

## Resource Mapping

So, you need a relay output instead of an open collector activation on decoding emergencies. Normally, an emergency message activates OUT1, which is an open collector transistor. If your application calls for relay contact closure, Resource Mapping will allow you to redirect the C Plus resources to accommodate unique needs.

### Using Resource Mapping:

Resource mapping allows you to map the output functions of OUT1, OUT2, KEY and MUTE between each other. If an individual wanted the mute relay to act like OUT1, they would enter `$$MAP OUT1 TO MUTE`. If it is desired to use OUT2 as the mute function, enter `$$MAP MUTE TO OUT2`.

The command structure is: `$$ MAP <function> TO <output>`

You can map one function to multiple outputs. The original <function> remains intact unless it is also mapped.

You can also map the C Plus input "CLEAR" to the function "CHBSY" (Channel Busy) by entering `$$MAP CHBSY TO CLEAR`. In this case, if the clear button is pressed, the display is cleared and the C Plus responds as though the input Channel Busy had been toggled. This will only be useful in modes that monitor Channel Busy like *Authorize* mode.

### Determining the condition of the Resources:

Once you have mapped a few outputs, you may lose track of just what goes where. Enter the command `$$MAP?` And the C Plus will output the following information:

<p><b>Output OUT1 is function OUT1</b></p> <p><b>Output OUT2 is function OUT2</b></p> <p><b>Output KEY is function KEY</b></p> <p><b>Output MUTE is function MUTE</b></p> <p><b>Function CHBSY is input CHBSY</b></p>
--

### See Also:

Communicating with the C Plus using Hyperterm

Channel Busy

External Alarm Activation

Mute Feature