

# Using your NanoCore12DXC32 USB QuickStart Package

## 1. Getting Started

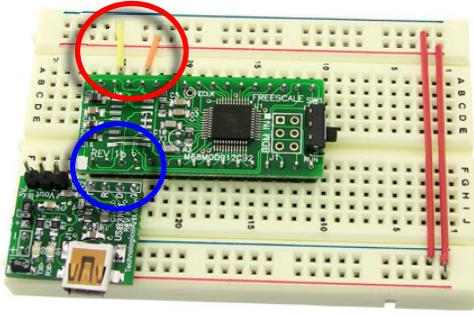


Figure 1

your breadboard, and wire it up as a Reset button by connecting one side of the switch to RESET\* (pin 30) and the other side of the button to Ground. Refer to Figure 2 for pin connection details.

### 1.2 Setting Up the Virtual COM port:

The USB2NCT module is based on the popular FTDI chip FT232RL. Usually, the operating system will recognize it when you attach it to a USB port on your computer. If not, you will need to download and install the appropriate driver from the FTDI website. Drivers are provided for various versions of Windows, MAC, and Linux. Download and install the correct version for your computer's configuration before proceeding. Then connect the supplied USB cable between a USB port on your computer and the USB2NCT plugged into your breadboard. If you're using Windows, it should report a new device has been found and proceed to initialize it. To determine which virtual comport it has been assigned to, right click the MyComputer icon on your Windows desktop, and select Properties from the popup menu.

Next, click on the Hardware tab, and click on the Device Manager button. Click on the + sign next to Ports (COM & LPT). The list of ports will be revealed. You will see one entitled USB Serial Port, with the assigned COM port number identified in brackets. To work with the tools we are going to use, the COM port must be 1, 2, 3, or 4. If it is higher than COM4, you'll need to change it. Do this by right-clicking the line "USB Serial Port" and select Properties from the popup menu. Click on the Port Settings tab, and then click the Advanced... Button. Select the COM port Number dropdown box and select a COM port number 2, 3, or 4 (usually COM1 is already in use). The port number may be reported as "in use" but you can disregard that at this point. Click OK and if Windows warns you about a duplicate, ignore the warning and click Yes. Click OK and then OK to the port settings. Then close the Device Manager and System Properties windows. At this point, you should have your hardware all set up and ready to test.

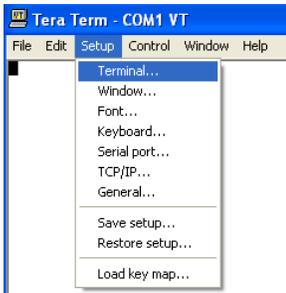


Figure 3

Each NanoCore12 module is shipped from the factory with a demo program loaded into Flash. To use the demo program, you will need to launch a terminal program on your PC. Some examples of terminal programs you can use are: HyperTerminal (included with Windows), TeraTerm, MiniIDE, and the terminal program inside ImageCraft's C compiler (ICC12 or ICC for CPU12). In this guide, we will use TeraTermPro, available on the Resources page of [www.NanoCore12.com](http://www.NanoCore12.com), or here: <http://support.technologicalarts.ca/docs/Third%20Party/TeraTermPro/Tera%20Term%20Pro%202.3.zip>

After unZIPPING and installing TeraTermPro, launch the program and select **Setup** from the drop-down menu. Select **Terminal** (see Figure 3) and click on the button next to **Term**

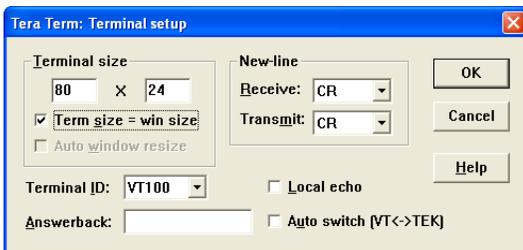


Figure 4

**size = win size**, then click OK (see Figure 4). Again from **Setup**, select **Serial Port** and select the com **Port** you are using, **9600** for **Baud rate**, **Data: 8 bit**, **Parity: none**, **Stop: 1 bit**, **Flow control: none** (see Figure 5). Now, make sure the **Load/Run** switch on NanoCore12DX is in the **RUN** position. You should immediately see a menu displayed in your terminal window (see Figure 6). If you don't see the menu, press the **RESET** button to momentarily bring pin 30 to logic low. If that doesn't work, re-read the above and verify your power supply voltage and connections.

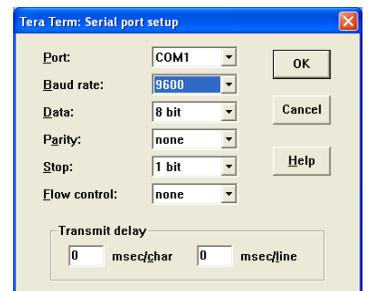


Figure 5

Now press the **Enter** key on your PC keyboard, and NanoCore12 will re-send the Demo Program Command Menu to your terminal window. If you got this far, it means that your module is successfully powered up and running, and communication is working in both directions (i.e. transmit and receive).

### 1.3 Using the Demo Program:

Now it's a good idea to attach a couple of LEDs to see the results of some of the Demo Program commands. Refer to Figure 7 for the connection details. The current-limiting resistor value should be no less than 680 Ohms, to avoid drawing too much current from the pin.

### 1.1 Hardware Setup:

Plug the USB2NCT module into your solderless breadboard so that the two downward-pointing power pins plug into the bus strips of your breadboard, but don't attach the USB cable yet. The pin marked + will supply the +5V bus, and the pin marked - is the Ground bus. Next, plug the NanoCore12DXC32ST module into the breadboard so that the first four pins (i.e. Pins 1 through 4) line up with the four communications pins of USB2NCT (see Figure 1). The USB2NCT module has a 3-pin power select jumper block. Place the jumper in the 5V position so that the USB port will supply 5V to your setup via the breadboard power buses. Connect a wire from the 5V bus to Vcc (pin 29) of NanoCore12DX, and connect a wire from the Ground bus to the Vss (Ground) pin (pin 31) of the module. Plug the supplied mini pushbutton into

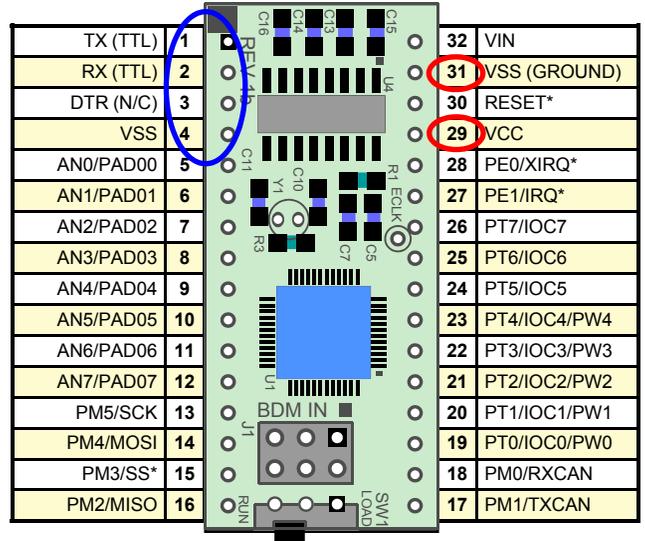


Figure 2

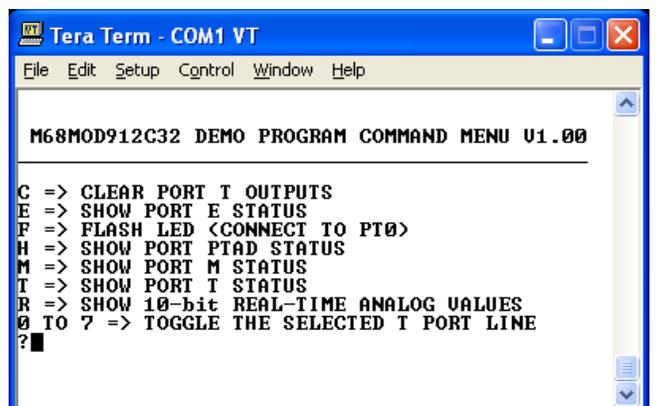


Figure 6

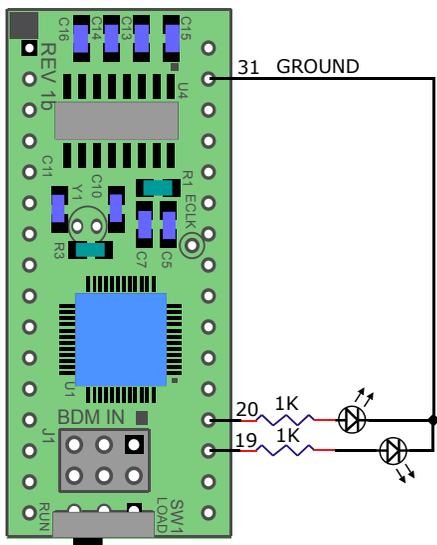


Figure 7 - LEDs on PT0 and PT1

- Type the digit 0 repeatedly and observe that the LED connected to PT0 (pin 19) toggles on and off.
- Type the digit 1 repeatedly and observe that the LED connected to PT1 (pin 20) toggles on and off.
- Type the letter F to cause the LED connected to PT0 (pin 19) to flash twice.

Typing E, H, M, or T will cause a hexadecimal number to be displayed, whose bit values represent the logic levels of the respective port pins. Those pins which are not connected to external circuitry (i.e. floating pins) will assume a logic 1 level, due to internal pullup resistors. You can connect any one or more of these pins to logic low (GROUND) and type the corresponding command to see how the displayed value changes.

#### 1.4 Loading a New Program Into Memory:

Whenever power is applied to the board, or whenever the Reset signal is brought to logic low and released, the microcontroller resets. The startup code in the Serial Monitor first examines the position of the LOAD/RUN switch to determine whether to run the monitor or to jump to the user program. If the switch is in the LOAD position, the Serial Monitor becomes active, waiting to receive commands via the com port. To use the Serial Monitor to send commands, you'll first need to install and launch uBug12.

##### 1.41 Using uBug12. Download uBug12 from

<http://support.technologicalarts.ca/docs/uBug12/uBug12.zip>, unZIP it, and install it. You can also download it from the link on the Resources page of [www.NanoCore12.com](http://www.NanoCore12.com).

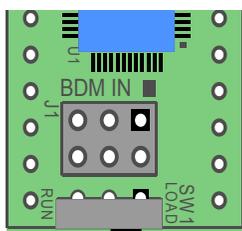


Figure 8 - Switch to Load Mode

1.42 Establishing a Serial Port Connection. Once you have launched uBug12, place the NanoCore12 switch in the **LOAD** position (see Figure 8) and reset your NanoCore12. Then click inside the command window of uBug12 (see Figure 9) and type the command `con 1` in order to establish communication between uBug12 and the NanoCore12 module. If you're using a com port other than 1, substitute the correct number (eg. type `con 3` if you are using com port 3). A **CONNECTED** message will appear in the uBug12 display window showing that the connection was successful.



Figure 9 - Command Window

1.43 Loading a Program. Loading a new program into Flash memory is a two-step process-- first you must erase any existing program from memory and then you load in your new program. Here's how to do it:

Type **fbulk** followed by <Enter>.

After erasing has finished, type **fload ;b** followed by <Enter>.

Refer to Figures 10 through 12. When the Windows file browser pops up, navigate to the file you want to load and click **OK**. After loading has finished, move switch SW2 to the **RUN** position and reset NanoCore12. If the program you just loaded uses the serial port (like the Demo Program, for example), you can just type the command **term** in uBug12 to launch a TeraTerm window (assuming you have previously installed it). To use a different terminal program, close uBug12 and open the terminal program of your choice.



Figure 10

For a list of all uBug12 commands, type **help** at the uBug12 command prompt. For more details on uBug12 and its commands, refer to the **ABCs of NanoCore12**, found on the Documentation page of [www.NanoCore12.com](http://www.NanoCore12.com).



Figure 11

1.44 Using the Sample Programs. Several simple programs have been created to get you started. They can be found on the Documentation page of [www.NanoCore12.com](http://www.NanoCore12.com), or here: <http://support.technologicalarts.ca/docs/NanoCore12/Code/SBASIC/Simple/>. Follow the steps outlined in 1.43 to load and run them one by one. Here are the names of the simple programs and a brief description of what they do:

- simple1:** turn on an LED connected to PT0
- simple2:** flash an LED connected to PT0
- simple3:** alternately flash LEDs connected to PT0 and PT1
- simple4:** send the message *Hello World!* out the serial port
- simple5:** combines the above into a single menu-driven program

1.45 Going Further. For troubleshooting help, or to start writing your own programs for NanoCore12DXC32, visit the Documentation page of [www.NanoCore12.com](http://www.NanoCore12.com). Not only does it cover all hardware details of the NanoCore12 family and accessories, but it contains thorough tutorials and numerous examples for using Assembler, BASIC, and C.



Figure 12

**For a free object-based programming Language custom-made for NanoCore12, Visit [www.nqBASIC.com](http://www.nqBASIC.com)**

Visit the Documentation page on [www.NanoCore12.com](http://www.NanoCore12.com) for updates to this document and related material

