



Global Caché Tech Tip

AMX IR Commands with the GC-100

When developing an AMX application, the developer will usually select IR codes from the AMX IREdit database and compile them into the application. Since AMX IR codes are not openly available to other vendors, the IR codes sent to a GC-100 Network Adapter must be learned in GC-100 format and included in the application as text strings. Codes are easily learned utilizing Global Caché's GC-IRL IR Learner or can be converted from known CCF files. Below is an example of how to define IR codes as strings and assign them as constants in NetLinx applications. It is recommend you place all of the IR codes in one file, and then include that file in the application project.

```
DEFINE_CONSTANT
// IR codes for GC Toshiba DVD player
char POWER_CMD[ ] =
'1,4,38000,1,1,343,170,21,64,21,21,21,64,21,21,21,21,21,21,21,64,21,21,21,
21,21,64,21,21,21,64,21,64,21,64,21,21,21,64,21,21,21,64,21,21,21,21,21,64
,21,21,21,21,21,21,21,64,21,21,21,64,21,64,21,21,21,64,21,64,21,64,21,1516
,343,85,21,3657,343,85,21,3818'
char PLAY_CMD[ ] =
'1,6,38000,1,1,343,171,21,63,21,21,21,63,21,21,21,21,21,21,21,63,21,21,21,
21,21,63,21,21,21,63,21,63,21,63,21,21,21,63,21,63,21,21,21,63,21,63,21,21,21,63,21,21,21,63
,21,21,21,21,21,21,21,63,21,21,21,63,21,21,21,63,21,63,21,63,21,1519
,343,86,21,3825'
char STOP_CMD[ ] =
'1,10,38000,1,1,343,171,21,64,21,20,21,64,21,20,21,20,21,20,21,64,21,20,21
,20,21,64,21,20,21,64,21,64,21,64,21,20,21,64,21,20,21,20,21,64,21,20,21,6
4,21,20,21,20,21,20,21,64,21,64,21,20,21,64,21,20,21,64,21,64,21,64,21,151
8,343,85,21,3663,343,86,21,3825'
```

The output of learned IR code from the GC-IRL is a comma delimited text string terminated with a carriage return. An example of a GC-100 IR command is given below.

```
sendir,4:1,4,38000,1,1,343,170,21,64,21,21,21,64,21,21,21,21,21,21,21,64,2
1,21,21,21,21,64,21,21,21,64,21,64,21,64,21,21,21,64,21,21,21,64,21,21,21,
21,21,64,21,21,21,21,21,21,64,21,21,21,64,21,64,21,21,21,64,21,64,21,64
,21,1516,343,85,21,3657,343,85,21,3818
```

The GC-100 IR command string begins with sendir, a module number, colon, and a connector number (4:1). The command and module numbers for the AMX driver module are not needed because they are filled in by the driver. Simply remove "sendir,4:" from the GC-IRL output and assign the result to a character string in the NetLinx application as shown above. Note the connector number is the first value in the text string included in the NetLinx application file. For more detailed information, please refer to "SEND_IR- command" in the "GlobalCache_GC100_InterfaceSpec" document included with the AMX GC-100 driver module.

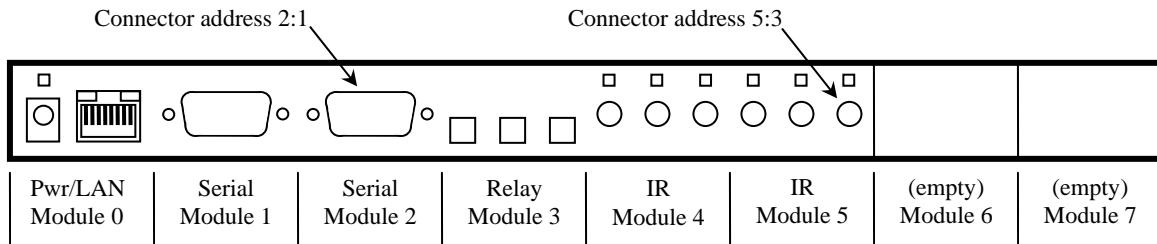
The GC-100's modular design provides for a variety of capabilities that can be combined within a single enclosure. Each module provides a particular function, such as infrared (IR), digital input, or relay closures. A module can support one or more connectors of the same type. For example, an IR module has three independent IR outputs; where as, a serial module has only one DB9 connector for serial data. The number of connectors a module can support is dictated by its 1.5 inch physical width.



Global Caché Tech Tip

A module's address is determined solely by its physical position within the GC-100 enclosure. Each module occupies 1.5 inches of front panel space, even if it's part of a larger printed circuit board containing other module types. At power on, module addresses are assigned starting with 0 for the left-most modules and increasing sequentially to the right until all module addresses are assigned (see GC-100-12 figure below). This concept presents a consistent programming interface as additional modules are added in the empty locations.

A connector's address is its position within a module, starting at 1 on the left and increasing to the right. A complete connector address includes the module address and the connector location within the module separated by a colon. See the GC-100-12 diagram below for examples of connector addresses. Note that a connector's address does not necessarily have to agree with the front panel label. Below, the IR connector at address 5:3 is labeled as 6 on the front panel of the GC-100-12.



GC-100-12

Sending an IR command is accomplished in the button event handler by calling the SEND-IR function in the driver support module. Below is an example of how the button event might process the POWER command.

```
BUTTON_EVENT[dvTP, nButtons]
{
    PUSH:
    {
        nIndex = get_last(nButtons)
        switch(nIndex)
        {
            case POWER_BTN:
            {
                send_command vdvDEVICE[IR_MODULE_ADDRESS], "'SEND_IR-
', POWER_CMD"
            }
        }
    }
}
```

For more information on using the NetLinx driver module with Global Caché GC-100 Network Adapters, please see the document "GlobalCache_GC100_InterfaceSpec" included with the NetLinx module.