Storno
RADIOTELEPHONES

CQM6000 Service Mode

Page Updated 9 Jan 2005

Discuss CQM6000 Service Mode

The Stornophone 6000 came with many advanced features and with this increased complexity came the requirement for better service tools. This lead to the development of a comprehensive method of directly controlling many aspects of the radio hardware using a special mode of operation known as Service Mode.

Getting into Service Mode

Although the obvious first question on your lips may be "So how do I get my Stornophone 6000 into Service Mode?" the answer is not the same in all cases. For the purposes of this article we will mainly be discussing Service Mode as it applied to the standard Opus/EC/EF/EL versions of CQM6000 firmware but you should bear in mind that there were also many variations on this theme. This applied particularly if your radio had system specific firmware or hardware and many applications did not directly support Service Mode in any form.

Another consideration is that some system specific radios also had other modes of operation particular to the system application. Examples included 'engineering mode' which could be invoked by a special keypad sequence or password, entry to which permitted engineers access to modify the radio's signalling ID or the famous 'Manuel Mode' which allowed a PhoneNet 2200 trunked mobile to be de-trunked and used as a conventional radio. The Stornomatic 6000 and other mobile phone variants also had special requirements, selection of their multiple power levels, country codes, etc.

One rough indicator to the presence of the comprehensive Service Mode was the size of the application in the main program EPROM. If this chip was a 27C256 it was unlikely to support many of the capabilities listed below, if it used a 27C512 then the chances were better.

Once the presence of a Service Mode was confirmed there were a number of possible different ways of getting the radio to enter that mode as follows:

Hardware method

The radio could be booted up in Service Mode by making a connection from the 'portable' input (normally used to detect when a CQM6000 was being used in a transportable cassette) on pin 3 of the rear D25 connector and linking it to the 'car radio mute' output on pin 16. Also as the mute used an open collector output, some form of pull-up was required, (either a 1k resistor pulling up to +12V or alternatively a 220 ohm resistor pulling up to the internal +5V from pin 9).

Care was required when using this method to ensure that these pins were isolated from external influence if the Service Mode link was used in conjunction with the standard connection arrangement.

The link was normally made using a momentary switch and it was necessary for the switch to be made during the power-up of the radio because the car radio mute output was only enabled for a very short period during the radio initialisation, during which the change in the portable input was noted. Pressing the switch after the self test period would not change the mode.

Codeplug method

The quickest and easiest method of getting a CQM6000 into Service Mode was simply to insert a Service Mode codeplug key into the socket on the control head. This worked even if the radio was already operating in it's normal mode and was a perfect technique for when a quick edit of just a few personality parameters was required. Normal operation was restored by switching the radio off and on again.

There were only a couple of complications about this method. Firstly, the majority of control heads did not have the optional codeplug socket fitted. Secondly, you could never find a Service Mode codeplug key when you really needed one. The reason for that was there was no straightforward way to make a Service Mode codeplug key from a user key. As will be shown below the contents of the EEPROM inside the codeplug key can be edited using Service Mode, however there was a 'feature' within the mask program in the control head processor that prevented the first 2 bytes being edited so the codeplug type was effectvely fixed.

If necessary, the codeplug type could be changed by connecting the codeplug directly to an I2C compatible programmer (or you could make your own). The first 2 bytes were programmed as follows:

Codeplug type	Address 00,01	Part Suffix
CQM6000 Standard User	00,01	G5
CQM6000 Service Mode	00,11	G6
Stornomatic 6000 NMT Service	00,81	G3

H-BUS method

This method used the same technique as the codeplug method above except that the insertion of a Service Mode codeplug was simulated by an external device connected to the radio's H-BUS. This method was used by automatic test equipment setups to control the radio when exercising test scripts and also by personality programming setups including the standard PC with H-RIB interface.

Personality method

Not suitable for performing personality editing but convenient for testing new unprogrammed sets or other cases when the radio had to be opened and serviced, the radio would always be in Service-Mode if the personality (E)EPROM was filled with null data (meaning all bytes programmed to 00, note that with all bytes set to FF the radio would boot normally but fail the self test giving an error message such as "PPROM EMPTY"). Another method was to remove the personality (E)EPROM from it's socket and start the radio with only the main firmware installed. Results for these methods vary according to version.

Keypad method

This method only worked for a few cases, mainly the Stornomatics and radiophone versions. The technique was to press down the 1 and * buttons on the keypad while switching on the radio.

Nothing else works method

This method was used for radios when the installed firmware did not support Service Mode but it was necessary to use Service Mode functionality and consisted of temporarily exchanging the main EPROM for a version that did what was required!

Service Mode operation

Having entered Service Mode you would be able to enter commands directly from the keypad of the control head. All commands were entered using 3 digits representing functions, subfunctions and paramaters. It should be noted that the selection of the digits was determined by the standard keypad layout, regardless of layout variations used on customised control heads.

Only the numeric keys were used plus the * key which was used to escape back to the first function digit entry in case of error or to abort from continuous read and edit functions.

Example: Command to display software package number

The command to display the part number of the firmware was command 0, subfunction 0, parameter 1 so to use the command you would simply use the keypad to enter:

0 0 1

This would cause the display to show the full part number and revision of the firmware:

Commands requiring Decimal data entry

Commands performing more complex functions requiring further entries such as the selection of channels, signalling systems, etc. provided prompts in the display and a flashing cursor where the next digit should be entered.

Commands requiring Hex data entry

Commands for entering signalling digits and reading or editing memory addresses and data required further entries in the form of a 2-digit decimal number representing each byte of address or data.

For example, to read the contents of the main EPROM at address **09AC** first the command was entered:

7 1 0

Then the cursor would prompt for the address A: at which point the Hex address could be constructed by entering the 2 decimal digits of each byte:

0 0 - (0)

0 9 - (9)

1 0 - (A)

1 2 - (C)

This would cause the display to indicate the current address at A: and the data at D:

Note that in this mode you could step through the following addresses by pressing a digit button of the required step size.

Function 0 - Special Functions

000	Reset function		EF
001	Display software package number		EF
002	Display radio type / Synthesizer Lock State		EF
004	Erase encryption keys		EL

Function 1 - Channel Functions

100	Reset all channel functions		EF
110	Select Rx channel from channel group 0		
111	Select Rx channel from channel group 1		EF
112	Select Rx channel from channel group 2		
113	Select Rx channel from channel group 3		
120	Select Tx channel from channel group 0		
121	Select Tx channel from channel group 1		EF
122	Select Tx channel from channel group 2		
123	Select Tx channel from channel group 3		
131	Rx = 66.510 MHz Tx = 66.460 MHz		
132	Rx = 74.520 MHz Tx =74.040 MHz	Alignment channels for CQM633X	EF
133	Rx = 87.060 MHz Tx = 86.970 MHz		
134	Rx = 77.000 MHz Tx = 77.000 MHz		
141	Rx = 136.900 MHz Tx = 136.900 MHz		
142	Rx = 151.150 MHz Tx = 150.400 MHz	Alignment channels for CQM611X	EF
143	Rx = 172.450 MHz Tx = 172.300 MHz		
144	Rx = 155.000 MHz Tx = 155.000 MHz		
161	Rx = 404.600 MHz Tx = 404.700 MHz		
162	Rx = 430.200 MHz Tx = 430.700 MHz	Alignment channels for CQM6662/3	EF
163	Rx = 467.300 MHz Tx = 467.400 MHz		
164	Rx = 436.500 MHz Tx = 436.500 MHz		
165	Rx = 404.600 MHz Tx = 404.700 MHz		
166	Rx = 430.200 MHz Tx = 430.700 MHz	Alignment channels for CQM6664	EF
167	Rx = 467.300 MHz Tx = 467.400 MHz		
168	Rx = 436.500 MHz Tx = 436.500 MHz		
171	Rx = 174.900 MHz Tx = 174.900 MHz		
172	Rx = 188.900 MHz Tx = 188.400 MHz	Alignment channels for CQM677X Lo	EF
173	Rx = 208.600 MHz Tx = 208.600 MHz		
174	Rx = 192.000 MHz Tx = 192.000 MHz		
175	Rx = 190.900 MHz Tx = 190.900 MHz	Alignment channels for CQM677X Hi	EF
176	Rx = 204.560 MHz Tx = 204.300 MHz		

177 Rx = 223.650 MHz Tx = 223.600 MHz

178 Rx = 207.500 MHz Tx = 207.500 MHz

Function 2 - Transmit Functions

200	Reset all Tx functions	EF
211	Select high Tx power	EF
212	Select low Tx power	EF
220	Select speech transmission in clear mode	EL
221	Select speech transmission in encrypted mode	EL
230	Select normal range (REX Off)	EL
231	Select extended range (REX On)	EL

Function 3 - Rx Functions

300	Reset all Rx functions	EF

Function 4 - Audio Functions

400	Reset all audio functions	EF
410	Volume level 0	EF
411	Volume level 1	EF
412	Volume level 2	EF
413	Volume level 3	EF
414	Volume level 4	EF
415	Volume level 5	EF
416	Volume level 6	EF
417	Volume level 7	EF
430	Squelch bypass off	EF
431	Squelch bypass on	EF
440	Select audio through DCAP	EF
441	Select standard audio paths	EF

Function 5 - Signalling Functions

500	Reset all signalling functions	EF
521	Enable decoding	EF
522	Select signalling system for decoder	EF
523	Select number of digits in received telegrams	EF
531	Send a telegram	EF
532	Select signalling system for encoder	EF
533	Select number of digits in send telegrams	EF
534	Enter digits of telegram (in Hex)	EF
540	CG encoder (CTCSS tone generation)	EF

541 CG decoder (CTCSS tone decoder - requires CG option board) EF

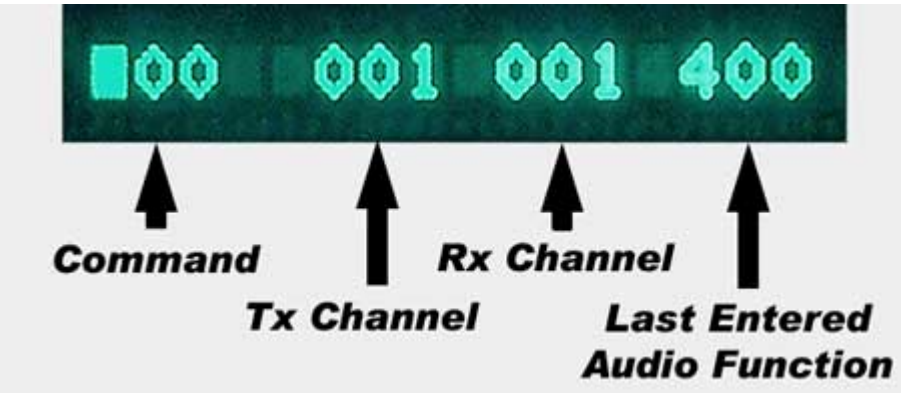
Function 6 - Input/Output Functions

610 Rear connector read pins EF
611 Rear connector reset specified pin EF
612 Rear connector set specified pin EF
613 Rear connector change specified pin EF
62X Output expander 0 commands EF
63X Output expander 1 commands EF
640 Input expander 1 read port EF
65X Internal port 1 commands EF
66X Internal port 3 commands EF
67X Output expander 2 commands EF

Function 7 - Memory & Programming Functions

700 Reset memory & programming functions EF
710 Read main EPROM at specified address in range 0000 - FFFF EF
720 Read personality (E)EPROM at specified address in range C000 - C7FF EF
730 Edit personality (E)EPROM at specified address in range C000 - C7FF EF
740 Read codeplug key at specified address in range 00 - 1F EF
750 Edit codeplug key at specified address in range 00 - 1F EF
760 Select codeplug device address for codeplug commands in range 20 - 2A EF
770 Re-initialize battery backed SRAM memory TRK

Function 1 Channel display



Function 5 Signalling system table

Signalling type	System
ZVEI - 1	01
ZVEI - 2	02

ZVEI - 3	03
CCIR	04
EEA	05
VDEW	06
Binary 1200 Baud ZVEI	09
Binary 1200 Baud EEA	11
Custom System	15

Watch out here for more information coming later...

Back to Stornophone 6000 Page