

CV29 Explained

By Scott Kurzawski

Before going on to the actual programming of CV29, let's go over the structure of CV29, since it has an entirely different meaning than all the other conventional CVs found on a standard mobile decoder. First, let's clarify that CV29 is same for all decoder manufacturers. It is part of the NMRA DCC decoder specification. Every CV is made up of a group of 8-bits. Think of a bit as a single-pole single-throw toggle switch, which has only two states. These states are either ON or OFF. In the case of conventional CVs, we didn't really care about the individual bits since we are just programming up a single numeric value, such as a loco's address. However, in the case of CV29, each bit has a unique meaning. Here is a listing of the bit meanings:

- Bit 0 ON [1] = Direction of operation is reversed
OFF [0] = Direction of operation is normal
- Bit 1 ON [1] = 28 Speed Step Mode (should always be enabled)
- Bit 2 ON [1] = Analog mode operation enabled
OFF [0] = Analog mode disabled
- Bit 4 ON [1] = Alternate Speed Curve Active
OFF [0] = Use table defined by CV 2, 5 and 6
- Bit 5 ON [1] = Use long address in CV17/18
OFF [0] = Use short address defined in CV1
- Bits 3, 6 and 7 are ignored by the DCC system firmware.

Now to transfer the state of all of the individual bits as a single numeric value is where the confusing part comes into play. What is required is a little binary mathematics to calculate the final numeric value but even that is pretty easy once you know how to do it. First, start by reading a bit string from **right to left**. As you move to each successive bit to the left, you will notice that the bit value doubles from the previous bit value.

Here's a table that defines each bit value:

	<u>Bit 7</u>	<u>Bit 6</u>	<u>Bit 5</u>	<u>Bit 4</u>	<u>Bit 3</u>	<u>Bit 2</u>	<u>Bit 1</u>	<u>Bit 0</u>
<u>Bit Value</u>	128	64	32	16	8	4	2	1

Now to figure out the numeric equivalent for bits that are ON, just add **ONLY** the respective values for the bits that are enabled [ON]. For example, if you wish to enable bits 1, 2 and 5, e.g. enable 28 speed step mode, analog operation and use long address, then to get the equivalent decimal value you would add $0 + 0 + 32 + 0 + 0 + 4 + 2 + 0 = 38$. As you can see when the bits are OFF, just place a zero in the formula for that bit. Therefore, in this example, CV29 would have a value of 38 programmed into it. Now the folks at NCE and System One made it easy for us by making each bit that we need to worry about in CV29 a Yes [**<1>**] or No [**<Enter>**] question, and so we just answer the questions in the SET CFG section and they take care of doing the binary math for us in the DCC system firmware.