# All Things Digital

## Amateur Radio for the 21st Century

Robert C. Mazur, VA3ROM

E: va3rom@gmail.com

W: http://www.va3rom.com

# JS8 Calling – Part 1



Figure 1: JS8Call Main Operating Screen

## Introduction

"JS8Call is an experiment to test the feasibility of a digital mode with the robustness of FT8, combined with a messaging and network protocol layer for weak signal communication on HF, using a keyboard messaging style interface. It is not designed for any specific purpose other than connecting Amateur Radio operators who are operating under weak signal conditions. JS8Call is heavily inspired by WSJT-X, Fldigi and FSQCall, and it would not exist without the hard work and dedication of the many developers in the Amateur Radio community." – Jordan Sherer, KN4CRD.

JS8Call (originally FT8Call) is a keyboard-to-keyboard (kb-kb) weak signal conversation (QSO) and texting/messaging (QTC) application created by KN4CRD. It was released to the Amateur Radio world in mid-2018 for initial testing, and by the time you read this the first general release version should be in use. From a handful of initial stations, JS8Call use has grown very quickly among data mode Amateurs, and some have established very active local area JS8Call radio groups/networks. A direct derivative of the Frank-Taylor eight (FT8) digital data mode and WSJT-X software, JS8Call, the application, and JS8 (Jordan Sherer eight), the mode, shares some similarities (see Table 1). JS8 has FT8's excellent weak signal performance easily exceeding that of most other popular Amateur Radio two-way data modes, plus it supports some features of the automatic packet reporting system (APRS) and standard packet radio (AX.25).

*Note: Jordan is still making changes to JS8Call (updated on a 10-day cycle) so there may be small or large differences in what you'll download and use as compared to the RC 2 version I'm currently using.*

## JS8Call Main Screen Descriptions (see Figure 1)

In the following text the numbers and letters refer to the items marked in Figure 1.

1) Transmitter's dial frequency (carrier) and base audio tone frequency shift added to the carrier.

2) Station identification and current computer UTC time/date.

3) Various transceive control and program functions.

4) All received JS8 signals decoding monitor.

5) Incoming/outgoing text/message JS8 signal monitor.

| Bandwidth Used | Modulation Type | Transmission Frame | Transmission Interval | Transmitter Duty Cycle |
|---|---|---|---|---|
| 47 hertz (Hz) | 8-FSK USB with 5.86 Hz tone spacing. | 12.6 seconds | 15 seconds | 84% |

| Packet Length | Transmission Speed | Decoding Capability | Encoding Method | Character Set |
|---|---|---|---|---|
| 75 bits per transmission frame. | Average 10-20 words per minute (WPM) but dependent on character encoding of each packet transmitted. | Multiple JS8 signals down to a signal-to-noise ratio (SNR) of -24 decibels (dB) in a 2500 Hz bandwidth. | Huffman bit code where characters are encoded by frequency of use (i.e. "E" is "100" and "/" is "11101010"). | Upper case letters A to Z, numbers 0 to 9, shift 0 to 9 characters and most common punctuation. |

Table 1: JS8 Mode Details

6) Text/messages entered here and waiting transmission.

7) Detailed station information (transferred from decoding monitor). Mouse right-click here to bring up message menu options. Hover mouse pointer over station call sign for hearing/heard by popup info box.

8) Transmit control programmed "macro" buttons (automatically made active or inactive, as required by JS8Call).

9) Computer aided transceiver (CAT) control status and audio level strength indicator.

A) Incoming signal audio waterfall monitor and other settings. Grey area between 500 to 1000 hertz (Hz) is reserved for "heartbeat" (HB) beacons and replies from JS8 stations hearing them, or replies to queries. Bottom 500 Hz (orange area) reserved guard band for JT65.

Do **not** transmit normal JS8 communications using audio tones below 1000 Hz unless absolutely necessary (setup option must be set to allow for this).

B) Transmitter output (audio tone) level control adjustment.

C) Time delta (timing error) correction/adjustment controls.

D) Current and past transceive activity status.

*Note: Even though they are "kissing cousins", JS8 and FT8 are not mutually intelligible or compatible and are recognized as two separate and distinct Amateur Radio digital data modes.*

Jordan has done a great job detailing the history and progress of JS8Call and JS8; he also provides excellent "how-to" YouTube video tutorials so I'm only going to cover some specific features most Amateurs don't fully exploit or don't really understand (based on my on-air experience). If you need just one multifunction, weak signal data mode program in your emergency communications (EmComm) and/or public service events toolbox then JS8Call is the one, even this early in its "evolution", in my opinion.

### General Information

A customized digital signal processing (DSP) 8-frequency shift keying (8-FSK) FT8 modulation scheme is used; technically it's 8-audio FSK (8-AFSK) because shifts are generated using soundcard audio tones. There's a base radio frequency (RF) transport method (carrier) for any radio data (packets), then a directed calling protocol is added supporting both free-form and directed messaging/relaying.

Just like FT8, timing is everything and your computer's real-time clock (RTC) must be kept as accurate as possible by synchronizing it using an Internet-based network time protocol (NTP) program like Dimension 4, NetTime, Meinburg, et al, or a GPS receiver and software (I use COAA's free GPStime) when the NTP service isn't available. However, JS8 seems to be far more timing "tolerant" as compared to FT8 because I've seen text/messages with more than two second time delta errors being decoded without a problem. But if you don't have either NTP or GPS time correction available, JS8Call has a feature to manually correct for timing errors by using other JS8 received signals for adjusting time delta. This can also be

used for correcting software defined radio (SDR) processing or latency delays that can make encoding and/or decoding very time critical data modes like FT8 problematic. As an aside, perhaps it could be further enhanced for calculating/plotting JS8 signal ionospheric propagation delays to determine short/long path propagation, space weather effects and/or diurnal effects on the same (if any), etc. But that's all up to Jordan because I may be opining about an idea added a very long and rapidly growing wish list of "would it be possible to…" but I digress…

JS8Call supports multiple operating systems (Windows, Linux and Apple), the Raspberry Pi μ-controller, and it even runs on early 21st century "obsolete" Linux/Windows computers. You don't need to tie up expensive, high-end transceivers and computer processors to build standalone Internet-connected JS8Call automatic APRS gating, PSKReporter spotting, packet radio messaging and relay (repeater) stations. Receive-only gating/spotting Internet connected JS8Call stations can be built using ubiquitous and inexpensive RTL-SDR v3 USB dongles coupled to a Raspberry Pi (something new for me to learn how to do).

### APRS Features

APRS is widely used by many digital Amateurs; it's been well established on the 2m and 30m bands for over 20 years, but wasn't designed for weak signal use and requires a much higher SNR or S/N, to work reliably meaning more transmitter power is required, especially on the 30m band. And an unconnected and unnumbered "All Stations" broadcast methodology is used making it an "all or nothing" data mode. APRS transmissions aren't addressed to anyone in particular, no error correction is used, there's no ability to request repeats for missing data, and no acknowledgements (ACKs) are transmitted (unlike standard packet radio). As a result, if just one APRS bit is corrupted, the entire packet is rejected as if it never existed!

Enter JS8 with its more "forgiving" weak signal APRS subset support thanks to assistance from Chris Moulding, G4HYG (APRS Messenger and Cross-Country Wireless fame). Any JS8 station can become an ad hoc APRS-IS (Internet Service) one-way Internet "gateway" station without the operator having to know anything about APRS and the APRS-IS because it's totally transparent to them. But a little knowledge is a good thing to have...

### 1) APRS Setup

By default, only JS8 signal reporting or "spotting" to the PSKReporter Internet network is enabled. To gate (spot) signals to the APRS-IS network, you must manually activate this feature by entering an APRS "passcode" in the JS8Call Reporting setup section (see Figure 2A, top). This passcode is created from your Amateur Radio root call sign (no suffixes or prefixes), and a Google search will point you to several websites to calculate yours (Windows EXE program available on my website).

Most data mode programs only express a station's geolocation as a 4- or 6-character alphanumeric group using the Maidenhead (England) grid square locator system (mine is EN58JK), which was developed by Dr. John Morris, G4AND, in 1980. But we have a problem because APRS expresses geolocations using latitude and longitude and this requires a 10-character grid locator (at minimum) to meet the required accuracy system standard (mine is EN58JK48TO) so support for this was added to JS8Call (see Figure 2A, bottom). A free Android application called "HamGPS" uses a smartphone's (or other Android device's) built-in global positioning satellite (GPS) receiver to determine latitude and longitude then converts it to the equivalent 10-character grid locator, along with providing other very useful information (see Figure 2B on the next page). There may or may not be an iPhone equivalent application available.
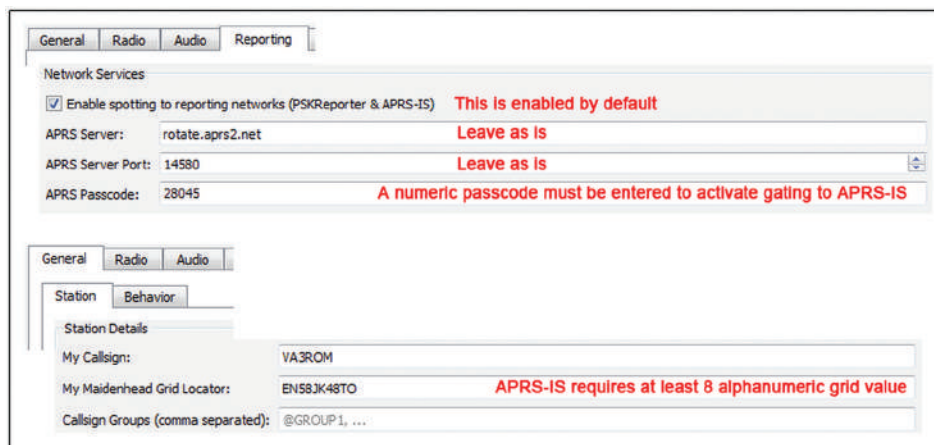
Figure 2A: APRS-IS Network Settings. Your passcode is unique to your call sign and shouldn't be used by anyone else. See http://n5dux.com/ham/aprs-passcode/ to create yours.

**Figure 2B: HamGPS**

## 2) APRS Beacons

When you want to transmit an APRS position (posit) beacon so others can see where you are on a real-time map (available via the open access aprs.fi or findu.com web servers), you simply send it using a pull-down message menu to the "@ALLCALL" group/network.
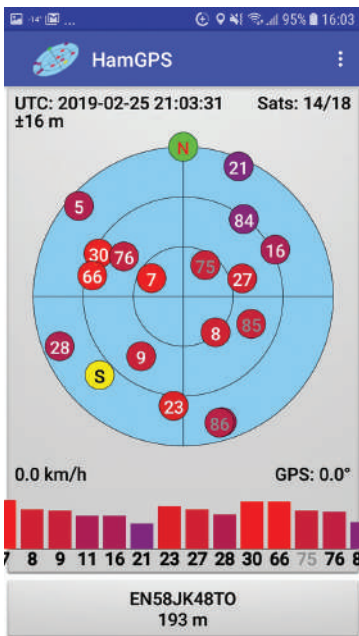
By default, every JS8 station is a member unless opted out (generally not a good idea). If it is received by another APRS passcode-enabled JS8Call Internet-connected station, it's automatically converted to a "proper" APRS-formatted packet then gated to the APRS-IS network (see Figure 2C).

## 3) Short Message Service (SMS) – Undocumented Feature

An APRS ancillary service is the ability to send a short text message (maximum 67 characters) via the APRS-IS over to the SMS gateway server (provided by Paul Dufresne, VE3OTB), which is connected to the commercial cellular phone system (for more details see http://smsgte.wixsite.com/smsgte). With dedicated APRS-to-APRS stations this can be used as a two-way system, but for JS8Call it's only one way.

But unlike the Winlink 2000 (WL2K) radio email system SMS support, with JS8Call you only need to know the destination smartphone's 10-digit number (it counts as part of 67 characters) and don't need to know and append the specific cellular service routing @address.

As of JS8Call RC 2, you have to manually type in the properly formatted APRS SMS text packet, but you can create and store a pre-formatted SMS template using a text editor then copy/paste that into JS8Call and make any required changes.

For example, my summer camp trailer site has no Internet, Wi-Fi, landline, electricity, etc., and my only means of communicating with the world is via Amateur Radio (using battery/solar power).



Figure 2C: APRS-IS Transmitting and Gating Example
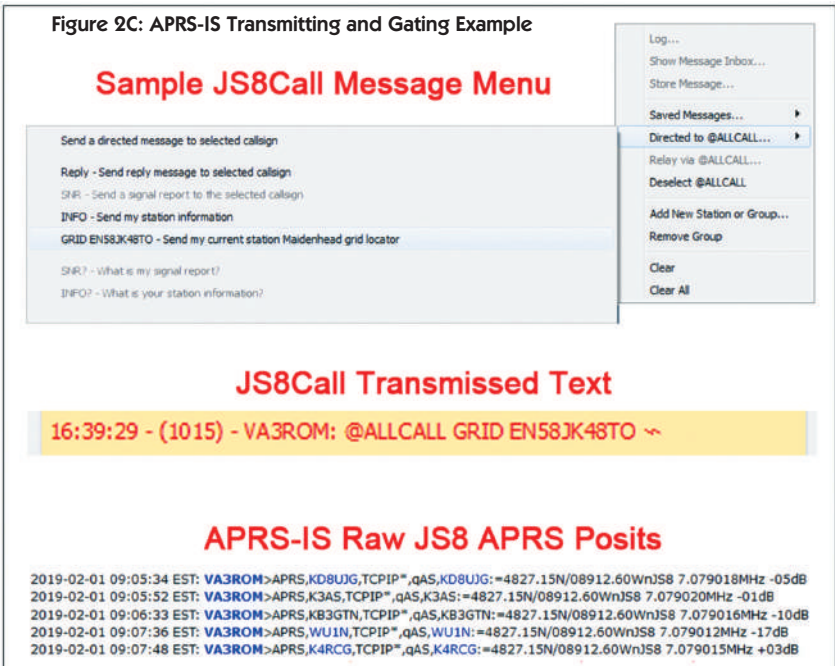
If I want someone to know my current whereabouts and/or status, I can use JS8Call to send an SMS text like this:

@ALLCALL APRS::SMSGTE :@8075551234 ARR CAMP ALL OK CU FRI ROBT

The originating phone number they'll see attached to your incoming SMS text will be that of the SMSGTE service provider so will you need to explain things beforehand or they may think it's phone spam mail, delete it and block the number (no SMS from you!), and/or try to reply back.

*Note: The routing information between "APRS::" (note the double colons) and the ":@" must be exactly 9 characters so spaces are added for padding. The "@ALLCALL" is used for an "All Stations" general service request or information broadcast, and is **not** intended to solicit calls from stations hearing it to initiate a QSO – that's the purpose of "CQ".*

Duplicate APRS SMS text packets gated by multiple stations are automatically rejected by the APRS-IS to ensure only one is passed on to the SMSGTE server. The only caveat is there's no guarantee your SMS text will be received and gated (no dedicated JS8 monitoring network). To solve this problem, I set up a full-time home-based 30 metre band 24/7 JS8 monitoring (receive only) Internet-connected station (my camp is within ground wave coverage so all the better). Based on my reception tests with other JS8 stations, only a handful of strategically placed stations would be required to virtually guarantee coverage of all of North and Central America.

*Note: The APRS-to-SMS service is free to use because expenses are covered by the SMSGTE provider, but if you do use this service you and/or your club/organization can contribute funds to help defray costs and show your support (there's a "Donate" link on the website). More than 16,000 SMS texts have been handled by the SMSGTE server since 2014 so imagine that phone bill!*

### Beacons Straight through the Heart

Unlike the weak signal propagation reporter (WSPR) one-way only propagation beacons, JS8Call supports a two-way beacon information and ersatz propagation system. JS8 beacons, called "heartbeats" (HB), are transmitted at specific or user-defined intervals and announce a station's availability for APRS gating, PSKReporter spotting, packet radio messaging and/or relaying (see Figures 3A and 3B on the next page).

On the receive end, all JS8Call stations with AUTO transmit mode enabled (and have transmit capability), and not engaged in another activity will automatically send an ACK back with the call sign of the station hearing your HB and your received SNR. Because JS8Call can decode multiple simultaneous JS8 signals (within a 2500 Hz pass band) you can often receive multiple ACKs back.

*Note: By default, a station's HB must be manually activated, and it's automatically deactivated whenever you're using the keyboard for various JS8Call activities. It also doesn't mean that the originating station sending a HB wants to engage in a QSO – again, that's the purpose of the CQ.*

**Figure 3A: Heartbeat Beacon Examples**





**Figure 4A: JS8Call Sample On-air Queries. Queries from stations are in the form of a question "?" appended using a mix of standard international Q-codes mixed in with English texting short forms, as required.**
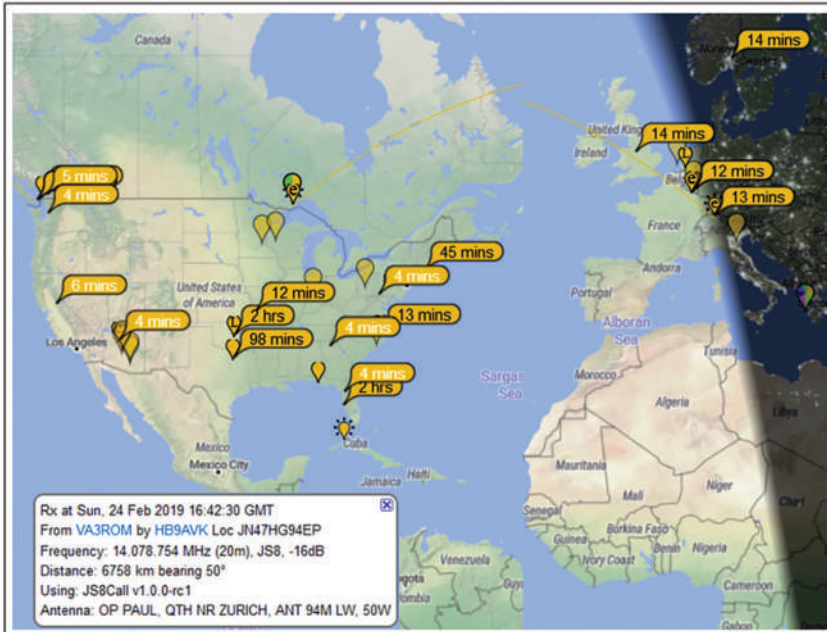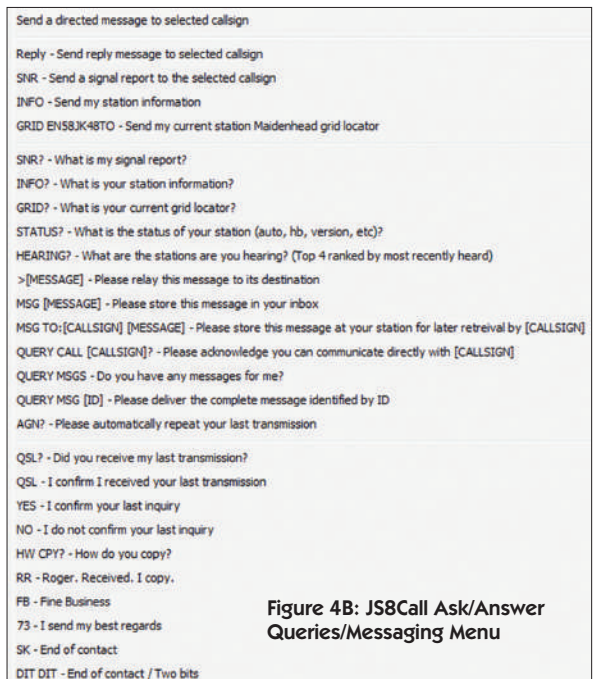


**Figure 3B: PSKReporter JS8 Stations Map. An edited image of JS8 stations I have heard and spotted or who have heard and spotted me on the 20m band (past three hours). Credit PSKReporter.info.**



**Figure 4B: JS8Call Ask/Answer Queries/Messaging Menu**

## Queries or Ask and Answer

JS8 stations can query each other for specific information. If a transmitted query is received by a station with transmit capability – and AUTO mode is active and not engaged in another activity – an auto-reply is sent back to the asking station with the requested information (see Figure 4A).

With reference to Figure 4B, JS8Call has a message menu list of various transmit functions you can access with the often used ones available as predefined macro buttons for faster access. But JS8 is less of a macro button mode than say PSK31 because of its much slower user operating speed (no touch typist and/or speed reader skills required), which gives you lots of time to think, type two-finger replies, and leisurely read text between exchanges (but it's not as painfully slow like JT65). Because of this, JS8Call is well suited to and works very well with text-to-speech, speech-to-text, text magnifiers and other adaptive/assistive technologies for those who need and use these such as Handihams.

*Note: Because some countries' Amateur Radio regulations prohibit unattended, automatic transceive operations (a control operator must always be present), JS8Call's default Behaviour > Idle Timeout is set to 60 minutes (you can change or disable this). If there's no keyboard input or mouse movement detected before the idle timer expires, the AUTO (transmit) mode is automatically turned off and your station operates in SPOT (receive) only mode.*

## My Final

Okay, I have to stop here. Hopefully your interest has been tweaked and perhaps you'll give JS8Call a try. Part 2 looks at its two-way packet radio messaging features, including local store and retrieve, remote store and forward, relays, plus its networking ability by creating custom @groups (e.g., @NWOARES, @ EOCSAO, @SIBLOPIT, etc).

This could be of particular interest to those involved in the Amateur Radio emergency service (ARES) and/or public service events where you need/want to keep radio traffic targeted to/from and within a specific group/network. – 73

TCA