Future Feature for the CS800D PLUS

The radio currently can only communicate using DMR and Analog. This document is going to show how we can extend it to also communicate using NXDN, FUSION, P25 PHASE I, P25 PHASE II and DSTAR.

NXDN, FUSION and P25 PHASE II uses the same AMBE vocoder used with the DMR. DSTAR and P25 PHASE I used a similar but different vocoder so the external computer will have to handle those two different vocoders. The patents for those two vocoders have long expired so there should be no legal issues in using the technology.

The basis for this technique is to have certain functions that the radio can do be done by the radio and other things that are beyond the capability of the internal processor because of speed or memory size, done by an external computer attached to the serial port and parallel port of the radio.

Conceptual Concept

The radio can do MDC1200 receive and MDC1200 transmit by using its firmware. MDC1200 is a digital format not too different than DSTAR. That means the radio has access to both the receive chain where it gets the raw data from an A/D converter and the transmit chain where a D/A converter generates a modulation waveform. Because the radio can do DMR, it has to have a AMBE II Vocoder. In this radio the AMBE II Vocoder is a firmware routine. Having the AMBE II Vocoder in the radio is important because the AMBE II Vocoder is still under patent protection and as such cannot be put in another product without proper licensing.

We now have everything we need in the radio's hardware and firmware to make a multimode digital radio apart from the firmware to process the different modes. In theory, that software can be put in the radio

but to do that we would need the source code of the radio. The manufacturer is not likely to release the entire source code but is willing to work with us to allow an external co-processor to do the protocol processing and the radio to process what the external co-process generates.

This document is going to specify the protocol between radio and the co-processor. It is not written in stone and as such can be modified to accomplish the task needed to make the best amateur radio possible.

Receive Mode

In the receive mode, we need two things. We need the raw data from the receive chain and a way of processing the vocoder. The raw receive data will be sent between the radio and the co-processor in 50 millisecond packets at 16-bit resolution. For DMR, NXDN, FUSION and P25 PHASE II, the firmware on the co-processor will recover the AMBE II data from the receive data stream and send it back to the radio to be processed by the on board AMBE II vocoder and eventually be output to the speaker. For DSTAR and P25 PHASE I, the co-processor will have the vocoder and will output data to the radio in 50 millisecond packets that will be then sent to the D/A converter in the radio and eventually be output to the speaker.

If you want to use a M17 vocoder, substitute the native vocoder for the M17 vocoder. For DMR, we will have a special mode where the co-processor acts as a vocoder for the radio instead of the built in AMBE II vocoder.

Transmit Mode

In the transmit mode, we need two things. We need to send raw modulation data to the transmit chain and a way of processing the vocoder. The raw transmit data will be sent between the radio and the co-processor in 50 millisecond packets at 16-bit resolution.

For DMR, NXDN, FUSION and P25 PHASE II, the radio will send AMBE II packets every 50 milliseconds that originated from the microphone of the radio. For DSTAR and P25 PHASE I, the radio will send raw voice data to be processed by the Vocoder in the co-processor.

In this mode, we are receiving microphone audio that has been processed by the AMBE vocoder if used for NXDN, Fusion or P25. If used for DSTAR, we are getting raw audio data that then gets compressed using the DSTAR vocoder on the external computer. The external computer then takes this voice data, puts it in packets, and modulates the radio by sending packets to the D/A converter on the radio.

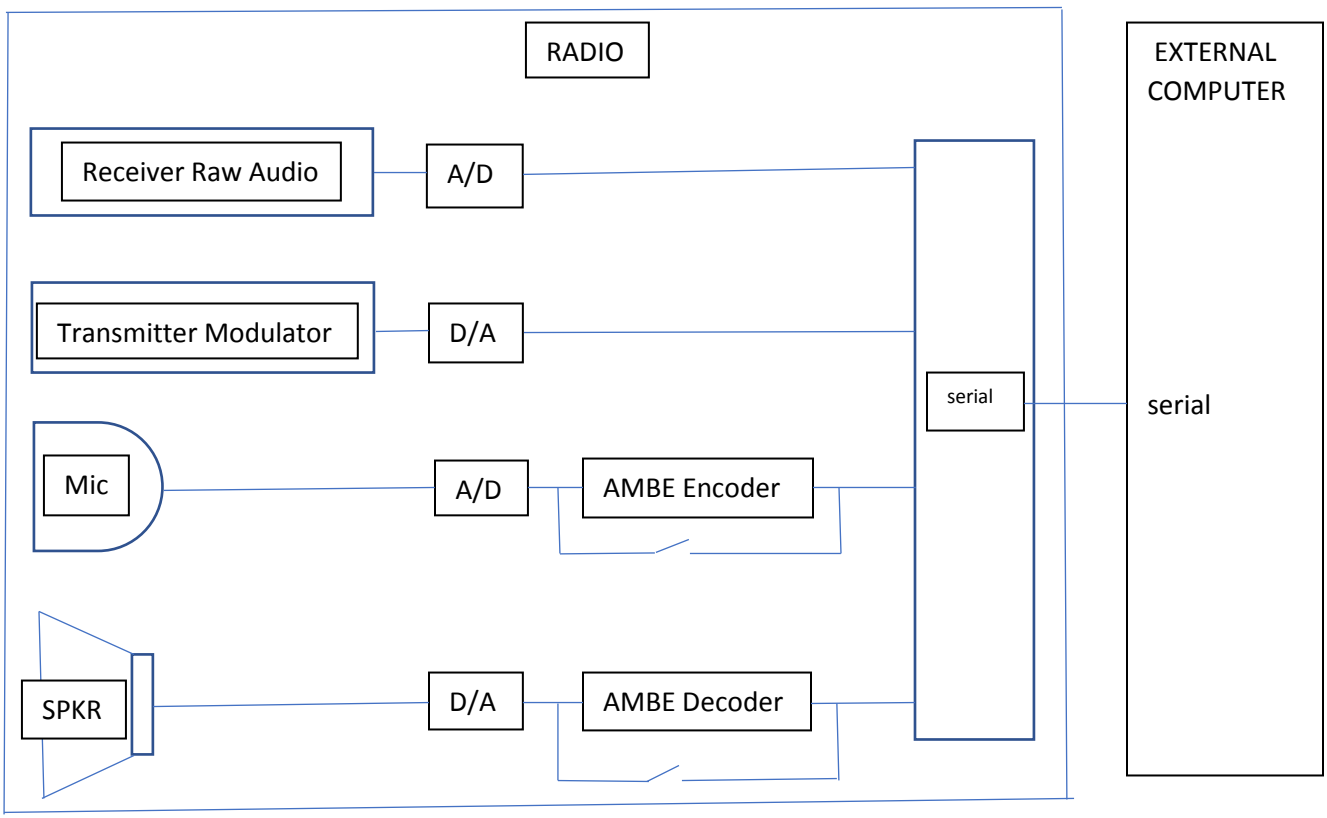
The co-processor now takes the vocoder data, puts it in the data stream of the specified protocol and generates 16 bit digital data to be processed by the D/A converter on the radio that will eventually modulate the transmit chain of the radio.

If you want to use a M17 vocoder, substitute the native vocoder for the M17 vocoder. For DMR, we will have a special mode where the co-processor acts as a vocoder for the radio instead of the built in AMBE II vocoder.

The co-processor

The external computer can be as simple as a single chip micro connected the DB15, a Raspberry Pi computer, or a standard PC. The key to this enhancement is the fact that the amateur community has already done most of the work already.

A block diagram of this feature is shown on the next page.



BLOCK DIAGRAM OF RADIO ATTACHED TO CO-PROCESSOR

API BETWEEN RADIO AND CO-PROCESSOR

Data from radio to co-processor

Raw Voice (microphone) Data

The radio sends 50 milliseconds of raw voice data from the microphone. The data is sampled at a rate of 8,000 samples per second at 16 bit resolution.

AMBE Voice (microphone) Data

The radio sends 50 millisecond of compressed data originating from the microphone of the radio. The compressed data is AMBE II format.

Raw Receive Data

The radio sends 50 milliseconds of raw receive data from the discriminator. The data is sampled at a rate of 8,000 samples per second a 16 bit resolution.

GPS location and Time

The radio sends once a second, the current location of the radio along with the time.

All Parameters from current channel

Each channel has up to 128 bytes of parameter data. This data is sent to the co-processor to be used

Parameters from specified protocol or mode

These parameters are in various screens of the CPS and stored in the radio. The protocol or modes are as follows:

1. Analog
2. DMR
3. DSTAR
4. FUSION
5. NXDN

6. P25 PHASE I
7. P25 PHASE II
8. Satellite
9. Hot Spot
10. Serial Data
11. APRS

Receive/Transmit mode

This tracks the PTT button.

Data from co-processor to radio

Raw Voice (speaker) data

The co-processor sends 50 milliseconds of raw voice data to the speaker originating from the data stream of the radio and possibly decompressed using a vocoder. The data is sent at a rate of 8,000 samples per second at 16 bit resolution.

AMBE Voice (speaker) data

The co-processor sends 50 millisecond of compressed data originating from the data stream of the radio. The compressed data is AMBE II format.

Raw Transmit Data

This is the raw modulated data going to the D/A converter in the radio and eventually being transmitted by the modulator in the radio. This data is sent every 50 milliseconds.

Enable/Disable GPS

If the GPS is enabled, the radio sends the location of the radio and the time every second. If the GPS is disabled, then the radio does not send GPS data to the co-processor. This does not affect the use of GPS in other modes.

Set to use external/internal vocoder

This parameter sets the switches on the AMBE II encoder and AMBE II decoder to determine if the data is being compressed or is raw.

Request Parameters from current channel

Gets all the parameters from the channel the radio is currently listening to.

Request Parameters for specified protocol or mode

The protocol or modes are as follows:

1. Analog

2. DMR
3. DSTAR
4. FUSION
5. NXDN
6. P25 PHASE I
7. P25 PHASE II
8. Satellite
9. Hot Spot
10. Serial Data
11. APRS

Set to Transmit/Receive/Disable

This allows the co-processor to override the main radio if set to transmit or receive. If set to disable, then the PTT button determines the mode.

Send Message to Display

Allows the co-processor to send messages to the display. The following commands are allowed:

1. Disable Radio from using the display
2. Enable Radio to use display
3. Clear screen
4. Send message to line 1
5. Send message to line 2
6. Save current screen
7. Restore saved screen

There is nothing in this protocol that prevents the co-processor from also sending a voice message to the radio for the visually impaired or for some other reason.