

# FST4およびFST4Wのクイックスタートガイド

Steve Franke, K9AN; Bill Somerville, G4WJS; and Joe Taylor, K1JT

WSJT-X 2.3.0では、特にLFおよびMF帯域用に設計されたデジタルプロトコルであるFST4およびFST4Wが導入されています。これらの帯域では、基本的な感度は同じシーケンス長の他のWSJT-Xモードよりも優れており、情報スループットの速度の理論上の限界に近づいています。FST4は双方向QSO用に最適化されていますが、FST4WはWSPRスタイルのメッセージの準ビーコン送信用です。FST4とFST4Wは、EbNautのようなモードの厳密で独立した時間同期と位相ロックを必要としません。

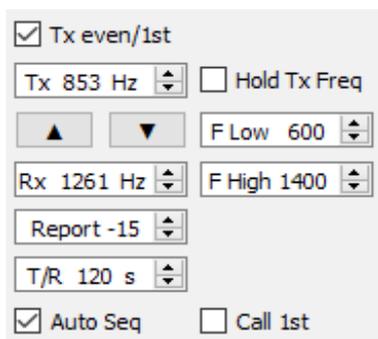
新しいモードは4-GFSK変調を使用し、メッセージのエンコードとデコードに共通のソフトウェアを共有します。FST4は15、30、60、120、300、900、および1800秒のT/Rシーケンス長を提供しますが、FST4Wは120秒より短い長さを省略します。サブモードには、FST4-60、FST4W-300などの名前が付けられ、追加された番号はシーケンスの長さを秒単位で示します。メッセージペイロードには、FT4、FT8、およびMSK144のように77ビット、またはFST4WのWSPRのようなメッセージの場合は50ビットが含まれます。ユーザーに表示されるメッセージ形式は、WSJT-Xの他の77ビットおよび50ビットモードのメッセージ形式と同じです。前方誤り訂正は、240の情報とパリティビットを備えた低密度パリティチェック（LDPC）コードを使用します。送信は160個のシンボルで構成されます。それぞれ2ビットの120個の情報伝達シンボルで、8つの事前定義された同期シンボルの5つのグループが点在しています。すべてのFST4およびFST4Wサブモードの基本的なパラメータを以下の表に要約します。加法性ホワイトガウスノイズ（AWGN）チャンネルでのシミュレーションを使用して、各サブモードのしきい値感度（2500 Hz帯域幅のSNRでデコードの確率が50%）を測定しました。WSJT-Xで最近開発された他のモードと同様に、アプリオリ（AP）デコードと呼ばれる機能により、標準の最小QSOまたはFST4W動作セッション中に情報が蓄積されるため、感度がさらに数dB向上します。

T/R period (s)	Symbol length (s)	Tone Spacing (Hz)	Occupied Bandwidth (Hz)	FST4 SNR (dB)	FST4W SNR (dB)
15	0.060	16.67	67.7	-20.7	
30	0.140	7.14	28.6	-24.2	
60	0.324	3.09	12.4	-28.1	
120	0.683	1.46	5.9	-31.3	-32.8
300	1.792	0.56	2.2	-35.3	-36.8
900	5.547	0.180	0.72	-40.2	-41.7
1800	11.200	0.089	0.36	-43.2	-44.8

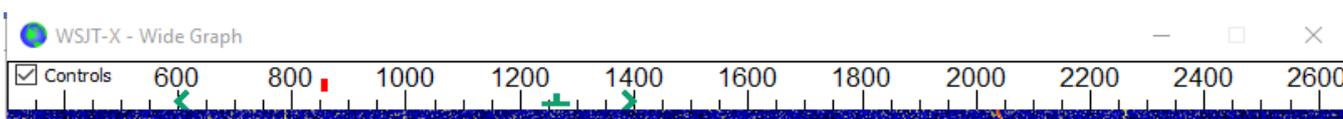
FST4-60は、主に必要に応じてマルチシンボルブロック検出を使用するため、JT9よりも約1.7dB感度が高くなります。FST4でのAPデコードでは、差は最大4.7dBになる可能性があります。FST4-120以降のシーケンス長は、比例して感度が高くなります。FST4W-120は標準のWSPRよりも約1.4dB優れており、30分のシーケンスでFST4W-1800はほぼ-45dBのしきい値SNRに達します。LFおよびMF帯域のJT9およびWSPRのユーザーは、代わりにFST4およびFST4Wの使用に移行することを強くお勧めします。

ここで説明したもの以外に、新しいモードの他のアプリケーションを考えるかもしれません。注意して頂きたいのはこれらは非常に狭帯域のモードであることです。表に記載されている感度を実現するには、オシレーターのドリフトとパスに起因するドップラーシフトが、シーケンスの全長にわたってトーン間隔よりも小さくしなければなりません。別のアプリケーションの一例として、ショートシーケンスサブモードFST4-15は、50MHzの電離層散乱経路で非常に効果的であることがわかっています。伝送長の反対の極限では、VK7MOとVK7ZBXは、非視線光散乱通信にFST4W-1800を使用し、LEDアレイ、フレネルレンズ、および光検出器を使用して153kmもの障害物のある経路にまたがって良好な成功を収めています。

WSJT-Xに精通しているオペレーターは、FST4とFST4Wを簡単に使用できることに気付くでしょう。画面上のコントロール、自動シーケンス、およびその他の機能は、他のモードと同じように動作します。LFおよびMF帯域の動作規則により、デコーダーが使用するアクティブな周波数範囲を設定するための追加のユーザーコントロールがあると便利です。[File]→[Settings]→[General]→[Single decode]がオフの場合、[F Low]および[F High]というラベルの付いたスピンボックスは、FST4デコーダーの周波数の下限と上限を設定します。

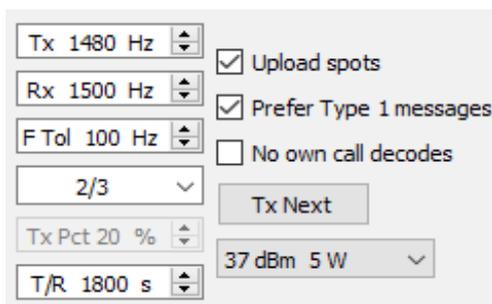


リミットは、ワイドグラフ周波数スケールで濃い緑色の山かっこ記号◇でマークされています。



Single decodeがチェックされている場合、F LowおよびF Highコントロールと緑色の< >マーカーが消え、デコードはRxFreq ± FTolの範囲でのみ行われます。

FST4Wの場合、デフォルトのRxFreqは1500Hz、FTolは100Hzであるため、アクティブなデコード範囲はWSPRの場合と同じ1400~1600Hzです。ただし、LFおよびMF帯域の動作規則に準拠するために、さまざまな中心周波数とFTol値を選択できます。F Tolの下の新しいドロップダウンコントロールは、FST4W送信をスケジュールするためのラウンドロビンモードを提供します。



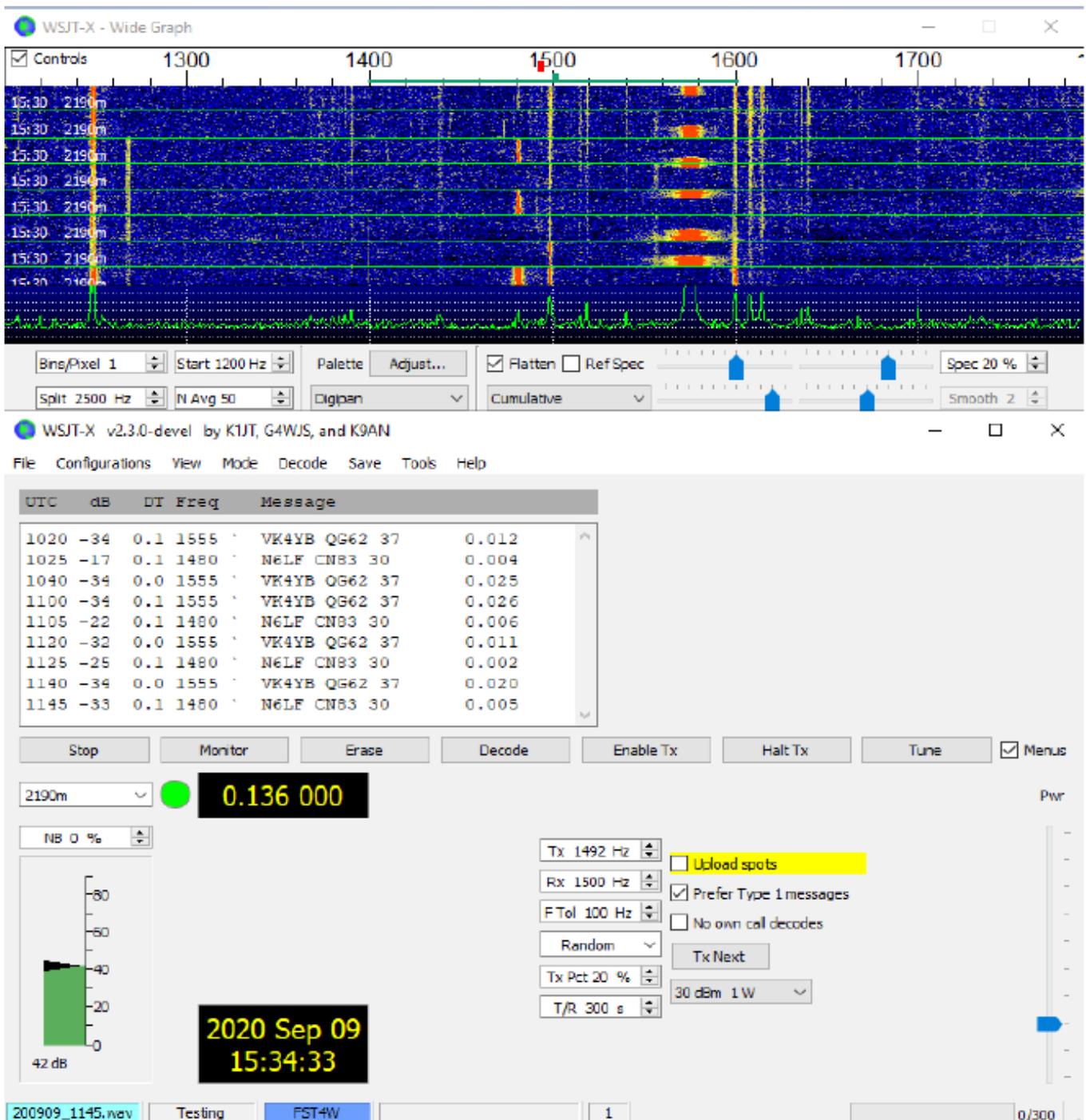
たとえば、3人のオペレーターがオプション1/3、2/3、および3/3を選択することに事前に同意した場合、FST4Wの送信は、2つのステーションが同時に送信することなく、固定された順序で行われます。シーケンス1は、00:00UTC以降の最初のシーケンスです。WSPRのようなスケジュールリング動作の場合、このコントロールで[Random]を選択する必要があります。

ドロップダウン帯域変更コントロールで現在提案されているデフォルトのFST4およびFST4W周波数を表示するには、1回のリセットを実行する必要があります。File → Settings → Frequencies に移動し、Frequencies テーブルを右クリックして、Reset を選択します。

オプションのノイズブランキングが利用可能であり、LFおよびMF帯域の大気ノイズの処理に効果的であることがわかっています。NB nn%というラベルの付いたスピナーコントロールは、FST4およびFST4Wのメインウィンドウのバンド変更コントロールのすぐ下にあります。このコントロールを、ブランクにするデータサンプルの推奨パーセンテージに設定します。5~15%の範囲のレベルは、LF/MFバンドの夏の条件でうまく機能することがわかりましたが、おそらく実験することをお勧めします。ウォーターフォールに表示され、.wavファイルに保存されたデータにはノイズブランキングが適用されていないため、事後に実験することができます。

実験的な機能では、NBの負の設定を使用して、ノイズブランキングへの「すべてを試す」アプローチをトリガーします。-1%はデコーダーに0、5、10、15、および20%のブランキングを試行させ、-2%は0、2、4、…、20%を試行します。FST4モードでは、ブランキング率がゼロ以外の試行は、周波数範囲  $RxFreq \pm Ftol$  でのみアクティブになります。NBの負の設定は非常に効果的ですが、デコード手順が大幅に遅くなります。

新しいモードの機能の例として、次のページのスクリーンショットは、2020年9月9日にNO3M（ロケータ-EN91WR）で2200m帯域で受信されたFST4W-300信号を示しています。関係する距離は、N6LFまで3501 km、VK4YBまで14,976kmです。デコードされた各行の終わりの数字は、Hz単位で広がったパストプラーの測定値です（この機能をアクティブにするには、現在の作業ディレクトリにplotspecという名前のファイルを作成し、そのコマンドラインからWSJT-Xを起動します）。原則として、デコードにはサブモードのトーン間隔よりも小さいドップラースプレッドが必要です。感度は、ドップラースプレッドがトーン間隔の1/8以下の場合に最適です。



## 付録A：FST4およびFST4Wのメッセージ形式

FST4メッセージのソースエンコーディングについては、参考文献[1]で説明されています。すべてのメッセージは77ビットのペイロードにエンコードされます。CQメッセージを送信するときにゼロの長い文字列を送信しないようにするために、アセンブルされた77ビットメッセージは、CRCおよびFECパリティビットを計算する前に、次の疑似ランダムシーケンスとビット単位の排他的論理和を取ります。

```
01001010010111101000100110110100101100001000101001111001010101011011111000101
```

受信ソフトウェアは、この排他的論理和手順を2回適用して、元の77ビットペイロードを復元します。

24ビットの巡回冗長検査（CRC）は、各77ビットの情報パケットから計算されて追加され、101ビットのメッセージプラスCRCワードが作成されます。CRCアルゴリズムは、多項式 $0x100065b$ （16進数）とゼロの初期値を使用します。

前方誤り訂正は、(240,101)LDPCコードを使用して実行されます。生成行列には、139行101列があります。これは、generator\_fst4.datファイルで定義されています。マトリックスの行*i*のゼロ以外の値は、*i*番目のパリティチェックビットを生成するために、101個のmessage-plus-CRCビットのどれを2を法として合計する必要があるかを指定します。139個のパリティビットが101個のmessage-plus-CRCビットに追加され、240ビットのコードワードが作成されます。コードワードビットのペアは、参考文献[1]の表3の3番目の列に示されているグレーエンコーディングを使用して、0~3の範囲の値を持つトーンインデックスにマップされます。結果として得られる120チャンネルシンボルのシーケンス、 $a_n$ ,  $n = 0, 1, 2, \dots, 119$ は、次の4つのグループに分けられます。

$$\begin{aligned}M_A &= \{a_0, a_1, \dots, a_{29}\} \\M_B &= \{a_{30}, a_{31}, \dots, a_{59}\} \\M_C &= \{a_{60}, a_{61}, \dots, a_{89}\} \\M_D &= \{a_{90}, a_{91}, \dots, a_{119}\}.\end{aligned}$$

同期は、送信されたフレームに5つの8シンボル同期ワードを埋め込むことによって実現されます。同期ワードは次のように定義されます。

$$\begin{aligned}S_1 &= \{0, 1, 3, 2, 1, 0, 2, 3\} \\S_2 &= \{2, 3, 1, 0, 3, 2, 0, 1\}.\end{aligned}$$

160チャンネルシンボルの完全なセットがシーケンスとしてアセンブルされます。

$$b_n = \{S_1, M_A, S_2, M_B, S_1, M_C, S_2, M_D, S_1\}$$

FST4Wメッセージのソースエンコーディングについては、参考文献[2]で説明されています。すべてのメッセージは50ビットのペイロードにエンコードされます。24ビットの巡回冗長検査(CRC)は、各50ビットの情報パケットから計算されて追加され、74ビットのメッセージプラスCRCワードが作成されます。CRCアルゴリズムは、多項式 $0x100065b$ (16進数)とゼロの初期値を使用します。前方誤り訂正は、(240,74)LDPCコードを使用して実行されます。生成行列には166行74列があります。これは、generator\_fst4w.datファイルで定義されています。マトリックスの行*i*のゼロ以外の値は、*i*番目のパリティチェックビットを生成するために、74個のメッセージプラスCRCビットのどれを2を法として合計する必要があるかを指定します。166個のパリティビットが74個のmessage-plus-CRCビットに追加され、240ビットのコードワードが作成されます。

240ビットコードワードは、FST4について前述した手順を使用して、160チャンネルシンボルにマップされます。

### 参考文献:

- [1] Steve Franke, K9AN, Bill Somerville, G4WJS, Joe Taylor, K1JT "The FT4 and FT8 Communications Protocols," QEX, July/August 2020, pp. 7-17.
- [2] Steve Franke, K9AN, Bill Somerville, G4WJS, Joe Taylor, K1JT, in preparation.