Those Strange Motorola Type II and Ericson EDACS Talkgroups

By David L. Wilson

Scanner manufacturers have spent a good deal of manual space explaining Motorola Type I trunk systems so that the scanner user can set up his scanner to understand such systems and scan the talkgroups of interest. Unfortunately, not only are Motorola Type II and Ericsson EDACS trunks systems only briefly described, the scanners themselves do not often indicate the most useful talkgroup designators (in the case of Motorola Type II) or even correctly identified designators (in the case of EDACS). For the casual listener, this information may not be of much interest; but for those interested in figuring out trunk systems, it is essential.

❖ Motorola Type II

Motorola Type II talk group numbers have mystified many scanner hobbyists for some time. Some may know they come from a set of twelve binary bits, of which the last four status bits indicate special purposes. The last bit before these in a Type II talk group is usually a 1, which is used to indicate talkgroup priority. The first three bits indicate which of the 7 blocks the talkgroup is in, for allocating that block as either Type I or Type II. But still, some mystery remains as to the rationale behind talkgroup assignments in each particular situation.

A scanner or scanning reference might show a Motorola Type II as 7D3 in hexadecimal or as 32048 in decimal. Some sources also would identify this talk group by its supposed Motorola designator 802003 (from 32048 divided by16 plus 800000). This talkgroup is the first of a series of used talkgroups. But why did the developer start with 7D3?

Unfortunately, none of these designators give an indication of the thinking of the system designer. However, as will be seen in the following example, there is another number for this talkgroup that *does* show the thinking that led to this talkgroup assignment.

If starting with the four hexadecimal designator (example, 7D30):

1. Drop the last character (7D3)
2. Convert the hexadecimal to binary (011111010011)
3. Drop the last bit (01111101001)
4. Convert the binary to decimal (1001 – it just happens that in this particular example results in a decimal that only has 0's and 1's).

Steps 2 and 4 can be done using the MS-Windows calculator by choosing the base clicking on the left side to choose hexadecimal (Hex), decimal (Dec), or binary (Bin), then entering the number in that base; and finally clicking on the left again to choose the desired new base.

Note that in the example, the result of 1001 is just 1 more than a thousand. The system designer decided to reserve the first 1000 talk groups for other purposes and started this set of active talk groups with 1001.

If one starts with the usual scanner decimal talkgroup designator instead (example, 32048), one does the following steps:

1. Divide by 16 and drop any remainder, as those are due to the final four status bits in the binary indicating a special purpose (2003)
2. Subtract 1 (2002)
3. Divide by 2 (1001 – the same answer we got when we converted 7D30 to decimal, so this is the same talk group).

Note that the first three possible talkgroups in a Type II system as displayed on a scanner are 16, 48, and 80, which now become 0, 1, and 2 – It now becomes evident these are the first three talkgroups in the system.

When one does the above, the first talk group in a cluster of active talkgroups is often found to be a number like 25, 26, 50, 51, 75, 76, 100, 101, etc. This is because the system developer divided up the talkgroups at the "nice" numbers of 25, 50, 75, 100 etc. to reserve space for other purposes or for expansion. Understanding this logic may aid one in deciphering the talkgroups of a Type II system.

❖ Ericsson EDACS

In the case of EDACS trunked systems, scanner manufacturers have generally allocated the EDACS AFS (Agency-Fleet-Subfleet) 11 bits as 4 agency bits (up to 16 agencies), 4 fleet bits (up to 16 fleets per agency), and 3 subfleet bits (up to 8 subfleets per fleet). This will be referred to below as AFS-4/4/3.

The problem is that systems may choose to allocate the 11 bits differently. Many systems allocated the 11 bits as 3 agency bits (up to 8 agencies), 4 fleet bits (up to 16 fleets per agency), and 4 subfleet bits (up to 16 subfleets per fleet). This will be referred to as AFS-3/4/4 in what follows. The impact of displaying as AFS 4/4/3 when the system is AFS-3/4/4 is that the AFS breakdown into agencies, fleets, and subfleets is incorrect and the logic of the designer of the system is no longer clear.

As an example, many EDACS systems on a scanner or in a scanning reference will indicate a busy talkgroup as 02-021. This is often the police or fire dispatcher. In AFS- 4/4/3, this would indicate agency 02, fleet 02, and subfleet 1. Note that 4 bits, being 0 to 15, will always be written as 2 decimal digits; and 3 bits, being 0 to 7, will always be written as 1 decimal digit. (Thus, for example, we wrote 02 instead of 2 for the agency number above, 02 instead of 2 for the fleet above, but 1 for the subfleet.) But then what happened to agency 01, fleet 01, and subfleet 1? We will soon discover.

Let us see what happens if we allocate the bits correctly as AFS-3/4/4.

1. First, break the AFS 4/4/3 designator into agency, fleet, and subfleet: 02-021 becomes 02-02-1
2. Now convert these to binary where 2 digits become 4 bits and 1 digit becomes 3 bits (use the MS Windows calculator if needed): 0010 0010 001
3. Now regroup the AFS-4/4/3 as AFS-3/4/4: 001 0001 0001
4. Convert the binary back to decimal (base 10) (again using the MS-Windows calculator if needed and using 1 decimal digit for 3 bits and 2 decimal digits for 4 bits): 1 01 01
5. Finally, as has been the tradition, place a dash after the agency and concatenate the fleet and subfleet: 1-0101

We have now discovered that our 02-021 is actually agency 1, fleet 1 and subfleet 1, not agency 02, fleet 02, as the 02-021 would lead us to think when it incorrectly grouped the bits between agency, fleet and subfleet.

Rewriting the talkgroup designators with the bits correctly allocated in AFS will often show the logic of the system's designer in setting up the various talkgroups in the system.

Finally, a quick look at information on the net may indicate that in some areas AFS-2/4/5, AFS-3/3/5, or AFS-3/5/3 are in use. It should be apparent how to modify Step 2 above to take care of these. If the agency, fleet and subfleet numbers seem strange for your system, it will be worthwhile to experiment to find the correct one to enable you to determine possible users of newly discovered talkgroups. An abundance of even numbered agency or fleet numbers is often an indication that the displayed AFS is not correct.

In conclusion, a little experimenting to determine the talkgroup number that the system designer used in designing a Motorola Type II system (or the correct agency-fleet-subfleet, in the case of an Ericsson EDACS system) will aid the listener in determining the use of the talkgroups that he hears.

This is your equipment page. Monitoring Times pays for projects, reviews, radio theory and hardware topics. Contact Rachel Baughn, 7540 Hwy 64 West, Brasstown, NC 28902; email editor@monitoringtimes.com.