v1.0 29-Dec-20

NetLinx Module Example

Description

This is a working example of a NetLinx module. The example is based on the VCR1 system call and demonstrates how to write a module to handle events. The main purpose of the example is to show correct module syntax.

The ability to reuse code is a desirable goal in software development; however, code reuse takes careful planning and organization. NetLinx provides tools such as functions and modules to promote reusability. Modules are NetLinx sub-programs designed to be "plugged into" a main program.

The module is basically a standard Netlinx program with MODULE_NAME instead of PROGRAM_NAME is the first line. The module also lists a series of parameters after its definition. The rest of the program looks like a standard NetLinx program. However, the name in MODULE_NAME and the file name must be the same. This is also the name used to call the module from the main program.

The module is then compiled into a <module name>.TKO and <module name>.TKN Sample code is file. The <module name>.TKO is an object file of the module. When a module is included in another program using the DEFINE_MODULE statement, the compiler creates a <program name>.TKO file of the code in that program only, then links the <program name>.TKO with the <module name>.TKO file to create a <program name>.TKN file. This file is downloaded to the master and contains the code from the main program and the code from the module. In order to compile, the <module name>.TKO and the <program name>.AXS must be in the same directory.

When adding modules to a NetLinx program, the DEFINE_MODULE definition can appear anywhere in the program. However, it is common practice to place DEFINE_MODULE before the DEFINE_EVENT section and after the DEFINE_START section.

Since the object file of the module is all this is needed for the final compile, modules can be used to protect any source code. Modules allow you to write a program which you can distribute in object (binary) format without giving away your source code.

Downloads: Sample Module code is in DL1 – ModuleExample.zip