Modification of the TK-931 to the amateur band

15-DEC-96

Dan Minear
N6RDK

1. Introduction

By examining the operation of the TM-441A/E and comparing it to the TK-931, it is possible to program the TK-931 to operate on the 902 Mhz amateur band.

This process consists of 2 parts: changing the receive frequency to go lower, and changing the repeater shift frequency from 39 Mhz to 25 Mhz for the California 902 band.

2. Documents

Kenwood TM-441A/E service manual B51-8061-00(S)955

Kenwood TK-931/931(HD) service manual B51-8086-00 (O) 1103

3. Kenwood TM-441A/E examination

3.1. Transmit/receiver PLL operation

The TM-441 has a common PLL that is programmed to transmit or receive a specific frequency. In the K or P model, the transmitter frequencies span from 438.000 Mhz to 449.995 Mhz. The receive PLL frequencies span from 407.175 Mhz to 419.179 Mhz. This allows for a first IF of 30.825 Mhz.

The PLL is programmed by shifting in serial data. These data consist of 2 counters: a 10 bit counter n, and a 7 bit counter A. The E or M models operate from 430 to 439.995 Mhz. From the manual,

> In 430 Mhz mode, Fvco (RX) is calculated by the following formula:
> Fvco = (430 - 30.825) = {(n x 128) + A} x Fosc / R
> where,
> Fvco: VCO output frequency
> n: Binary value of the 10-bit programmable counter
> A: Binary value of the 7-bit programmable counter
> Fosc: 12.8 Mhz reference frequency
> R: Binary value of the 14-bit programmable counter
> 2560 (5, 10, 15, 20, or 25 kHz step mode)
> 2048 (12.5 kHz step mode)
> In 5, 10, 15, 20, or 25 kHz step mode, n = 623 and A = 91
> Therefore, Fvco is calculated as follows:
> Fvco = {(623 x 128) + 91} x 12800 / 2560
> = {79744 + 91} x 5
> = 399175 = 399.175 Mhz

There's a few things to observe about this process. Since A is a 7 bit counter, it can only span the values 0 - 127. Because the above equation multiplies n by 128, n can be thought of as a most significant portion, and A as a least significant portion. Therefore, you can view n and A as a *continuous* 17 bit counter and not worry about the dual counter process. Furthermore, depending on the value of R, a single step of the 17 bit counter will increase the PLL frequency by the desired 5, 10, 15, 20, or 25 kHz.

The concept here is that by storing the 17 bit PLL counter value, you are storing a particular frequency, and each step in the counter steps by the desired channel spacing.

4. Kenwood TK-931

4.1. Schematic examination

An examination of the TK-931 shows that the basic radio architecture is the same as the TM-441. The PLL is changed by a serial bit stream, and the PLL also has a 12.8 Mhz reference oscillator. 12.8 Mhz / 2048 = 6.25 kHz, and 6.25 x 2 = 12.5 kHz, so if you compute 12.8 Mhz / 1024 = 12.5 kHz, which is the 900 business band channel separation. We must be getting close.

4.2. 93CS66 serial EEPROM

The TK-931 has a 93CS66 4096 bit, or 512 byte EEPROM. The KPG5D programming software makes a data file of 560 bytes, enough for the 512 bytes and some other extra info about the file itself.

4.3. KPG5D programming software

If you save a programming file with the test frequencies set to certain values, and then change the test frequencies and resave to a different file, you can then compare the two programming files that you have.

What you find is the location in the file of the test frequency data, and furthermore you find the proportional relationship constant between the stored value and the actual frequency. For example, if you start test frequency 1 with 935.0125 Mhz and change to 935.0250 Mhz, the values in the programming file at address 0x01A are hex 0x9801 and 9802 respectively. You can start to build a translation table for values to frequency:

hex	decimal	freq
0x9801	38913	935.0125 Mhz
0x9802	38914	935.0250 Mhz

Here we find evidence that a single step in the stored value corresponds to a single channel spacing increment.

4.4. The programming equation

Continuing with other frequencies, we hypothesize an equation:

F = N x .0125 + c, where
N is our stored EEPROM value
F is the transmit frequency
c is some value that offsets the calculation to the desired frequency (we find it to be 448.6 Mhz)

This is similar to our TM-441 PLL equation, but there is a constant that is always added here. Oh well, we're up in the 900 Mhz band, so why not?

4.5. Searching for the repeater shift

Now we see that the stored value sort of represents a 12.5 kHz step. We'll take a long shot and see what 39 Mhz translates to: 39 Mhz / 12500 = 3120, or 0x0C30 hex. We'll take a longer shot and go back to our programming files and search for a 0x0C30. We find it at address 0x00E.

If we take our 93CS66 and modify address 0x00E from 0x0C30 to 0x07D0 (0x07D0 = 2000, and 2000 x 12500 = 25 Mhz), we find that the radio is now transmitting 25 Mhz down from receive in non talk around, but the receive frequency has moved down by 14 Mhz (39 - 25 = 14) from the programmed frequency.

I've currently just programmed the receive frequency 14 Mhz higher than I actually want and everything works out. However, I believe this can be cured with some more work (my friend took his radio back once the thing started working!).

4.6. Curing the 14 Mhz problem

I have yet to change some value near the repeater shift value by 14 Mhz / 12500 = 1120 in the EEPROM. I suspect that once this is done the 14 Mhz quirk will disappear and the radio will work as programmed.

4.7. Programming the amateur frequencies

You can find the repeater frequency locations by comparing two stored files as before. Once the 14 Mhz quirk is fixed, the EEPROM value will be

    (F - 448.0) / .0125, where
    F is in Mhz

For now I've just programmed the receive frequency to be 14 Mhz above the actual frequency.

5. Summary

The TK-931's serial EEPROM can be modified to make the radio transmit in the 902 amateur band with the desired 25 Mhz shift for California. Other repeater shifts are possible by adjusting the values above. I have yet to address the 14 Mhz offset.

By the way, I don't know where the EEPROM's repeater shift is getting set to the 39 Mhz value. If you try to program a doctored up programming file, the shift gets set back to 39 Mhz. This is why I have been using a device programmer for my editing. I suspect the value is being set from the KPG5D software, so I need to make a fake uploading program that just spits the raw data to the radio without changing it.