9700, IC-7300, IC820 等 ICOM リグ	19
5.2.3 FTDX10, FT-991A 等 Yaesu リグ	20

5.2.4 Kenwood 社リグ	21
5.2.5 QRP-Labs QCX-mini 等	21
5.2.6 マニュアルリグ	22
5.2.7 Alinco DJ-G7	22
5.3 ヘッドホン切替の接続	22
5.4 マイク・P T T切替の接続	23
5.5 パドルの接続	23
5.6 起動後設定	23
6 基本操作	26
6.1 運用画面とキーボード操作	26
6.2 CW キーイング	29
6.2.1 和文 CW	31
6.2.2 F2A	32
6.3 バンドマップ	34

6.4	マルチチェック	36
6.5	運用状況確認画面	37
6.6	リグ (Radio) の切り替え.....	37
7	コンテスト運用	39
7.1	コンテストの設定	39
7.2	ログの初期化.....	42
7.3	その他の設定.....	42
7.4	デュープチェックについての注意.....	43
7.5	CQ/S&P 切替.....	44
7.6	コンテスト QSO	44
7.7	コールヒストリ検索.....	45
7.8	スーパーパーシャルチェック	46
7.9	運用続き・・・SO2R まで	47
7.10	過去ログの参照・編集.....	50

7.11 運用 Radio 切替、ヘッドホン切替	51
7.12 運用情報を調べる・・・	51
7.13 運用終わって、ログ吸出し.....	52
8 サテライト運用	54
8.1 衛星の選択 準備	54
8.2 衛星の AOS 検索と選択	55
8.3 運用	55
8.4 衛星リスト	57
9 信号強度測定、その他の制御	58
10 ネットワーク接続	59
10.1 zserver への接続(zLog ネットワーキングへの参加).....	59
10.2 TCP サーバー.....	59
10.3 Web サーバー.....	60
10.3.1 ログダウンロード	60

10.3.2 POTA 支援機能.....	64
10.3.3 SOTA 支援機能.....	66
10.3.4 Web での運用機能	67
10.3.5 Web リグ設定.....	68
10.3.6 Web 口ガー設定.....	69
10.3.7 ファイルアップロードについて	70
11 ふつうの運用.....	72
11.1 Remarks に記録されるデータについて	73
12 音声入出力関連機能.....	75
12.2 F2A・音声合成・イヤホンマイク入出力.....	75
12.3 音声データの作り方.....	76
13 機能拡張・プログラムの変更などについて	79
14 表	81
表 14.1 コマンド一覧（Rcv コールサイン欄にキー入力）	81

表 14.2 キーボードショートカット一覧	85
表 14.3 運用ウインドウ一覧.....	90
表 14.4 モード名称.....	91
表 14.5 キー CW 符号対応表.....	91
表 14.6 CW メッセージマクロ一覧.....	92
表 14.7 PC terminal からのコマンド一覧 (help で表示)	93
表 14.8 リグ設定文字列の意味	97
15 ハードウェア構成について（回路図）	99
15.1 Mini 版 (revD, revE)	99
15.2 サブ CPU 書き込み改造について	105
15.3 Wide 版(revF, revG)について	106
16 本品について	114
17 謝辞	114

1 はじめに

みなさん運用ログ、コンテストログなどは PC を使って取っていると思いますが、

「設定が間違ってログが保存されてなかった！」「運用中にログファイルが失われた！」

「PC の設定が難しくて、ちゃんとコンテストやれるかわからない・・・」

「リグコンの設定が思うようにいかない、コンテスト中におかしな動作になってしまった！」

「CW を PC から打つのがやりづらいから、キーヤーの並列つなぎが手放せない。」

「野外運用で PC ロギングをしてみたいけど、PC は重いし電池が持たない・・・」

「リグを複数つないで、切り替えながら運用したい (SO2R)」

ということはないでしょうか？

本器は、そのような悩みへの一つの答として開発いたしました。私 JK1DVP は、2021 年秋の開発開始からこれまで、ほとんどの場合、これを使ってコンテストに参加してきましたし、いくつかのコンテストでは、ロケの要素が大きいところですが、部門第一位を得ることができましたので、コンテストをより楽しむための道具としてご活用いただければ幸いです。

本年は、昨年版をベースとしてパドルコネクタを備えながらもよりコンパクトになった mini 版に加え、見やすい大画面、リグ・キーボードインターフェース他の追加、メモリ拡張により高機能となった Wide 版を頒布いたします。

jk1dvp 荒木 2025/8/15 更新

2 2024 年版からの更新について

ハムフェア 2024 に頒布した版をベースとした mini 版、大画面の Wide 版を開発、ハードウェアは、下記のアップデートが施されています。

2.1 mini 版

パドル接続端子が付きました。今後拡張のため、サブ CPU のプログラム書き込みが行えるよう配線を行いました（詳細、回路説明を参照ください）。昨年版と比べてより携帯に適した小さなサイズの 3D プリントしたケースに収めました。また、ケースにはビルトインのスタンドが備わり、よりフィールドでのセットアップが楽になりました。

2.2 Wide 版

2.4" OLED 2 画面を搭載し、メイン/サブ C P U を含め 4 層基板 1 枚にまとめています。端子はパドル等 mini 版に備わっているものはすべて備わっていますが、リグインターフェースを、Yaesu/Kenwood 向け C A T 端子を 2 系統に増設。また、キーボード専用 USB 端子を追加したため、リグの接続をこれまでの USB 端子を使って行えるようになりました (IC-905 等)。また、音声関係は 2 系統のリグヘッドホン入力に加え、リグ 1 のマイク出力を備えています。

イヤホンマイクを接続すれば、リグにイヤホンマイクの音声と、本器で再生する F2A 電信や音声読み上げ（フォネティック）音声をミキシングしてリグに送れます。リグの音声関係 I/F はヘッドホン/マイクとも 3 系統の切り替えができる拡張設計となっていますが、本器には、そのうちヘッドホン 2 系統、マイク 1 系統を備え、フィールド運用での多くをカバーする設定としました。さらに、Wide 版では、PSRAM を 8Mbytes 増設しており、多局のデュープチェックや、高速のスーパーパーシャルチェック、クラスタ情報等、メモリ不足のため実現困難(または、安定稼働が困難)なアプリケーションが可能となりました。また、地味ですが、CPU モジュールを組み込んだことで USB-C に完全対応しました（これまで USB-C – USB-A ケーブルでの接続が必要でした）。

3 本器で何ができるのか？

- ・手のひらに乗るサイズで、コンテストログとして必要な機能が凝縮・網羅。
- ・手のひらサイズのモバイルバッテリーにつないで 1 日以上連続で稼働が可能な低消費電力。山岳移動にも対応。
- ・ログは内蔵 microSD カードに記録され、トラブル・ミスなどで QSO 記録が失われることはありません。
- ・クラスタに自動接続、クラスタ情報を取り込んだバンドマップで自在に呼びまわり可能。
- ・2 系統のリグを接続、自在に切り替えながら運用が可能。
- ・リグコントロールと合わせ、2 系統のリグのヘッドホンを切替、同時に受信など可能。
- ・CW キーイングを接続したキーボードで自在に行える。もう CW キーヤー + パドルは必要ない！
- ・リグコンの設定が容易。リグをリストから選ぶだけ。
- ・zserver (zLog のデータベースサーバー) に接続、運用ログの同期が可能。

マルチオペのノード子機として使用可能。

- ・サテライト通信に対応。自動でネットから TLE をダウンロード・更新、AOS の計算、アップリンク・ダウンリンク周波数の補正コントロール・ローテーター制御（開発中）など。IC-9700、IC-820、IC-705 2 台、FT-991AM など制御。
- ・F2A 電信ができます。
- ・RTTY のキーイングもできるようにしていますが、AFSK、FSK などリグごとの差異が大きいため、完成度はこれからです。
- ・地味な売りですが、本器には RTC が付いており、ネットワーク(NTP nict.jp サーバー利用)で自動的に時計が合わせられるとともに、山でネットが使えない環境でも正しい時刻を刻みます。

4 同梱品・別途用意品

頒布品は、mini 版、Wide 版とも組み立て完全動作確認済みです。

運用のためには、本器に

・ **リグとの制御通信（リグコン）接続ケーブル** 下記は、JK1DVP が動作検証をしたものです。

アイコム IC-705 不要（Bluetooth で接続）

アイコム IC-9700, IC-820 CI-V 3.5mm ヘッドホンケーブル

QRP-Labs QCX-mini 等 CAT 3.5mm ヘッドホンケーブル

ヤエス FT-991A, FTDX10 等: CAT DB9-3.5mm ステレオヘッドホンケーブル シリアル通信 3.5mm RS232C tip:TX(LOG → Rig, DB9-pin3 SERIAL IN) ring:RX(Rig → Log, DB9-pin2 SERIAL OUT) sleeve: GND のように結線をしてください。ボーレートは 38400bps timeout を 100ms 程度に設定してください（長すぎても短すぎても通信がうまくいかないようです）。

あるいは、FTDI 社製 シリアル-USB ケーブルを USB2 ハブを介してリグと接続することも可能ですが、あまり動作をしっかり確認できていません。

Kenwood 社製リグ：DB9 シリアル端子がある機材では上記 YAESU 機と同様に DB9 – 3.5mm ヘッドホンケーブルでのシリアル通信で接続が可能だと思いますが、Kenwood の最近のリグを持っていないので確認ができていません。

- ・ **CW キーイングケーブル** 3.5mm ヘッドホンケーブル tip -GND 間でフォトトランジスタで断続を行っています。Tip 側に負極性をかけることができませんので、昔の管球リグにつなぐには間に回路が必要です。
- ・ F2A は各リグのマイク端子—3.5mm オーディオケーブル(MIC, PTT)が必要です。F2A 運用時には、KEY1 端子の ring に 600Hz の音響信号が出力され、tip (通常 CW キーイング)で PTT を出力しています。開発中の拡張マイク切替基板では、リグへのマイク接続そのままで F2A を運用できるようにしています。
- ・ **リグ—ヘッドホン切替接続** は 3.5mm ステレオヘッドホンケーブルを使ってつなぎます。ステレオにした場合 1 番が左、2 番が右に聞こえるようにしています。このように、接続コネクターは、入手性を優先して 3.5mm ステレオが多いです。秋月などで用意してください。ヘッドホンの切り替えは、口ガードとは GND が絶縁されていますが、ヘッドホン端子相互は GND がつながっていますので、どちらか片側のリグへの接続は、グラウンドループアイソレーター (Amazon などで買えます)を入れたほうが良いと思います。
- ・ **PC との接続や、モバイルバッテリー等からの動作電源供給** は本体 USB-C コネクターで行います。Mini 版では、USB-C-USB-A 接続ケーブルを用意してください。USB-C - USB-C ケーブルを使った場合は、採用した内蔵の esp32 devkitC の作りのせいで、給電がなされないようです (devkitC ボード上に 1005 チップ抵抗をはんだ付けすれば動作するようには変更がおそらく可能ですが、未検証)。Wide 版では、USB-C に完全対応しましたので、USB-C 直繋ぎで OK です。
- ・ **キーボード** USB 有線キーボードか USB ポートに Bluetooth ドングルをつけた場合 Bluetooth キーボードでの接続を確認していますが、USB 有線

キーボードがおすすめです。好きなキーボード（英語・日本語）をお使いください。JK1DVP は Thinkpad Trackpoint 英語キーボード を使っていますが、Elecom 等の日本語キーボードでも動作は確認しています。

- ・ mini 版、Wide 版どちらも、ビルトインのスタンドが備わっています。
必要に応じ開いて使ってください。マジックテープ等でリグ、ポール等に貼り付けるのもおすすめですが。

5 接続・設定

5.1 全体端子構成

5.1.1 mini 版

正面には、2画面の1.3"OLEDディスプレイがあります。背面には、収納できるスタンドが備わっています。



左面には、キーボード接続用のUSB-Aコネクタ、その下にパドル端子、左からヘッドホン出力、ヘッドホン1入力、ヘッドホン2入力、CI-V接続、CAT(シリアル)接続と3.5mmオーディオ端子が並んでいます。



右側には、モバイルバッテリーや PC と接続するための USB-C コネクター、
CW キーイング端子(右から KEY1, KEY2)があります。

5.1.2 Wide 版

正面には 2.4" OLED 画面が 2 画面あります。背面には、折り畳み式のスタン
ドが前方と後方の 2 か所が収納されています。



右面には、右から、イヤホンマイク端子 (4pin) , ヘッドホン 1 入力、ヘッド
ホン 2 入力、マイク 1 出力、CI-V 端子, CAT1 端子 (3pin) , CAT2 端子
(3pin) と並んでいます。



左面には、右から USB-C 電源・PC 接続端子、USB-A 端子、キーボード専用 USB 端子 (KBD)、パドル端子(PDL), CW キー端子(KEY2, KEY1)の順に並んでいます。

5.2 リグの接続

まず、各リグ別に口ガーとの接続方法を記します。

5.2.1 IC-705, (IC-905)

Bluetooth でリグと接続します。IC-705 の MENU-SET-Bluetooth 設定-ペアリング/接続-機器検索で DVLogger 内に内蔵された Bluetooth モデム (ESP32-BT) を検索し、接続してください。接続がされたら、ESP32-BT (接続中) と出ると思います。自動接続を ON にしておくと、リグの電源を入れなおした際には自動でつながるので便利です。ただし、口ガーの電源を入れなおした際には、切断されますので、手動で再接続が必要となります。また、外部端子-CI-V の設定は、CI-V アドレスは A4h CI-V トランシーブは OFF としてください。

CW keying は KEY2 を使うように設定されています（変更可能）。

口ガーを立ち上げたら、Radio0（左）に設定する場合は、**Alt-HOME**, Radio1（右）に設定する場合は、**Shift+Alt-HOME** で IC-705 を選びます。正しく接続がされたか、周波数・モードがログでリグと同じ表示になっていることを確認してみてください。

Alt-Delete キーでリグの設定状態などを見られるようになっています。リグの横に * が付いていない場合には、リグコンが有効になっていません。

Alt-End キーで * を付けてください。

また、右画面で **TAB** を押して **Spec** を選ぶと、現在のリグ設定の詳細を確認するとともに変更（Enter で反映）することができます。CW キーのポート、CI-V アドレス、シリアルボーレート、またリグの名前（NAME）など変更が可能です。

名前を変更（**NAME:**）できますので、こちらで新しい名前を上書き定義して、自分用のリグの設定を作ることができます。

また **RIG** には選んだリグの名前が表示されていますが、こちらにリグの名前を入力して Enter を押すと、リグのリストから検索を行い、最もよく一致したリグを設定します。



図 5.1.1 リグ設定画面

左画面)

リグ運用モード(SO1R/SAT/SO2R) 本器 IP アドレス

Rig0~2: リグ名 *はリグコン有効

Callhist: コールヒストリファイル名

Settings: 設定ファイル名

右画面) Tab 押して **Spec** を選ぶと Radio 毎に設定されたリグの設定を見ることができる。内容を変更した場合には Enter で変更が反映される。

CW:1, TP:3_4, P:-2 … リグ設定、詳細は **表 14.8 参照**

IC-705 に関しては、SG-Lab 社製 1200MHz 帯トランシーバー(IF 145MHz RF 1295MHz)の設定、コスモウェーブ社製 2400MHz 帯トランシーバーの設定がされています。**Alt-X** を押すことで、IF-RF の周波数変換が自動的になされて、トランシーバーを使った SHF 運用が快適に可能です（後述）。

また、**Wide 版**では、**IC-905 や IC-705 を USB 端子に接続して運用することもできます**。これは衛星通信、SO2R 等 2 台の IC-705 を同時に運用するのに便利です。この場合、IC-705 の USB 端子は、本器の電源投入後 USB に接続するようにしてください（初期化の問題かと思いますが今のところ修正できていません）。また、リグの選択は IC-705-USB という名前で登録されています。接続後、リグを **IC-705-USB** と選択、Enter すると接続できるはずです。今のところソフトウェアの問題から、多少手順が煩雑ですが、接続できるはず。IC-905 も同様に、RIG から IC-905 を入力して選択してください。

5.2.2 IC-9700, IC-7300, IC820 等 ICOM リグ

背面の Remote(CI-V)コネクターから 3.5mm オーディオプラグを使って、CI-V コネクターに接続します。RIG は IC-9700 や IC820, IC-7300

を選択してください。(ほかの ICOM リグでも CI-V アドレスを変更（リグ側だけでなく口ガード側で変更が可能）することで接続が可能だと思います。IC-9700 については、サテライト関連や、リグの S メータの dBm 変換の設定が特別にされています。

また、ID-31 等の CI-V コネクターがあるリグについても、IC820 等を選択し、CI-V アドレスを Spec 欄で編集して変更することにより接続できます。ID-52 等の Bluetooth で接続できるリグについては、IC-705 を選択し、こちらも CI-V アドレスを変更することによって接続が可能かと思います。また、IC-7300 以前のリグでは、通信ボーレートの上限が 19200bps のようですので、そのあたりも参考にしてください。

5.2.3 FTDX10, FT-991A 等 Yaesu リグ

背面の RS232C コネクターから DB9F-3.5mm オーディオプラグケーブル（図 5.1.2）をつけて、CAT コネクターに接続をしてください。RIG は FTDX10 や FT-991A-SER を選択してください。FTDX10, FT-991A などの違いは、運用可能バンドの違いが反映されるようになっているほか、リグの S メーターから dBm を読み取るためにリグごとに異なる感度設定情報が反映されるようになっています。YAESU リグの場合、リグの ID を読み取って、感度設定があるリグのみ dBm 値を読むようになっていますが、他のリグの場合でも、S メーター読み取り値は記録されます。CAT のボーレートは、38400bps に設定をしてください。また、CAT のタイムアウトを設定できるかと思いますが、これは 100ms 程度にしてもらえると接続ができると思います。

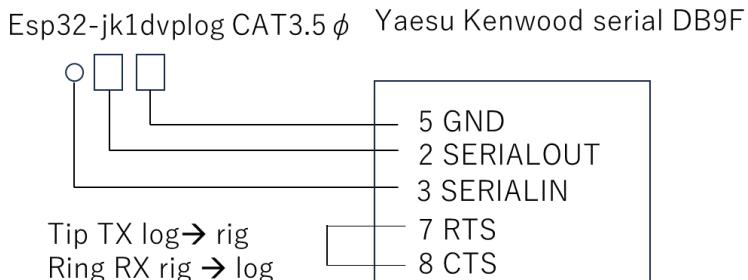
CW キーイングは、ICOM リグと同様に KEY1 または KEY2 から KEY 端子に接続を行ってください。

Yaesu リグ(FTDX10)に関しては、電源投入時に CAT ケーブルを繋いでいると、リグがファームウェアアップデートモードになることがあるようです。このあたりの動作がはっきりしないのですが、ご注意のほどお願いします。

また、FTDX3000 等古めの Yaesu リグでは CAT コマンドの周波数桁数等のルールが異なっています。これは FTDX3000 とリグ名を入力することで対応しています。

5.2.4 Kenwood 社リグ

Kenwood リグは、Yaesu リグと同様に CAT ポートに RS232C ポートから接続をしてください。リグは KENWOOD-SER を選んでください。



Kenwood では特に RTS-CTS の結線が必要です。

図 5.1.2 CAT シリアル接続ケーブル配線（例）

5.2.5 QRP-Labs QCX-mini 等

本体の CAT コネクターから 3.5mm オーディオステレオプラグで CAT コネクターに直結できます。QCX-MINI を選んでください。Yaesu, Kenwood 等は RS232C レベルですが、こちらは TTL レベルで、またシリアルデータ

タのビット極性が異なりますが、つながるように対応をしています。

5.2.6 マニュアルリグ

リグコンができないハンディ機などのリグも MANUAL を選ぶことで使用できます。ヘッドホン切替と、ログ入力が同期しますので、便利です。

リグのモード、バンド、周波数などは、Alt-m, Alt-'<,' , Alt-'>.' , 周波数 (kHz) 入力などで合わせてください。

また、他のリグでリグコンがうまくいっていない場合など、さしあたり MANUAL を選ぶことでログや CW キーイングが可能です。

5.2.7 Alinco DJ-G7

こちらもマニュアルリグですが、コスモウェーブ社製 5.6GHz トランスバーターの使用ができるようなトランスバーター設定がされています。DJ-G7 を選んでください。フィールドデーコンテストの際、こちらで 5.6GHz 帯で F2A 運用を楽しみました。

5.3 ヘッドホン切替の接続

リグのヘッドホン出力をヘッドホン 1, ヘッドホン 2 に接続することで、ロガーで切替 SO2R 運用が可能です。ステレオで同時に聴く場合には、Radio0 がヘッドホン 1 で左側、Radio1 がヘッドホン 2 で右側から聞こえるようになっていますので、Radio0 に設定するリグを左側に配置するのが

おすすめです。

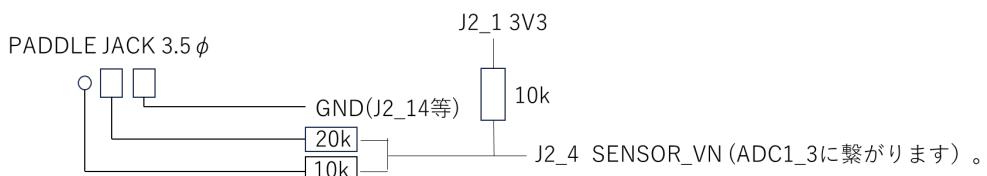
5.4 マイク・PTT切替の接続

Wide 版では、ヘッドホンと同様にリグへの MIC・PTT 接続も切り替え SO2R 運用ができます。F2A, AFSK なども MIC から接続されるようになります。

本体には MIC1 のみが装備されています。配線は TIP が MIC RING1 が 8V (マイク電源) 入力 RING2 が PTT(GND に落ちると送信), SLEEVE が GND です。RING1 の 8V 電源入力ですが、内部にマイクと TONECW・音声出力のミキサアンプが入っていますので、MIC1 からの電源入力は必須です。

5.5 パドルの接続

ハムフェアでの声を受けてパドルを下記の配線で繋げられるようにしました (本年配布版ではコネクター含め実装されています)。



パドルは IAMBIC-A/B, リバース/ノーマルの切り替えが可能です。

5.6 起動後設定

基本、キーボードを使った操作となりますので、6. 基本操作も含めてご覧ください。

Tab/Shift-Tab で項目間を移動します。

Wifi の SSID, Password を設定し、各項目で Enter で Wifi ネットワークに接続します。

本器はこの手順を繰り返すことで複数の SSID を設定することができます、最も強い Wifi に接続します。設定された SSID PASSWD は、/wifiset.txt というファイルにアップデートされていきます。電源投入時に設定された wifi 設定を自動的に読み込みます。もし、WIFI 設定をリセットしたくなつた場合は、PC 接続して reset_settings コマンドで設定ファイルを消去されますので、reset_settings をして save するとよいでしょう。

設定は **SAVE** コマンドで保存できます。SAVEname とすると名前付けしたファイルに設定を保存できます。設定は無名の設定ファイル(settings.txt)が起動時に読み込まれます。ので、こちらにコンテストの設定を保存しておくことをお勧めします。LOAD コマンド（または、LOADname）で保存した設定内容を読み込めます。

また、他に、設定が必要な項目は下の通りですので、起動時に設定し、SAVE することをお勧めします。

自局コールサイン, コンテスト送出ナンバー[Snt]

電力コード [Pwr.] JARL コンテストの際に送るコードですが、バンド毎に変更できるようになっています。

Grid Locator [G.L.] サテライト通信で自局位置の設定に使います。

JCC/JCG コード [JCCG] オプション CW で情報の自動送信に便利です。

cluster アドレス、ポート[Clist] (jg1vgx をデフォルトで入れてあります)

zserver アドレス

CW 送出メッセージ (ファンクションキーF1-F6 に登録) Alt-C CW キーイングの項参照

RTTY 送出メッセージ

コンテスト種別(Ctrl-2 で設定)

SO1R 等種別(Ctrl-5 で設定)

ハンドル名[Name] CW送出マクロあり。ハンドル名を入れます。

6 基本操作

6.1 運用画面とキーボード操作

右側の画面（図 6.1.1）は運用のための画面、左側は、情報表示（バンドマップ、ヘルプ等）の画面です。



図 6.1.1 右画面) 運用画面 左画面) 情報表示画面

右画面では、上から、

1 行目) 周波数・モード

2 行目) 現在選択のリグ#(0,1) 現在時刻、QSO 数 (SEQNR) /S 信号強度、

3 行目) 入力項目名 (Rcvd, Sent, RST, ...) CW キーヤーモード (K)、マルチ、デュープの状況

4 行目) 文字入力 Callsign ナンバ 左右 2 つ または 1 行遠し

5 行目) 現在送信されつつある CW/RTTY 文字列が流れます。

というように表示がされます。

入力項目名それぞれにキー入力ができるようになっていますが、入力できる文字の種類などは、入力項目によって異なります。例えば、Callsign 入

力欄では、コールサインは必ず大文字で表示されますが、Remarks 欄では、スペースなどすべての文字が入力可能となっています。

項目間の移動は TAB (前進)、Shift+TAB (後退) で行いますが、コンテスト運用では、スペースで移動できると楽ですので、**Callsign と Rcvd の間はスペースを押すと行ったり来たりできるようになっています**。他の欄もスペース、Shift+スペースは TAB と同様に移動するのですが、スペースが入力できる欄に関しては、この機能が無効になっています。

BS, DEL, カーソルキーで入力文字列をエディットできます。が、PC のようにキーリピートは実装されていませんので、1 個 1 個入力をしてください（実装を後回しにしてますので、今後実装されるかも）。

また、**CapsLock キーは Ctrl として働きます** (JK1DVP の趣味で)。

Ctrl-C でコールサイン入力欄 Ctrl-E で受信ナンバー欄、Ctrl-R で Remarks 欄、Ctrl-S で受信 RST 欄、Ctrl-A で送信 RST 欄に直接飛びます。

Ctrl-B でパーシャルチェックを実行。

相手コールサイン入力ウィンドウ(Rcv)は、特別なウィンドウで、リグの制御やコマンドを入力するためにも使います。例えば、ここに、周波数を kHz 単位で入れて Enter を入力すると、リグの運用周波数を変更します。また、モード (FM, USB, LSB, CW) を入力すると、リグのモードを変更します。ここに入力するコマンドは、ログの読み出し(READQSOLOG)など **表 14.1** のように様々な動作をさせるためのコマンドがあります。

TAB を押していくと、ロガーの動作パラメターを入力する欄がたくさん

でできますが、多くの欄では Enter を入力することで、入力したパラメタ一を適用して動作をさせるようになっています。これは表 14.3 にまとめます。

キーボードショートカットキーが多数あります。これは表 14.2 にまとめますが、運用ログ関係が Ctrl+, リグ・周波数・モード・バンドマップ関係が Alt+ に集めてあります。また、Shift は Radio 番号に関係と、CW キーイング関係あります。

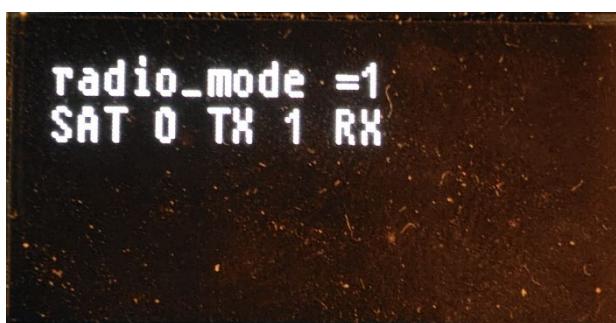


図 6.1.2 運用モード表示 1 サテライトモード Radio0 TX Radio 1 RX となっている。

Ctrl-5 で運用モードを SO1R/Satellite/SO2R で切り替えます。SO1R と SO2R の違いは実はあまりありませんが、CQ を出す際に受信リグを切り替えるときには関係してきます。

BANDENhex コマンド、BANDMASKhex コマンドは、各 Radio で運用可能なバンドを決めます。リグによっての制約もあると思いますが、運用可能なアンテナが付いているかどうかによっても変わります。hex の指定がないときには、現在の設定を示します。

BANDMAP コマンドによって、クラスタからバンドマップに取り込む対象

とするバンドを絞ることができます。

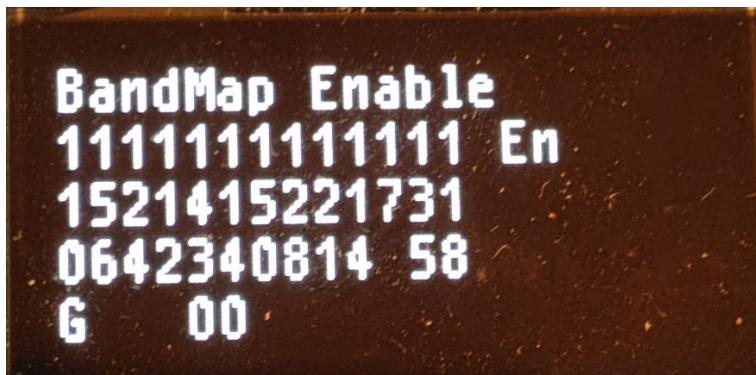


図 6.1.3 リグごとの運用許可バンド表示 $En=1$ となっているバンドが運用できる。

6.2 CW キーイング

CW キーイングは zLog, n1mm+ 等と同様に Ctrl-K でキーヤーモードに入ることで、入力した文字がキーイングされますし、キーヤーモードかどうかは、画面に K で示されています。が、キーヤーモードって、ミスが起きやすく自在にキーイングするのは難しいかなと思い、本器では他であまり例を見ないですが、**Shift を押しながらキーを押すと、そのキーが CW で直接送出される**ようにしました。これを使うと、ちょっとしたメッセージのやり取りなど、パドルを使わずとも電文を送信できます。

JK1DVP はこれを実装してからは、キーヤーモードは 1 度も使っていませんし、パドルも 1 度も使っていませんが、なれるとかなり便利です。

ただし、キーボードを見ればわかる通り、よく使う?と/は同じキーに割り当てられているため、この方法が使えません。そのため、**/については、隣のキーに割り当ててあります(Shift+' (英語), Shift+ バックスラッシュ (日本語))**。ちょっと慣れが必要ですが。その他、BT, AR なども表 14.5 の通りキー割り当てがされています。

日本語キーボードについては、**KBDJPN** コマンドを打つことでキーマップを切り替えます。**KBDENG** コマンドで英語キーボードに戻します。また、この変更は **SAVE** コマンドで保存されます。

キーイングの速度は、**PGUP, PGDN** で 10-35WPM の範囲で変更できます。

ファンクションキーで CQ など CW のメッセージを送出できます。その割り当ては n1mm+と同じです。

F1 CQ

F2 コンテストナンバー

F3 TU Callsign TEST

F4 自局コールサイン

F5 相手コールサイン+ナンバー

F7 まで定義できるようになっています。

定義は Alt-C を押してから、Tab で項目を選び編集できます。このメッセージで使えるマクロ(\$C など・・・)は表 14.6 にまとめています。もう一度 Alt-C を押すと、RTTY のメッセージの編集ができます (RT の表示に変わる)。もう一度 Alt-C で抜けます。



図 6.2.1 CW メッセージ編集中 (F1 キー)

繰り返しになりますが、**Alt-C** をもう一度押すと、**RTTY** の送出メッセージも同様に編集が行えます。こちらは PTT の制御などが必要なため、マクロが少し追加になっています。もう一度 Alt-C を押すと、通常入力に戻ります。

メッセージの送信中などに、**Esc** を押すと、CW メッセージの送出をバッファに溜めている文字も含め停止します。

また、Shift+BS で送出されつつあるメッセージの 1 文字削除などが可能ですが、うまく使いこなすことは困難かと思います。

ファンクションキーは Ctrl を押しながら押すと、繰り返しリピート送出が可能となっています。CQ をただ繰り返し出したい時などに使えます。

呼ばれるなどした場合には ESC を押して繰り返しを止めてください。

また、Shift+ファンクションキーは、SO2R モードでは送出しているときに、別の Radio を聞くようになっています。Shift+Ctrl+ファンクションキーで繰り返しそのような形で CQ などを送出できます。

6.2.1 和文 CW

'{' (英語) @ (日本語) キーを CW で入力することで、ホレを打って和文 CW を入力するモードにスイッチします。再び '}' (英語) [(日本語) でラタを打って英文に戻ります。右画面にはキーヤーモードの時には K で表示していますが、和文モードに入ると J になります。また、Shift+文字で CW を入力しているときに和文モードに入っているときには、j で和文モードであ

ることを表示します。

和文モードの時に'(',')'で英文/和文を切り替えます。

入力はローマ字を訓令式で入れると、和文カナに変換して和文を送信しますが、下には和文でなくローマ字のまま送信されますので、違和感があるかも。ローマ字和文変換は訓令式ですが、反響を見てヘボン式でも打てるようになります。和文オペする方のご意見をいただきたいところです。

NNで'ん'を送信します。

6.2.2 F2A

Alt-g で、**Tone keying** を有効にすると、**mini** 版では、**KEY1 端子の R (Ring)** に **600Hz** トーンが出力され、**L(Tip)** は、**PTT** として動作します。図 6.2.2.1 のような回路 (DJ-G7 用) の接続ケーブルで KEY1 端子とリグの MIC 端子を接続することで F2A で CW 運用をすることが可能です。

DJ-G7 で試したところ、頭切れがありましたので、PTT ON からキーイングまで 200 ms のディレイを入れてあります。また、打鍵終了後 150ms で受信に戻ります。また、Tone keying を ON にしているときには、リグのモードが電話モード (FM, AM, SSB 等) か CW モードかにかかわらず、ログには CW として扱われるようになっています。また、RST には 599 が自動的にセットされます。また、ログには、運用が F2A 電信等であることを示すために、Remarks に F2A と記録されます (AM、SSB 等に設定していたときも F2A と出ます。手抜き Hi)。



図 6.2.2.1 DJ-G7 F2A 接続ケーブル回路（例）

Wide 版では、Alt-g で F2A 等 TONE CW が運用できるのは同じですが、TONE が出力されるのが MIC 端子なのが違います。 MIC 端子は、ほぼ IC-705 互換で図 6.2.2.2 に示す通りですが、図に示したような PTT、サイズ変換ケーブルを使うことで IC-705 にイヤホンマイクからのマイク信号と混合して TONE・音声を送り込むことが可能です。

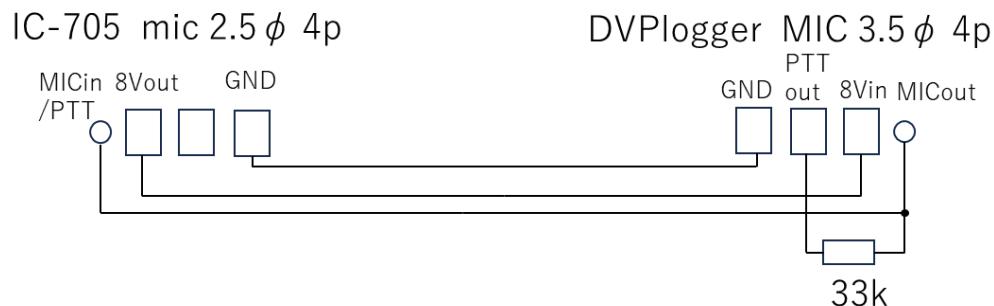


図 6.2.2.2 IC-705 MIC 接続ケーブル回路（例）

6.3 バンドマップ[°]

コンテストに限りませんが、左画面にクラスタや運用状況を反映したバンドマップが表示されます。

2014.0	JK1DVP	10
2019.0	JN7DOR	0
2030.4	JH3QFY	0
2019.6	JA2BCQ	1<
2016.2	JL3JRY	2
2016.5	JH7TXX	5

図 6.3.1 バンドマップ表示画面

バンドマップは基本は運用中のバンドを示しますが、**Alt+右左方向キー**で表示バンドを切り替えることができます。表示は左から周波数(kHz) カールサイン スポット情報からの経過時間(分)です。

運用周波数($\pm 100\text{Hz}$ 内)に局がいる場合には、1行目に下線を引いて表示します。すでに **QSO** がされている場合には、——線で消して、**QSO 済み**であることがわかるようになっています。また、**スポット情報が更新されてからの分数を右側に示**しています。デフォルトでは上から一番新しいスポットを表示していますが、**Alt-b** で周波数並び替えと切替ができます。また、一番右側にくで、カーソルが示されていますが、**Alt-上下矢印**で動かせます。これで選んで **Alt-SPACE** を押すと、その周波数に運用周波数を変えます（また、**S&P** になる）。また、選んだコールサインが **Callsign** 欄に自動入力されます。すでに **QSO** の実績がある場合には、**Exchange** も取り込まれます。

最終スポットから 20 分以上経過した情報は消去されます。コンテストの場合と 10 分以上前のスポット情報が有効である局は稀です。

6.4 マルチチェック

マルチチェックが行われた際には左画面にマルチチェック画面が表示されます。

Ctrl-T, Ctrl-Y でコンテスト情報、マルチ画面を出すことができます。繰り返し **Ctrl-Y** でマルチ画面をスクロール表示できます。**Ctrl-T** で戻ります。



図 6.4.1 コンテスト情報、マルチ表示ウィンドウ (左画面)

KCJ コンテストを選択

D : NG C/P は CW/Phone の重複 (Dupe) が認められない

D: OK C/P ですと、CW/Phone 両方とも QSO 可能を示す。

Mult: Kanagawa は入力欄に入力したマルチ(KN)が Kanagawa であることを示す。

Multi in 7MHz は 7MHz で獲得マルチを * で示す (徳島、愛媛、高知、福岡、長崎、熊本マルチはとれたが、佐賀、大分が取れていない・・・ことを示す)

6.5 運用状況確認画面

Ctrl-U で各バンドの運用 QSO 数、マルチの状況、バンドマップ上での未 QSO 局数などを見ることができます。Ctrl-U を続けて押すと各バンドを切り替えて表示します。

1.8	7Q> 0	7M
3.5	55Q> 0	24M
7	129Q>16	39M
14	0Q>18	0M
21	0Q> 5	0M
28	00> 9	0M

図 6.5.1 運用状況確認画面 バンド名 Q で QSO 数 > の右側の数字は、バンドマップ上で未 QSO のスポット数 M で獲得マルチ数

6.6 リグ (Radio) の切り替え

¥キー (英語キーボード)] (日本語)を押すと、リグが Radio0, Radio1 で切り替わります。右画面でどちらの Radio が選択されているか表示されますので確認してください。また、同時にヘッドホンも切り替わります。

`/~キー(英語 1 の左, keycode 0x35)を押すと両方のリグが右左に振り分けられて同時に聞こえるように切替ができます。また、Shift-¥ (英語)Shift-](日本語)を押した場合には、Radio0, Radio1, Radio2 の順で順番に切り替えることができます。Radio2 は多少面倒ですが manual リグを設定するなど便利に使えるかと思います。

当初、あまり考えずに作ったため、音声切り替えは Radio0 が Phone1

に Radio1 が Phone2 に対応します。キーイングは任意の組み合わせを可能とした (Spec 欄で設定) のですが、こちらも一番最初のキー端子が KEY1 となり番号はそれに対応しています。

7 コンテスト運用

7.1 コンテストの設定

本器は、n1mm+やzLogなどのように、立ち上げ時にコンテストを選んだりする必要はありません。ですが、デュープチェック、マルチチェックのために、対象コンテストを選ぶ必要があります。ショートカットキー(Ctrl-2)を押すと、左ウインドウでコンテストの情報が表示されますので、対象コンテストを選んでください。また、Tabを押していく、TEST欄でコンテスト名を入力すると入力した文字列と先頭から比較してマッチしたもののが選ばれます。コンテスト中もいつでも切り替えが可能ですので、例えば2つのコンテストに同時に参加してログを取る、ということも比較的容易にできますし、普通のQSOを途中挟む、なども可能です（が、送出NRの設定を変更していないと記録が混乱する可能性はありますので注意）。

選べるコンテストは、現在のところ下記の通りです（リクエストあれば増やします）。

表 7.1.1 コンテスト一覧

NOMULTI マルチチェックしません（デュープチェックはあります）。コンテスト外運用などはこれが良いと思います。また、下記のコンテストで対応が困難な場合もこちら。

JANoPwr JARLの都府県マルチで、電力コードがないコンテストはこれ。

AllJA JARLの都府県マルチ+電力コードに対応

ACAG 全市全郡コンテスト

FD フィールドデーコンテスト JARL 主催コンテストナンバー、2400MHz
以上は市郡区マルチに対応しています。

CQWW CQZone マルチをチェックしているだけです。

ARRLDX ARRL international contest 地域マルチ 電力を省略して K
などを認めるようにしています。

ARRL10m ARRL 10m コンテスト地域マルチ こちらは DX は serial です
ので、全部が数字のみ入力できるようにしました。

以下口カコンです。

TAMAGAWA 多摩川コンテスト

TOKYOUHF 東京コンテストのマルチなので、東京 CW、東京コンテストなどにも対応

Saitama オール埼玉コンテスト

KCJ KCJ コンテストマルチ

KantoUHF 関東 UHF コンテストマルチ

KanagawaInt オール神奈川県内局

Yokohama オール横浜コンテスト

UEC 電通大コンテスト

Tsurumigawa 鶴見川コンテスト

HSWAS 広島 WAS コンテスト GL マルチ+広島県内市郡区マルチ

TOKAIQSO 東海 QSO コンテスト

UEC_VUS 電通大 VUS コンテスト

ACAG (NoMulti) 全市全郡ですが（電力コード送出）、マルチチェックをしません。

Yamanashi 山梨コンテスト

MusashinoL JR 武蔵野線コンテスト

KCWA KCWA CW コンテスト

以下は、歴史的な経緯に関係したコンテストです。

JA8(int)コンテスト JA 8 コンテストの道内局マルチですが、2023 年のコンテストの時に設定したので、マルチまた変わりましたね（汗 頑張って作ったので、記念に置いています。役に立たない。。。

ACAGnochk 末尾に電力コードが付いていますが、マルチチェックをしません。こちらも歴史的な経緯（市郡区データ入れていない）で残しています。

このように、JK1DVP の趣味で設定しているわけですが、類似のコンテストを選択することで、おおよそ常識的なコンテストでナンバー交換はできるようになっていると思います。ただ、ポイント・得点などの計算は、それこそコンテストごとにかなり違いますし、運用中にこれらの要素を気にすることはほとんどないと思われる所以、あまり真面目に対応していません。CTESTWIN, zLog などにおんぶにだっこです。

7.2 ログの初期化

コンテストを選んだら、これまでの QSO データを別ファイルに移して、今回のコンテスト運用のためのログを初期化するために、**NEWQSOLOG** コマンドを打ってください。NEWQSOLOG をしても、これまでの QSO データはバックアップファイルとして残っており、コマンドで読み出しが可能です。

7.3 その他の設定

あとは、**送出ナンバー**、**コールサイン**の設定が必要です。これは、**Tab を押していくと Snt の欄**に設定します（デフォルトではダミーを入れてあります）。

以前の DVLogger では、この送出ナンバーの欄でマクロが使えませんでしたので、JARL コンテストでは、電力コードを送出しますが、電力コードを除いたものを設定（例えば 11M だったら 11 のみ）してください、とお願いしていましたが、現バージョンでは、**この仕組みをログ記録、CW 送信などで一貫した扱いができるよう変更しました**のでご注意ください。

具体的には、Snt 欄は、\$W マクロで展開される情報を書くようになっています。また、Pow 欄は \$P マacroで展開されるバンド毎のコードを書くようになっています。そのうえで、送信 CW コードは Alt-c で設定する各 Function キーの記述に従って送出されます。例えば、F2 は送出ナンバーですが、**JARL コンテストの場合 県ナンバー+電力コード**ですので、**F2 の欄には、\$V\$W\$P と記載してください**。これで例えば CW で 5NN11P などと送り、ログには 59911P と記載されます。**電力コードのないコンテスト**ですと、**ここを \$V\$W と設定します**。デフォルトでは電力コードを送出

する設定になっていますので、口力コンなどではこれを編集する必要があるということです。

また、**Snt** 欄で設定される\$W マクロは、, または; で区切って **1.8MHz-1200MHz のコードと 2400MHz 以上のコードを切り替えられる** ようになっていますので、Snt 欄に例えば 11, 1115 などと書けば、神奈川県、伊勢原市と切り替えて送信・ログ記載をしてくれます。

Pow 欄の電力コードは、**1.8MHz から各バンドでのコードを羅列**してください。例えば、430MHz まで 50W, それ以上で 5600MHz まで 1~2W のような感じの場合。

MMMMMMMPPPP

と入れてください。これは\$P で展開され、CW 送信、ログ記載に反映されます。その他、**マクロ展開については、表 14.6 をご覧ください。**

7.4 デュープチェックについての注意

MAKEDUPE コマンドで、これまで記録されたログを読み込んでデュープチェックリストなどを再構築することができます。これは、**何かの理由** でログを中断して、電源を切ったりして再起動した場合に、再開するために必要です。これをしないと、シーケンシャル番号が 1 に戻ってしまいますと、デュープチェックが過去の QSO を反映しないのが問題となります。また、**デュープなのに呼んでしまう、などの原因になります**ので、お忘れなく。自動でやればいいのではないか? という意見があるかと思いますが、これを自動でやらないのは、フィールドで何かログが破損する等した場合に、自動で読み込むようにしていると無限におかしなことが続くなどの、絶対に起こってはいけない事態を避けるためです。ご理解ください。

7.5 CQ/S&P 切替

さて、コンテストでは CQ、S&P を切替ながら運用をしますが、本器でも Alt-Q (**n1mm+**と同じ) で CQ, S&P を切り替えます。切り替わると CQ, S&P それぞれの運用周波数がリグに送信されます。**F1 (CQ) を出すと、CQ に切り替わり CQ の周波数を記憶**します。また、運用周波数をリグなどで変更をすると、CQ から S&P に切り替わりますが、CQ を出していった周波数を記憶しているので、再び Alt-Q を押したときに CQ を出していった周波数に戻ってくることが可能ですし、また、Alt-Q で S&P に切り替えることで、その直前に聞いていた周波数を聴くことが可能です。

また、**Alt-Z** で CW、Phone を相互に切替ることが可能です。本器では、CW, Phone それぞれで CQ, S&P の周波数を記憶しており、これによって、Phone, CW を自在に行ったり来たりしての運用が可能です。

PHONE で運用している際に F1 を押すと、CW と同様に CQ になり、現運用周波数が CQ 周波数として登録されます。また、**Ctrl-v** で VOICE メモリーを有効にしている場合には、F1 キーなどを押すとリグの音声メモリーが再生されます。F 2～F 6 でもそれぞれに対応した音声メモリー（M 1～M 6）が再生されます。また、**Wide 版**では、**音声読み上げ機能**が使えます。

7.6 コンテスト QSO

さて、QSO して相手のコールサインを入力しますと、**3 文字以上入れたところでデュープチェック**が働きます。**デュープの場合には、右側画面に D**と知らせます。本器の場合、デュープであっても QSO は普通に行え記録もされます。

7.7 コールヒストリ検索

今回、SD メモリに登録したスーパーパーシャルチェックファイルを読み込み、スーパーパーシャルチェックを行う様変更をしています。ファイル形式は name.PCK で、Web アップロード（9.3 章参照）でアップロードしたファイルを CALLHISTname コマンドを入力することで、設定できます。スーパーパーシャルチェックの読み込みは、Ctrl-4 で ON/OFF 切替によって行えます。

mini 版では、メモリが少ないため、コールヒストリ検索で読み込むとその大きさにもよりますが、不安定→再起動の可能性が増えると思います。将来的には、サブ CPU のメモリがだいぶ余っているので、こちらに機能を移していくと考えているのですが、注意して使ってください。

下記は、旧版の記述ですが、そのまま残しておきます。

また、Ctrl-4 で callhist (コールヒストリ)検索を ON/OFF できます。コールヒストリは、過去に QSO した局とナンバのデータベースを参照する機能です。Esp32 のメモリの制限から、microSD を都度読んで検索をしているため、コールヒストリの検索は遅いです。また、コールヒストリの登録は、PC からシリアルで本器に接続し、callhist_set [*callhistory_filename*] でコールヒストリファイルの書き込みをはじめ、*end* が入力されるまでに入力された、*callsign exchange* 行群で登録を行います。ただし、この *callsign exchange* 行の群は、あらかじめ、*callsign* の *tail letter* または、/エリアのエリア番号の数字で、並び替えてある必要があります（これは少しでも検索を速くするためです）。例えば

JK1DVP/1 11P

JF3TBL 27P

JA1ZLO 10H

JK1DVP 11M

というような順番（1, L, O, P が tail letter）に並び替えておく必要があります。このようにソート処理したコールヒストリは、github で配布する、Python スクリプトなどを使って作成することができます。

さらに、注意点としては PC シリアル接続の場合、microSD への書き込み速度の関係上、PC からの送信に行ごとにディレイを入れるようにしてください。しないと文字化けしてしまうと思います（というわけで遅いし使いにくいので、JK1DVP はあまり使ってません）。次項のスーパーパーシャルチェックのほうがおすすめです。

7.8 スーパーパーシャルチェック

下記のスーパーパーシャルチェック機能は、上の SD メモリのファイルのスーパーパーシャルと統合をしています（たぶん）。

スーパーパーシャルチェックはコールサインを 3 文字入れると発動し、部分マッチがあったコールサインとナンバーを左画面に最大 6 局表示します。

‘キー（英語キーボードでは; の右）：（日本語）を押すと、〈で示されるカーソルが下に動きます。Ctrl-‘（英語）Ctrl-：（日本語）を押すと、〈で示

**されたスーパーパーシャルチェックの項目が運用画面に取り込まれます
(CALLSIGN、ナンバーとも)。**

下記、旧版の記載なのですが、一部有効だと思いますので、残しています。後で、検証して直します。

SD にファイル登録をするコールヒスヒストリのほかに、プログラムに組み込んだコールヒストリを作成することが可能です（こちらは検索が高速、かつ、partial check に対応しています）。大変強力なのですが、これを使うには、本器のプログラムにコールヒストリを文字列の配列 callhist_list として変数に格納する C++ プログラムを加えコンパイルしなおす必要です。

7.9 運用続き・・・SO2R まで

CQ で運用中、コールサイン入力後、；（L の右側のキー）を押すと、相手のコールサインとナンバーを送出(F5 に登録されたもの)し、ナンバー入力画面に移ります。これは n1mm+ の ESM (Enter Sends Message) モードと同様になっています。

また、ナンバー入力を行うとマルチチェックを行います。マルチリストにナンバー入力していた場合には、M と出て、確定ができません。デュープかつマルチも有効でない場合には！と示します。

Enter で確定し、CW などの場合には、F3 のメッセージ（TU コールサイン TEST）を送出し、また、コールサイン入力欄に遷移します。もし、TU を送出せずに確定したい場合は、Shift+Enter を押してください。

また、n1mm+などと同様に、ナンバー送出後にコールサインを修正した場合には、相手の訂正したコールサインの後に TU callsign を送出します。

一方、S&P では、これらの自動メッセージ送信は働きませんので、F4 でコールサイン、F2 でナンバーを送ったりして QSO を進め、Enter で確定してください。F4 でコールサインを送出した際に bandmap に登録されます。また、Alt-n でバンドマップへ CALLSIGN 欄に書いた局として登録することも可能です。

「これは本当の ESM じゃない」という声をもらいましたので、ESM コマンドによって (SAVE で保存されます)、n1mm+と同様の ESM モードになります。具体的には、S&P では Call 欄で Enter を押すと F4 (my callsign) 送出します。コール入力、Space などで Exch 欄に先方のナンバーを入力し Enter を押すと、即確定、かつ、F2 (自局 Exchange) 送出します。CQ では、Call 欄が空欄のときに Enter を押すと、CQ (F1) を送出します。Call 欄にコールサインを入れて Enter を押すと先方の Callsign+Exchange を送出し、Exchange 欄に移動します。ナンバーを入れて Enter を押すと TU (F3) 送出して確定は同じです。

PHONE の場合も同様に Ctrl-v で VOICE メモリを有効 (TU) にした場合には、; でナンバー入力欄に遷移しますが、確定で F3 (M3 ボイスメモリ 3 番目 TU メッセージ) を再生し、コールサイン欄に戻ります (ちょっとやりすぎ感がありますので JK1DVP はあまり使っていませんが)。

さらに、今回 Wide 版では、キーボードから入力したキーを読み上げる機能を実装しました。Ctrl-v を押していくと (TU) の次に (Voicegen) と出

ますが、この時、CQ(F1)では、Alt-cで設定したCW等のメッセージを音声読み上げ機能により送出します。これによって、リグに関わらずボイスメッセージによるCQが可能です。また、Callsign入力後 ';' キーや、Enter(ESMモード時)にはCWと同様に相手のコールサインをフォネティックで読み上げ、ナンバーを和文読みで送ります。具体的には、「じゅりえっとあるふあーわんずるーりまーおすかー ごーきゅーひとひとまいく」などと読み上げます。CQWWではこれでは困ると思いますが、英語版はまだ実装していません。必ず和文読みとなります。さらに、TU送出時にも、音声読み上げを行います（「ありがとうございました。じゅりえっときろわんでるたびくたーぱぱ こんてすと」のように）。音声データは、「四国めたん」のものを添付していますが、データの作り方は、[12.3章](#)で説明します。

S&Pでコールサインを調べたら DUPE、コールサインなどを消す必要が出てきますが、これは Ctrl-W でワイプします。かなり頻繁に使うキーなので慣れておきましょう。

運用時に Ctrl-5 で SO1R/(Satellite)/SO2R の切り替えができますが、下記に SO2R のときの振る舞いについて説明します。

SO2Rのときは、Radio0/1で運用している際には、CQ, Call+Exchange, TU送出時に Radio を切替（Radio0でCQ送出しているときに Radio1 選択）して、他方の Radio で S&P の探索、QSO を行えるようになっています。そのため、CQ, Call+Exchange, TU 送信の音声はヘッドホンで聞こえません（Stereo 設定にすれば聞こえますが）。送出の状態については CW, Phone ともに Radio 1で送出時に下線、Radio2 で送出時に二重下線で表示していますので、参考にしてください。リグ切

り替え時には、バンドマップも切り替わります。このあたりの動作は今後見直しする可能性がありますが、現状そのようになっています。その他、SO2R ではリグの切り替えを頻繁に行いますので [7.11 章](#) も見てください。

7.10 過去ログの参照・編集

過去ログの参照は **Ctrl-o** (**origin**:最後に記録した QSO), **j(current**:現在、最後に表示した QSO), **p** (**prev** : ひとつ前 QSO), **n** (**next** : 次の QSO)で左画面に表示できます（図 7.10.1）。



図 7.10.1 過去ログ表示画面

1 行目) 日付 時間 QSO 番号(Q) 2 行目) 周波数 モード
3 行目) 相手 Callsign ナンバ 4 行目) 自局 Callsign ナンバ
5 行目) Remarks 記載内容 CQ/SP QSOID(zserver 用) S:数字は相手局
信号強度(dBm など)

表示されている **QSO を編集するには、Ctrl-‘-’** で運用画面に取り込みます。編集が終わったら QSO と同様にナンバー入力のところで Enter を押して確定すると、編集が microSD に書き込まれます（**ログデータには*E** が **Remarks** に追加され、追加書き込みがされます）。**編集を破棄したい場合**には **Ctrl-W(ワイプ)**で破棄されます。本ログでは、Q S O の同一性を Q S

○時刻で判断する関係で、QSOの時間を編集することはできません。また、安全性の観点からデータを消去することはできません。QSOを消去する扱いとしたい場合には、Remarksなどに”del”などを書き込みをしてログ提出の際に検索・編集してください。

7.11 運用 Radio 切替、ヘッドホン切替

運用中の Radio(0, 1)は、\ (バックスラッシュ)キー(英語 KBD)] (日本語 KBD) で切替ができます。ヘッドホンもリグの切り替えに合わせ、切り替わります。ヘッドホンは、半角/全角 (日本語 KBD) ~ (1の左側 英語 KBD) を押すと、ステレオで1, 2両方ミックスと切替ができるようになっています。

7.12 運用情報を調べる・・・

デフォルト（またはSMETERコマンド）では右画面の2列目右側に、Radioでの受信信号強度を dBm 単位で示していますが、SEQNR コマンドで、シーケンシャル QSO 番号を表示できます。また、さらに INTERVAL コマンドを打つと、前の確定 QSO からの経過時間（秒）を示します。これで、QSO レートなど把握ができるかと思います。

運用中、各バンドの QSO 数などを確認する場合には、Ctrl-U で表示できます。続けて繰り返し押すと、表示されていないバンドの状況も表示することができます。また、こちらには、クラスタから取り込んだ局のうち未 QSO の局数も表示されていますので、時々チェックすると、他のバンドの状況に気が付くことができると思います。

7.13 運用終わって、ログ吸出し

コンテストの運用が終わったら、書類提出です。ログの回収は、

- 1) DVLogger の Web サーバー機能に接続([10.3 章](#))してダウンロード。
いくつかのフォーマットでダウンロードもでき、速度も速いので**こちらがおすすめ**です (JK1DVP は今やもっぱらこれでやっています)。
- 2) PC に USBシリアル接続、READQSOLOG コマンドによってテキスト形式でログを読み出して、CTESTWIN でファイルのインポート→TXT 形式ログファイルを開く。ログの読み出し、記録は、私は teraterm のログ機能を使って行っていますが、表示させてコピペでももちろんよいでしょう。
- 3) zserver にログをマージして、zLog を使ってその後のログ処理を行う(すみません。これまだ完全にできていません)。
- 4) DUMPQSOLOG コマンドで、内部ログファイル(QSO.TXT)を読み出し PC 上で python で動作するログコンバーターqso2txt.py で同様に変換して CTESTWIN にインポートするなどする(細かい運用周波数などを見たい場合には、これが良いです)。

という方法があります。

それぞれ一長一短ありますが、1, 2)の txt をインポートする場合の注意は、**CTESTWIN はファイルをインポートしただけでは、デュープチェックをして反映してくれないので、その他→入力済みデータの Dupe 確認をすること**が必要です。

また、**本器でログを編集した場合には、元の QSO データはそのまま残**

り、編集した QSO は時刻、SEQNR が同じで Remarks に *E というフラグが付加されて、編集されたことを示しているので、インポートする前に該当のデータを探して削除・編集することが必要です（入力済みデータの Dupe 確認で、デュープとして処理されるとは思いますが、時刻が同じなのでどう扱われるか不明です）。

8 サテライト運用

8.1 衛星の選択 準備

Rcv 入力欄に **SATELLITE** コマンドを入れ Enter すると、衛星データ (TLE) を読み込みます。もし 3 日以内に更新がない場合には、インターネットからダウンロードを行います。また、G.L. 欄に運用地点の**グリッドロケーターを入力**、Enter で設定を行います (save してある場合には、その内容が入力されているはずです)。また、**IC-705 では、GPS 出力を CI-V に出力することができ、その場合、自動的に G.L.欄がセットされます。**

その後、**Sat.** 入力欄に衛星名 (たとえば RS-44) を入力すると、その衛星が選択され、トラッキングが開始されます (図 8.3.1)。Sat.入力欄を空欄にして Enter を押すと、衛星トラッキングをやめます。

トラッキングの方法は、リグによって異なりますが、2つの無線機をつないでいてそれぞれアップリンク・ダウンリンクの周波数に設定されている場合には、自動的にそれらを2台使用することを前提としたトラッキングモードになります。そうでない場合には、単一リグモードにセットされます。

周波数オフセットですが、衛星ごとに異なりますが、Shift+上下矢印で 100Hz 単位で変更が可能です。

8.2 衛星の AOS 検索と選択

次に衛星の AOS-LOS 時刻を検索できます。Alt-A を押すと、現在時刻以降の衛星ごとの **AOS-LOS 時刻を検索して表示**します（計算に少々時間がかかりますが、Calc next AOS-LOS と表示されたのち、計算次第表示されます）。計算済みの AOS-LOS 時刻表は、Shift-Alt-A で表示されます。一度に 6 衛星分表示されますが、Shift-Alt-A を押すと、さらに先の時間の AOS-LOS 表を表示します。表示は 5 秒間で消え、通常のトラッキング画面が表示されます。

また、Alt-P で次に AOS となる衛星を選択、トラッキングできます。

8.3 運用

Alt-B でビーコン周波数を受信するようにセットします。

Alt-F で現在トラッキング中の衛星のトラポン中心周波数にセットします。

Alt-R でリグコンのモードを変更します。トラッキング画面でそのモードが示されますが、

RATB: 単一リグモード 受信 VFOA 送信 VFOB

RBTA : 単一リグモード 受信 VFOB 送信 VFOA

ROT1 : 複数リグモード 受信 Radio 0 送信 Radio1

R1T0: 複数リグモード 受信 Radio1 送信 Radio0

複数リグモードは、2 台のリグをそれぞれ ANT につなぎ、それぞれで送信、受信を行うことを想定したトラッキングモードを示します。

單一リグモードですが、今のところ IC-705 と IC-9700 に対応しており、IC-9700 では、main でしか送信できませんので、main は TX、sub は RX となるように設定されます。このあたり、衛星運用をあまりやっていないので、適切かわかりません。。(運用しながら直すつもりでそのまま)。

Alt-S で衛星トラッキングモードを受信固定・送信固定・衛星固定・トラッキングなしの順で変更できます。実際のところ衛星ごとに決まったトラッキングモードになるべきだと思いますが、未実装です。

衛星運用のログは、contest を NOMULTI に設定していれば普通にログが取れると思いますが、ログには衛星トラッキングをさせているときには、Remarks の欄に衛星名、運用 GL、周波数オフセット値が追加されるので、ログの整理に便利です。



図 8.3.1 衛星画面

左画面) トラッキング状況

衛星名 運用 GL (ローテーター現在方位角)

AZ : 衛星方位角 (°) EL : 衛星高度 (°) リグコンモード

TX : TX 周波数

RX * : RX 周波数 *は受信 (RX) 固定を示す。

S : 衛星周波数

Ofs:衛星ごとの送受周波数オフセット(Hz)

右画面)

Sat. S&P Sat (衛星トラッキングモード) M

衛星名入力(RS-44)

XW-2B	18:27-18:37	66
AO-27	18:28-18:37	6
XW-2D	18:30-18:40	60
XW-2C	18:41-18:51	25
JO-97	18:58-19:03	2
AO-07	19:11-19:18	1

図 8.3.2 AOS 計算結果画面

左画面) AOS 検索

衛星名 AOS 時刻-LOS 時刻 分

8.4 衛星リスト

以下は Sat 欄に入力して運用が可能な衛星のリストです。すでに飛んでない衛星もあるかと思いますが、記念です Hi.

FO-29 AO-73 AO-07 XW-2A XW-2B XW-2C XW-2D MESAT1

RS-44 EO-88 CAS-4A CAS-4B JO-97 FO-99 HO-113 ISS IO-117

FO-118 CAS-10

9 信号強度測定、その他の制御

本器はリグからSメーター信号を連続的に取得しており、その情報を使って、QSO中の相手局の信号強度を記録する機能を持っています。

SMETER コマンドによって、信号強度は右画面に S???? と表示されます（デフォルト）。設定リグによっては、dBm に変換して表示しています。変換は、アンテナやプリアンプのゲインを考慮して行っています。また、各局の信号強度は、CQ出して呼ばれたQSO中ではコンテストNTRをこちらから送信して受信に移った後からQSO確定までの間の受信信号の最大値を取るようにしておらず、表示では下線を引いて示します。下線がない場合には、現在の S 値ということになります。

そのほか、ローテーター制御、アンテナ切替制御など私の趣味で作った機能がいくつかありますが、特定のハードウェアのみで動作するため、割愛します。コマンド一覧にそのコマンドは記載しています。

10 ネットワーク接続

10.1 zserver への接続(zLog ネットワーキングへの参加)

TAB で Zsvr を選び、zserver の IP アドレスを入れて Enter を押すことで、zserver と接続ができます。

(zLog ネットワークにメッセージを Remarks に書き、Ctrl-z を押すことで送ることができます)。また、zserver 接続時に zLog からメッセージが broadcast された場合には、左画面に表示します。

QSO 確定時に内蔵 SDメモリに QSO データを書き込みますが、同時に zserver に QSO データを送信します。また、現在運用バンド・周波数を zserver に通知しています(30 秒ごと)。

これは、zserver から QSO データをダウンロード・マージできればよいのですが、まだ実装されていません。

10.2 TCP サーバー

Alt-Delete で示された IP アドレスに telnet 接続(port23)すると、本器にシリアル接続するのと同様の制御などが行えます。つまり、PC から USB シリアル接続してデータを読み出すこともできますし、TCP 接続でもデータを読み出せるということです。

10.3 Web サーバー

Alt-Del キーで本器の IP アドレスを調べ、スマホなどから、

http://ipaddress/ , または、http://dvlogger.local/によって

DVPlogger の Web に接続できます。(https には対応しておりませんの

で注意)。

ここでは、さまざまな機能にアクセスできるリンクが集まっており、ま

た、DVPlogger にパーシャルチェック、サブ CPU プログラム等のファイル

をアップロードできるようになっています。

10.3.1 ログダウンロード

10.3.1.1 TXT 形式

/readqso は、現在のログを txt 形式でダウンロードします (ctestwin で

txt インポートできる形式)。(図 10.3.1)

▲ 保護されていない通信 192.168.1.2/readqso							
2025/08/16 21:46:45	JK1LSE	7.0	CW	599KN	599ST	SP	705
2025/08/16 21:49:26	JR6CSY	7.0	CW	599KN	599NS	SP	787
2025/08/16 21:52:00	JK1DVU	7.0	CW	599KN	599KN	CQ	730 S:-8.0
2025/08/16 21:53:07	K6RB	7.0	CW	599KN	5993	CQ	741 S:-67.9
2025/08/16 21:53:28	JF4ETK	7.0	CW	599KN	599HS	SP	737 S:13.4
2025/08/16 21:54:21	JH7TIX	7.0	CW	599KN	599IT	CQ	735 S:8.7
2025/08/16 21:54:39	K6XX	7.0	CW	599KN	5993	CQ	777 S:-66.3
2025/08/16 21:55:08	JA1IAZ	7.0	CW	599KN	599KN	CQ	739 S:-68.4
2025/08/16 21:55:49	JK2BAP	7.0	CW	599KN	599SO	CQ	756 S:-54.7
2025/08/16 21:56:12	JF1JDG	7.0	CW	599KN	599KN	CQ	745 S:-67.4
2025/08/16 21:56:48	JA9MAT	7.0	CW	599KN	599TY	CQ	718 S:-68.9
2025/08/16 22:05:43	JK8PBQ	7.0	CW	599KN	599IS	SP	776
2025/08/16 22:06:41	7L1ETP/1	7.0	CW	599KN	599ST	SP	740
2025/08/16 22:07:22	JF3IYW	7.0	CW	599KN	599OS	SP	718
2025/08/16 22:08:25	J11RXQ	7.0	CW	599KN	599IB	SP	712
2025/08/16 22:09:07	JA1TMG	7.0	CW	599KN	599TG	SP	749
2025/08/16 22:10:51	JH3HGI	7.0	CW	599KN	599HG	SP	759
2025/08/16 22:11:30	JA2NPS	7.0	CW	599KN	599SO	SP	784
2025/08/16 22:12:57	JH4UYB	7.0	CW	599KN	599HS	SP	705
2025/08/16 22:13:56	JF1NHD	7.0	CW	599KN	599IB	SP	730
2025/08/16 22:14:43	JA5CBU	7.0	CW	599KN	599KA	SP	759
2025/08/16 22:15:33	JR30QJ	7.0	CW	599KN	599HG	SP	766
2025/08/16 22:16:00	JJ1LBJ	7.0	CW	599KN	599KN	SP	769
2025/08/16 22:16:26	JE7DMH	7.0	CW	599KN	599FS	SP	753
2025/08/16 22:17:39	JR5MBI	7.0	CW	599KN	599KC	SP	731
2025/08/16 22:18:43	JJ7DTQ	7.0	CW	599KN	599FS	SP	730

図 10.3.1 txt 形式でログダウンロード

10.3.1.2 JARL ログ入力サービス

/jarllog :JARL のログ入力サービスにログを直接アップロードすることができます。運用中のログが JARL ログシート入力画面に転送されます(図 10.3.2)。この Web ページで提出用ログを編集して保存、提出することができます。

← → ⌛ contest.jarl.org/cgi-bin/logsheetsform.cgi

ログシート入力画面 v1.32e (Jul. 8, 2025)

- ALL ASIAN DXコンテストのマルチ・得点にも対応しました（「ALL ASIAN DXコンテストモード」にチェック、「マルチ再計算(上書き)」押下）
- Turbo HAMLOGで作成したCSVファイルの読み込みに対応しました
- 得点欄一括修正機能を追加しました

ヘルプ

入力内容をサマリーシート作成ページへ送る

サマリーシート作成ページへのデータ送信時刻：(未送信) 行

ローカルファイルへ保存 logsheet-20250817.txt	保存	ローカルファイルから読み込み ファイルを選択 選択されていません 読込む	Turbo HAMLOGで作成した CSVファイルを読み込む
--------------------------------------	----	---	--------------------------------

得点欄一括修正	<input checked="" type="checkbox"/> 新規入力行に前行の一部をコピーする	<input checked="" type="checkbox"/> 受信ナンバーの数字部分をマルチとする
マルチ再計算(上書き)	<input checked="" type="checkbox"/> モード欄を1キーで入力する(RST連動)	<input checked="" type="checkbox"/> 入力文字を大文字にする
得点計算(内容チェック)	<input type="checkbox"/> コールサイン入力時に時刻欄を更新する	<input type="checkbox"/> ALL ASIAN DXコンテストモード
テキストプレビュー	<input checked="" type="checkbox"/> マルチ欄に自動的に書き込む	<input type="checkbox"/> QSOパーティーモード
フィールドデータコンテストの局種係数： <input type="text"/>		

全時刻欄を9時間進める 全時刻欄を9時間戻す

	DATE (JST)	TIME	BAND	MODE	CALLSIGN	SENTNo	RCVDNo	Mlt	Pts	行挿入	行削除
1	2025-08-16	21:46	7	CW	JK1LSE	599 KN	599 ST			行挿入	行削除
2	2025-08-16	21:49	7	CW	JR6CSY	599 KN	599 NS			行挿入	行削除
3	2025-08-16	21:52	7	CW	JK1DVU	599 KN	599 KN			行挿入	行削除
4	2025-08-16	21:53	7	CW	K6RB	599 KN	599 3			行挿入	行削除
5	2025-08-16	21:53	7	CW	JF4ETK	599 KN	599 HS			行挿入	行削除
6	2025-08-16	21:54	7	CW	JH7TIX	599 KN	599 IT			行挿入	行削除
7	2025-08-16	21:54	7	CW	K6XX	599 KN	599 3			行挿入	行削除
8	2025-08-16	21:55	7	CW	JA1IAZ	599 KN	599 KN			行挿入	行削除

図 10.3.2 JARL ログシート入力画面

10.3.1.3 CSV 形式

/csv で、TurboHAMLOG でインポート可能な CSV ファイルを出力します。（そりゃバグ取り前でした。どうも、思っていたのと一列ずれています。）（ハムフェア後にアップデートを出します（汗））

10.3.1.4 ADIF 形式

/adif で ADIF 形式でのログダウンロードが行えます。

10.3.1.5 ダンプ形式

/dumpqso で、 QSO データの生ファイルを取得することができます。

10.3.1.6 過去 QSO ログの取得

readqso, jarllog, dumpqso はそれぞれ

http://ipaddress/readqso?num=001 のように num=バックアップ

QSO ファイル番号 を付けた時には、対象がバックアップ QSO ファイル

になります。つけない場合には、現 QSO ファイルを対象とします。過去

のデータを読み出したいときには?num=バックアップ QSO ファイル番号

を付けて読んでみてください。

10.3.2 POTA 支援機能

/potahelp で図 10.3.3 の画面が出ますが、ここで、一番上のボックスに QRV した Park 番号を入力することで、QRV した Park (JCCG 欄に POTA/park で設定、Remarks に POTA_MY:で取り込まれる) のログのみを ADIF でダウンロードできます。また、Open POTA uploader のボタンを押すことで、ダウンロードした POTA adif ログを POTA uploader に送ることができます。

また、下側の BOX に GridLocator を入れ、Find を押すことで、近隣の Park 3 か所の検索ができます（リンク先は POTA の Park 説明ページです）。IC-705 等で GPS を CI-V に送る設定をしている場合、GPS が受信できれば、この項目は自動的にあります。この検索機能を使うためには、Web アップロードで、POTA-JP.CSV をアップロードしておいてください（ない場合）。

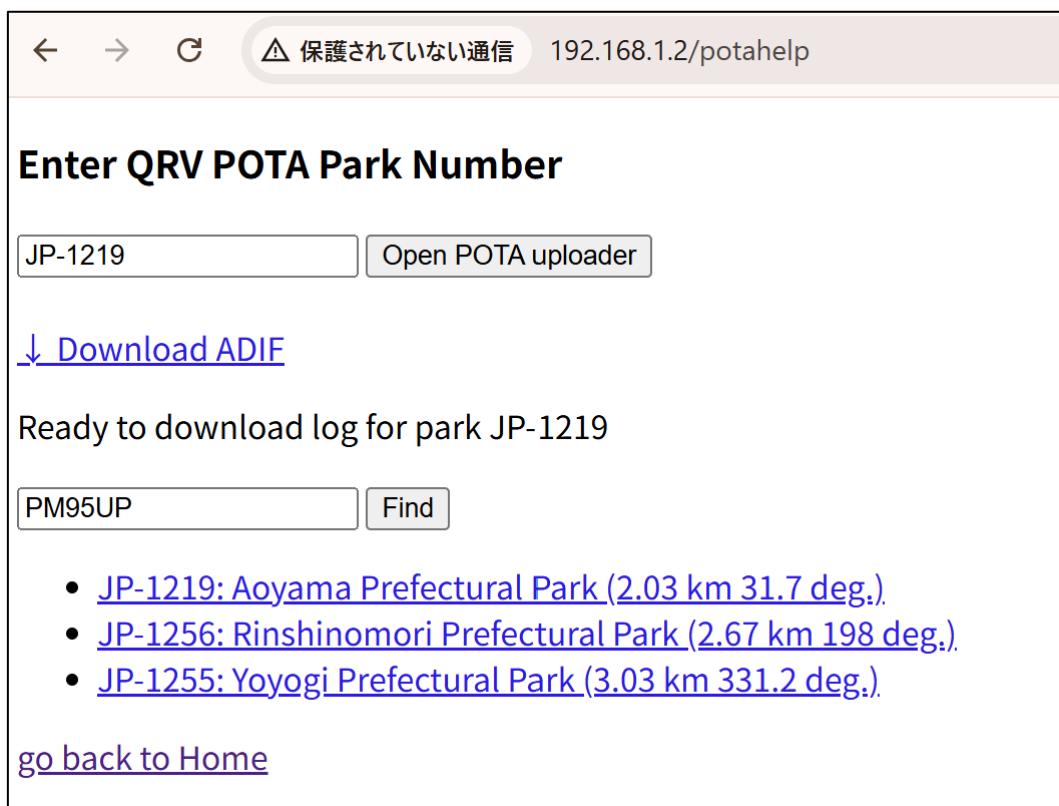


図 10.3.3 POTA 支援画面

この仕組みを使うと、**自局の運用 park** が JCCG に POTA/park(JP-1219..)

として書かれ、QSO 確定時に POTA_MY:pota_park と書かれます。

Remarks に POTA:相手の park# と書くことで、追っかけの場合も ADIF などに反映させることができます。

10.3.3 SOTA 支援機能

/sotahelp で図 10.3.4 の画面が出ます。こちらで、運用 Summit の設定、近隣の Summit 検索が行えます。このため、Web アップロードで、JA_SOTA.CSV をアップロードしておいてください（ない場合）。POTA 同様に、近隣の Summit 検索、運用 Summit 設定と ADIF によるログ作成・SOTA updaterへのアップロードが可能です。



図 10.3.4 SOTA 支援画面

10.3.4 Web での運用機能

/op からは図 10.3.5. のように Web ページから DVlogger を運用することができます（不安定なのでおすすめしませんが・・・）。

The screenshot shows the DVlogger Operation web interface. At the top, there are buttons for selecting the radio (0, IC-705, 1, FTDX10, 2, DJ-G7), mode (CW, USB, LSB, FM), and band (1.9, 3.5, 7, 14, 21, 28, 50, 144, 430, 1200). A status message at the bottom left says "09:15:27 Radio:1 S&P Freq: 7.014.50 Hz Mode: CW". Below the radio buttons are function keys: ESC, F1 CQ, F2 Exch, F3 TU, F4 MyCALL, and F5 Call&Exch. There are also fields for "Call:" and "Recv:" with associated buttons. Under "MyCall:", the values are JK1DVP, 599, and KN. Under "Contest:", the value is KCJ. At the bottom, there is a note "Band map test" and a link "go back to Home".

図 10.3.5 Web 運用画面

10.3.5 Web リグ設定

/rigs でリグの設定を行うことができます。設定を確定させるには、ボックスで

Enter の入力が必要です。

The screenshot shows a web browser window titled "DVlogger RIG Settings" at the URL "192.168.1.2/rigs". The page displays configuration parameters separated by commas:

- CW:cwport (1-3)
- B:baudrate
- P:catport_number ((-2:Manual) -1:USB, 1:Bluetooth, 2:CI-V, 3:CAT, 4:CAT2)
- ADR:CI-V_address
- NAME:rig_name
- XVTR:transverter_frequencies Frequencies joined by `_` from index 0: IFlo_0 IFhi_0 RFlo_0 RFhi_0 IFlo_1 ...
- R:CAT_reverse_polarity (0/1)
- BM:band_mask in_hex
- TP:cat_type_rig_type
(cat_type 0: ICOM CIV, 1:Yaesu(New), 2:Kenwood 3:Manual(NoCAT) 4:Yaesu(old))
(rig_type 0:IC-705 1:IC-9700 2:Yaesu 3:Kenwood 5:IC-7300)

Below the configuration parameters, there is a note: "Press Enter in input box to reflect changes." and a link "go back to Home". At the bottom, there are two buttons: "Save RIGs" and "Load RIGs".

Three sections show examples for different rigs:

- IC-705**: CW:1,TP:0_0,P:1,ADR:A4,B:115200,BM:1800,PTT:2,XVTR:2424000000_2430000000_434000000_440000000_1294000
- IC-9700**: CW:2,TP:0_1,P:2,ADR:A2,B:19200,BM:FC7F,PTT:2,NAME:IC-9700,
- FT-991A**: CW:2,TP:1_2,P:-1,B:38400,BM:1E00,PTT:2,NAME:FT-991A,

図 10.3.6 リグ設定画面

10.3.6 Web ロガー設定

/settings からは、リグの設定を行えます。Save, Load ボタンは SAVE, LOAD コマンドと同等の動作をします。設定を確定させるには、ボックスで Enter の入力が必要ですが、スマホでは難しいかもしれません。ここは改善を予定しています。

The screenshot shows the DVLogger Settings page with the following configuration:

- My Call:** JK1DVP
- Sent Exch:** KN
- Contest Name:** KCJ
- Power Code:** MMMMMMMMMPPP
- JCC/JCG POTA/ SOTA/** (Empty field)
- Wifi_SSID:** testap
- Wifi_Passwd:** testtest
- Cluster Name:** arc.jg1vgx.net:7000

図 10.3.7 DVLogger 設定画面

10.3.7 ファイルアップロードについて

下記のように内蔵 microSD カードにファイルのアップロードが可能です。

File Upload

Free Storage: 2.47 GB | Used Storage: 4.06 MB | Total Storage: 2.48 GB

選択されていません

バージャルチェックのファイルはname.pck (8.3形式)でアップロードしてください。
CALLHISTnameとコマンドを入力すると、name.pckを読み込みます。

ファイルアップロードの開始終了は表示されませんので、ファイルリスト更新までお待ちください。

After clicking upload it will take some time for the file to firstly upload and then be written to SPIFFS, there is no indicator that the upload began. Please be patient.

Once uploaded the page will refresh and the newly uploaded file will appear in the file list.

If a file does not appear, it will be because the file was too big, or had unusual characters in the file name (like spaces).

You can see the progress of the upload by watching the serial output.

Name	Size
QSO.TXT	36.75 KB
SETTINGS.TXT	698 B
RIGS.TXT	1.26 KB
POTA-JP.CSV	101.80 KB
JA_SOTA.CSV	205.42 KB

図 10.3.8 ファイルアップロード画面(Web Top 画面の下方)

下記にアップロードして使用するファイルの説明を書きますのでご参考ください。

QSO.TXT 36.75 KB QSO ログデータ

SETTINGS.TXT 698 B 設定ファイル

RIGS.TXT 1.26 KB リグ設定ファイル

POTA-JP.CSV 101.80 KB 国内の POTA PARK ファイル

JA_SOTA.CSV 205.42 KB 国内の SOTA Summit ファイル

APP0.BIN 889.69 KB サブ CPU アプリケーション
BOOTLOAD.BIN 25.03 KB サブ CPU ブートローダーファイル
PARTITIO.BIN 3.00 KB サブ CPU パーティションファイル
SPIFFS.BIN 1.62 MB サブ CPU SPIFFS ファイル
KCJ.PCK 7.17 KB スーパーパーシャルチェックファイル(KCJ コンテスト)
CALLSIGN MULTI の列
TLE.TXT 15.70 KB 衛星軌道ファイル
SATINFO.TXT 114 B 衛星情報ファイル

11 ふつうの運用

コンテスト運用とあまり変わることはありませんが、[JCCG] に入力した JCG/JCG コードを、\$J で CW 等のメッセージで送出可能です。

また、送る QSL カードの種別などは、ショートカットキー(Ctrl-Q) で **Buro, hQSL, NO** から選んで QSO データの Remarks に書き込むことができます。また、QSL カードの選択状況は、DupeCheck の状況の左側で **J: JARL h:hQSL** によって示されます。

Remarks に市町村名を入力（Remarks 先頭からスペースまでの文字列を検索の対象としますので、マルチ名のスペースは'_'で置き換えてください）し、Ctrl-f を押すと、マルチを逆引きできます（入力した文字数までの部分検索なので、一部を入力して検索し、検索文字を追加入力などすると見つけやすいです）。左画面には、マルチが見つかった場合には、
found=マルチ、マルチの名前が表示され、見つからなかった場合には not found と表示されます。逆引きで検索されたマルチはマルチ欄に入力され、マルチ入力に移ります。コンテストに NOMULTI を選んでいる場合も、全市全郡マルチ（つまり、市郡区）が入力されており、市郡区で検索ができますので、使いようによっては便利でしょう。また、同じ Ctrl-f でも Exchange 入力欄で押した場合には、強制的にマルチチェックを実行します。NOMULTI でもこちらから、JCC/JCG 番号から市郡区名を検索することができます。全市全郡の名前は、「市」を C、「郡」を G、「区」を K の各 1 文字で省略していますので覚えておくと便利です。

また、運用では、QTH と QRA を Remarks に入れることが多いと思いますが、先に QTH（わからなければ_などとりあえず入れる）、スペースを入

れて JCC, スペースを入れて QRA を入れるなどすると、上記の逆検索機能が有効に使えるのでよいかなと思います。

Ctrl-g でコールサインから DX entity を検索し、その結果を左画面に表示されます。たとえコンテストを CQWW に設定していても、この内容で、得点計算されませんが。。。

それから、私は国試の時に勉強して以来、和文 CW をやっていないので、使い物にならないかも知れないのですが、キーボードで入力したローマ字から和文コードを送出する機能を作つてみました。和文モードの場合は、キーヤーモードの表示に J/j(japanese の略)が付きます。移動運用でサクッと和文でご挨拶などに使えるといいかな？ と思っています。

11.1 Remarks に記録されるデータについて

QSO ログを確定されるときに様々な情報を記録します。特に、Remarks には様々な情報を記録していますので、下記をご覧ください。

11.1.1 CQ/S&P の別

CQ, SP と記録されます。

11.1.2 自運用場所

Jccg の欄に入力した文字列に SOTA/summit が入っていた場合には、QSO

確定時に *Remarks* に *SOTA_MY:summit* が入り、自運用 *Summit* を記録することができます。*POTA/park* が入っていた場合には、*POTA_MY:park* が入ります。一方 *Jccg* の最初に書かれた番号は常に *JCC/JCG* 番号として認識され、*Remarks* には *J:jcc/jcg_number* として記録されます。

11.1.3 SMeter

また、Smeter の記録は *Remarks* に **S:**として記録されます。

11.1.4 QSL カードの選択

JARL , hQSL のように記録されます。

12 音声入出力関連機能

12.1 CW デコード機能

ヘッドホン端子の出力 R 側を AD 変換できます。この入力を使って CW デコーダーを実装しています。つまりヘッドホンで聞いている信号をデコードしています。デコーダーは、 **DECODER** コマンドを入力すると開始。デコードされたら左画面に表示します。 **DECODERSTOP** コマンドで終了します。今のところ、デコーダーの負荷が重く、デコーダーを動かした場合 Wifi 等の動作が止まってしまうため、改良が必要な状況ですが一応動きます。

12.2 F2A・音声合成・イヤホンマイク入出力

また、mini 版では、Key1 端子の ring に TONE (600Hz) CW を出力します。

Wide 版では、MIC 端子に TONE CW を出力するようにしました。また、TONE CW に加え、Phone モードのときは、MIC 端子に読み上げ音声を出力できます。AD 変換で出力しているので、トーンもテープを付けた高品質なものになっています。MIC 端子は、IC-705 と類似のピン配置で図 5.2.2.2 のようなケーブルで IC-705 の MIC 端子に直結できます。さらに、Wide 版のイヤホンマイク端子は、pin4 のマイク端子が本器の MIC 端子に本器からの音声出力とミキシングして出力されるようになっているため、イヤホンマイク端子に接続したヘッドセット等を使ってハンズフリー運用が行えます。 **TOGGLEPTT** コマンド等を併用すれば、キーボードでの PTT 操作を合わせたハンズフリー運用がフィールドで楽しめます。

また、コンテスト運用のところで述べましたが、**音声合成機能**もこちらに出力されます。Ctrl-v でコンテスト QSO の流れに沿った音声再生ができるほか、CW と同様に Shift を押しながら入力したキー入力の内容をフォネティック

クで再生ができます。音声は「四国めたん」の音声データを添付していますが、下の通り自分の音声も作れます。

12.3 音声データの作り方

本器の再生機能で再生できるデータは、Audacity などで作成することが、で
きます。Audacity すべての音声が入った wav ファイルを 8kHz モノラル、
signed 16-bit PCM で作成してください(ファイル-オーディオをエクスポート)。

録音が必要な音節は、A-Z (エー～ゼット), 0-9 (ゼロ～ナイン), _0～_9 (マ
ル、ヒト、フタ、サン、ヨン、…キュウ), a-z (アルファード、…ズールー),
_C (CQ コンテスト), _c (コンテスト), _D(どうぞ), _A(ありがとうございました), _M(もういちどおねがいします), _H (ほかいらっしゃいますか), / (ポ
ータブル) です。

現状プログラムの作りこみの関係と、サブ CPU の SPIFFS の容量の関係
(2Mbytes)のため、同時に 1 種類のファイル名が metan.wav と、そのファイ
ル上の音節の始まりと終わりの秒数が書かれたラベルファイル(Audacity で
は、解析-サウンドから自動ラベル付け、等で作ったラベルを秒+ミリ秒フォ
ーマットで出力 (編集-ラベルエディタ-エクスポート)) した metanidx.txt を

まとめた spiffs.bin (spiffs_data/の下に入れて、dvlogger_ext でビルドすると自動生成します) をサブ CPU に書き込んでください。

0.039312 0.491400 _1

1.030303 1.303849 _2

1.784603 2.120000 _3

2.912367 3.190000 _4

表 12.3.1 ラベルファイル metanidx.txt の内容 開始 (秒) 終了 (秒)

ラベル

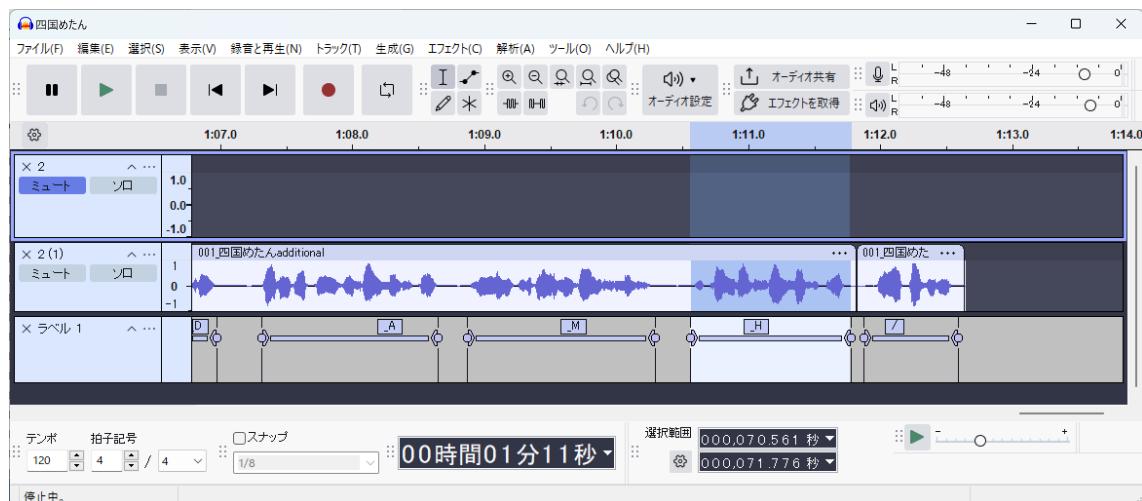


図 12.3.1 Audacity で音声ラベル付け中。



図 12.3.2 サウンドラベルエディット中の様子。

13 機能拡張・プログラムの変更などについて

DVPlogger は、機能拡張を行うための端子(2x5 ヘッダ*1, 1x8 ヘッダ・部品未実装)を備えており、機能拡張基板をスタックしての、 SO2R でのマイク (PTT) 切り替えの実装が可能です。

- ・本器の動作プログラムは esp32 esp-idf v4.4.7 + Arduino esp32 v2.0.17 で実装しており、ソースコード、システムプログラムと関連プログラムは、下記のサイト(github)で取得することができます。

<https://github.com/JK1DVP/dvplogger>



- ・本器へのプログラムの書き込み方

こちらも、github で案内いたします。

- ・プログラムのソースコンパイル、スーパーパーシャルチェック用のデータ取り込み

本器はメモリの都合でスーパーパーシャルチェックなどの場合、プログラムに過去 QSO データを取り込むことが必要になります。その場合、プログラムをデータを含めコンパイルする必要があります。github で必要なモジ

ユールを公開します。過去 QSO データをソースコードに変換するのは古めかしい Tcl コードですが、summary2callhist.tcl というプログラムを用意しました。ソースからのコンパイルは、きれいにする時間が足らず github での公開まで間に合いませんでした。スーパーパーシャルチェックは強力なので、できるだけ早くユーザーの皆さんのデータを組み込めるような方法を整備して、これも github でお知らせするようにいたします。

- ・本器の回路図・端子機能については本マニュアルに記載しましたので、他のプログラムを記述・ロードすることで自分好みのプログラム実装も可能と思います。例えば、本器のプログラムを変更することで、SO2Rmini (<https://nn1c.org/so2r/>) 相当の動作をさせて、PC の周辺機器として使うことなどが可能です（ハードウェア構成が異なるため多少書き換えが必要です）。
- ・バグレポート・機能実装要望などは、メール (jk1dvp@gmail.com) で受け付けます。フィードバックを歓迎いたします。
- ・関連情報を JK1DVP のブログ (<https://jk1dvp.blogspot.com/>) にて出しますので、そちらも併せてご覧ください。



14 表

表 14.1 コマンド一覧 (Rcv コールサイン欄にキー入力)

DISCONN クラスタ接続を断

PRINTKEY デバッグ用入力キー表示

SATELLITE サテライト TLE を読み込み、必要ならネットから更新。

NEXTAOS 次の AOS のリストを表示。

ESM Enter Sends Message モードに入る/抜ける (Enter で QSO を進めます)。

DECODER CW デコーダーを起動 文字をデコードすると左画面で表示されます。

DECODERSTOP CW デコーダーを止める。(現在のところデコーダーを動かすと、Wifi の動作などに問題が出ますので、使わないとき STOP がおすすめ)

SMETER SEQNR 欄に S から始まる信号強度値(dBm)を表示。QSO 中は _____をつけて相手局の最大強度を示す。show_smeter=1,2 は、dBm に変換するかどうかの違い。

INTERVAL 最後に確定した QSO からの経過時間を表示。(SEQNR を入力の上)

SEQNR QSO シーケンシャル番号を表示

NEWQSOLOG 新しく QSO ログを初期化

READQSOLOG 現在運用中の QSO を ctestwin で import できる形式で

出力

MAKEDUPE これまでの QSO ログに基づきデュープチェックリストを作成

KBDJP 日本語キーボード設定に

KBDUS 英語キーボード設定に

KEY[num] リグの接続された CW ポートを変更 KEY1, KEY2

STRAIGHT ストレートキー モード切替 (右 Shift キーで縦振打鍵できます)

TOGGLEPTT 右 SHIFT で PTT をトグルできるようにするか切替。

(STRAIGHT でストレートキー モードとした場合にはそちらが優先)。

IAMBICA パドルを IAMBIC-A の設定にする。

IAMBICB パドルを IAMBIC-B の設定にする。

PADDLEREV パドルをリバースにする。

PADDLENOR パドルをノーマルにする。

SAVE 設定を microSD に保存。SAVE*name* とすると設定を指定した名前で保存することができる。

LOAD 設定を microSD から読み込み。LOAD*name* おなじ。

ROT ローター コマンド ROTEN1 ローター 制御 有効 ROTEN0 ローター 制御 無効

ROTTYPE*n* ローターの型式 設定

ROTNORTH 現在方位を NORTH とする。

ROTSOUTH 現在方位を SOUTH とする。

ROTAZn ローターターゲット方位を n° に設定。

ROTSTEPn ローターを n° ステップで動かす。

ROTSWEEPaz 現在から ROTSTEP で設定した角度ずつ az° になるまでローターを回転させる（アンテナ特性測定用） ROTSWEEP のみで回転キャンセル

ANT0n アンテナ 0 制御 RELAY0 n=0 OFF n=1 ON

ANT1n アンテナ 1 制御 RELAY1 n=0 OFF n=1 ON

ANTALTsig 信号が sig 以上を受信したら ANT を切り替える。

NEXTRIG / PREVRIG 次の・ひとつ前のリグを選択

ENABLERIG / DISABLE RIG リグを有効・無効にする。

NATTO dit dah 比率を変更して「ねばる」キーイングとする。

CWNORMAL 普通の 1 : 3 に CW キーイングを戻す。

VERBOSE デバッグのために冗長な出力をターミナルに出力

SO2RMINI SO2RMINI を USB ポートにつなぐことで SO2R を実装（古いので機能しなくなりました）

CALLHIST[*name*] コールヒストリを読み込んで検索を開始。*name* が指定された場合にはそれを読み込む。

LISTDIR microSD に記録されたファイルのリストを表示。

HELP ヘルプメッセージを表示 繰り返すことで複数ページのヘルプを表示。

HELP_R 逆方向にヘルプメッセージのページをめくる。

ADCSTAT AD 変換器の入力信号を検査する（左画面で一定時間表示）

WIFI WIFI 有効無効を切り替える。

DUMPQSOLOG 過去に記録した QSO データをダンプする(PC ターミナル接続に)。

EXITEMU telnet での運用画面・情報画面工ミュレーションを抜ける。

RADIO[n][cmd] リグの設定を変更。n = radio#(0,1,2), cmd 0
disable 1 enable 2 switch+ 3 switch-

BANDEN[hex] hex を指定した場合には、現在運用中の RADIO で運用（バンド選択）できるバンドを指定。指定しない場合には現在の設定を左画面に表示

BANDMASK[hex] バンドマスク（バンド選択できるバンド）の設定
BANDEN の反転ビットを指定

BANDMAP[hex] バンドマップにクラスタからの情報を反映させるバンドのビットを設定。

モード文字列(CW, FM, AM, LSB, USB, RTTY 等) リグモードの変更 表
参照

周波数 (kHz 単位) リグ周波数の変更。

RESETDISP 画面不具合が発生した場合の画面リセットを行う。

DISPTYPE0 1.3" OLED を使う。 設定は SAVE できます。

DISPTYPE1 2.4" OLED を使う（Wide 版）。 設定は SAVE できます。

DISPTYPE2 1.3" OLED を使う(mini 版は、こちらです)

MUXTRANS サブ CPU への通信をマルチプレクスする（これをすることによって CAT2、USB キーボード端子が利用可能となる） 設定は SAVE できます。

NOMUXTRANS サブ CPU への通信をマルチプレクスせず、平文で行う。（デフォルトで MUXTRANS となるはずですが、うまくいかないときなどは試してください）。

SAVERIGS リグの設定を保存。

LOADRIGS リグの設定を読み込む。立ち上げ時には、自動で読み込むようになっています。

RESETRIGS リグの設定を出荷時設定に戻す。SAVE しない限り電源再投入でもとに戻ります。

表 14.2 キーボードショートカット一覧

Shift-下矢印 衛星ごとのオフセット周波数を 100Hz 下げ

Shift-上矢印 衛星ごとのオフセット周波数を 100Hz 上げ

Alt-HOME Radio0 のリグを変更

Alt-END Radio0 のリグをイネーブル・ディセーブル

Alt-Shift-HOME Radio1 のリグを切替

Alt-Shift-END Radio1 のリグを Enable/Disable

Alt-Ctrl-HOME Radio2 のリグを切替

Alt-Ctrl-END Radio2 のリグを Enable/Disable

各リグの変更で、他の RADIO で登録されているリグと CAT ポートなどの競合がある場合にはスキップします。

Alt-'>,' '<.' manual band 変更

(* MANUAL リグ以外では、リグコンで周波数を変更することによってバンドチェンジしているため、リグコンが働いていない場合にはバンドは変わりません)

Shift-右 左 バンドマップのエントリ変更

Alt-上 下 バンドマップのエントリ変更

Alt-'/' ローテータースイープを一旦停止

Alt = アンテナリレー トグル開始

Alt--' リグのスペクトラムスコープ周波数幅をモードに合わせ適切に設定。

Alt-t チューニングのためキーイングトグル

Alt-x トランスバーターON/OFF

Alt-1 リグの優先度を現在選択中のリグに設定

Alt-z CW↔ PHONE 切替

Alt-q CQ/S&P 切替

Alt-a NEXTAOS を表示

Alt-p NEXTAOS に一番近いサテライトを選択

Alt-p クラスタの表示情報のレベルを変更

Alt-b ビーコン周波数に設定（サテライト）

Alt-b バンドマップを周波数ソート/スポットが新しい順にソートでトグル

Alt-c CW message edit → RTTY message edit (RT) → exit

Alt-d bandmap のエントリーを削除。

Alt-n 現在の周波数を bandmap に CALLSIGN 欄に書かれた局として登録

Alt-e USB キーイングを ON/OFF (DTR でキーイング予定ですが、実験中の機能で働きません)

Alt-g トーンCWキーイング ON/OFF

Alt-SPACE 現在バンドマップで選ばれているスポットを選択

Alt-Del リグの情報などを表示。

Alt-f サテライト運用周波数をトラポン周波数帯の中心に設定

Alt-r サテライトVFOモードを変更

Alt-s サテライト周波数制御モード変更 受信固定・送信固定・衛星固定・トラッキングなし

Alt-w 現QSOデータのクリア（ワイプ）(Ctrl-wと同じです)

Alt-'>,'<.' バンド変更 上下

Alt 上下 バンドマップ選択カーソル変更 (?)

Alt 左右 バンドマップのバンド変更

Alt -m モード変更

Ctrl-Funckeу リピート送信

Ctrl-r Remarks

Ctrl-e 受信 exch ^

Ctrl-c 相手コールサインへ

Ctrl-a こちらから送った相手の RST レポートへ(カーソルは信号強度 S のところに行きます)

Ctrl-s 相手からもらった R S T レポートへ(カーソルは信号強度 S のところに行きます)

Ctrl-f Remarks に入れた名前から該当マルチを検索

Ctrl-g CALLSIGN から DX entity を検索し左画面に表示

Ctrl-k キーヤーモード切替

Ctrl-w Q S O ワイプ (Alt-w と同じです)

Ctrl-v ボイスメモリー送信 OFF/ON (noTU) /ON (TU) 切替

Ctrl-j 現在見ている過去 Q S O エントリーを左画面に表示

Ctrl-p ひとつ前の Q S O データを表示

Ctrl-n 次の Q S O データを表示

Ctrl-o 最後に記録された Q S O データ表示

Ctrl--(ハイフン) 現在の表示している Q S O データを編集(確定で編集終了)

Ctrl-1 CW/PHONE Q S O 有効無効切替

Ctrl-2 コンテスト変更

Ctrl-3 バンドマップのマスクビット（現在運用中）を切替

Ctrl-4 コールヒストリ検索をON/OFF

Ctrl-5 リグの運用モード（SO1R, Satellite, SO2R）切替

Ctrl-t, Ctrl-y マルチリスト表示 (worked が*)

Ctrl-u 各バンドのQSOS数マルチ数の表示

上下 周波数変更 (100Hz)

**Shift +上下 モード別により大きな周波数で変更 (CW500Hz, SSB
3kHz, FM 20kHz)**

PRTSCREEN, INS, HOME, END 未定義

表 14.3 運用ウインドウ一覧

Rcv 相手のコールサイン ナンバー Space で左右を移動 Tab で次に Shift+Tab で戻る

RST Snt 相手に送る R S T Rcv 相手から送られたこちらの RST

Snt 自分のコールサイン ナンバー

Rmks (Remarks)

Test コンテスト名 コンテスト名を入力 Enter でコンテストの設定が行えます。

Sat. 運用サテライト名 (TLE と同じ名称) 空欄の場合は通常運用となる。

G.L. Grid Locator 設定 Enter で運用位置設定 (サテライト)

JCCG JCC/JCG 番号

Name (オペレータハンドル名前) Ron 等

RIG 設定されているリグの名前(IC-705, QCX-MINI 等)

リグの名前を入力 Enter でリグの設定を行うことができます (リグの名前は部分検索できます)。

Spec リグの設定情報一覧

Clist cluster アドレス:ポート

CCmd クラスタ接続時に送信するコマンド

Mail メールアドレス

Pow JARL コンテスト電力コード(P, L, M, H, Q (QRP splint)) 左から 1.8MHz …に対応

SSID WIFI SSID

Pass WIFI password

zSvr zserver IP アドレス

表 14.4 モード名称

LSB USB AM CW RTTY FM WFM CW-R RTTY-R DV

表 14.5 キー CW 符号対応表

'/' / '"/' / (Shift +/ が? のため、キー割り当てを移動)

'.' カンマ符号 '.' Dit

'+' AR '=' BT '&' AS '<' AR '>' SK '-' BT '_' BT

(このあたり、日本語・英語キーボード対応で混乱している可能性があります。しっかり検証できていません。)

'(' KN 和文モード→英文遷移 ')' KK 英文モード→和文遷移

'{ (英) @ (日)ホレ 打電の上和文モード '}' (英) '[' (日)ラタ 打電の上英文モード

表 14.6 CW メッセージマクロ一覧

\$I 自分のコールサイン(Snt 欄) \$C 相手のコールサイン(Rcv 欄)

\$U “CQ” (CW モード) または _C “CQ コンテスト” (Phone)

\$T “TEST” (CW モード) または _c “コンテスト!” (Phone)

\$W 送るコンテストナンバー 電力コードは含みません Sent 欄で
11,1115 等と書くと、1.8-1200MHz では 11 2400MHz- では 1115
が \$W に入ります。

\$J JCC/JCG ナンバー(JCCG 欄) \$P 電力コード(Pow 欄でバンド毎に設定
される)

\$V 送信 RST (送信 CW では 9 は N に変換されます)

\$Q 各バンドでの Sequence Number (001~)

\$S Sequence Number (1~) \$N 自分の名前(Name 欄)

'_ (アンダースコア) は、CW 送信では短いスペースとして働きます(zLog
と同じ)。

音声読み上げ備考

-D (どうぞ) _A(ありがとうございました) _M(もう一度お願いします) -
H (ほかいらっしゃいますか) うまくファンクションキーの定義に入れて
使ってみてください。

\$R (RTTY) Rmks (Remarks)に書かれた内容

\$L (RTTY) New Line

表 14.7 PC terminal からのコマンド一覧 (help で表示)

Available Commands:

emu (entering terminal emulation EXITEMU to end)

verbose[num]

help (show this)

loadsat/nextaos/satellite

// operation commands

decoder/decoderstop CW デコーダーON/OFF

playstring 音声読み上げ (小文字でフォネティック, 大文字でそのまま, 数字は英語読み _数字で和文読み数字 _A ありがとうございました。)

playcwstring TONECW 送信 (Wide 版)

playwpmSpeed TONECW の速度 (WPM) を与える

playq 現在演奏中の TONECW/音声読み上げのキューを表示

// QSO log commands

newqsolog (start new log QSO.TXT)

makedupe (dupe/multi check from QSO.TXT)

dumpqso[num](dump raw qso file [num] if read backup files

num がない場合には QSO.TXT (現運用 RAW ログデータ) をダンプします。

num が付いている場合には、NEWQSOLOG コマンドで過去ログに

rename された QSO.001 などをダンプします。

readqso (read log QSO.txt in ctestwin txt imporable format)

listdir (show directory content)

dumpcur 現在 QSO 中のデータをダンプ

dumptop 最初の QSO データをダンプ

dumpnext/dumpprev 次の/前の QSO データをダンプ

dumplast 最後の QSO データをダンプ

dump *num num* 番目の QSO データをダンプ

// CLUSTER commands

Dx de (push cluster information)

// RADIO remote control commands

setstninfo{Callsign} (set target to work station information)

status (get status of the radios)

switch_radio {radio#}/enable_radio {radio#}

focus {radio#} (change focused radio)

show_summary QSO サマリ表示

show_bandmap バンドマップ表示

// SETTINGS commands

save[name] (save settings)

```
load[name]

settings (show settings information)

assign{variable_name} {value} (assign settings variables)

post_assign (perform post assignment tasks)

// CALLHISTORY commands

callhist_set [callhist_filename] (start setting callhist information,input
lines of callsign exchange,end will end setting.

callhist_open [callhist_fn] (set callhist filename and open)

callhist_search (start callhist search, input callsign to search, end will
end search session.)

// MAINTENANCE commands

flashersd boot part app spiffs SD カードに書かれた app0.bin
partitio.bin bootload.bin spiffs.bin をサブ CPU に書き込む。

flasher 最低限のサブ CPU プログラムをサブ CPU に書き込む。

addap ssid password Wifi アクセスポイント設定

disptype0 1.3" OLED 設定に (original 版)

disptype1 2.4" OLED 設定に

disptype2 1.3" OLED (mini 版)設定に

muxtrans MUXTRANS コマンドと同様

time yyyy-mm-ddThh:mm:ss RTC クロック設定

reset_settings 設定を出荷時に戻す
```

restart_dvplogger DVLogger 再起動

reset_display 画面をリセット（不具合が起こった場合の対応のため）

// development commands

gpio[port] [value](write mcp gpio port)

serialシリアルポートの割り当て状況表示

send *string* *string* を Serial2（サブ CPU 通信）に送信

i2c_scan i2c バススキャン

usb_desc USB デバイスの表示

ntp_stat NTP の状態表示

表 14.8 リグ設定文字列の意味

CW : CW ポート 1: KEY1 2:KEY2

TP:m_n m: CAT_TYPE 0:ICOM CI-V 1: Yaesu 2: Kenwood 3: NO CAT

4:Yaesu OLD (周波数の桁数が少ない FTDX3000 等)

n: RIG_TYPE 0: IC-705 1: IC-9700 2:Yaesu 3:Kenwood 4: Manual

P: CAT ポート番号 -2: Manual -1: USB 0: PC console (設定できません)

1: Bluetooth 2: ICOM CI-V 3: CAT シリアルポート

4: 拡張 CAT2 シリアルポート (新規追加)

ADR: CI-V アドレス

B: ボーレート

NAME: リグの名前 (この名前で検索を行えます)。接続ポートが区別できる
よう、FTDX10, FTDX10-2 などと枝番でポート名を入れるとよいでしょう。

XVTR: トランシーバー周波数定義 RFLO_RFHI_IFLO_IF_HI と_で繋げて
周波数関係を記述します。最大 3 件定義でき、複数の定義は_で繋げて記述し
ます。

R : CAT ポートが極性リバースの場合 1 を設定してください。QCX-mini 等は
0 YAESU,KENWOOD リグは R:1 となります。

PTT : PTT の方法の記載 0~3 (USB など区別できるように設定を置いていま
すがこちらは現在のところ使われていません。)

BM: リグが運用できるバンドのマスクビット (運用できるバンドを 0 にする)
で、HEX 値を指定。FE00=1.8~430 ,FF80=1.8~50MHz ディフォルト値が
間違っているものがあるようです。

15 ハードウェア構成について（回路図）

15.1 Mini 版 (revD, revE)

ハードウェアは図 15.1(DVPlogger 本体)に示すような回路図の通りです。表 15.1 に部品表（本体）を示します。以下、解説します。

CPU には、esp32 を使っています。これは、マイコンの中では、内蔵メモリがかなり大きいことから選択しました。コンテストログは、デュープチェック、マルチチェックが必須ですが、QSO の情報を記憶しておき、スピーディーにチェックすることが必要で、他のマイコンでは、コンテストの QSO 数を考えると収まりません。そもそも、DVPlogger を作ろうと考えたきっかけが、zLog の最初期の PC で搭載されていたメモリにかなり近い量のメモリが搭載されているということでした。esp32 の Wifi 機能は、クラスタやサテライト情報の取得、zserver や telnet ポートなど外部との通信に使っています。esp32 は devkit-C という開発モジュールを使っています。これはプログラム変更など外部との通信が USB シリアルでできる必要性があることや入手性からこれを採用しました。ただ、本品についているモジュールは、一般的に流通している devkit-C は microUSB コネクターが付いていますが、これは USB-C なのが、少しユニークです。一方、モジュール調達後、調べたところ、5V ラインに挿入されているショットキーダイオードが不適切なものが付いていることがわかりましたので、devkit-C で通常使われているものに交換をこちらでしています。モジュール交換等の際には注意が必要です。

また、devkitC のほかに 30 ピンのモジュールが多数流通していますので、今回基板はどちらのモジュールでも実装可能としました。

キーボード等とのインターフェースには、esp32 に SPI 接続した USB HOST I/F チップが搭載された USB host mini モジュールを使っています。採用したモジュールはなぜか、USB 電源出力が 3.3V につながっているという不思議な設定なのですが、モジュールのパターンカットを 1 か所することで 5V を USB bus に接続対応しています。なので、こちらもモジュールの取り換えには注意が必要です。USB host には、キーボードを接続するほか、ハブを入れて FTDI 社製の USB シリアルケーブルをつなぐと、Yaesu, Kenwood など RS232C ポートとの通信ができるようにしてあります、が、安定性がいまいちなので、あまりお勧めしません。開発中は IC-705 もここで繋がるようにプログラムを書いたりしたのですが、IC-705 など多くのリグの USB 接続は、内部にハブが入っていたり、複数ポートのデバイスだったりで、この USB HOST チップのライブラリでディフォルトで対応コードが書かれていらないものが多いです。ということで、IC-705 ではより確実、安定接続（しかも無線）ということで、Bluetooth での接続を採用しています。

データの記録には、micro SD カードを SPI 接続して使っています。esp32 のブート時の flash 電圧設定に SD につないでいるピンが使われて干渉するため、esp32 の flash 電圧を 3.3V に設定するよう、efuse の書き込みをしてあります。esp32 のモジュールを交換するなどした際には注意が必要です。本品には 16GB の microSD を取り付けていますが、64GB 以上の microSD は、ファイルシステムの関係で動作しませんので、注意が必要です。16GB の容量は、コンテストのログ等の情報を一生分貯めても大丈夫かと思いますが。

CWキーイングなどは esp32 の GPIO でオプトトランジスタ (TLP293-4) を駆動する仕組みです。2 系統の CW ポートを用意しましたが、esp32 としては

3 系統のキーイングができるようにポートを設定しています。同じオプトトランジスタで RELAY1, RELAY2 と外部制御ポート(JST XH-4pin に接続)を用意しましたが、TLP293-4 の I_c が 50mA しかないので、リレーの直接駆動はできません（注意）。また JST の XH コネクターで RELAY1,RELAY2 は引き出しありますが、本ケースでは、防水性を考えて、このコネクターに対応した孔をあけていません。ノックアウトできるように薄く造形していますので、お使いの方は適宜ケースを加工してください。

KEY1 の tip には、オプトトランジスタのキーイングがつながっていますが、今回 F2A 対応のため、ring に esp32 の DAC 出力がコンデンサ(1uF)を介して繋いでいます（mini 版対応）。DAC では、F2A では 600Hz の正弦波を発生しています。また、RTTY では AFSK のトーンも同様の仕組みで発生しています。開発中のマイクの切り替え拡張ボードを接続した場合には、こちらではなく、直接マイクのほうに信号が出力されるようにする予定です。

ディスプレイは I2C バス接続の 2 枚の 1.3" OLED を利用しています。OLED のチップ抵抗でデバイスのアドレスを 0x78 と 0x7A で選べるようになっていますので、同じバスに 2 枚 OLED を接続しています。こちらのデバイス電圧は 3.3V です。また、本ボードでは、4 ピンは GND VCC SCL SDA の順に並んでいます。市場には、これと配列が異なる OLED が流通しており、本頒布品も一部の OLED がそのようなものでしたが、OLED の基板上の半田ジャンパーで変更対応をしております。また、revD 以降では、基板上でどちらのタイプでの OLED でも実装が行えるようにしております（SV3, SV4 を追加）。

同じ I2C バスに RTC(DS1307Z)を接続しています。これによって、山中など

インターネットがつながらない場所でも時刻が正しくセットされます。こちらのデバイス電圧は 5V のため、I2C バスに MOSFET(BSS138)による電圧レベル変換を挟んでいます。RTC には、バックアップのリチウム電池 (1620 サイズ)を取り付けています。

リグコンのインターフェースは 3 系統のシリアル I/O を用意しました。まず、1 つ目は、Bluetooth モデムを介して IC-705 と通信するためのポートです。Bluetooth モデムは、6pin コネクターで接続していますので、Bluetooth モデムが必要ない場合には、このコネクターを使って、他種のリグとの接続に使うことも可能です（が、esp32 に直結なので気を付けてください）。Bluetooth モデムは、市販品は、高価であるか、技適がないものが非常に多いです。そのため、今回、安価にすると同時に、十分な性能を持たせるために、秋月の esp32 モジュールを使って、Arduino のスケッチによる Bluetooth モデムを作成して取り付けました。IC-705 と CI-V の接続ができるとともに、他の CAT リグなども(Bluetooth のインターフェースがあればですが)無線接続できるはずです。

2 つ目のシリアルポートは、ICOM の CI-V ポートを構成しました。古いものを含め IC-9700 等広範な ICOM リグとこのポートで接続が可能です。

3 つ目は、Yaesu, Kenwood 等の RS232C ポートと接続が可能な C A T ポートです。このポートは、QRP-Labs の QCX, QCX-mini 等の TTL レベルのシリアルポートにも接続ができます。RS232C レベル、TTL レベルそれぞれかなり電圧範囲が異なるため、両方に接続できるようなハードウェアの定数調整に苦労しましたが、安定して接続できるようになったと考えています。

ヘッドホンの切り替えは、2 系統のヘッドホンをオプトリレー(TLP241A)で

L/R ステレオ、切替で接続できるようにしています。このオプトリレーの切り替えは esp32 の GPIO が足りませんので、I2C エキスパンダ(MCP23017SS)を使って制御しています。MCP23017SS は 16 ポートの制御ができますので、あまりは 2x5 外部拡張ヘッダに接続しています。これらは、現在開発中のマイク切替、アンテナリレー切替等に使用する予定です。

また、ヘッドホンの出力は、コンデンサを介して esp32 の ADC に接続をしています。

24 年のハムフェアでパドルを入力したいというリクエストがありましたので、ADC を使って J6 に接続されたパドルの接点を読み取る回路を追加しました。C13 は回路図上では 0.1uF となっていますが、値が大きすぎるため現在除去しています。

外部の拡張のために、10 ピンの拡張ヘッダを用意しました。こちらには、ADC, DAC , I2C, キーイング GPIO 1 ポート, 5V 電源を引き出しています。こちらは、開発中のマイク切替等の拡張ボードで使用します。

revE と revD の違いは、基板上のイヤホンマイクのコネクターを 4pin にしたほか、基板内での回路端子の有無ですが、mini 版の機能構成の上では、関係ありません。

それから、今回頒布の mini 版で大きな改良は、Bluetooth モデムのみに使っていたサブの esp32 のプログラム書き込みを本体 CPU からできるようにしたことです。これは I2C GPIO の 2 ピンを秋月 esp32 モジュールの EN と BOOT ピンに配線して実現しています。書き込みの手順については??章で説明しています。この変更によって、これまでサブ CPU は Bluetooth モデムの働

きしかしていませんでしたが、これを拡張しています。Mini 版では端子配線がされていませんが、wide 版での音声再生はサブ C P U で行っているので、配線をすれば、活用は可能です。また、サブ C P U のメモリを活用してデュープチェック、パーシャルチェックを行うことも可能なはずなのですが、未実装です。このように、やりようによっては mini 版でもかなり wide 版と同様のパフォーマンスに近づける可能性はあります。また、wide 版、mini 版ともにサブ CPU とメイン CPU の間の通信をこれまでの 115200bps から 3Mbps に高速化しています。これによって、mini 版でも多様なアプリケーションの可能性があると考えています。

15.2 サブ CPU 書き込み改造について

mini 版では昨年度版からサブ C P U のメイン C P U から書き込めるように改造を行っていますが、昨年度版も図 15.2.1 のように 2 本の配線（10pin コネクターの図の一一番下の列の 2 ピンと秋月モジュール 0、EN 端子の間を接続）を行うことによって、同等に出来ます。

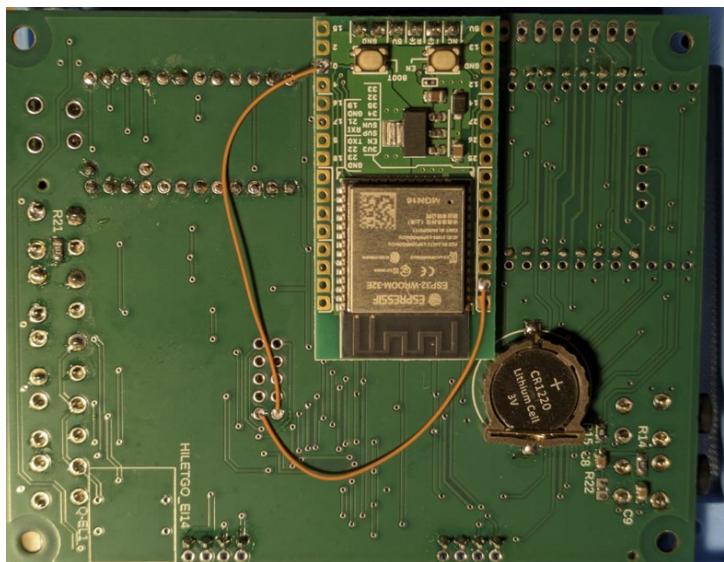


図 15.2.1 サブ CPU 書き込みのための改造ポイント

15.3 Wide 版(revF, revG)について

Wide 版は、新設計の CPU モジュールを搭載した 1 枚の 4 層基板(revG 回路図 : 図 15.2)で構成されています(写真 : 図 15.4)。頒布品の中には revF があるかと思いますが、ジャンパー等によって、revG と機能同等としてあります。

昨年の頒布時には、入手性を考えて、部品は極力、秋月で多量に扱われているものを選択し、esp32-devkitC, OLED, USB host mini などは aliexpress で入手したのですが、秋月以外で調達した部品に仕様不明点や、不良品が多く動作作品を作成するのに困難を感じたため、今年の頒布のために新規設計した revF, revG では、極力外部モジュールを削減すべく取り組みました。基板にメイン・サブ CPU (ESP-WROOM-32D)、serial-USB(CH340C)等の周回路、USB-C 端子を備えることで、市販の esp32 モジュールを全廃しました。また、メイン CPU には 8Mbytes の PSRAM が接続され、これまでの版では不可能なメモリを多く使う機能が実現できました。また、細かい点ですが、RTC を DS1307Z から DS3231 に変更したこと、RTC の動作がより安定したものになりました (部品コスト増)。

Wide 版では、2.4"の OLED の接続を前提として設計していますが、1.3"の OLED をこれまで通り搭載することもできるようになっています。1.3"の OLED 端子には VCC 3.3 V を供給しますが、2.4"の OLED には 5V/3.3V の VCC 切替ができるようになっており、頒布品は 5V VCC で動作しています。

Wide 版では、サブ CPU に USB キーボードインターフェース (CH9350L)、2 つ目の CAT (Yaesu/Kenwood) インターフェース、DAC 出力、ADC 入力 (予備)、KEY3 出力を持たせています。実際のところ、メイン CPU 側には PSRAM を接続したことによって、端子に余りが全くありません。そのため、mini 版ではメイン側の DAC で F2A トーンを出力させていました

が、Wide 版では、この機能を削除して、サブ CPU 側の DAC で実施するようになります。ただ、サブ CPU 側に移行するのはもったいない、ということで、F2A のトーン信号をデジタル生成したものにしたほか、音声再生機能を作りこみました。Wide 版の F2A はこのように機能拡張しており、現在のところ、作りこみが完全に終わっていないのですが、和文についても対応するものとなっています。

マイク切替回りは、最も今回の頒布で苦心したところです。最終的には、音声とイヤホンマイクをミキシングするミキサーマイク基板（図 15.3）を追加製造して取り付けています。音声レベルが適切となるよう、ミキサーマイクのゲイン定数は設定してありますが、必要に応じ 0603 抵抗を変更することで対応ができます。ただ、この**ミキサーマイク**の電源は、リグ側から出力される電源で動作します。リグ側からマイク端子に出されている標準的 3.3V では場合によってアンプが十分な動作しません。固定リグや、IC-705 等可能な場合は**極力 8V 程度を MIC 端子に出力して使うようにお願いします。**

これら機能、メモリ追加のため、Wide 版では CPU の入出力ピンアサイン等が変更になっています。具体的にはキー端子、USBhost mini の割り込み端子です。そのため、ソフトウェアは Wide 版と mini 版でコンパイルオプションを選ぶようになっており、それぞれ別のバイナリ配布しております。ご注意ください。

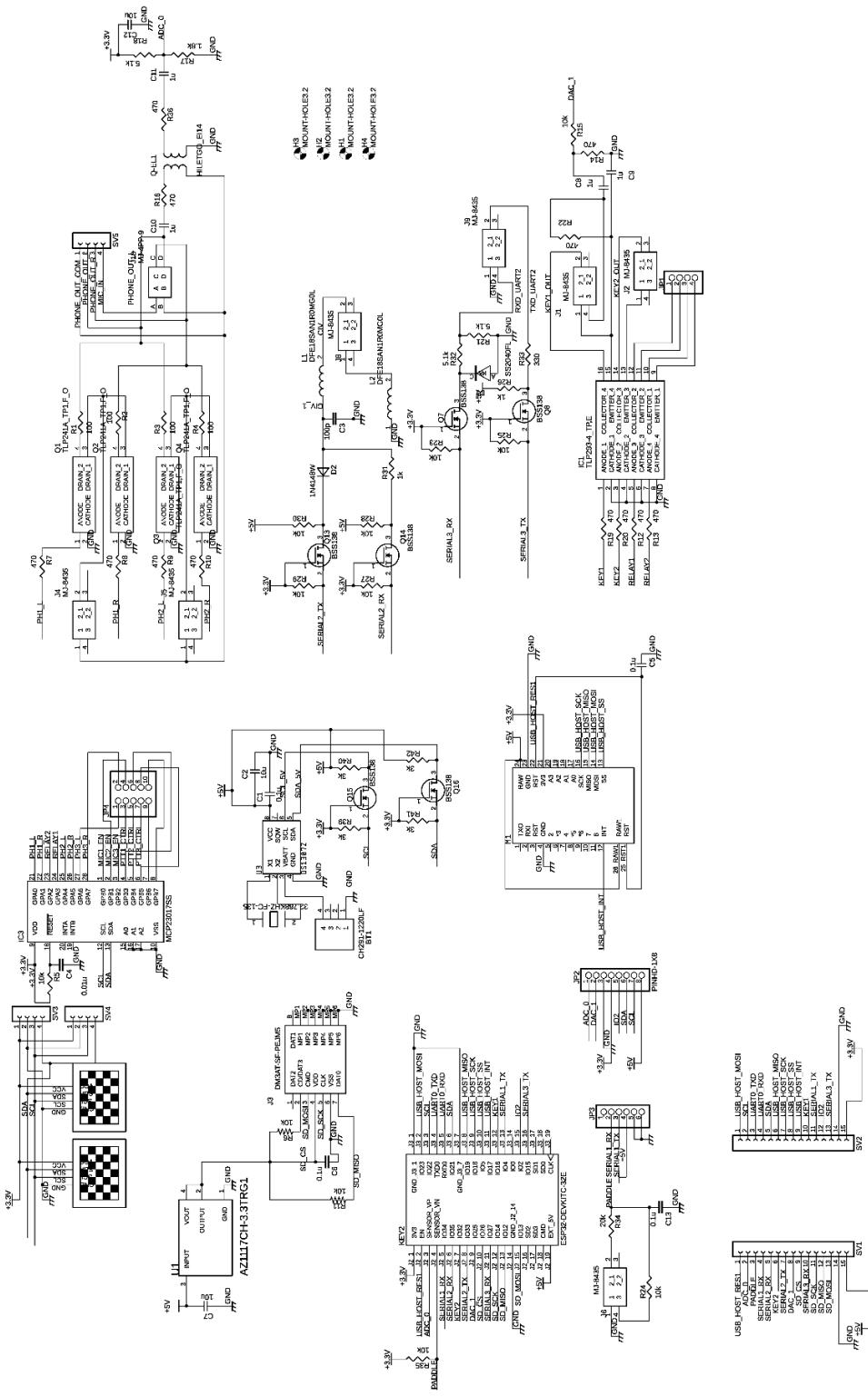


図 15.1 DVlogger 回路図 (revD) mini 版対応

Qty	Value	Device	Package	Parts	Remarks
1	USB_HOST_SHIELD_MINI_NOPATCH	USB_HOST_SHIELD_MINI_NOPATCH	USB_HOST_SHIELD_MINI_NOPATCH	M1	USB Host Mini
2	OLED	OLED	OLED	U\$1, U\$2	1.3" I2C OLED
1	microSD				16GB
1	ESP32-DEVKITC-32E	ESP32-DEVKITC-32E	MODULE_ESP32-DEVKITC-32E	KEY2	
1	0.01u	C-EUC0603	C0603	C4	
2	0.1u	C-EUC0603	C0603	C6, C13	
2	0.1u	C0805-2012	C0805	C1, C5	
1	1.8k	R0805-2012	R0805	R17	
4	100	R0805-2012	R0805	R1, R2, R3, R4	
1	100p	C-EUC0603	C0603	C3	
2	10k	R-US_R0603	R0603	R6, R11	
10	10k	R0805-2012	R0805	R5, R15, R23, R24, R25, R27, R28, R29, R30, R35	
3	10u	C0805-2012	C0805	C2, C7, C12	
1	1N4148W	1N4148W	SOD123	D2	
2	1k	R0805-2012	R0805	R26, R31	
4	1u	C0805-2012	C0805	C8, C9, C10, C11	
1	20k	R0805-2012	R0805	R34	
1	32.768KHZ-FC-135	32.768KHZ-FC-135	32.768KHZ-FC-135	U\$5	
1	330	R0805-2012	R0805	R33	
4	3k	R0805-2012	R0805	R39, R40, R41, R42	
12	470	R0805-2012	R0805	R7, R8, R9, R10, R12, R13, R14, R16, R19, R20, R22, R36	
3	5.1k	R0805-2012	R0805	R18, R21, R32	
1	AZ1117CH-3.3TRG1	AZ1117CH-3.3TRG1	AZ1117CH-3.3TRG1_0	U1	
6	BSS138	BSS138	SOT95P240X120-3N	Q7, Q8, Q13, Q14, Q15, Q16	
1	CH291-1220LF	CH291-1220LF	CH291-1220LF_1	BT1	1220電池
2	DFE18SAN1R0MG0L	DFE18SAN1R0MG0L	INDC1608X100N	L1, L2	
1	DM3AT-SF-PEJM5	DM3AT-SF-PEJM5	DM3ATSFPEJM5	J3	microSD
1	DS1307Z	DS1307Z	SO08-0.15	U3	
1	HILETGO_EI14	HILETGO_EI14	HILETGOEI14	Q-EL1	アイソレーション シトランス 600:600
1	MCP23017SS	MCP23017SS	SSOP28	IC3	
1	SS2040FL	SS2040FL	SOD123	D1	
4	TLP241A_TP1,F_O	TLP241A_TP1,F_O	SOP254P1000X425-4N	Q1, Q2, Q3, Q4	
1	TLP293-4_TP,E	TLP293-4_TP,E	SOIC127P700X220-16N	IC1	
1	MJ-4PP-9	MJ-4PP-9	MJ-4PP-9	J10	ヘッドホン端子
7	MJ-8435	MJ-8435	MJ-8435	J1, J2, J4, J5, J6, J8, J9	
4	MOUNT-HOLE3.2	MOUNT-HOLE3.2	3.2	H1, H2, H3, H4	タッピングネジ M3 基板
1	PINHD-1X8	PINHD-1X8	1X08	JP2	拡張基板接続
3		FE04-ROUND	FE04-ROUND	SV3, SV4, SV5	OLED, 拡張基板接続
2		FE15-1-ROUND	FE15-ROUND	SV1, SV2	30pin esp32モジュール接続
1		PINHD-1X6R	1X06/90	JP3	Bluetoothモデム 基板接続
1		PINHD-2X5	2X05	JP4	拡張基板接続
1		S4B-XH-A	S4B-XH-A	JP1	Relay1Relay2接続
1	秋月esp32ボード				Bluetoothモデム のため
2	3D printed case				上下
1	アクリル窓				
105					

表 15.1 DVLogger 部品表 mini 版対応

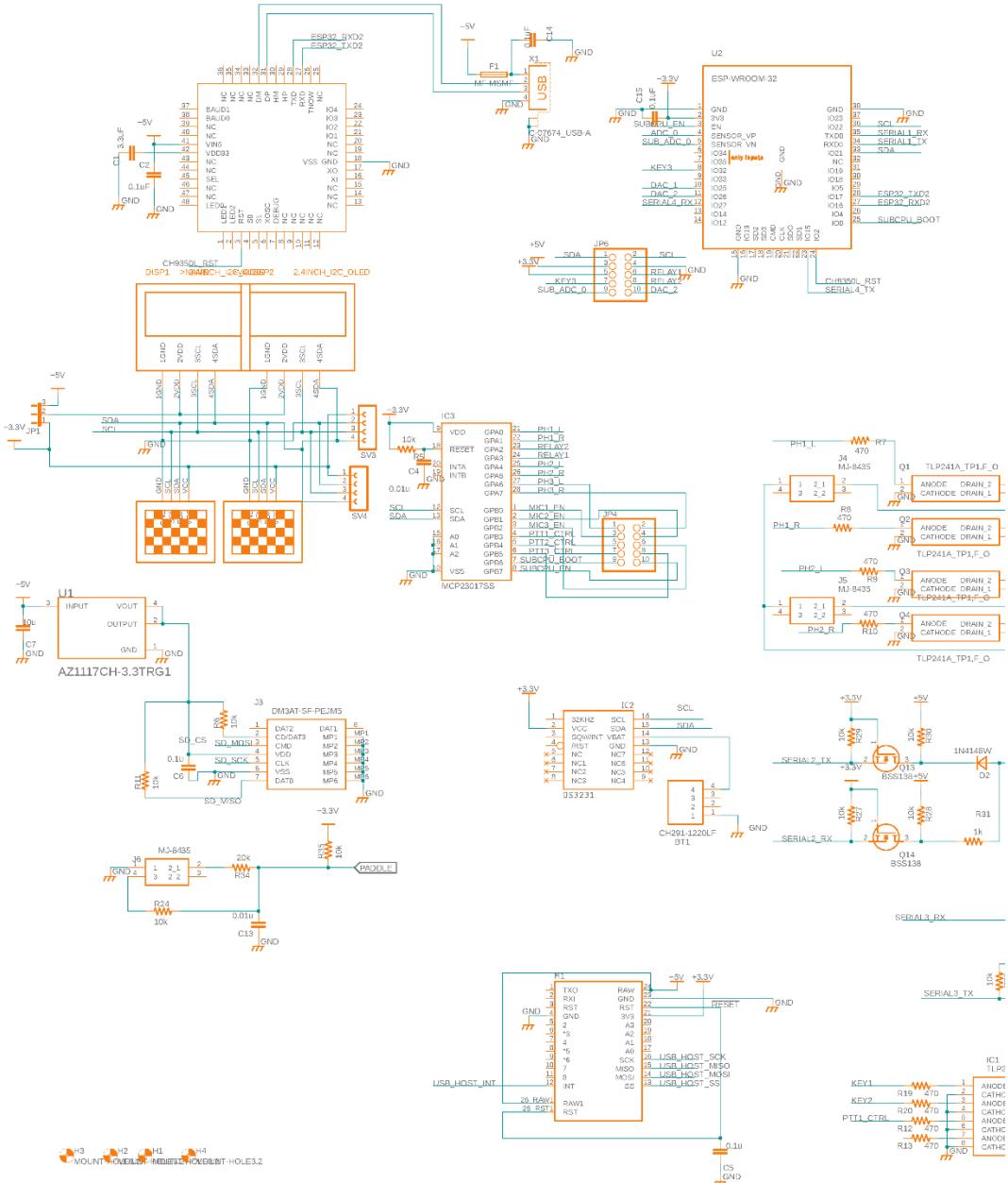


図 15.2 DVlogger rev G 回路図 (Wide版対応) (1/3)

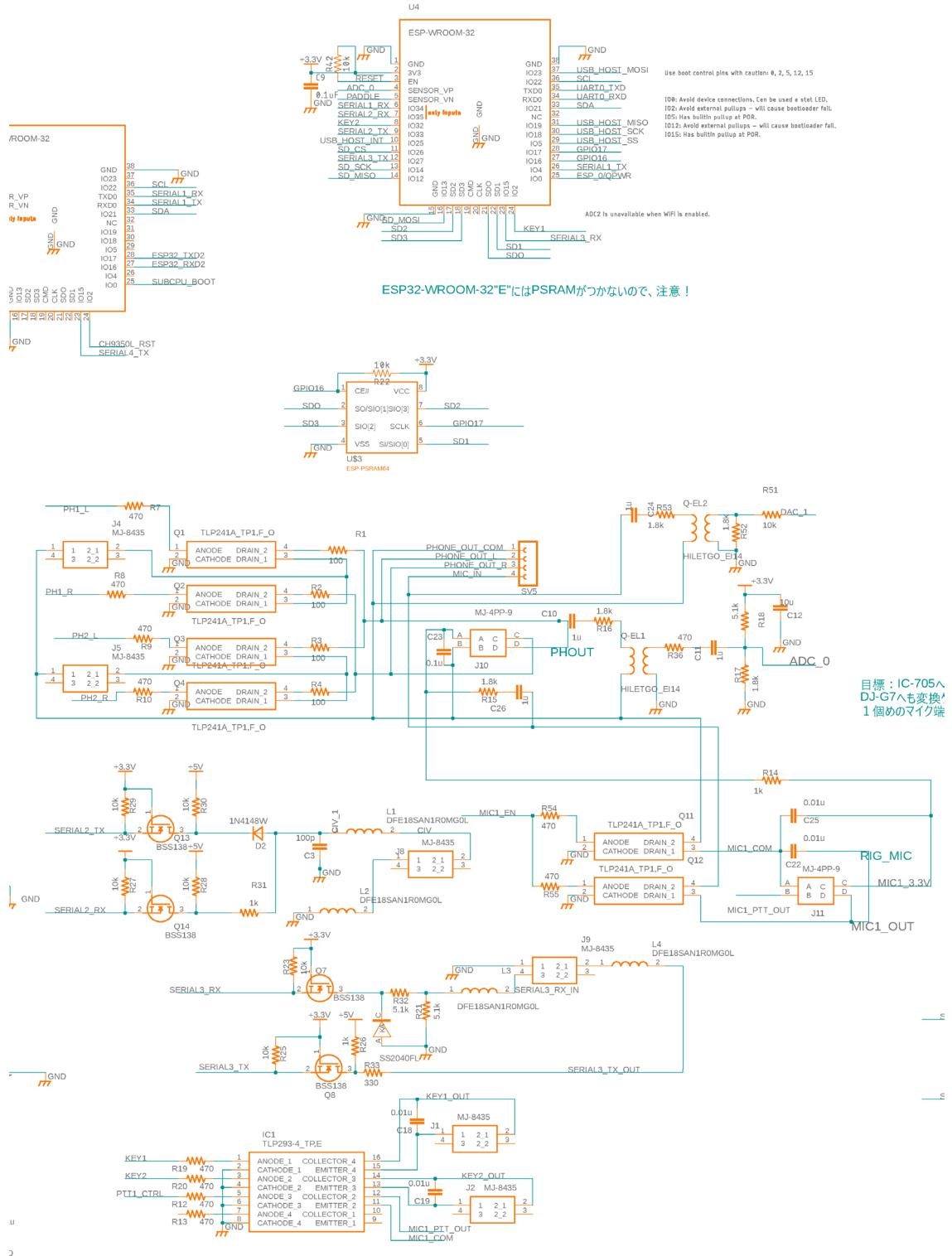
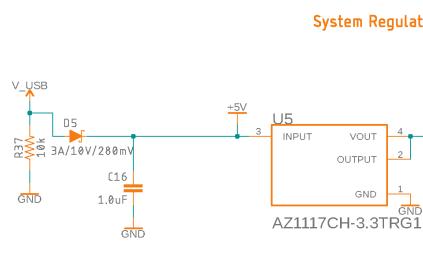
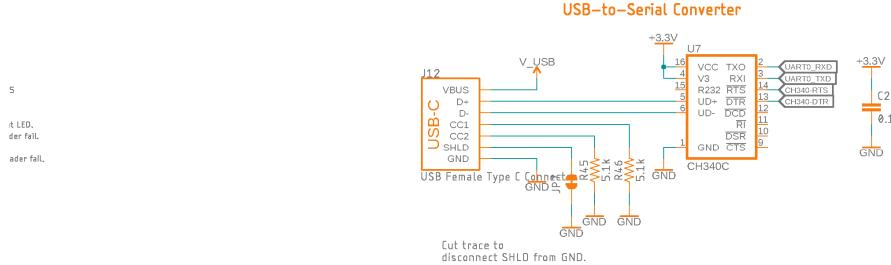


図 15.2 DVLogger rev G 回路図 (Wide 版対応) (2/3)



—
Auto-Reset

目標 : IC-705へ直結できる。
DJ-G7へも変換ケーブルを作れば繋げられる、ようにする。
1 個目のマイク端子として切替ができるようにする。

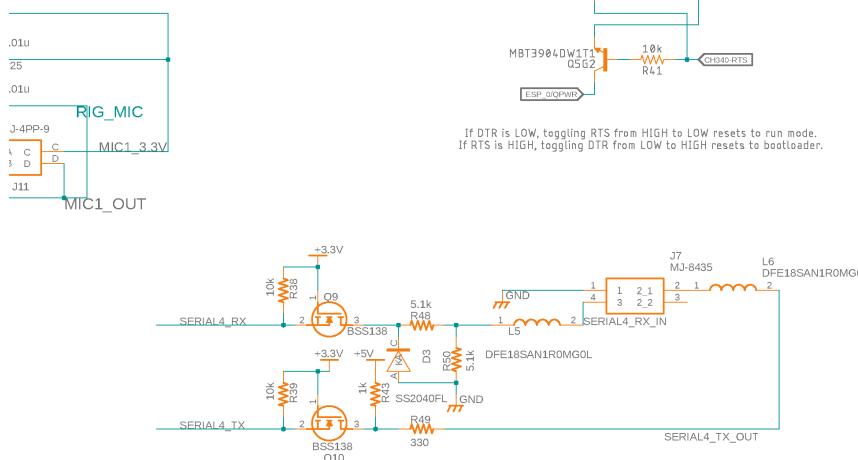


図 15.2 DVLogger rev G 回路図 (Wide 版対応) (3/3)

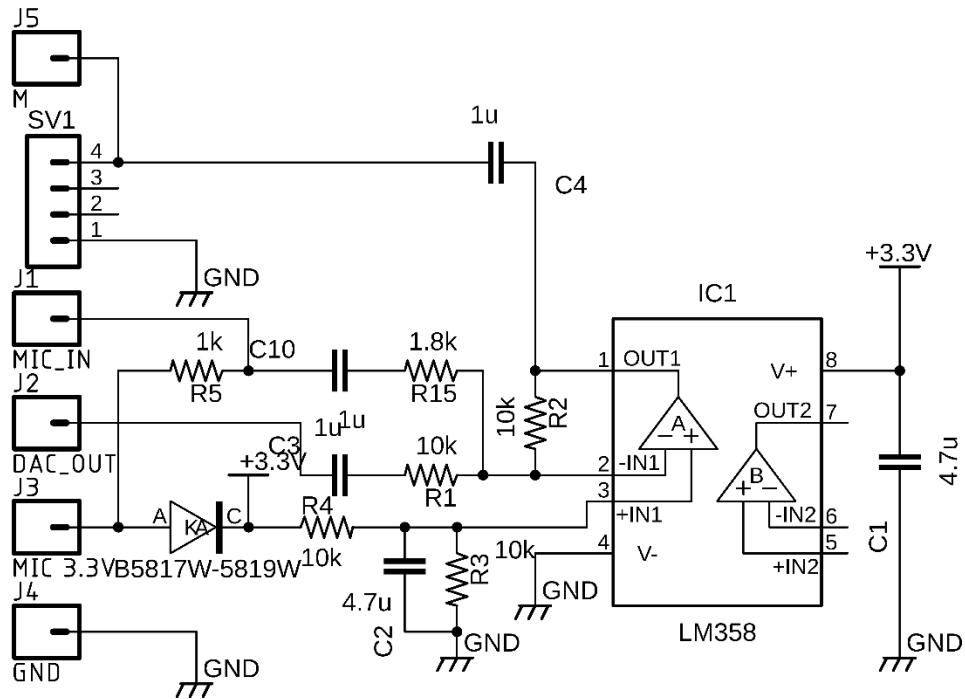


図 15.3 DVlogger マイクミキサー基板 回路図(Wide 版対応)

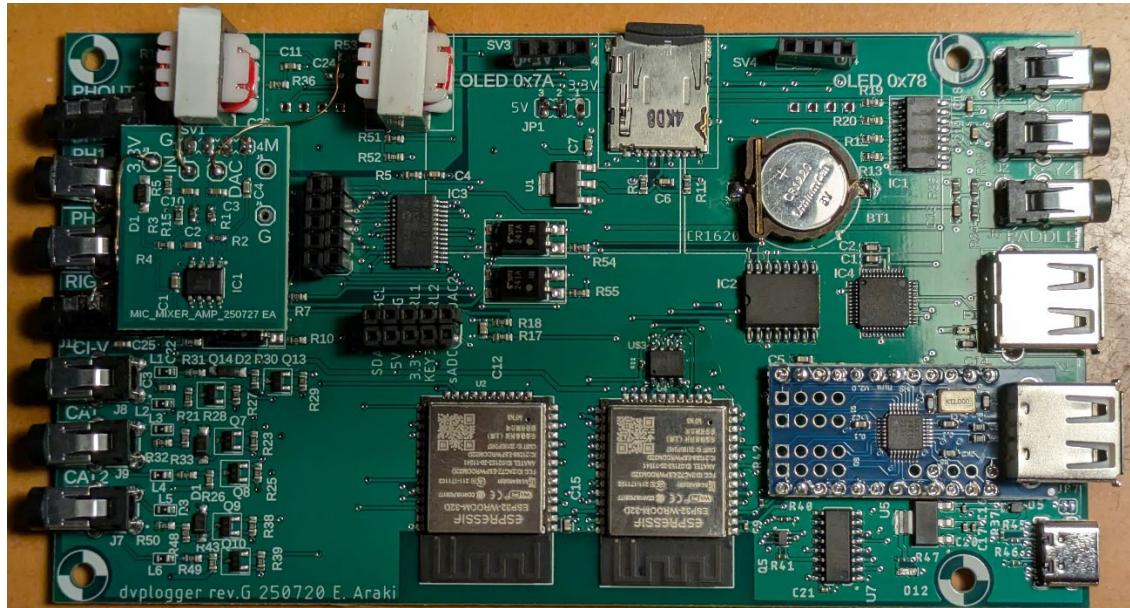


図 15.4 DVlogger Wide 版 実装写真

16 本品について

本品は、アマチュア無線の運用を楽しむために個人アマチュア(JK1DVP)が設計・製作したものです、できるだけ良好に動作するよう心掛け、開発試験、個々の動作確認もできる限り行ってはおりますが、ハードウェア・ソフトウェアともに無保証であり、本品を使用して発生したいかなる損害も補償することはできません。また、本品ハードウェアおよび資料の許可のない複製、再頒布、商業的販売は禁止しております。

17 謝辞

本品の開発・改良にあたっては、JN2MRJ, JG8GZJ, JN1MSO、JK1EDC 他各局のご意見フィードバックをいただき反映させることができました。ありがとうございました。