

# CWP Help

Klaus von der Heide, DJ5HG

## 1. Introduction

CWP is a computer program that enables communication by classic Morse telegraphy at very low signal to noise ratios. The program itself offers detailed help to all buttons and fields of the graphic display. The goal of this draft is to add information that is more general and that, therefore, could not be integrated in the program. For the basic principles of CWP see [1].

## 2. Synchronization

The receiver in a digital communication system must know the clock of the incoming sequence of symbols, and it must know where the sequence starts and ends. Usually this information has to be obtained from the received signal. The well known weak-signal mode JT65 by Joe Taylor [2] transmits a pseudo-random pattern uniquely for synchronization. That is a simple and very reliable way to achieve both, packet synchronization and symbol synchronization. But it costs half the transmitted energy. Since CWP is based on the classic Morse telegraphy, it must reconstruct the synchronizations without such features.

Loosing the packet synchronization in a weak-signal communication is catastrophic. Therefore, all modes used in ham radio are based on a fixed packet timing aligned at the minutes of UTC. Of course, there remains an uncertainty on the exact position due to skew of the computer clocks at both stations and partly due to the delay caused by the radio path.

Unfortunately, computer clocks are not very precise. Even worse, precise synchronization of external devices such as a sound card with the computer clock within the environment of an operating system that is not made for real time applications cannot be achieved in a manner that runs on nearly all computers. That is the reason, why CWP is based on a quite different concept. The advantage is that many subsequent CWP pathes can be accumulated without sufficient signal for an individual synchronization. The drawback is the necessity of a precise measurement of the sampling rates once for each sound card. CWP does work without these measurements. But in that case the computer clock should be set every hour or so and then the CWP synchronize button must be pushed to resynchronize with the computer clock.

CWP uses the input sampling rate of the sound card as its master clock. The nominal sampling frequency is 8000 samples/second. The actual sampling frequency differs somewhat. It should be measured in advance, and the frequency has to be set in the setup menu of CWP. The computer clock should be correct within about one second at CWP program start when the alignment of the packet with UTC is defined. The running program then entirely ignores the computer clock. It only counts about 480000 input samples per minute with the exact value depending on your settings of the input sampling rate.

Unfortunately, many sound cards do not use the same sampling rate for input and output. Therefore, we also have to measure the output sampling rate, and we must set this value in the CWP-setup.

### 3. The CWP-Program

#### 3.1. The User-Interface

The user interface of CWP is shown below in figure 7.

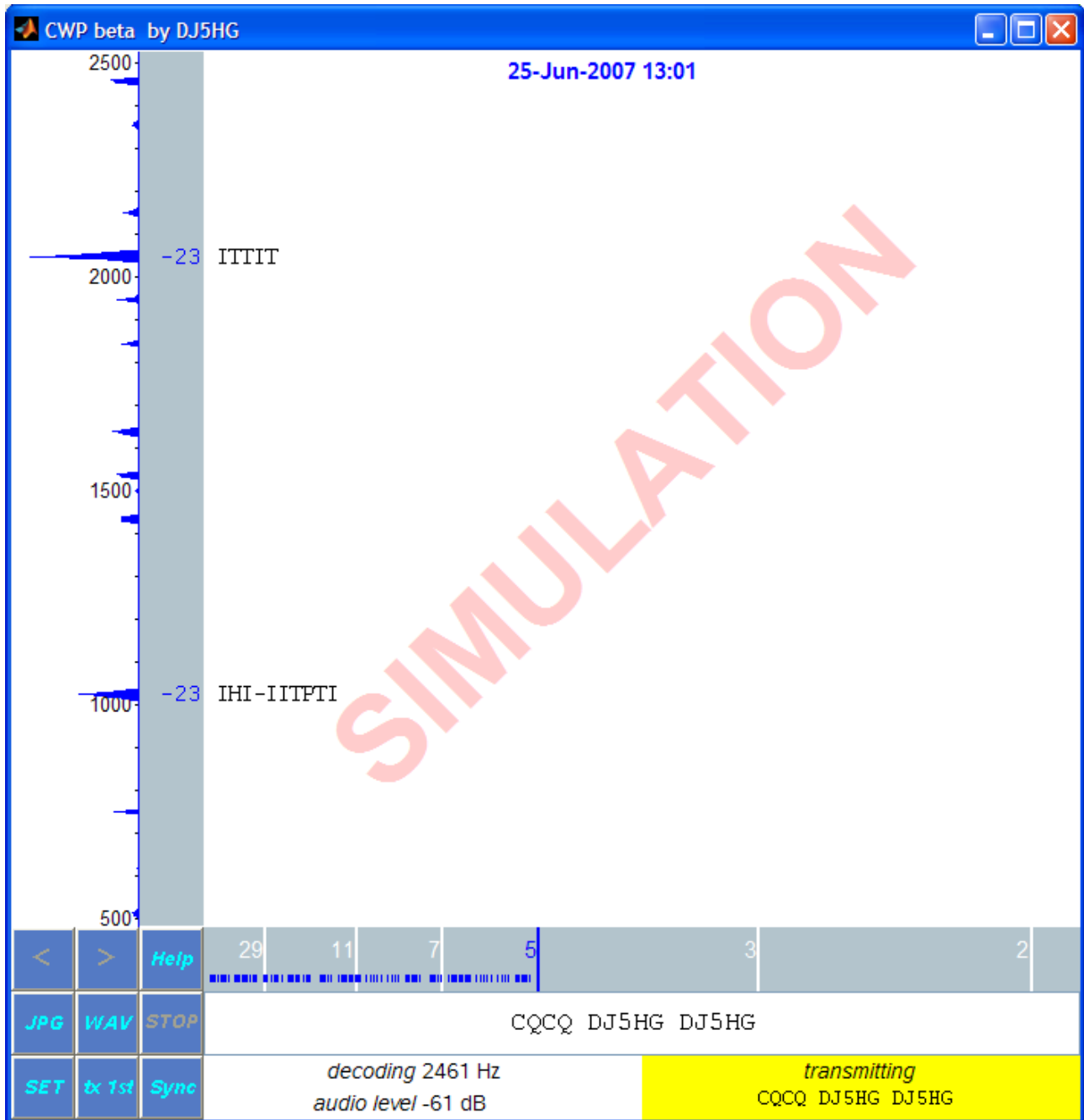


Figure 3.1. The user interface of CWP is segmented into (1) the receiver output with the spectrum to the left, the decoded text to the right (with all garbage here), and the snr between, (2) a control field with 9 buttons, (3) a transmission segment with the text input edit field (with the CQ DJ5HG in this case) and the corresponding Morse code, and a status field indicating what is going on. No transceiver was connected here, so the audio level is very low.

After program start you will get a window like that shown in figure 3.1. You will get help on all details by shifting the mouse pointer over the detail in question and then typing the letter „h“ or „H“ on the keyboard (the window must of course be in focus).

### 3.2. The Setup

Pushing the SET button in the user interface window will open the CWP Setup shown in figure 3.2. Some basic parameters can be specified with this setup window.

Most of the interface fields of the setup window are text edit fields. Set the mouse pointer into the field and click the left mouse button to set the cursor into the field. Then use the keyboard to type your specification and enter it with the ENTER key.

The fields named "Simulation" and "NoiseBlanker" are simple toggle Buttons, and those named "COM Port" and "PTT" are selection fields.

The screenshot shows the 'CWP Setup' window with a blue title bar and standard Windows window controls. The interface is organized into four main sections:

- Operator:** Includes text fields for 'My Call' (DJ5HG), 'My Locator' (JO53IM), and 'My CQ' (CQCQ DJ5HG DJ5HG).
- Rig Parameters:** Includes 'UTC - Clock' (-1), 'Tx Audio Frequency' (1234), 'NoiseBlanker' (OFF), 'LowerBandLimit' (480), 'UpperBandLimit' (2530), 'COM Port' (none), and 'PTT' (RTS).
- Radio Channel:** Includes 'Mode' (EME) and 'Coherence [sec]' (0.8).
- Simulation:** Includes 'Simulation' (OFF), 'SNR [dB]' (-20), 'Coherence [sec]' (0.2), 'Freq. Drift [Hz/min]' (0), 'Input Sample Rate' (7999.844), 'Output Sample Rate' (7999.356), 'Accumulation' (off), and 'Computer' (medium).

Fields are represented by text boxes, toggle buttons, or selection boxes with dropdown arrows.

**Figure 3.2. The setup interface. Click into the My Call field etc. and then type in your call, locator, and preferred CQ.**

## Operator

The three fields for the **operator** are not used directly by the program. These texts are linked to the characters **<**, **>**, and **|**. If you type these characters as shortcuts into the edit send text field of the CWP main window, then they will be replaced by the fulltext of MyCall, etc.. when you type the ENTER key.

**MyCall:** Type here the full callsign including special prefixes etc., **YL/DJ5HG/P** for example.

**MyLocator:** Type here the actual Maidenhead-locator in the format you want to send. Examples are: **JO53** or **JO53IM**.

**MyCQ:** Before typing here your favorite CQ call type it in the edit send text field of the CWP main window and enter it with the ENTER key. The CQ text should be repeated as often as possible. For example, type **CQ DJ5HG**. This will be repeated 7 times, but it leaves a significant time unused. Therefore **CQ CQ DJ5HG** is better, but the callsign is sent 7 times as before. **CQCQ DJ5HG DJ5HG** is repeated 5 times, but it sends the callsign 10 times. If **CQ CQ DJ5HG DJ5HG** is used then the repetition results in **CQ CQ DJ5HG DJ5HGCQ CQ ...** which may cause some irritation.

Hit the keys "R" or "r" when the setup window is in focus to get help for the rig parameters, or hit the keys "S" or "s" when the setup window is in focus to get help for the simulation parameters,

## Rig Parameters

**UTC - Clock:** Set here the difference between your computer clock and UTC in hours. East of Greenwich are negative differences, west are positive. This value is used to display the time in UTC. The file names of stored screenshots and wave files use UTC also.

**Tx Audio Frequency:** This is the audio frequency of the generated transmit signal. Be aware of the fact that you transmit in USB or LSB mode. So the actual HF-frequency is the carrier frequency of your transceiver plus this audio frequency (USB) or the carrier frequency minus this audio frequency (LSB). A good choice is 1500 Hz since that is in the center of the receiver bandwidth. If you set the  $\Delta$ Tx of your transceiver to -1500 Hz (in USB mode) the transceiver will display the correct CW-carrier frequency without shift.

**Noise Blanker:** Switch ON/OFF a simple noise blanker.

**Lower and Upper Band Limits:** The CWP-decoder only decodes signals between these limits.

**COM Port and PTT:** You can control the PTT of your transceiver via the good old serial interface. Scroll through the options using the little up and down buttons and select by a click into the field. The PTT control only is active if simulation is „off“. You can test the settings in the mode „txonly“ and then trying the different settings. If all is ok the transmitter will be „on“ the first 56 seconds of every minute and „off“ for the last 4 seconds.

**Important:** The current version of CWP needs a restart of CWP after a change of the COM port.

**Sample Rates:** Type in here the sample rates of your soundcard as precise as possible. Be aware of the fact that the CWP program uses the input sample rate as its master clock. See the chapters 2 and 3 for further information.

## Radio Channel

**Mode:** This is a toggle button for switching between „terrestrial“ and „EME“. The mode only specifies the channel delay to the values 0 or 2.5 seconds, respectively.

**Coherence:** This is an important parameter which significantly influences the sensitivity of the receiver. This parameter should match the actual coherence time as far as possible. If there is very slow fading or even none at all then the coherence time should be set to „inf“. In tropo scatter on 2 m 0.2 s is a good choice. The coherence time roughly is half the smallest observed fading period. If the value is lower than that of the actual channel then the sensitivity is some dB lower than it could be. On the other hand, if the value is significantly larger than that of the actual channel then the probability of a correct decode may be very low. To get skilled with this parameter you should use the simulation option with  $\text{snr} = 0$  and different coherence times and listen to the generated signal with earphones.

## Simulation

The CWP simulation should not be transmitted by your transceiver. It is for a simulated communication of two computers crosswise linked by two audiocables (audioout of computer A to audioin of computer B and viceversa).

**Simulation:** This is a toggle button for ON/OFF switching.

**SNR [dB]:** Klick into the field and edit the SNR value. The generated Additive Gaussian Noise has a bandwidth of 2500 Hz. The total signal power within the path is adapted such that the specified SNR exactly is reached even in the case of fading.

**Coherence [s]:** Coherence time of simulated Rayleigh fading. 100 divided by the frequency in MHz gives an approximation to the real situation in EME (so 0.7 for 144 MHz, 0.2 for 432 MHz, 0.08 for 1296 MHz ...). The fading simulation takes considerable time on the computer. If the status field of CWP indicates "output too late" then the output signal had to be cut at the first second in order to remain synchronized to the computer clock. The fading simulation is switched off by the value „inf“.

**Freq.Drift:** This specifies a frequency drift of the carrier frequency. The CWP receiver can cope with the usual EME frequency drifts of  $\pm 6$  Hz per minute on 144 MHz. The simulator can simulate larger values also.

## 3.3. The Hardware Interface to the Transceiver

The interface between your computer and your transceiver is the same as for WSJT or similar programs. The audio output of the receiver must be connected to the audio input of the soundcard, and the audio output of the soundcard modulates your transmitter. The transceiver mode is SSB. Both, USB and LSB are possible. CW cannot be used.

Please be aware of the fact that the carrier frequency of your transmission is the sum (USB) or the difference (LSB) of the frequency shown on your transceiver and the tx audio frequency set as a Rig Parameter in the setup window. With a  $\Delta\text{Tx}$  at your transceiver set to -1500 Hz (USB) and the tx audio frequency set to +1500 Hz you will get a real CW feeling with the carrier sent on the

frequency shown on your transceiver, and the receiver listening in a band  $\pm 1000$  Hz around.

If you have an RS232 serial interface you can use the lines RTS or DTR normal or inverted (indicated by /) to switch your PTT. You have to set the COM port and the PTT in the setup window appropriately. If Simulation is ON the serial interface is switched off. See chapter 3.2. for more information.

### **3.4. Running a Simulation with CWP**

The simulation option is for use with two cooperating computers. The interface is made by two crossed audio cables from the analog audio output of one computer to the analog audio input of the other computer and vice versa.

As a first step set the clocks of both computers equal within about one second. Then choose snr and coherence time in the setup window and push the simulation button. The coherence time in the simulation section controls the actually simulated fading. The coherence time in the radio channel section is the value the receiver assumes the fading to be. Now start CWP on both computers by choosing the mode such that both programs can interact: tx1st + tx2nd or txonly + rxonly.

With the sampling rates set correctly a simulation should run at least for 24 hours without resynchronization. All parameters and the send text can be changed as in a QSO.

### **3.5. Running a QSO with CWP**

The recommended operating procedure is as follows:

- (1) Set your computer clock as precise as possible. An accuracy of 1 second is sufficient.
- (2) Start CWP or push the Sync Button of CWP.
- (3) Set the parameters, especially set Simulation „OFF“ and choose channel delay and coherence.
- (4) Push the Mode button of CWP until it runs in „rxonly“ mode.
- (5) Now wait for the first path to be recorded by CWP. Then CWP will inform you on the audio level. If it is larger than 0 dB then reduce the audio level until it is somewhere between -15 dB and 0 dB. The level is not critical.
- (6) If a CQ is decoded unambiguously, i.e. at least two times the same callsign, then do the following:
  - (a) Click on the decoded text with the left mouse button to decode on this frequency first.
  - (b) Type your reply into the send text field.
  - (c) Push the Mode button of CWP until it shows the correct mode, i.e. „txodd“ if the CQ was received in an even minute (time in the decode field).
- (7) The transmitter now will automatically send your reply.
- (8) The other operator will see your message decoded on his computer screen when he is already sending CQ again or replying to another station. So please be patient, CWP is a very slow mode.

- (9) The CWP report could be the SNR value without the sign, i.e. 21 for a signal decoded at -21 dB. But also the O-report and RO are possible. If there is no ambiguity (clear frequency) the **RO**, **RRR**, and **73** or **TNX** could be sent without a callsign. This works down to about -26 dB. But it generally is recommended to combine these messages at least with the own callsign.

### 3.6. Operating Hints

#### Simulation OFF

In contrast to the simulation in a real QSO only one computer is active, of course. The simulation button in the setup window must be „off“. Otherwise the simulated noise also would be sent. Furthermore, the transceiver control via the COM port is switched off in the simulation mode.

#### Transceiver Frequency

For monitoring a frequency band of 2 kHz choose the mode „receive only“. If you want to start with a CQ call then choose „txeven“ or „txodd“ to send in even minutes or in odd minutes respectively. The CWP receiver uses a bandwidth of 80 Hz centered at each carrier frequency found within the SSB bandwidth. If you reply a CQ call of some other station choose a frequency that will be within the SSB bandwidth on the other side. Tuning the frequency to fit that of the caller probably will lead to interference with other stations replying. If the other station replies to someone else, i.e. not to your call, then stay calling. Other than JT65, CWP is a narrow-band mode that usually does not lead to interference with others. The SSB bandwidth is good for up to 10 parallel QSOs. Outside of contests a single frequency for CWP should be appropriate. Never change the frequency within a running QSO.

#### Send Text

The send text must be entered with the ENTER key latest in second 55 of a minute. The characters <, >, and | will be replaced then by the text you have defined as your call sign, your CQ call, and your locator, respectively. Please note the generated Morse code above the send text input field. To optimize the weak-signal behavior, the Morse code should be repeated as often as possible. So it should be short. On the other hand long free space between the end of the Morse code and the marked repetition bar leads to an unkeyed carrier within this gap. The receiver at the other end then will not be able to reconstruct the carrierphase for the last and the first Morse symbol in an optimal way. Since the CWP receiver listens at the margin it usually fills blank space with **Es**. It therefore may be better not to transmit the absolutely shortest message. Other than JT65, CWP is a narrow-band mode. Therefore, there is no ambiguity on which decoded text belongs to what station as long as frequencies are not changed. Although it is a good practise in CW to send the calls in every path this is not necessary. Especially at marginal conditions it is possible to send simply **RO**, **RRR**, and **73** which results in a gain of about 3 db.

#### Frequency Stability

The receiver adapts to signals with slowly driftig frequency up to 6 Hz per minute. The generated signal must be as free of phase distortions such that the phase variation within a single Morse symbol is not larger than 90°. Modern transceivers with signal generation by DDS are as accurate. But the phase distortion of the channel may be large. This is surely the case for EME on 1296 MHz and higher, and also in tropo scatter conditions on 2 m. Depending on the fading situation the parameter coherence time in the channel parameter section must be set properly.

### Decoded Text

CWP sends every message repeatedly. Depending on the accuracy of the clocks used at both ends the decoded message may be wrapped around. The message **THIS IS A CWP TEXT** may be received as **TEXT THIS IS A CWP** or similar. It is to the operator to find the correct alignment.

As mentioned above the CWP receiver tries to read CW at the absolute margin. As a consequence it sees a signal where there is none. Especially unkeyed gaps often will be filled by **E**, **I**, or **T**. You should be aware of the fact that the confidence into an **E** is less than into **M** for example.

The most likely errors of CWP are dot shifts. The callsign **DJ5HG** for example often is corrupted into **N25HG**, **BO5HG**, **DJH5G**, **DJ5SP**, **DJ5HME**. Especially the latter case occurs when the message **CQ CQ DJ5HG** is sent which has two unkeyed blanks at the end.

The confidence of the individual characters is grey-coded.

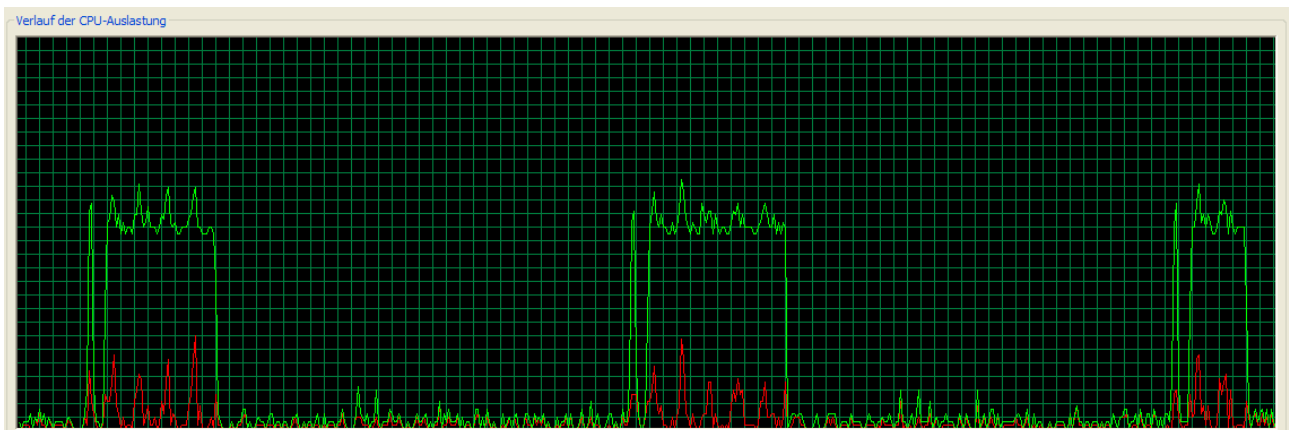
### Skin

All windows generated by CWP (main window, setup window, and help windows) can be resized with font sizes proportionally adapted. Fit them to your personal preferences.

## 3.7. Known Problems

### Blocked Buttons

The CWP decoder is a CPU-intensive task. Even in a multi-processor system (CWP then uses all processors) the buttons of the main window may not react immediately when the decoder is running. The same is true for the setup interface. Figure 3.4 shows the processor activity on a 3 GHz Dual Processor system under WinXP.



**Figure 3.4. Processor activity of CWP within 4.5 minutes. The short peaks left from the longer ones are due to the computation of the signal to be sent. Three decoding periods are visible with 4, 5, and 2 signals decoded.**

## 3.8. Files

All pathes to files are printed into the black window at startup.



## 4. Remarks

These hints only are vague forecasts. They are written before any QSO ever had been run with CWP. CWP is an experimental software. Problems should be reported to the author.

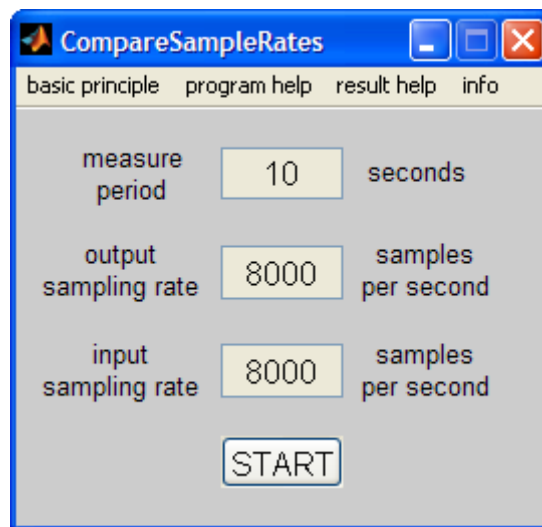
## Literature

- [1] Klaus von der Heide, *CW for Weak Signal Applications: CWP*, DUBUS 36, No 3, pp. 52-58.
- [2] Joe Taylor, <http://physics.princeton.edu/pulsar/K1JT/>

## Appendix A. Measuring the Sampling Rates of a Sound Card

### A.1. Precise Measurement of the Difference between Input and Output Sampling

The relation of the sampling rates of output and input can easily be measured by generating a tone on the output and feeding it back on the analog path via an appropriate audio cable to the input. A DSP program can determine the phase difference of both, the outgoing and the incoming, waves, and compute the relation of the sampling rates. The author offers the program *comparesamplerates* for this purpose. Its user interface is as follows:

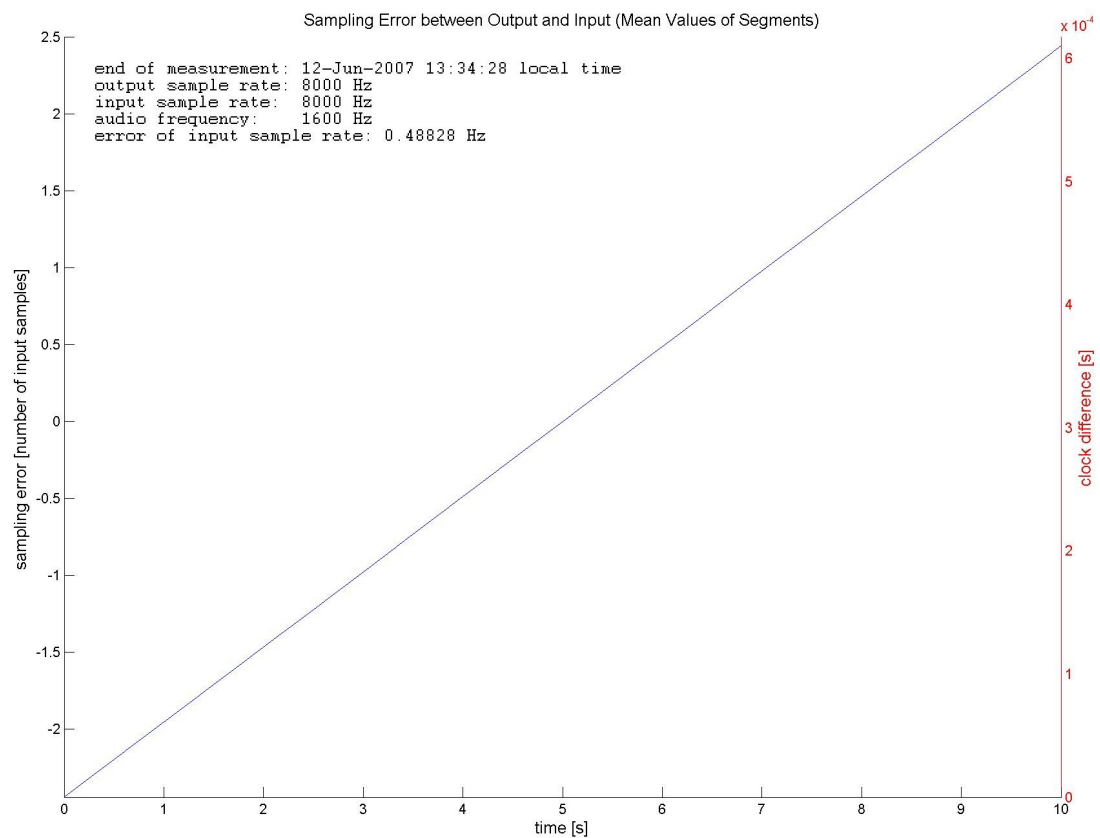
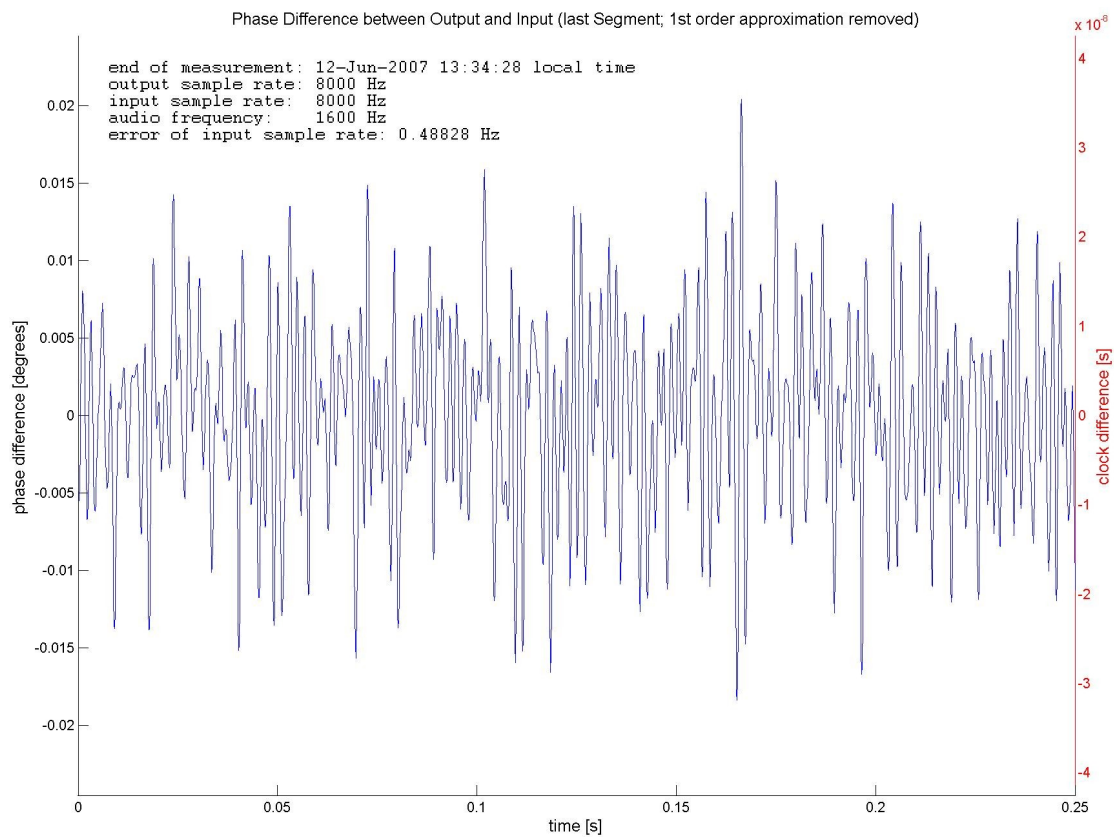


**Figure A.1. User Interface of the program *comparesamplerates*. Use the values shown here for your measurements. Push the START button after you have connected the analog audio input with the analog audio output of your soundcard with an appropriate audio cable.**

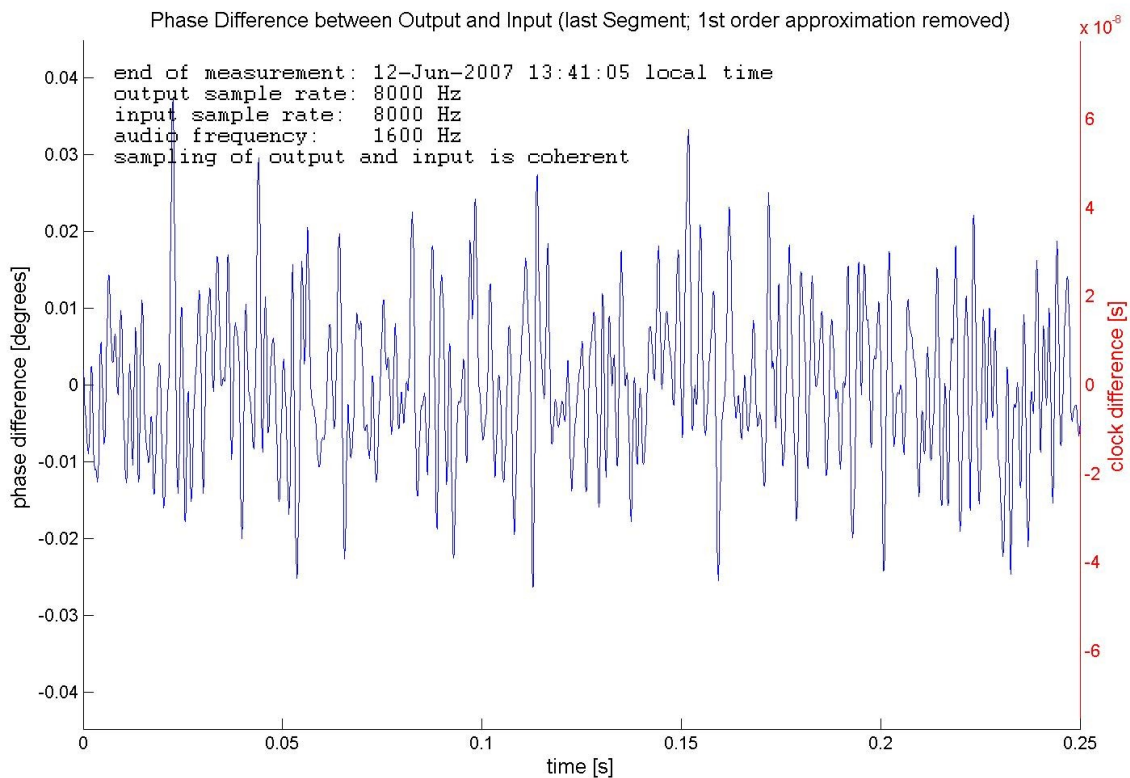
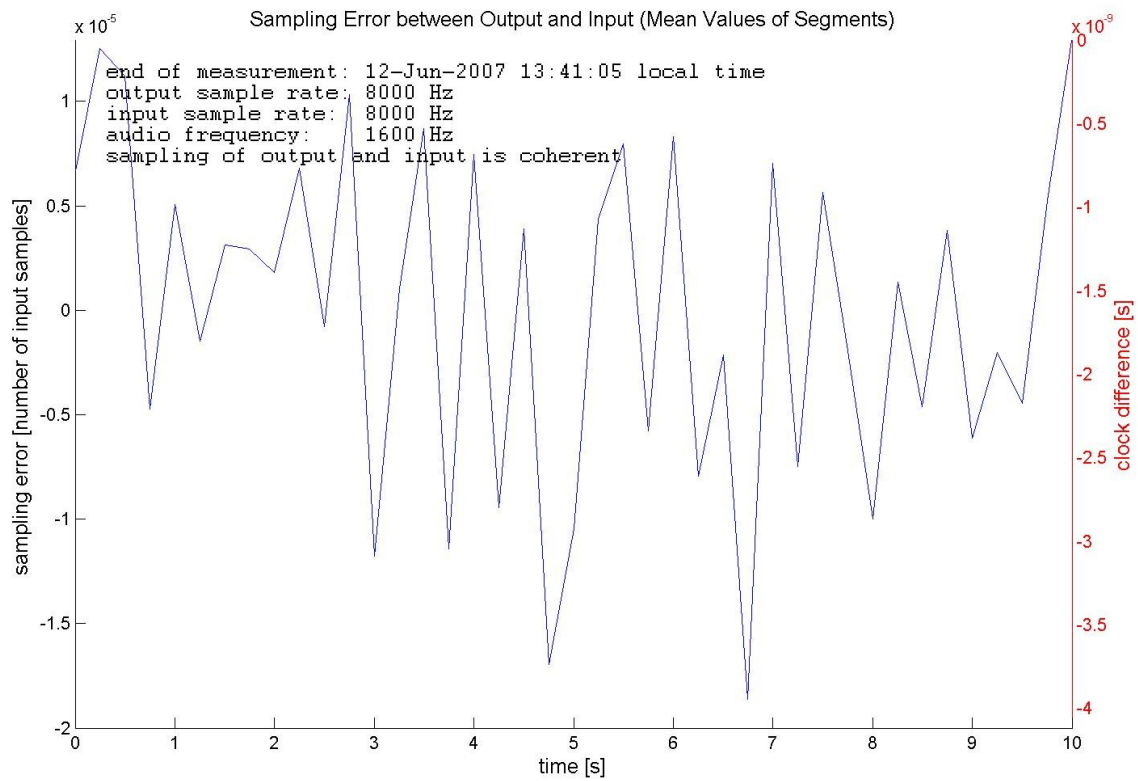
Use the help menu of the program for further information. Type in just the values shown and enter them by the ENTER key. Be sure to have the audio cable installed correctly, and to have set the output and input volume appropriately. Then push the START button and wait the measure period to get the result. It may look as shown in figure A.2. The lower figure clearly indicates a linear increase of the phase difference which corresponds to a frequency difference. So, on this sound card the sampling rates for input and output are different. With a measure period of 10 seconds three decimals right of the decimal point of the computed frequency difference should be correct. Two correct decimals are sufficient for optimal use of CWP.

If the sound card uses the same sampling rates for input and output, the result may be similar to that shown in figure A.3.

This program only determines the relation of both sampling rates, not the absolute values. That is the task of two further programs: *measuresamplerate* and *samplingrate* (see chapter A.2).



**Figure A.2. A linear phase shows that the sampling of input and output are not identical.**



**Figure A.3. The phase shows noise here within 0.04 degrees.  
 This indicates identical sampling at input and output.**

## A.2 Precise Measurement of the Output Sampling Rate

This measurement needs some extra hardware: (1) A longwave receiver that can receive a frequency normal as DCF77, HBG, MSF etc. and (2) a very simple interface that distorts the audio output to produce harmonics in the long wave near the received frequency normal. A circuit diagram of the interface is shown in figure A.4.

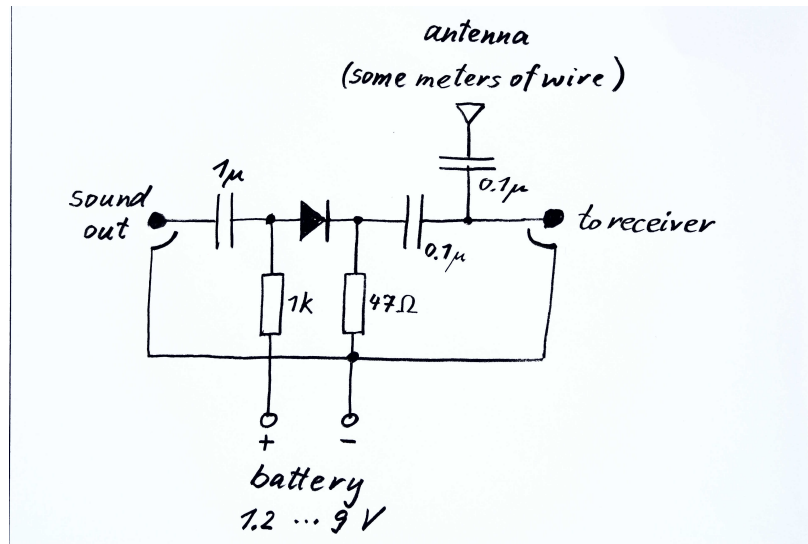


Figure A.4. Circuit diagram to generate harmonics.

As a first step tune to a frequency normal with frequency less than 100 kHz (the harmonics of the audio will not reach further). If you try CW mode then the frequency setting should be the nominal frequency, 77500 Hz for DCF77 for example. If you prefer USB or LSB tune 1.0 kHz lower or 1 kHz higher resp. If you can hear the signal that's fine, but the signal may be weak. Now connect the receiver audio output with the sound card audio input, start the program, and type the necessary information into the interface fields of the program window. The audio signal of the frequency normal should be within the audio passband. Then push the START button and wait for the results. It of course is a nonsense measurement since the interface circuit is not installed. But there should be a peak at the same frequency in all four resulting spectra indicating the carrier of the frequency normal. For confidence, switch the antenna off and push the START button again. This time the peaks should be vanished. If all was o.k. until now, assemble the interface circuit and install it with a second audio cable from the sound card output to the interface circuit. For a first test use earphones at the receiver output instead of the cable to the computer. Set the audio output volume to maximum and push the START button. The generated harmonic should be hearable. Otherwise take the antenna off and try again. Optimum results will be obtained if the amplitudes of the frequency normal and the harmonic are comparable. If you succeeded until now install the cable from the receiver to the soundcard input again and push the START button. If no generated harmonics are visible then the audio output amplitude is too low or the antenna should be reduced. If the peaks of the harmonics are much greater than those of the frequency normal then reduce the audio output volume until all looks like the following figures A.5 and A.6 which were taken of DCF77 with with an ICOM IC-736 plus 3m of wire as antenna.

Figure A.5 shows the first trial with a large 300 Hz-CW-bandwidth and 10 seconds period. In the second trial the band is considerably narrowed, the output sampling rate is changed in the corresponding edit field of the program into the result of the first trial 7999.340 Hz, and the period is set to 20 seconds. The result is shown in figure A.6.

The accuracy of the measured output sampling rate only depends on the period. CWP needs an accuracy of 0.01 Hz which easily is obtained with an arbitrary receiver using a period of 30 seconds. Since the program stores all sampled data in main memory you could encounter an *out of memory error* at very long periods (1000 s). The stability of the sampling rates usually is very good. The author measured drifts lower than 0.0001 Hz per hour for the rate 8000 Hz.

Figure A.7 shows the results of measurements with large periods at three different frequency normals. The measured sampling frequency remains the same within 0.001 Hz.

With the absolute output sampling rate and the relation between output and input we now have all necessary information: Set the output sampling rate into the corresponding setup field of CWP and set the measured output sampling rate minus the measured difference from (2.1) into the field for the input sampling rate.

If you do not have an appropriate longwave receiver or if no frequency normal can be heard or if you had no success with the interface circuit, then read the next paragraph A.3.

MeasureSamplingRate

help info

LF QRG

77500

Hz

sampling frequency

8000

Hz

audio passband

550 850

Hz

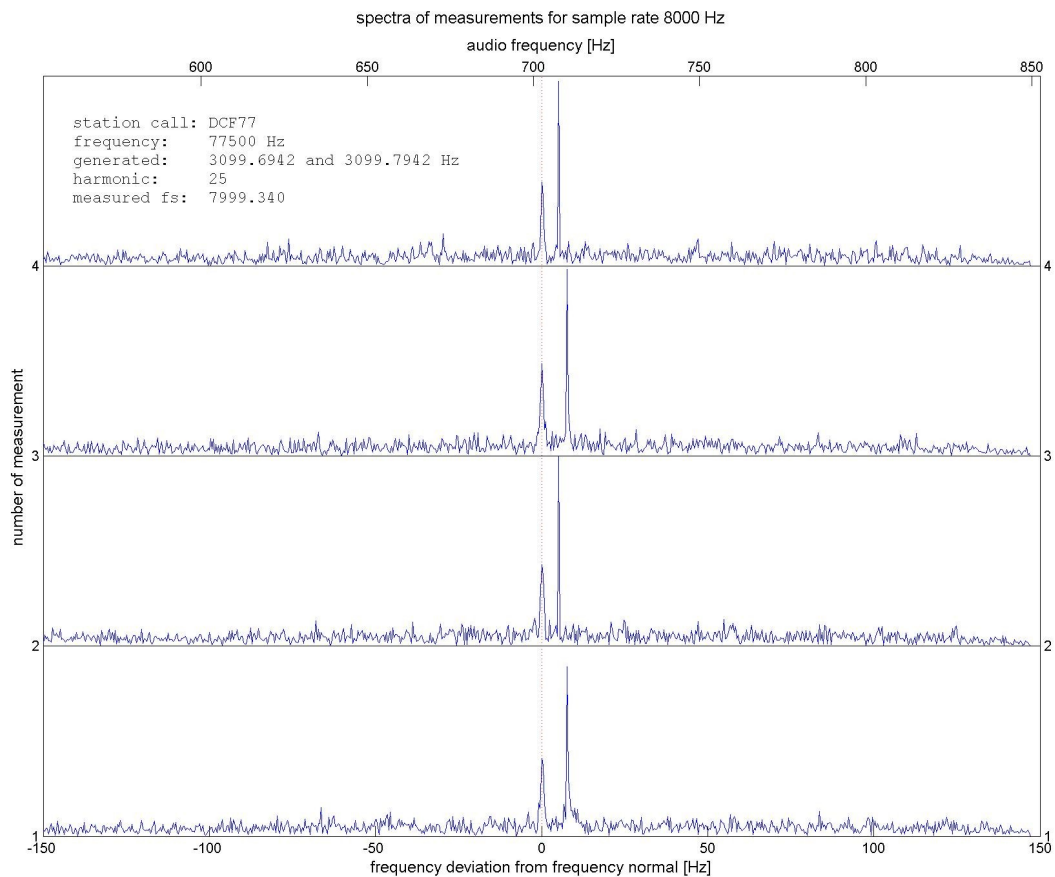
measure period

10

s

START

START



**Figure A.5.** The carrier frequency of DCF77 is typed in, and a wide input band is chosen for a first attempt. A period of 10 seconds is sufficient for the first start. The generated harmonics are all to the right of the frequency normal which indicates that the assumed sampling rate is too large.

MeasureSamplingRate

help

info

LF QRG

77500

Hz

sampling frequency

7999.340

Hz

audio passband

690 715

Hz

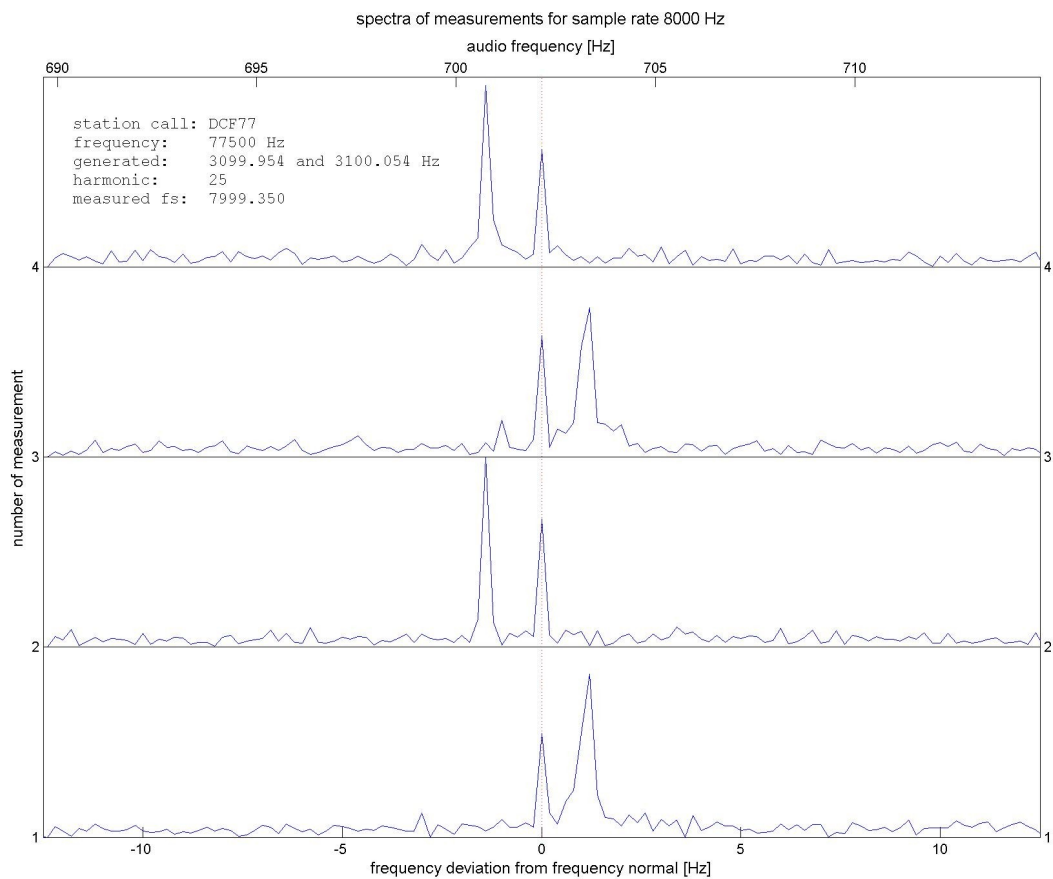
measure period

20

s

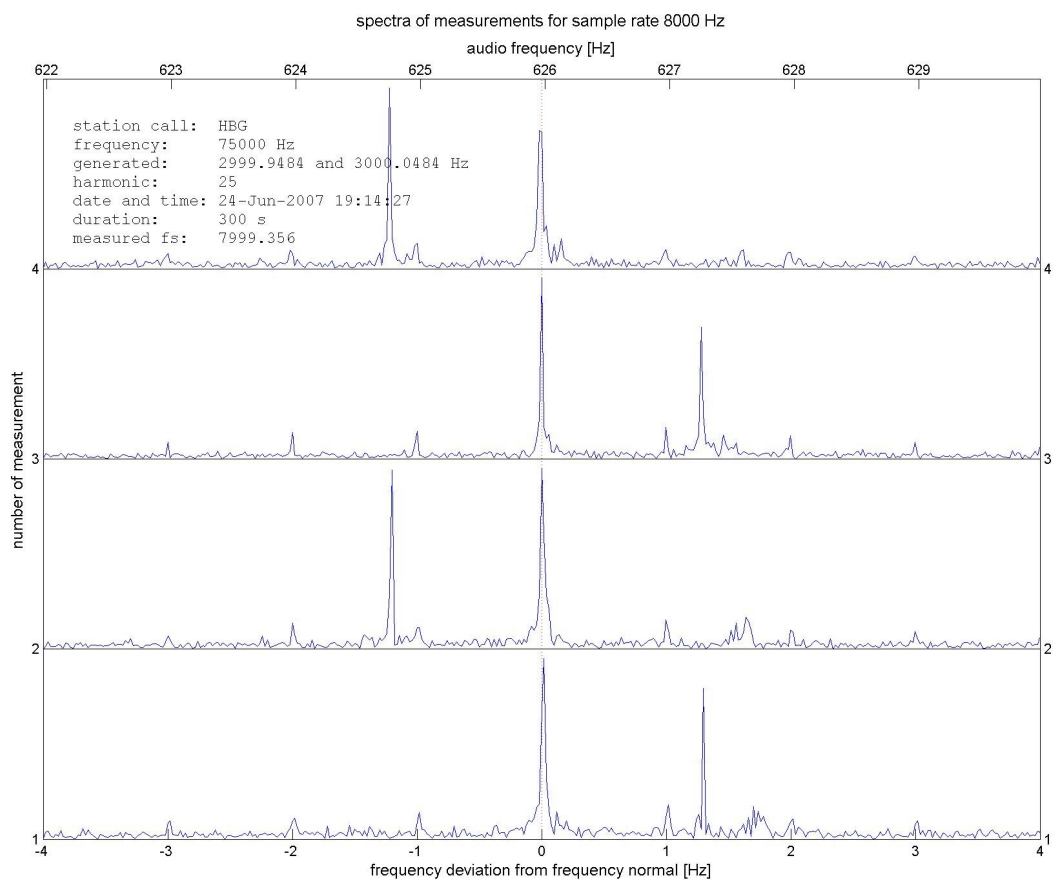
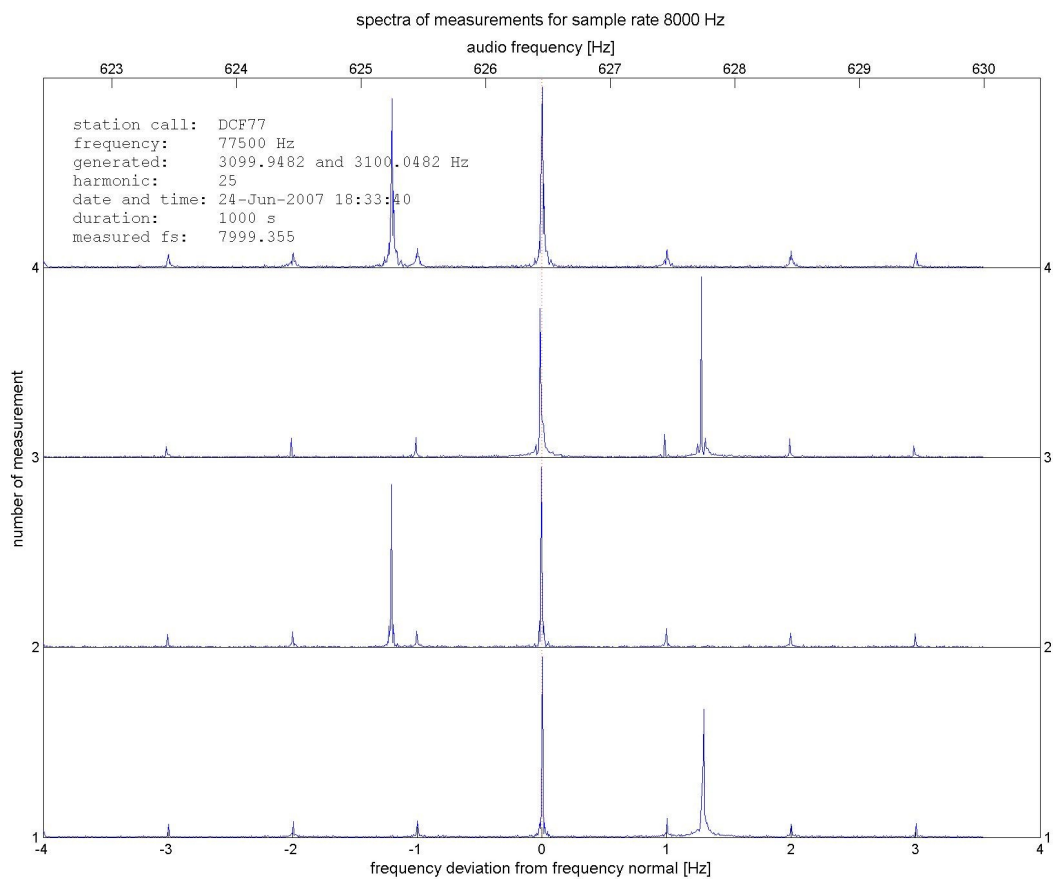
START

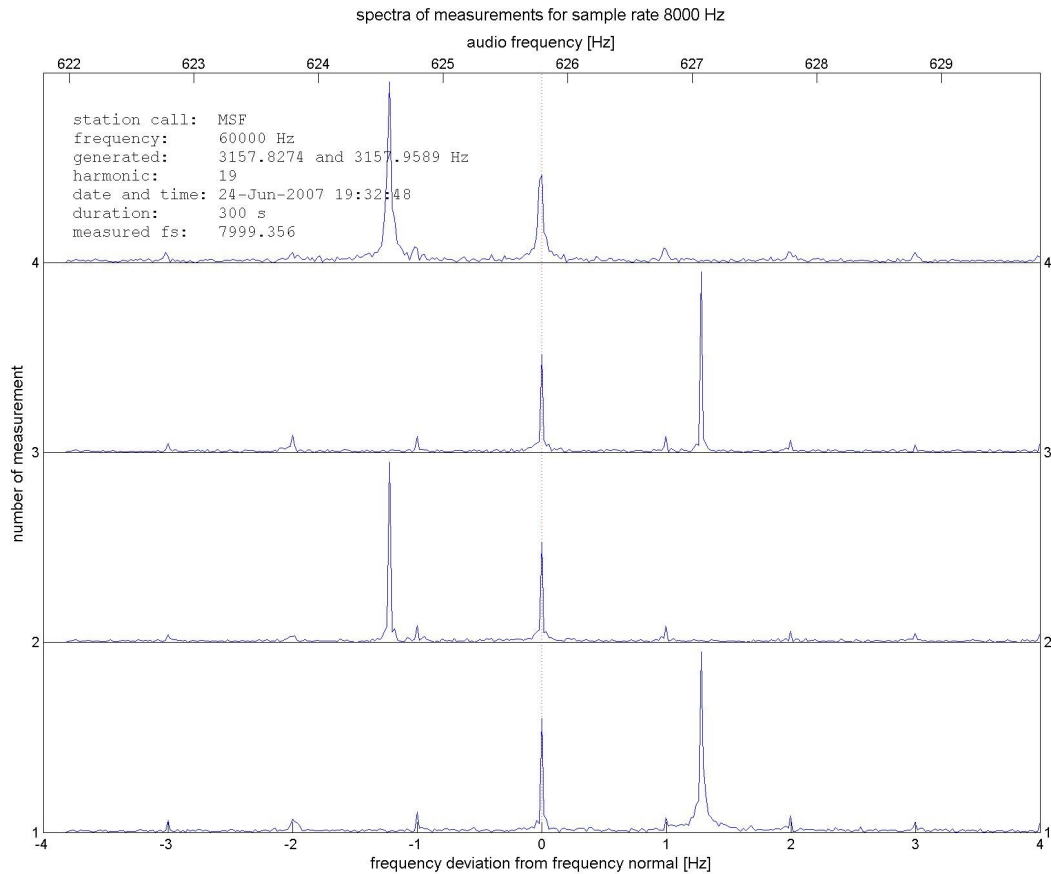
START



**Figure A.6. The assumed sample rate was corrected here into the result of the first measurement. Also the band is narrowed to give a better resolution, and the period is 20 seconds now.**







**Figure A.7. Three subsequent measurements using the different frequency normals DCF77 on 77500 Hz, HBG on 75000 Hz, and MSF on 60000 Hz. The antenna was a 10 m wire 3 m above ground. The measured sampling frequency remains constant within 0.001 Hz. The receiver drifted down by 0.7 Hz. The program measures this drift and compensates it as long as the drift is constant in time. The weak comb-like structure of the spectrum with the small peaks at 1.0 Hz distance is caused by the amplitude modulation of the frequency normal station with negative pulses every second.**