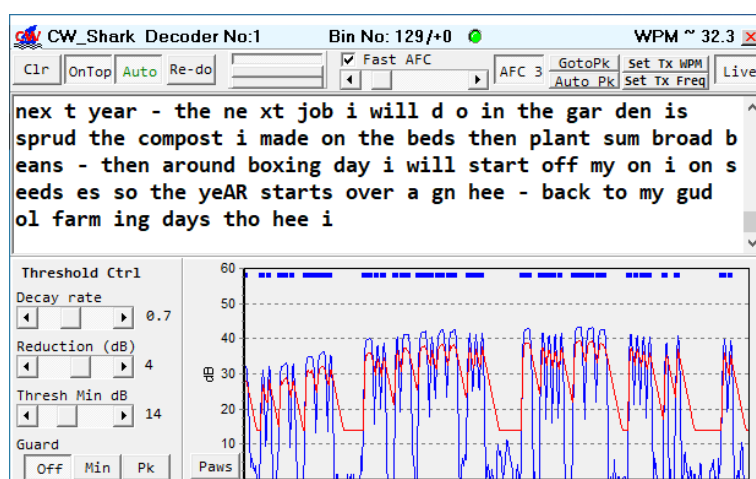
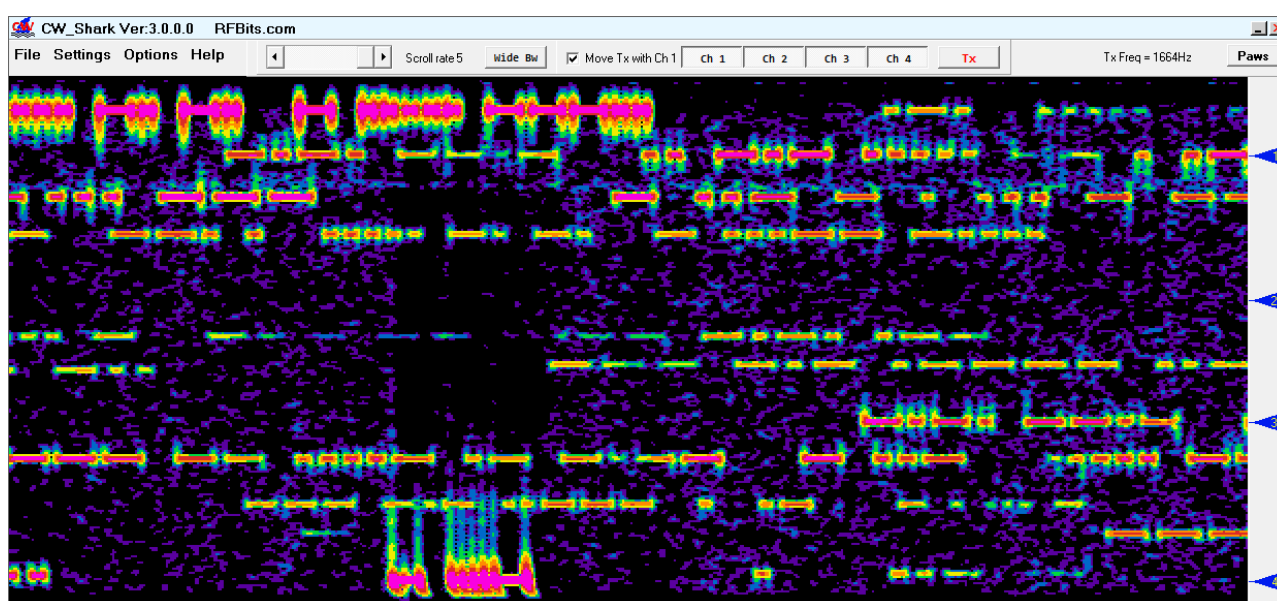


CW_Shark Manual

A 32 bit Windows morse engine

RFBits.com



Issue 3.0 27th Oct 2025

CONTENTS

1.0	CW_SHARK END-USER LICENSE AGREEMENT ("AGREEMENT")	6
1.1	Limit of liability	6
1.2	License	6
1.3	Restrictions	6
1.4	Term and Termination	6
1.5	Severability	6
1.6	Amendments to this Agreement	6
1.7	Important Information	6
1.8	Contact Information	6
1.9	Steinberg ASIO	7
2.0	ABOUT CW_SHARK	8
2.1	Decode Only.	8
2.2	Keyboard QSO	8
2.3	Straight or Paddle Key QSO	8
2.4	Straight or Paddle key practice	8
2.5	Discussion	8
3.0	MAIN FORM	9
3.1	Menu	10
3.1.1	File	10
3.1.2	Settings	10
3.1.3	Options	11
3.1.4	Help	11
3.2	Scroll Rate	12
3.3	Wide Band	12
3.4	Move Tx with Ch1	12
3.5	Decoder Channel Buttons	12
3.6	Tx Button	12
3.7	Tx Freq	12
3.8	Paws Button	12
3.9	Minimise	12
3.10	Close	12
3.11	Ch4 Marker	12
3.12	Ch3 Marker	12
3.13	Tx Marker	13
3.14	Ch2 Marker	13
3.15	Ch1 Marker	13

4.0	SETTINGS FORM	14
4.1	Sound Card	14
4.1.1	Morse Source	14
4.1.2	Tx Output	14
4.1.3	Tx AF Buffers	14
4.1.4	Refresh Drivers	14
4.2	Aux AF	15
4.2.1	Enable Aux AF	15
4.2.2	Aux AF Out	15
4.2.3	Aux AF Buffers	15
4.2.4	ST Vol	15
4.2.5	Rx Vol	15
4.2.6	Raw / Filtered	15
4.3	Tx Ctrl	16
4.3.1	Tx Output level	16
4.3.2	PTT & Com port settings	16
4.4	Tx Monitor	17
4.4.1	Enable Tx Monitor	17
4.4.2	CRs before and after	17
4.4.3	Delay compensation in	17
4.4.4	Delay Compensation out	17
4.5	Text	18
4.5.1	Rem Extra E's	18
4.5.2	Mark Runs	18
4.5.3	All Caps	18
4.5.4	Font Size	18
4.6	Waterfall	18
4.6.1	Top	18
4.6.2	Step	18
4.7	ASIO	19
4.7.1	Paddle Side Tone	19
4.7.2	Keyboard Keyer ST	20
4.7.3	Rx Vol	20
4.7.4	Raw / Filtered	20
4.7.5	ASIO Engine	20
4.7.6	ASIO Ctrl panel	20
4.7.7	Win Snd Panel	20
4.8	Keyer	21
4.8.1	Key Type	21
4.8.2	Side Tone Freq	21
4.8.3	Key PTT hold time	21
4.8.4	KeyTims	21
4.8.5	Lax	21
4.8.6	Com Port Selection.	21
4.8.7	Swap	21
4.8.8	Invert Signal	21
4.9	Key Tims	22
4.9.1	Form Scale	22

4.9.2	Dot Paddle	22
4.9.3	Dash Paddle	22
4.9.4	CW Signal	22
4.9.5	Speed WPM	22
4.9.6	Example IambicA 'CW'	22

5.0 THE DECODER FORM 23

5.1.1	Decoder Instance No	23
5.1.2	Bin No	23
5.1.3	Decode LED	23
5.1.4	WPM	23
5.1.5	Clr	23
5.1.6	Ontop	23
5.1.7	Auto	23
5.1.8	Re-Do	24
5.1.9	Show Sig Lev	24
5.1.10	Show Dot Dash Workings	25
5.1.11	Show mark Gap History	26
5.1.12	AFC Search Depth	26
5.1.13	Fast AFC	26
5.1.14	AFC On/Off	26
5.1.15	Goto Peak	26
5.1.16	Auto Peak	26
5.1.17	Set Tx WPM	26
5.1.18	Set Tx Freq	26
5.1.19	Live	27
5.1.20	Close	27

6.0 THE KEYBOARD TX FORM 28

6.1	Stop	28
6.2	Speed Control	28
6.3	Memory Buttons	28
6.4	Tune Button	29
6.5	Blue text edit.	29
6.6	White text edit.	29
6.7	Tx Text Window	29
6.8	Tx character set	29

7.0 HOW IT WORKS 30

7.1	Receive	30
7.1.1	Audio processing.	30
7.1.2	The Mark Gap Amplitude Decision.	30
7.1.3	The Timing Thresholds.	30
7.1.4	The Dot Dash decision.	31
7.1.5	About the Amplitude threshold.	31
7.2	Transmit	32
7.2.1	Text to audio stream.	32
7.2.2	Paddle Keyer	32
7.2.3	Tx Monitor	32
7.2.4	PTT	32

8.0	THE ASIO PADDLE KEYER	33
8.1	Overview	33
8.2	Enabling the ASIO keyer	33
8.3	Installing and Configuring the ASIO4ALL driver to the correct soundcard.	34
9.0	ABOUT SOUNDCARDS	36
9.1	About External SoundCards	36
9.2	About Bluetooth Sound devices	36
9.3	About virtual audio cables	37
10.0	ABOUT SERIAL PORTS (COM PORTS)	38
10.1	Motherboard Serial ports (PTT or paddle keyer)	38
10.2	USB Serial ports (PTT or paddle keyer)	38
10.3	Virtual Serial ports for PTT	39
11.0	TIPS ON RUNNING CW_SHARK	40
11.1	Receiver Bandwidth - Audio level	40
11.2	Receiver AGC	40
11.3	CPU load	40
11.4	Real-Time Priority	40
11.5	Windows 11 Admin mode	40
11.6	Some notes on ASIO Sidetone latency	40

1.0 CW_SHARK END-USER LICENSE AGREEMENT ("AGREEMENT")

In order to use the CW_Shark programme, the user must agree to be bound by the following Licence Agreement.

If you do not agree to the terms of this Agreement, do not use the programme.

1.1 Limit of liability

This software is intended for educational and personal use only. There is no guarantee that the software works or that it cannot damage your data or hardware. Use of the software is solely at the operator's risk.

1.2 License

The author grants you a revocable, non-exclusive, non-transferable, limited license to download, install and use the Application solely for your personal, non-commercial purposes strictly in accordance with the terms of this Agreement.

1.3 Restrictions

You agree not to, and you will not permit others to:

- a) License, sell, rent, lease, assign, distribute, transmit, host, outsource, disclose or otherwise commercially exploit the Application or make the Application available to any third party.
- b) Modify, decompile or reverse engineer the Application for any reason.

1.4 Term and Termination

This Agreement shall remain in effect until terminated by you or the software author.

The author may, in his sole discretion, at any time and for any or no reason, suspend or terminate this Agreement with or without prior notice.

This Agreement will terminate immediately, without prior notice from the author, in the event that you fail to comply with any provision of this Agreement. You may also terminate this Agreement by deleting the Application and all copies thereof from your device.

Upon termination of this Agreement, you shall cease all use of the Application and delete all copies of the Application from your devices.

1.5 Severability

If any provision of this Agreement is held to be unenforceable or invalid, such provision will be changed and interpreted to accomplish the objectives of such provision to the greatest extent possible under applicable law and the remaining provisions will continue in full force and effect.

1.6 Amendments to this Agreement

The author reserves the right, at his sole discretion, to modify or replace this Agreement at any time.

1.7 Important Information

CW_Shark at this time is still in development, 27th Oct 2025, and presently has an expiry date of 1st Jan 2026, after which it will refuse to run. A new version should be available from the author before this date.

1.8 Contact Information

If you have any questions about this Agreement, please contact the software author at:-

RFBits.com@gmail.com

1.9 **Steinberg ASIO**

This programme uses parts of the Steinberg ASIO Licenced Software Developer Kit.



ASIO is a trademark and software of Steinberg Media Technologies GmbH.

2.0 ABOUT CW_SHARK

CW_Shark is a 32bit windows programme, design to analyse, decode and encode morse code.

The programme can be used in four modes :-

2.1 Decode Only.

This is the simplest option and only requires the audio from the radio to be fed to the soundcard of the PC. Not recommended but it will work using the microphone in a laptop to listen to the audio from an HF radio, (disable any noise reduction dsp on the laptop microphone).

2.2 Keyboard QSO

For transmission you need the output of the soundcard to feed the mic input of the radio. Additionally, PTT must be enabled either by VOX or preferable via the RTS of a COM port. If you have a working setup for digital modes such as FT8, then those settings should work for this programme.

2.3 Straight or Paddle Key QSO

To enable the paddle keyer, the ASIO4ALL driver must be installed for a 2nd soundcard, which will provide a low latency sidetone. See section 8.0

2.4 Straight or Paddle key practice

CW_Shark's Tx monitor, allows hand sent morse to be decoded and analysed for timing accuracy. Further analysis is shown the 'Key Tims' form shows the timing of the paddle keys and the resultant CW.

Details of the soundcard and Com port setup are given later in this document.

2.5 Discussion

CW_Shark is intended for personal morse code practice, in the hope that the programme is no longer needed.

The first version of the programme was started in 2005, but was only resurrected in June 2025.

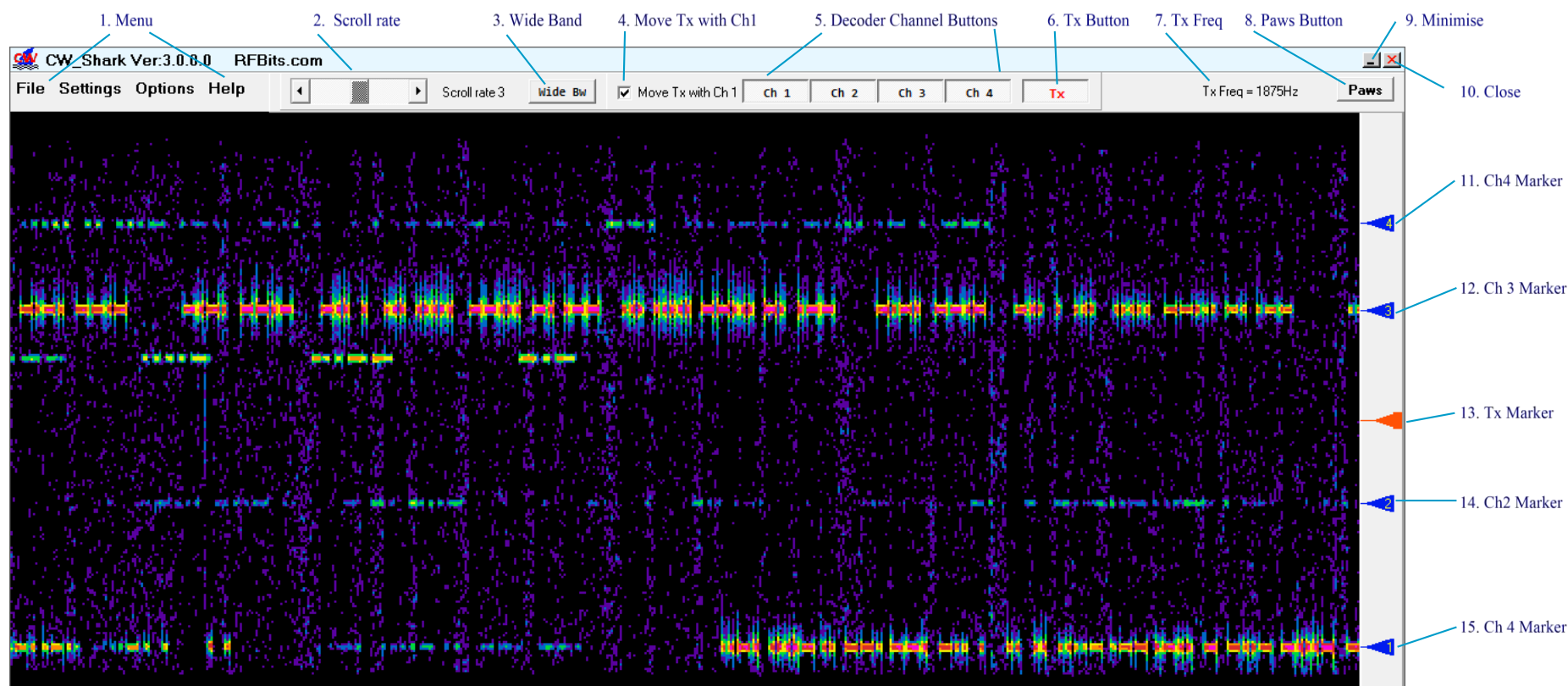
3.0 MAIN FORM

The main form of the programme is shown below. It consists of a set of controls around the main horizontal waterfall.

The horizontal axis of the waterfall is time, the vertical is audio frequency ranging from DC to 4kHz. If you radio has a narrower filter you will not see the all these frequencies.

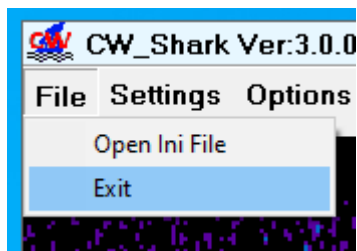
A double mouse click on a line of CW will open decoder 1, showing the last 60 secs decode on that frequency. Shift, Ctrl and Alt with double clicks will open decoders 2, 3 and 4 respectively. A right mouse click will decode to a hint window.

The controls are numbered here and the following section details their functions.



3.1 **Menu**

3.1.1 **File**



3.1.1.1 **Open Ini file**

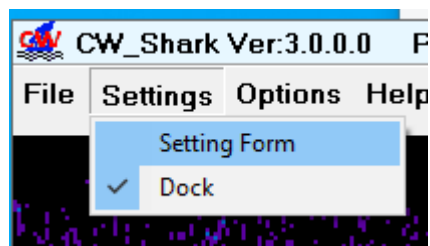
Opens the ini file where the settings are stored.

Delete this file to go back to default values.

3.1.1.2 **Exit**

Closes the Programme.

3.1.2 **Settings**



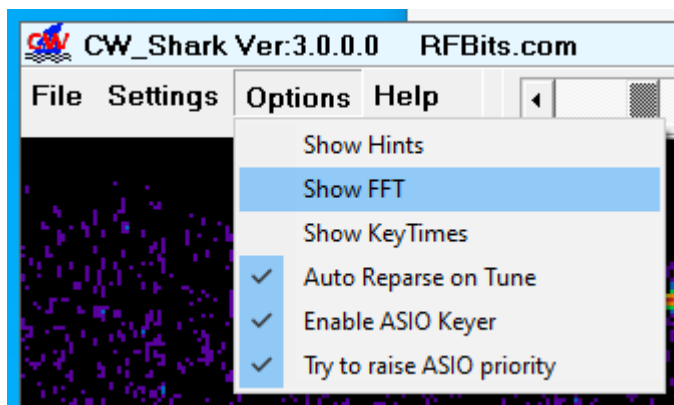
3.1.2.1 **Setting Form**

Opens the settings form see section 4.0

3.1.2.2 **Dock**

When checked the settings form will stay in the top left corner of the waterfall.

3.1.3 Options



3.1.3.1 **Show Hints**

Turns on/off the control hints.

3.1.3.2 **Show FFT**

Extends the main form to display the current FFT values.

3.1.3.3 **Show Key Times**

Opens the 'Key Tims' form see section 4.9.

(Only available if ASIO keyer is enabled.)

3.1.3.4 **Auto Reparse on Tune**

If checked, when a decoder is re-tuned it will clear the text window and reparse the last 60 secs of that frequency.

3.1.3.5 **Enable ASIO keyer**

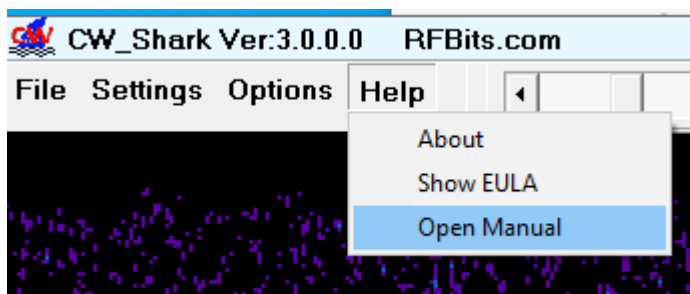
Enables or disables the ASIO paddle keyer. (Effective only after programme restart.)

See section 8.0 before enabling this option.

3.1.3.6 **Try to raise ASIO priority**

If checked the programme will try to raise the priority of the ASIO thread. This can help with drop outs on the ASIO sidetone. (Effective only after programme restart.)

3.1.4 Help



3.1.4.1 **About**

Opens a simple about form.

3.1.4.2 **Show EULA**

Shows the End User Licence Agreement

3.1.4.3 **Open Manual**

Open this document in PDF viewer.

3.2 Scroll Rate

This slider sets how fast the main horizontal waterfall scrolls to the left.

Rate 1 - 60 secs of history, (this is the size of the Reparse buffer).

Rate 2- 30 secs;

Rate 3- 15 secs;

Rate 4- 7.5secs.

Rate 5- 3.25 Secs. (Note, the faster scroll rates do hit the CPU)

3.3 Wide Band

Depress this button to double the filter bandwidth, useful for CW rates above 36 WPM.

3.4 Move Tx with Ch1

Check this box and the Tx frequency will move with the Ch1 marker.

3.5 Decoder Channel Buttons

Depressing these buttons will open and activate the relevant decoder form, note the decoder does not run unless open.

See section 5.0 for full description of the decoders.

3.6 Tx Button

Depressing this button will open the Tx form.

See section 6.0 for a full description of the Tx form.

3.7 Tx Freq

This label shows the current setting of the Tx audio frequency. This value can be used to manually set a radio's 'Pitch Freq', if you want to use the radio's CW keyer.

3.8 Paws Button

Depress the 'Paws' button to pause the sound card input.

All functions will stop.

3.9 Minimise

This will send the programme and all its windows to the task bar, the programme will still continue to run.

3.10 Close

This will close the programme.

3.11 Ch4 Marker

This shows the current frequency of Ch4. Use the mouse to drag the marker to chosen frequency. If the Ch4 is not active then this marker is not shown.

3.12 Ch3 Marker

This shows the current frequency of Ch3. Use the mouse to drag the marker to chosen frequency. If the Ch3 is not active then this marker is not shown.

3.13 Tx Marker

This shows the current frequency of the Tx audio. Use the mouse to drag to chosen frequency, can also be tied to Ch1 Rx, or any decoder can use it's 'Set Tx freq' button to net the frequency.

If the Tx form is not visible then this marker is hidden. Note this marker is red and slightly bigger than the Rx markers so it can always be seen.

3.14 Ch2 Marker

This shows the current frequency of Ch2. Use the mouse to drag the marker to chosen frequency. If the Ch2 is not active then this marker is not shown.

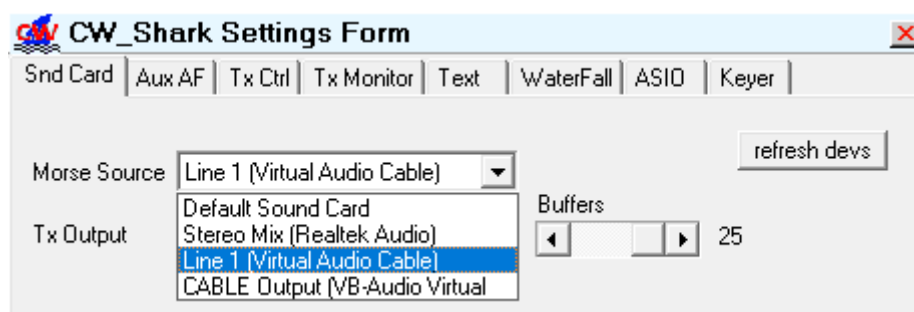
3.15 Ch1 Marker

This shows the current frequency of Ch1. Use the mouse to drag the marker to chosen frequency. If the Ch1 is not active then this marker is not shown.

4.0 SETTINGS FORM

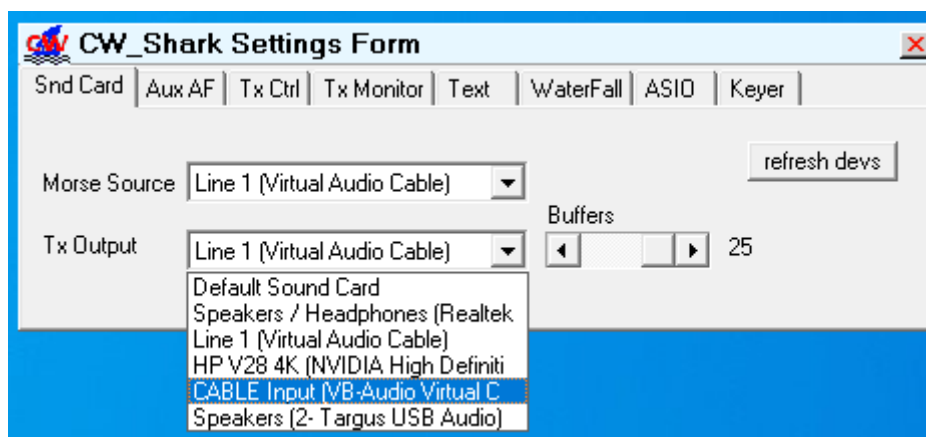
4.1 Sound Card

4.1.1 Morse Source



Select the sound card that the radio's audio output is connected to, this can be a real, USB or virtual sound card.

4.1.2 Tx Output



Select the sound card where the Tx audio is to be sent, to modulate the transmit.

4.1.3 Tx AF Buffers

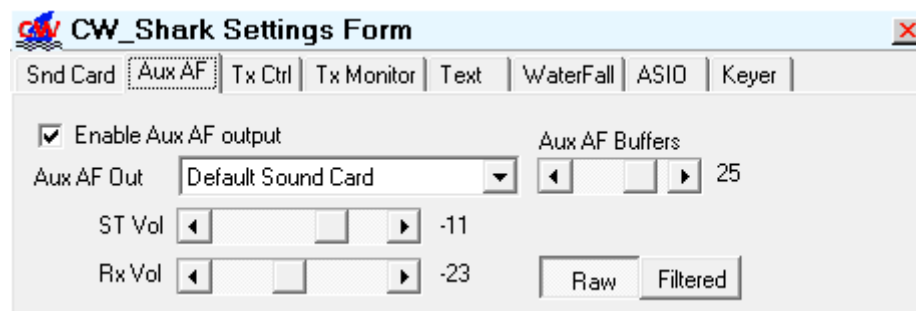
The control sets the number of buffers used to send the audio out. Too small a number and the audio will break-up/stutter, each buffer adds 10mS delay to the audio leaving the PC.

4.1.4 Refresh Drivers

Pressing this button will close all WM audio, re-populate the device lists and then re-open the selected device.

4.2 **Aux AF**

The Aux AF output is intended to send a copy of the Rx audio to PC speakers, a Tx sidetone is added to this audio, so you can hear the Tx'd CW. If the radio only has one AF output and this has been used to feed the 'Morse Source', then this output is very useful.



4.2.1 **Enable Aux AF**

Check this to enable this output.

4.2.2 **Aux AF Out**

Select the desired destination.

4.2.3 **Aux AF Buffers**

This control sets the number of buffers used to send the audio out. Too small a number and the audio will break-up/stutter, each buffer adds 10mS delay to the audio.

4.2.4 **ST Vol**

Adjusts the level of keyboard and paddle keyer sidetone fed to the Aux AF output.

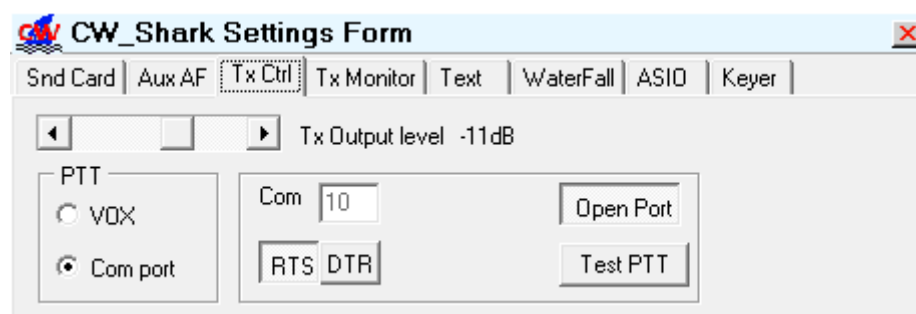
4.2.5 **Rx Vol**

Adjusts the level of the Rx audio On this output.

4.2.6 **Raw / Filtered**

Select filtered and the audio output is taken from an inverse FFT of the Decoder No1. This forms a very narrow filter ~16Hz, that follows the tuned frequency of the decoder No1.

4.3

Tx Ctrl

4.3.1

Tx Output level

This control sets the level of the audio morse code sent to the radio via the sound card. It is calibrated in dB, where 0dB is full scale output of the sound card.

This level needs to be adjusted to match the radio, the radio should have some AGC to prevent overdriving, but the control can also be used to reduce the output power of the radio when set below the AGC start level.

4.3.2

PTT & Com port settings

If VOX is selected the Com port PTT options are not shown.

If Com port is selected, then the correct Com port number must be entered into the text box.

Also select either RTS or CTS as the com port control line used for PTT.

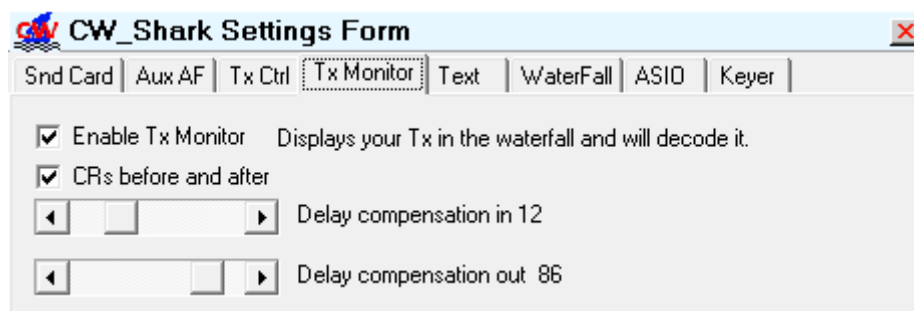
Pressing the 'Open Port' button will try to open the selected port, if the button does not stay down then the operation failed, check the port number.

If the button stays down then the port has opened correctly, and you can then press the 'Test PTT' button. This will put the radio into Tx mode, although there is no audio sent.

If the port was opened successfully next time the programme is run it will automatically open the same port for PTT. If it fails to open, a warning should be shown.

4.4 **Tx Monitor**

The Tx monitor sends a sample of the transmit audio back to the decoders whilst in Tx mode. If a decoder is tuned to the Tx frequency it will decode it and add the text to the receive text panel, in a pinkish colour.



4.4.1 **Enable Tx Monitor**

Turns the Tx monitor on or off.

4.4.2 **CRs before and after**

If checked then the decoded Tx text starts a new line in the decoded text panel.

4.4.3 **Delay compensation in**

This delay compensates for the time taken for the sound buffers to reach the radio, so that the radio is keyed just before the audio arrives.

If using Com port PTT, as recommended, then this delay value should be about 12.

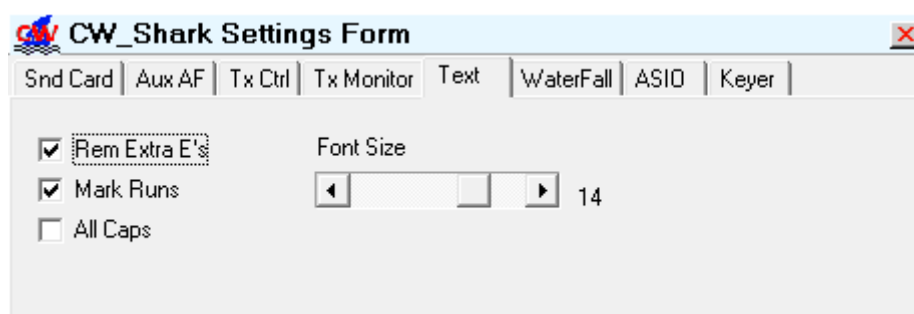
Too long and the first dots and dashes will be missed from the Tx, too short and the end of the transmission will be lost. It is important to set this delay correctly before any real Tx.

To test this setting, a 2nd receiver should be used to listen the Tx (into a dummy load) and the delay should be adjusted to ensure the 1st dot any Tx is sent correctly. The setting will be remembered next time the programme is started.

If using VOX PTT, then the value may need to compensate for any look ahead delay in the radio.

4.4.4 **Delay Compensation out**

This delay is to allow blanking of any AGC burst that can occur when the receiver is re-enabled. The detection threshold is raised for a short period after any Tx.

4.5 **Text**4.5.1 **Rem Extra E's**

Check this to removed extra 'E's decoded from the noise floor.

4.5.2 **Mark Runs**

Check this and where the programme made a 'guess' at splitting character runs, will be highlighted in red.

4.5.3 **All Caps**

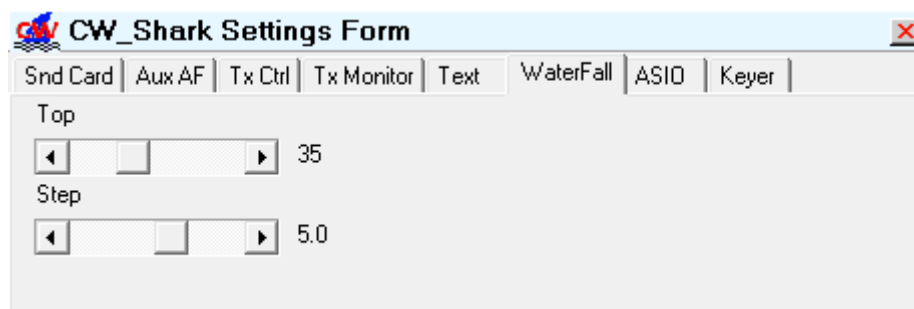
Decoded text will be all capitals.

4.5.4 **Font Size**

Adjusts the font size in the Decoder and Tx text panels.

4.6 **Waterfall**

Allows adjustment of the horizontal waterfall colours.

4.6.1 **Top**

Adjusts the level in dB that sets the top colour.

4.6.2 **Step**

Adjusts the dB thresholds for the colour thresholds.

4.7

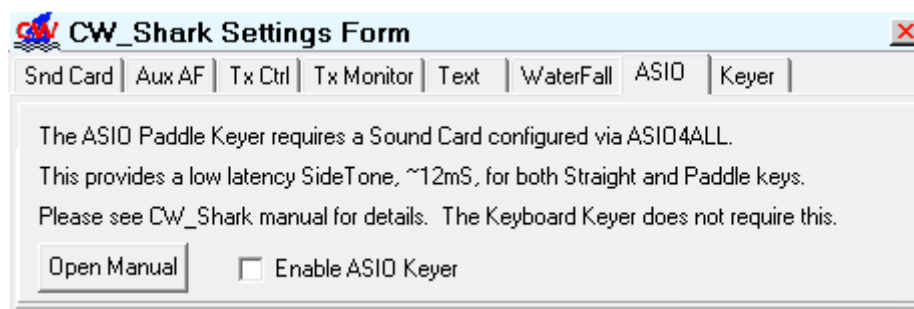
ASIO

If you do not want to use the paddle keyer of this programme then you **do not** need to enable the ASIO keyer, or install the ASIO driver.

The keyboard keyer will still work, and the keyboard sidetone can be heard in the Aux AF output.

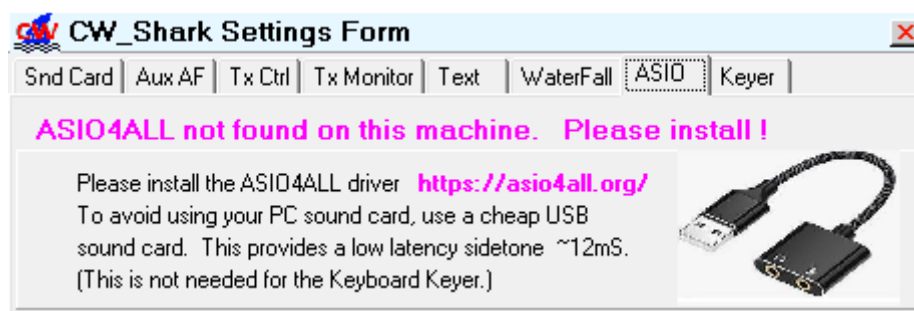
ASIO stands for Audio Stream Input/Output, and it refers to a computer audio interface protocol developed by **Steinberg**. It provides a low-latency, high-throughput pathway for audio applications to communicate directly with sound cards, bypassing the operating system's standard audio path.

Initially CW_Shark does not try to use the ASIO4ALL driver and the ASIO settings tab shows the following:-



After enabling the ASIO keyer, the next time the programme runs it will look for the ASIO4ALL driver.

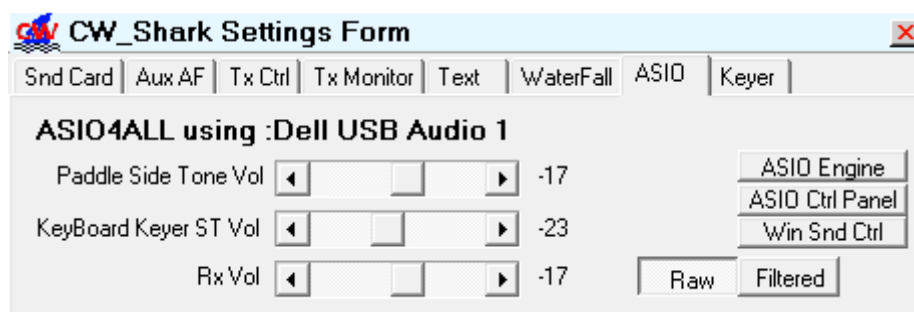
If you have not installed the universal ASIO4ALL driver then you will see this message.



Installing the ASIO driver enables the paddle keyer and the low latency sidetone monitor. Visit <https://asio4all.org/> to learn about ASIO4ALL.

Please read section 8.0 for notes on setting up and selecting the ASIO device.

With the ASIO driver installed you will see :-



The bold text shows the device currently used by the ASIO4ALL driver.

4.7.1

Paddle Side Tone

Sets the level of the Paddle side tone sent to the ASIO device.

4.7.2 Keyboard Keyer ST

Sets the level of the side from the keyboard keyer to the ASIO device.

4.7.3 Rx Vol

Sets the level of the Rx audio fed to the ASIO device.

4.7.4 Raw / Filtered

Select filtered and the audio output is taken from an inverse FFT of the decoder No1. This forms a very narrow filter ~16Hz, that follows the tuned frequency of the decoder No1.

4.7.5 ASIO Engine

Opens the CW_Shark ASIO engine form, this engine is used to automatically open the ASIO4ALL device at program start. It is visible for a brief time at programme start.

4.7.6 ASIO Ctrl panel

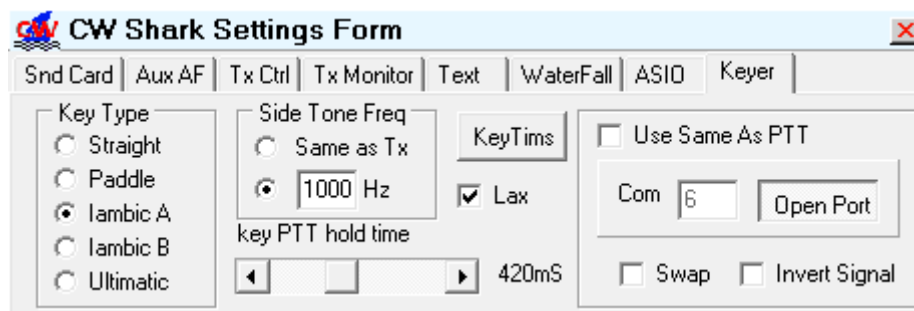
This will open the ASIO4ALL control panel.

4.7.7 Win Snd Panel

This will open the 'Windows Sound Control Panel'. This is useful to manually reset the default windows audio device if Windows has changed it.

4.8 **Keyer**

This controls the settings of the paddle keyer, only available if the ASIO4ALL driver is installed. The ASIO keyer uses a Com port CTS and DSR control lines to read the paddle state.



4.8.1 **Key Type**

Use this to set the key type required.

4.8.2 **Side Tone Freq**

The paddle sidetone is either that of the Tx audio or a fixed frequency set here.

4.8.3 **Key PTT hold time**

This sets how long the radio is kept in Tx after the dit or dah has been sent.

4.8.4 **KeyTims**

This button opens the 'Key Paddle Analysis Form', see section 4.9 for description.

4.8.5 **Lax**

Setting this checkbox relaxes the timing for the Iambic keyers. This was added to make the keyer function as close to the iambic keyer in the 'Thetis', the main difference is that the paddles are read during the intra symbol gap, not just at the end.

4.8.6 **Com Port Selection.**

If 'Use Same As PTT' is checked then the programme will use the same comport as selected for the PTT signal.

Otherwise enter the Com port number to be used and press 'Open Port'. If the button does not stay down then the port failed to open, check that nothing else is using that port.

4.8.7 **Swap**

This will swap the left right Paddles.

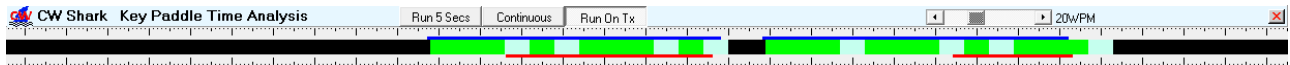
4.8.8 **Invert Signal**

This will invert the signal logic.

4.9 **Key Tims**

The 'Key Tims' form shows the timings of the left right paddles and the RF signal. The idea of this experimental form is to analyse and thus help practice in paddle timings.

The details of the form as explained here:-



4.9.1 **Form Scale**

The scale of the form is fixed and is shown above and below the signals.

Small tick = 10mS.

Medium tick every 50mS in between.

Larger Tick every 100mS

The paddles are read every 10mS, the RF is sampled every 2.5mS

4.9.2 **Dot Paddle**

Times where the dot paddle is active are shown by the lower Red line.

4.9.3 **Dash Paddle**

Times where the dash paddle is active are shown by the upper Blue line.

4.9.4 **CW Signal**

Green	:Carrier is on, (either a dot or a dash);	(Timed by PC)
Light Blue	:Intra Character Gap, no Carrier.	(Timed by PC)
Black	:Inter Character or Inter Word Gap, no Carrier.	(Timed by Operator)

4.9.5 **Speed WPM**

This control sets the paddle key WPM, it is echoed on the Tx form.

4.9.6 **Example IambicA 'CW'**

The screen grab above shows the characters 'CW' being sent at 20WPM using Iambic A squeeze mode.

It starts off by the Blue dash paddle being active first, this immediately starts a Dash to be sent.

Next because the Dot paddle is pressed before the end of the Dash time, (which includes the intra Symbol gap in light Blue), after the Dash is completed a Dot is sent.

After the end of the Dot time, the paddles are checked. Here it sees both paddles active so it alternates to sending a Dash this time.

After the end of this 2nd Dash, the paddles are checked again. As they are still both active it alternates to send a dot next.

At the end of the 2nd dot time, the paddles are checked again. This time no paddle is active so in IambicA mode that is the end. We enter a gap time shown in black, and will stay that way until a paddle is pressed again. This is the inter Symbol time, and this time is controlled by the operator.

The next paddle to activate is the Dash Paddle, this starts a Dash, at the end of the Dash time the paddles are checked, this time only the Dash paddle is active. This starts a second Dash.

When it comes to read the paddles at the end of the 2nd Dah it finds both paddles are active. This causes an alternate and a Dit is started.

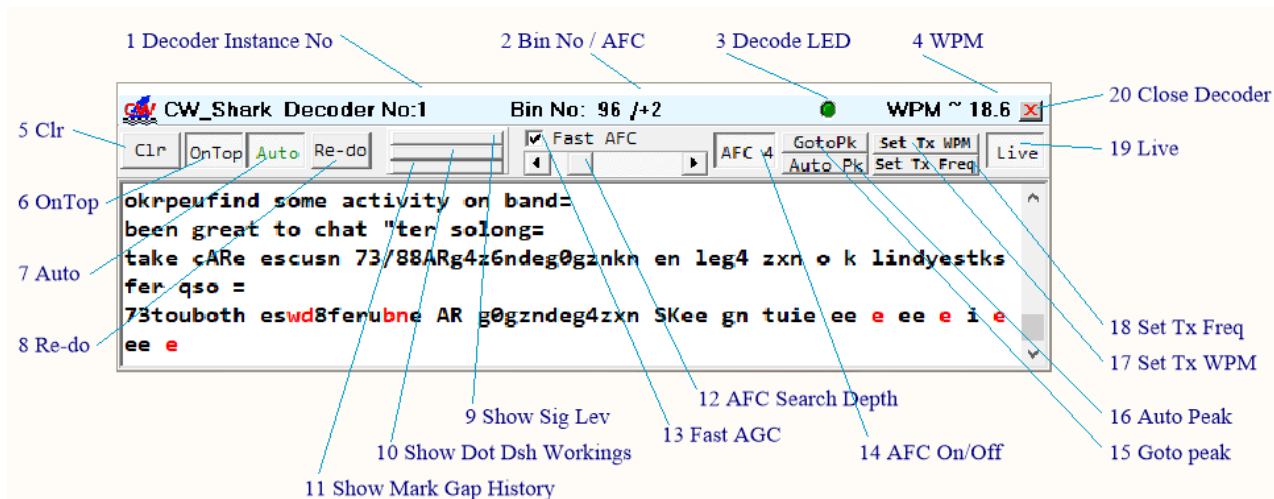
At the end of this Dit when the paddles are read it again sees both are press so it alternates to a Dash.

At the end of this Dash it finds no paddles are active, so it enters a Gap time, shown in black.

5.0 THE DECODER FORM

The programme allows up to 4 decoders to run at the same time. Each one is fully independent and has its own blue tune marker in the main waterfall form, this marker displays and controls the tuned frequency of that decoder.

The form controls are numbered here and the following section details their function.



5.1.1 Decoder Instance No

Along with the programme title, the decoder Ch or instance number is shown.

5.1.2 Bin No

This shows the current FFT bin the decoder is tuned to. The 2nd number shows the current AFC correction in FFTs bins. Note an FFT bin is 15.6Hz wide so an FFT bin number of 154 corresponds to 2402Hz. If the AFC has reached it's limit, then the AFC correction is shown in **red**, to indicate that the AFC may need to be increased.

5.1.3 Decode LED

Lit when more than e's and t's are being decoded. This signal is used to determin if Auto Peak should continue to count down.

(Not visible unless Auto Peak is active.)

5.1.4 WPM

This shows an estimate of the current decode speed in words per minute.

5.1.5 Clr

This button simply clears the decode text window.

5.1.6 Ontop

With this button pressed, the programme will try to keep this form on top of others.

5.1.7 Auto

With this button pressed the programme will try to automatically find the best timing thresholds for the current signal.

Releasing the button freezes the time thresholds at their current values, it will also enable the preset WPM the 'Dot Dash workings' panel. See 5.1.10.2

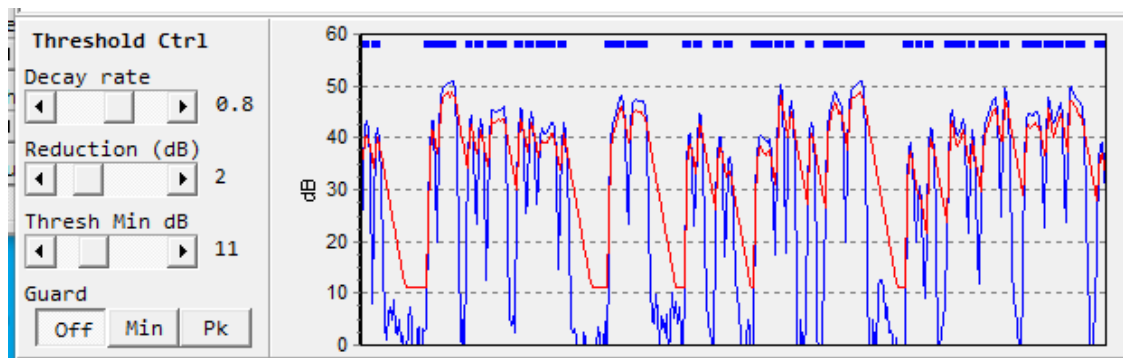
5.1.8 Re-Do

The programme holds a 60 sec history of the state of all FFT bins, pressing the Re-Do button will firstly clear the decode text then reprocess the last 60 secs history at the decoder's frequency.

This gives the opportunity to 're-parse' the last 60 seconds with or without AFC or Auto timings, this can sometimes produce a better decode.

5.1.9 Show Sig Lev

This button opens the Signal Level panel, which shows the last 6 seconds of signal amplitude, threshold value and Mark-Gap decision.



The threshold controls set the behaviour of the mark-gap, (signal – no signal), decision.

The threshold is displayed as the red line in the above panel.

Each decoder has its own set of threshold controls, which are saved in the 'ini' file.

There are three parameters that control the threshold.

5.1.9.1 Decay Rate

The decay rate sets the rate of decay of the threshold from it's peak detect. This can be seen as the slope of the red line when it is falling. The slope must be great enough to cater for rapid QSB fades of a signal. The threshold will fall no lower than the 'thresh min' setting. This must be set above the noise floor, otherwise you will see lots of random e's and t's.

5.1.9.2 Reduction

This is how far in dB, the threshold is lowered from the peak detect of the signal.

5.1.9.3 Thresh Min and Guard

To try to estimate the noise floor, the decoder uses two 'guard channels' separated above and below the set frequency. It looks for the noise floor in these channels, assuming there is no signal on those frequencies then it is a good estimate of the noise floor on the tuned frequency. You can select either the peak or the minimum of the guard channels. The threshold minimum is then set to that level plus the level from the 'Thresh Min' control.

The 'Guard' can also be turned off, in which case the threshold minimum is simply the level of the control setting. (This seems necessary if the waterfall is very busy).

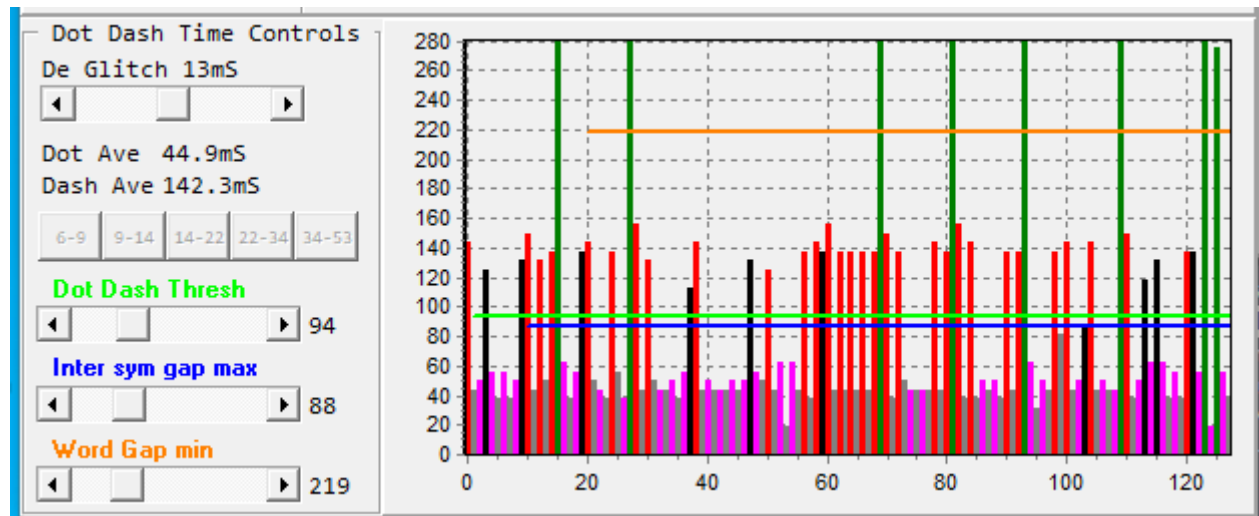
See section 7.1.5. for more details of the threshold controls.

This panel can be hidden so reduce CPU usage.

5.1.10 Show Dot Dash Workings

This button opens the 'Dot Dash workings' panel.

All timings shown in this form are in mS.



The controls of this panel are explained here:-

5.1.10.1 De Glitch

The deglitch control attempts to remove amplitude decision glitches caused by noise.

This control seems to have some use at low speeds, but at high speeds >40 wpm, where the dots and inter-sym gaps are small it will make things worse.

5.1.10.2 Pre-Set Speed Buttons.

If the 'Auto' button at the top of the form is not depressed, these buttons are enabled.

Pressing one of these buttons will then put the time thresholds to suitable values that the WPM range indicated on the button shows.

(Note at this time the 'Auto' function fails for speeds below 12 WPM, so this is the only way to decode 6 WPM.)

5.1.10.3 Dot Dash Threshold

This control will manually set the Dot-Dash time threshold. Marks shorter than this are taken as Dots, marks above as Dashes.

5.1.10.4 Inter sym gap max

This control will manually set the Inter-Sym time threshold. Gaps shorter than this are taken as being within the same morse character, gaps above start a new character.

5.1.10.5 Word Gap min

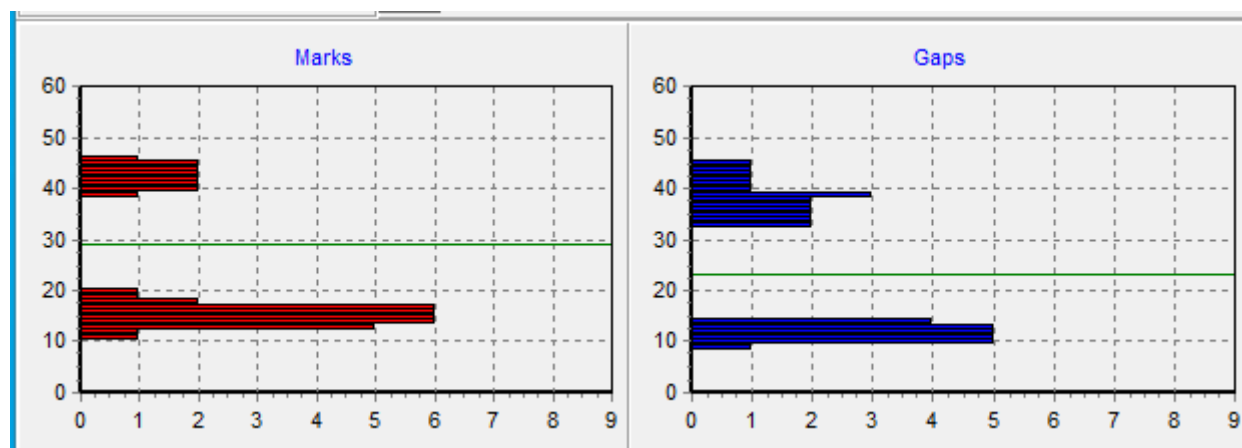
This control will manually set the word gap min time threshold. Gaps greater than this value will cause a space to be output to the decoded text, to start a new word.

See section 7.1.4 for a more detailed description of the time thresholds.

In normal operation this panel should be left hidden, to save CPU effort in drawing the graph.

5.1.11 Show mark Gap History

This button opens a mainly debug panel, that shows the Mark and Gaps history, see 7.1.3 for explanation.



In normal operation this panel should be left hidden, to save CPU effort.

5.1.12 AFC Search Depth

This control sets the $\pm$ search depth for the automatic frequency control. If AFC is active then programme will look in the FFT bins either side of the tuned frequency, if it sees a stronger signal, it will move the tuning to that bin.

5.1.13 Fast AFC

Checking this box and the programme will re-tune more quickly after the RF level falls.

5.1.14 AFC On/Off

This button turns the AFC function on or off.

Note that if the button is released with the left mouse button, AFC is turned off and the tuned frequency reverts to centre bin. If released with the right mouse button the AFC is turned off and the centre frequency is reset to include the current AFC offset.

5.1.15 Goto Peak

This will tune this decoder channel to the strongest signal in the 0 .. 4kHz audio band.

5.1.16 Auto Peak

When active the programme will automatically do a 'Goto Peak' after 30 secs of no valid decodes being sensed.

When depressed the button shows the count down to the auto peak, this count is reset to 30 secs when a valid decode is made.

5.1.17 Set Tx WPM

Pressing this button will set the Tx form's speed to match that of the current decode.

5.1.18 Set Tx Freq

Pressing this button will net the Tx frequency to match the set frequency of this decoder.

5.1.19 Live

Releasing this button will stop the decoder updating, note that the re-do button will still update the last 60 secs.

5.1.20 Close

Closes this decoder, does not run in the background.

6.0 THE KEYBOARD TX FORM

The Tx form is initially hidden.

To show the Tx form depress the **Tx** button on the water-fall form.

The controls of this form are as follows:-

6.1 **Stop**

Hitting the stop button will stop any transmission as quick as possible and the Tx text window is cleared.

6.2 **Speed Control**

This slider control sets the speed of the morse transmission in words per minute,

6.3 **Memory Buttons**

There are 10 memory buttons, to edit the memory use a mouse right click to open the edit window.

The hint for the buttons shows what is in that memory.

Any empty memory contains 'MT', the memories are held in the 'ini' file.

6.4 **Tune Button**

Depress this button to send a continuous tone to the radio, use this to check Tx audio level, power levels and Tx frequency.

6.5 **Blue text edit.**

Enter text in here you might want to repeatedly send. Text will be transferred to the Tx text window on pressing enter, this text is not cleared automatically.

6.6 **White text edit.**

Text entered here will be sent on pressing return, this text is then cleared, ready for the next sentence.

6.7 **Tx Text Window**

Type here and the characters will start to be sent immediately.

As the text is sent it turns grey, the current sending character is **red**.

You can use back space (del), to remove characters that have not yet been sent.

To clear the window, press the **Stop** button.

6.8 **Tx character set**

As well as the 26 letters and 10 numerals, the following symbols are encoded:-

=	- . . . -	-	- -
?	. . - - . .	/	- . . - .
,	- - . . - -	ö	- - - .
.	. - . - . -	ä	. - . -
+	. - . - .	ü	. . -
@		(	- . - - .
\$	. . . - . -	)	- . - - . -
”	. - . . - .	!	- . - . . -

Other symbols are sent as spaces.

7.0 HOW IT WORKS

7.1 Receive

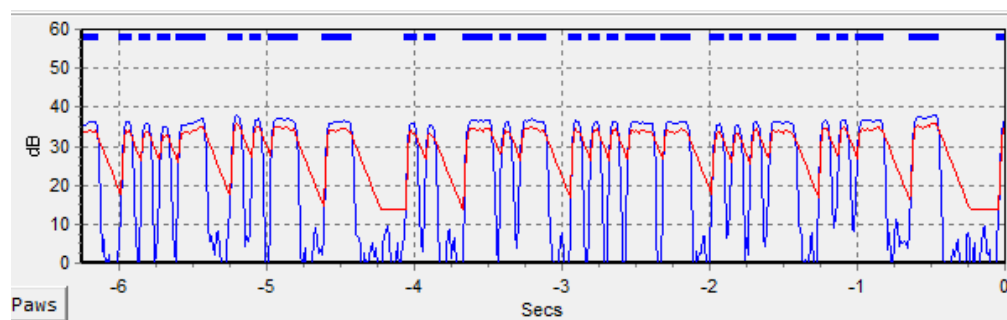
7.1.1 Audio processing.

The programme takes its input from the sound card of the PC. Samples are taken at a fixed 8K per second. These samples are then fed to an overlapping 512 point FFT. The FFT output consists of 256 'bins' each 15.6Hz wide and the data rate out of the FFT is 160 amplitude samples per second for each bin.

It is this data that is displayed in the horizontal FFT and any selected bin can be passed to one of the four decoders. A record of the last 60 seconds worth of the FFT output is held in a recirculating buffer, that can be used to decode or 'reparse' the last 60 seconds worth of any bin.

7.1.2 The Mark Gap Amplitude Decision.

The decoder determines if the amplitude is above or below a threshold. The threshold (red), signal level (blue) and decision (blue) can be seen in the 'sig lev' panel. Note that the vertical amplitude scale has accurate dB calibration, the signal shown below shows a S/N ratio of ~28dB in the 15.6Hz FFT bin bandwidth.

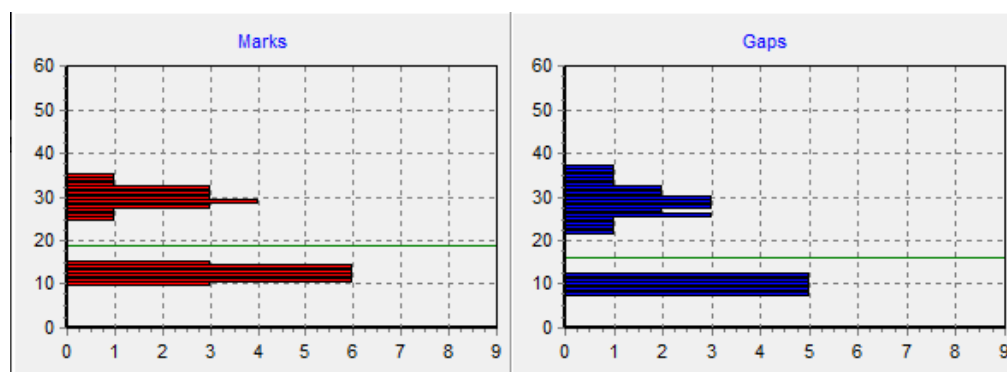


7.1.3 The Timing Thresholds.

The decoder further measures and logs the time in either the mark state (signal above threshold) or gap state (signal below threshold). If it can determine that there are two distinct groupings of the mark timings, it then places the dot/dash threshold midway between.

In a similar manner it logs the timings of the gaps, if it can see two distinct groups it puts the inter-symbol threshold midway between the groups. IE if the gap was less than this threshold, we are still decoding the same character, if greater we have started a new one.

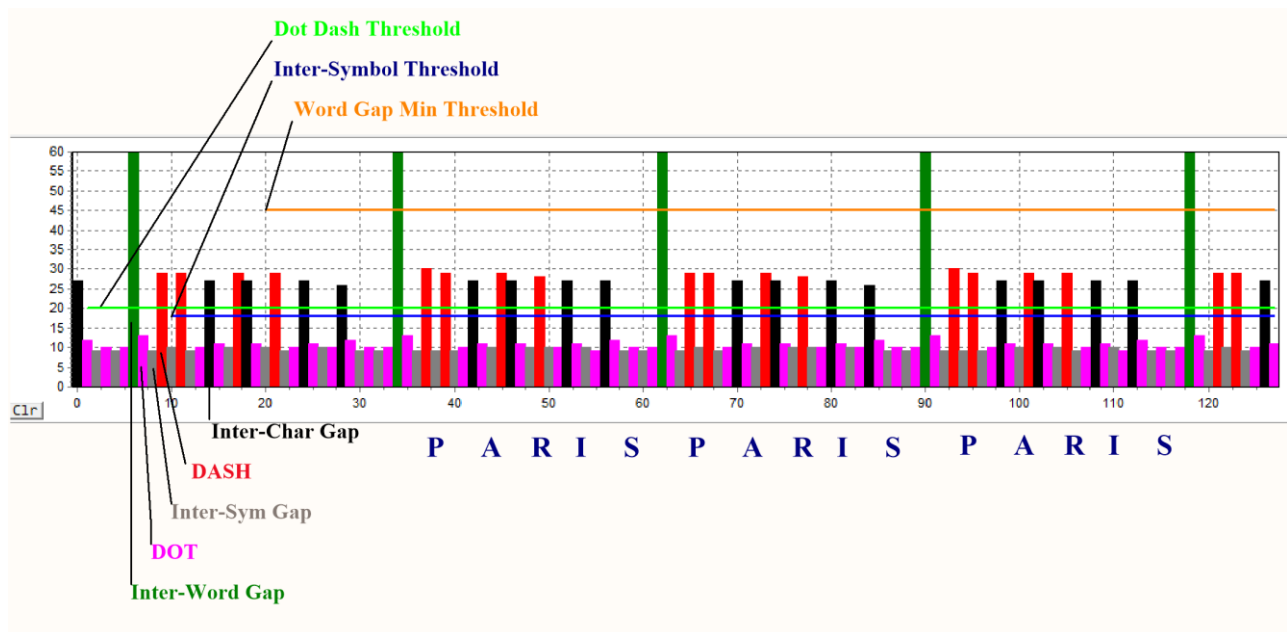
These workings can be seen in the 'mark gap history panel'.



There is a third time threshold, the word gap minimum. If a gap exceeds that time, then a new word is assumed and a space is added to the decoded text. This threshold is automatically set to 2.5 times the inter-symbol threshold.

7.1.4 The Dot Dash decision.

The decoder then uses these thresholds to determine if we have just had a Dash, a Dot, an inter-symbol gap, an inter-char gap or an inter-word gap. These decisions and their thresholds are shown in the 'dot dash workings' panel.



After every Inter-Char gap the programme looks to see if it can find a match for the preceding dot dash pattern. If it does it outputs the corresponding character to the decode text, if it fails to find a match it outputs a '*'.

After every Inter-word gap, a space is added to the decoder text.

If you examine the above diagram, you can see how the test word PARIS is decoded.

7.1.5 About the Amplitude threshold.

Critical to the decoding process is the Mark Gap amplitude threshold. The programme uses a peak detect to initially put the threshold at set dB below the peak. It then allows the threshold to decay at a fixed rate, unless it is again pushed up again by the peak detect. Most importantly the threshold is not allowed to fall down to the noise floor, otherwise random e's and t's are mostly detected. Falling into the noise can also upset any timing thresholds.

To determine the noise floor two 'guard' bins are used. Assuming that there is no signal in these bins the 'Thresh Min' is set above this level.

Optionally the programme can use either the peak or min of these guard bins, or it can just use a fixed threshold.

The controls for these settings are found 'Sig Lev' panel in the decoders. Each decoder uses its own set of values.

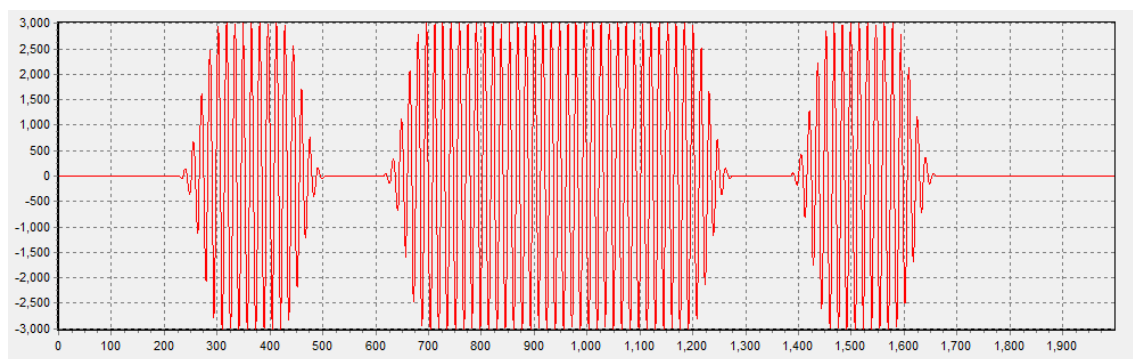
7.2 **Transmit**

7.2.1 **Text to audio stream.**

The Tx starts by reading any unsent characters in the Tx window. The character is converted to marks and gaps by using a look up table. This on/off stream is in turn converted to an on/off audio tone by an NCO, controlled by the Tx AF frequency, TX level and Tx WPM settings. The sample rate of this Tx stream is same as the receiver audio, at 8k sample per second.

Importantly before this audio stream is passed to the Tx sound card buffers, it is passed through a narrow bandpass filter tuned to the audio frequency of the Tx. This effectively removes key-clicks on the transmitted signal.

The shaping of the Tx signal at 50 wpm is shown here as it sends an 'r'.



If there is no unsent text in the Tx window the sound card is fed with 0's, the stream to the sound card never stops, even during receive.

7.2.2 **Paddle Keyer**

The paddle keyer uses the timings from the ASIO4ALL driver, every 10mS the state of RTS and DSR of the selected Com port are read. Depending on the state of the pins, the operational mode (straight, or paddle variants), and the keyer WPM, tones are generated at the Tx AF frequency and sent the same route as the text to audio stream.

The tones passed to the low latency ASIO4ALL sidetone are **not** passed through the AF filter and can also be set to an independent fixed frequency.

7.2.3 **Tx Monitor**

Samples of the Tx audio buffers are also passed to the receiver, if these buffers are filled with Tx sound, even the spaces between words, (but not the 0's of no data), they are marked for Tx.

If the receiver sees this mark it will feed these buffers to the receiver instead of the those of incoming AF from the sound card. This 'sidetone' can then be read by the decoder in the normal manner. Such decoded text is marked in a pinkish colour in the decoder window.

7.2.4 **PTT**

Preferably the radio PTT is controlled by either the RTS or CTS control lines of a comport. This may be a real RS232 Com port or a virtual comport. The programme uses the same mark in the buffers sent to the receiver to switch the state of the control line for the period of transmission.

It is also possible to use a radio's VOX to control the T/R state. If VOX is used then the 'Delay compensation in' may need to be extended, for the Tx monitor to work correctly.

8.0 THE ASIO PADDLE KEYER

8.1 Overview

This section describes the optional ASIO paddle keyer.

In order to provide a usable sidetone for the Paddle keyer, the programme uses the ASIO4ALL driver. Using the standard Windows API, which is used for the Tx and Aux AF outputs results in a high a latency. A delay of 200mS in the audio reaching the transmitter is not ideal, but does not really matter unless QSK is required. A 200mS delay in the sidetone when sending CW by hand is however a disaster, the delayed audio makes sending impossible, with a straight key or paddle.

Using the ASIO4All driver the latency has been reduced to about 12mS, ie the time from the key switch to the audio. This certainly allows sending up 25 WPM, apparently most people cannot detect a latency of up to around 10-15mS, so it may work for higher speeds.

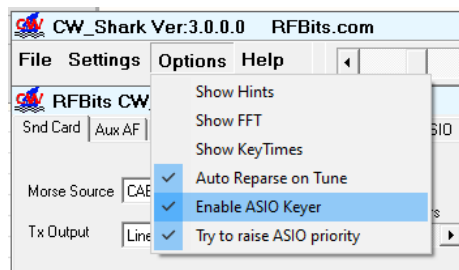
The ASIO4ALL driver works with almost any Windows soundcard, it does not replace your existing sound card drivers or 'mess with them in any way'. See <https://asio4all.org/> to read about and download the ASIO4ALL driver. The programme was developed with ASIO4ALL V2.16, but it should work with later versions.

The programme opens the ASIO4ALL driver with a sample rate of 8K, this seems to be allowed on all the sound-cards tested.

8.2 Enabling the ASIO keyer

It is not recommended to install the ASIO4ALL driver until familiar with the operation of the rest of the programme and the sound card interfaces for the 'Morse Source', 'Transmit' and 'Aux AF' have been sorted.

Initially the programme does not even look for the driver, this must be enabled either in the Setting/ASIO tab or via the Options menu.



The option can be turned off any time later via the pop-up menu.

With this option selected the next time the programme is run, it will look for the ASIO4ALL driver.

Note the change will only occur when the programme is restarted.

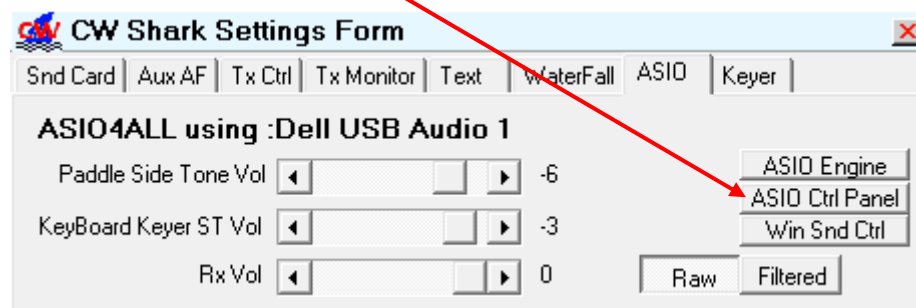
8.3

Installing and Configuring the ASIO4ALL driver to the correct soundcard.

Once the ASIO4ALL driver is installed, when the programme starts it attempts to open the ASIO4ALL device.

The first time it runs it will probably pick your default soundcard to try to open.

To see what it has opened and what is available press the 'ASIO Ctrl Panel' button in the ASIO tab in the setting forms.



This will open the Steinberg ASIO4ALL control panel.



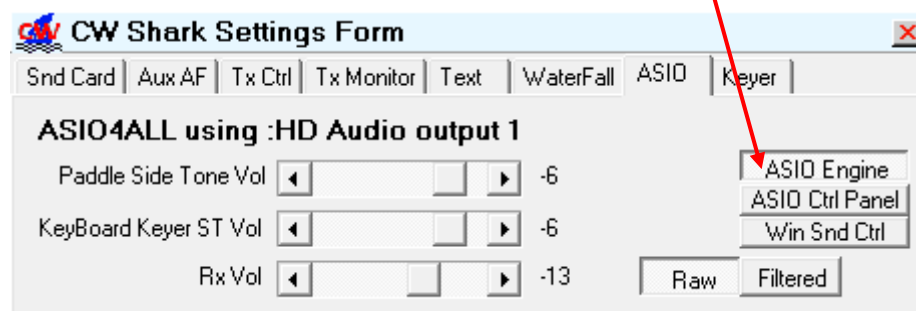
Here it shows the list of soundcards available and it shows that it has connected to the Realtek Audio device. This is the sound card on the mother board and not the one desired.

To change the device, click just to the left of the device name.



This will move the  marker to the wanted device, although the Realtek device is still being used at this time.

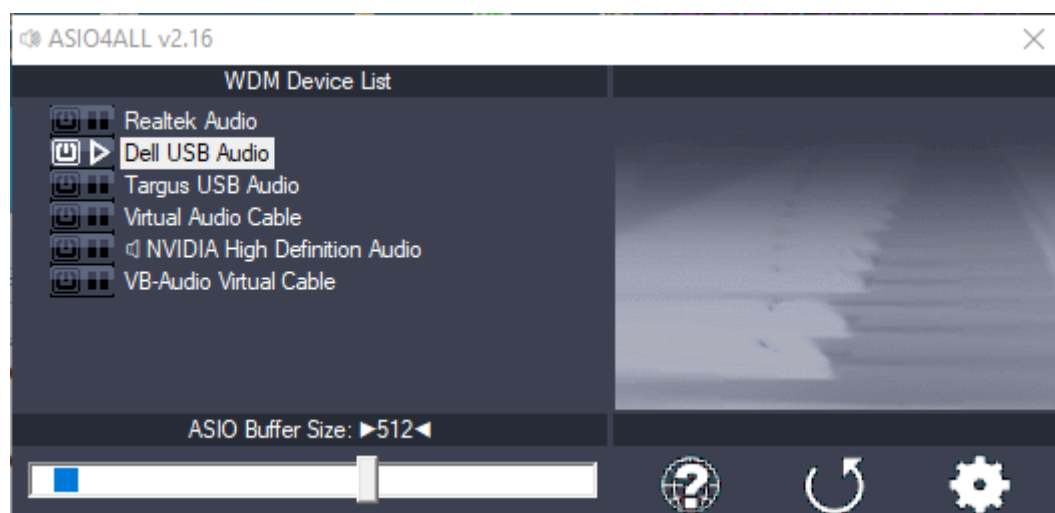
Next press the 'ASIO Engine' button on the settings form.



This will open the 'CW_Shark ASIO Open Engine' form. On this form press the 'Re-Start' button.



The ASIO4ALL driver will then close the current device and open the new selection. This is shown in the ASIO4ALL control panel.



Amazingly the ASIO4ALL driver seems to remember this choice and the next time CW_Shark is run, this process is not necessary.

The ASIO4ALL control panel has some more advanced features, but these are not necessary, as CW_Shark does the rest of the configuration of the buffer sizes etc.

These panels can now be closed.

9.0 ABOUT SOUNDCARDS

9.1 About External SoundCards

A standard PC has a single soundcard, although it is possible to configure the programme to run with a single soundcard it is recommend to install a 2nd soundcard dedicated to this programme. This ensures sounds from other programmes do not interfere with reception, or get transmitted.

Some modern radios include a USB soundcard, if so, this should obviously be used for the 'Morse Source' and 'Tx Output'.

External USB soundcards come in many flavours; it does not need to be an expensive item. This item costs £7 from Amazon and works well as the audio interface to the radio. It also works with the ASIO4ALL driver to provide a low latency sidetone for the paddle keyer.



AB13X USB Audio soundcard

If using an older USB soundcard, such as Creative Sound Blasters, ensure that you have the latest creative drivers, as they do not behave with the stock Windows 10/11 drivers, <https://support.creative.com/downloads>.

It is also worth checking that there is not an unused soundcard already in the system, in a monitor or a docking station.

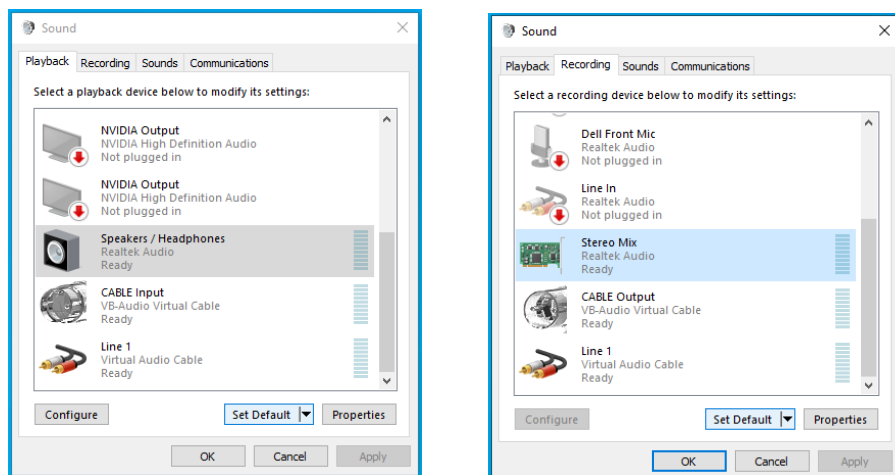
Some soundcards do not show themselves unless there is a cable plugged in.

9.2 About Bluetooth Sound devices

Connected Bluetooth speakers or headphones, should show up in the drop-down list for the audio output devices and will also show in the list of ASIO4ALL devices.

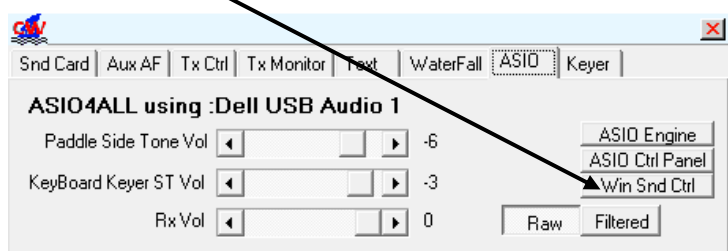
They are suitable to use for the Aux AF output, as a little more delay won't really matter for just monitoring. But they are not recommended for the ASIO output as the added Bluetooth delay makes the output useless as a working sidetone.

Note that Windows has a very annoying habit of assigning a new USB sound card as the default device. To correct this, open 'Windows Sound Control Panel' and reset the default to the original soundcard.



What is even more annoying is that Windows will often put the new device as the default, next boot. Even when it has been corrected. It can even change the default device when a cable is plugged in.

To open the 'Window sound control panel' you can enter 'mmsys.cpl' into a cmd window, or press this button.



9.3

About virtual audio cables

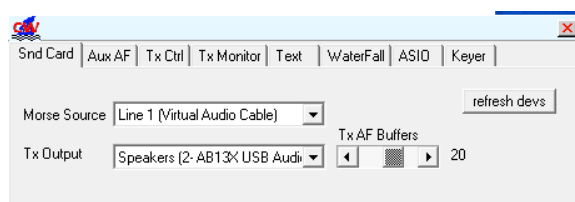
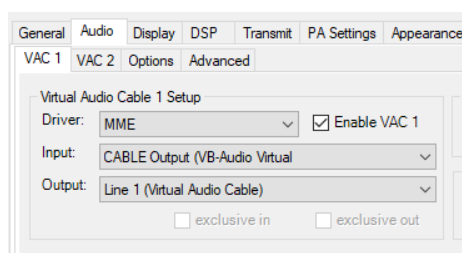
If using a PC based DSP radio such as Thetis, the audio connection to the software should be made through a virtual audio cable (VAC).

There are several available, examples are:-

<https://vac.muzychenko.net/en/>

<https://vb-audio.com/Cable/>

Once installed, select the VACs in the radio and CW_Shark options, this is the setup for Thetis :-

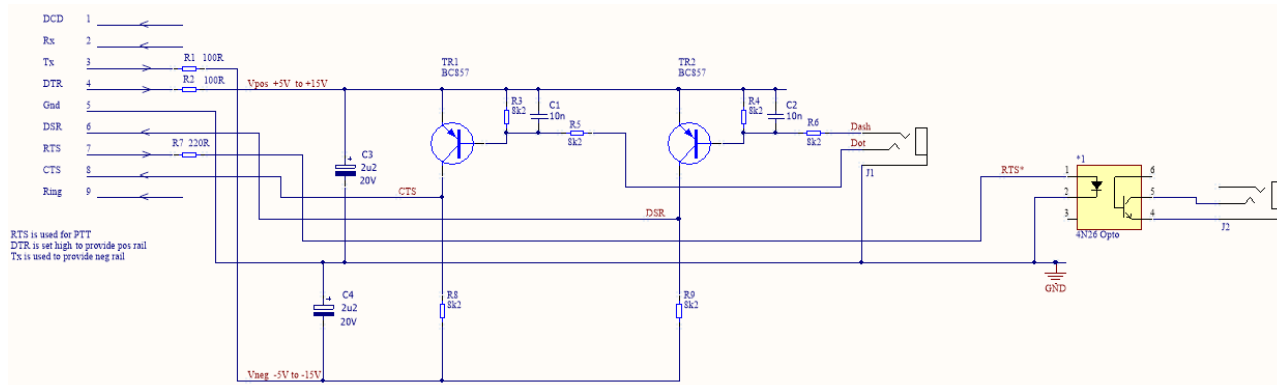


10.0 ABOUT SERIAL PORTS (COM PORTS)

The programme uses serial ports to control the PTT output to the transmitter and for the dot and dash inputs for the paddle keyer. The PTT and keyer can be uses the same port or different ones.

10.1 Motherboard Serial ports (PTT or paddle keyer)

These ports use $\pm 5V$ to $\pm 15V$ for signalling, they need a voltage translation circuit to interface with a key to ground. The RTS line provides just enough drive to switch an opto isolator which can key most modern radios. This is an example circuit :-



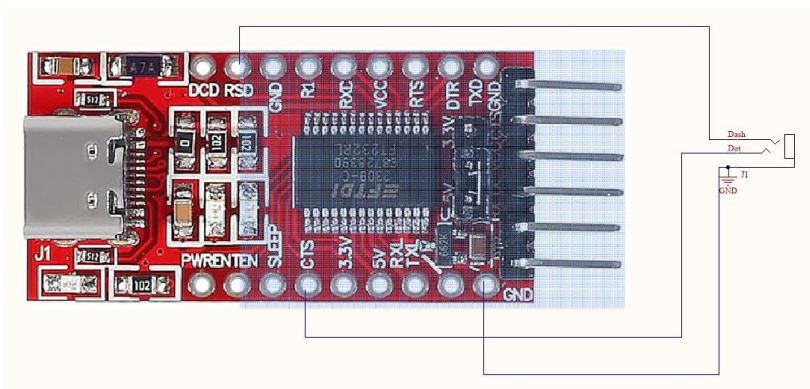
10.2 USB Serial ports (PTT or paddle keyer)

Some USB serial ports include charge pump inverters and provide the full RS232 voltage levels, these are generally sealed cables such as :-



These need to be treated as ports on the motherboard, and require the same voltage translation interface, see above.

Others provide 3V3 or 5V voltage logic levels, these are usually bare PCBs such as :-



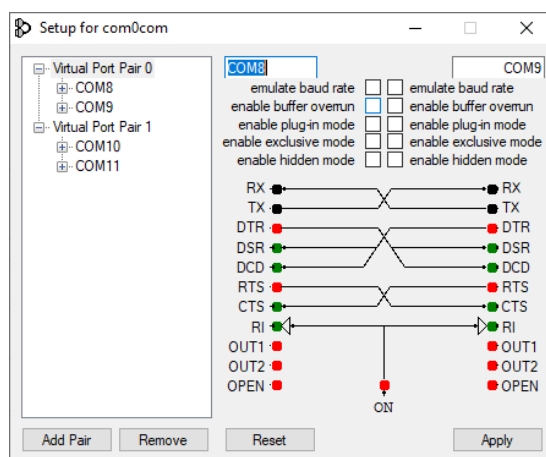
These are generally cheaper and easier to interface. As there is an active pull up on the CTS and RTS lines, the paddle key can be directly connected to CTS and DSR (sometimes marked RSD), see above.

For either type try to use one that uses the FTDI chip, these seem more consistent.

10.3 **Virtual Serial ports for PTT**

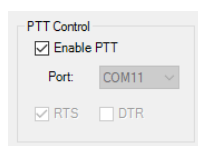
To control the PTT of a Software radio such as 'Thetis', the best option is to use a virtual com port pair such as 'com0com'.

https://src.koda.cnrs.fr/laplace-service-commun-electronique/membres/arnauld-biganzoli/emulateur-de-ports-serie-virtuels/-/tree/main/com0com/com0com-2.2.2.0-x64-fre-signed?ref_type=heads

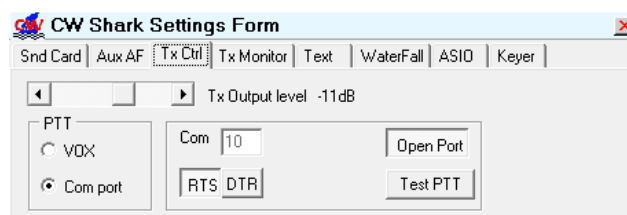


Once installed, configure at least one pair of ports.

In the radio software select one of the ports as the PTT input, either CTS or DSR, they are fed by RTS or DTR of the other port.



In CW_Shark select the other port of the pair and select RTS or DTR to drive the PTT of the software Radio.



11.0 TIPS ON RUNNING CW_SHARK

11.1 Receiver Bandwidth - Audio level

The CW_Shark will use an audio bandwidth of up to 4kHz, ideally the receiver should have a 4kHz bandwidth to match. Set the receiver to SSB mode either USB or LSB and select the widest filter up to 4kHz available.

If the receiver band width is too low, < 1kHz, then the CW_Shark AGC will not function correctly and the audio level fed to the programme may need to be reduced.

The transmit audio bandwidth should be set to match.

11.2 Receiver AGC

If the receiver has an AGC gain control, this should be set so that the strongest signal in the passband, just starts to lower the receiver gain. This will ensure that other stronger signals will not affect the level of the wanted signal.

11.3 CPU load

Most of the CPU load of the programme is consumed by the graphics.

CPU load can be reduced by :-

- Lowering the Waterfall scroll rate.
- Only showing the text box in the decoders.

11.4 Real-Time Priority

If this option is selected under 'More ...' settings, the programme will try to raise the thread priority for the ASIO keyer to 'real-time'. This can help in drop outs on the Paddle sidetone. This operation will fail, unless the programme is run in admin mode, no error message is reported.

11.5 Windows 11 Admin mode

Under Windows 11, it is recommended to run CW_Shark in administrator mode.

Although the programme does not make use of any privileged instructions, (except for raising priority if selected), when running in admin mode the programme runs faster and uses less CPU.

This effect is not seen in Windows 10.

11.6 Some notes on ASIO Sidetone latency

Not all soundcards perform the same, here are some test results made with several different soundcards, all using the ASIO4ALL driver.

CW_Shark ASIO sidetone tests mS			
		min	max
RealTek mother board		12	22
Nvidea High definition USB monitor		12	22
C-Media CM108 USB		20	30
Dell USB dock		30	40
AB31X USB		20	30
Targus USB dock		25	35
TI PCM2902 USB (PnP)		22	32
Creative Blaster SB0490 USB		52	62
Creative Blaster SB270 USB		30	40

Note that the Creative soundblasters are both very old obsolete devices.

Tests were also made comparing motherboard serial ports against USB serial ports, here there was no noticeable difference.