Using the Lego NXT with Labview

http://www.legoengineering.com/component/content/article/105

The Lego NXT

http://thenxtstep.blogspot.com/

32-bit ARM microcontroller - an Atmel AT91SAM7S256. Flash memory/file system (256 kB), RAM (64 kB) and the USB Device interface.

4 analog inputs for a variety of sensors, plus ability to interface to any analog-out device

3 PWM motor driver outputs with built-in encoders (1° resolution)

Bluetooth communications (NXT to NXT, NXT to computer)

LCD display and 4 front-panel buttons.

The Lego NXT

http://thenxtstep.blogspot.com/

Advantages:

1. Plug and play… it works. No soldering required.
2. Relatively cheap ($250 for LEGO MINDSTORMS Education NXT Base Set).
3. Mechanically open-ended.
4. Extensible (e.g. mindsensors.com)

Disadvantages:

1. Perception – It's a toy!
2. Programming – support for the NXT as a target for embedded Matlab is limited. Labview support is now good.
3. Not truly real-time.

The Lego NXT – an option

What's in a kit?

Three DC motors with built-in encoders.

Two "bump sensors" (switches).

A light intensity sensor.

An ultrasound sensor.

A microphone.

Loads of gears.

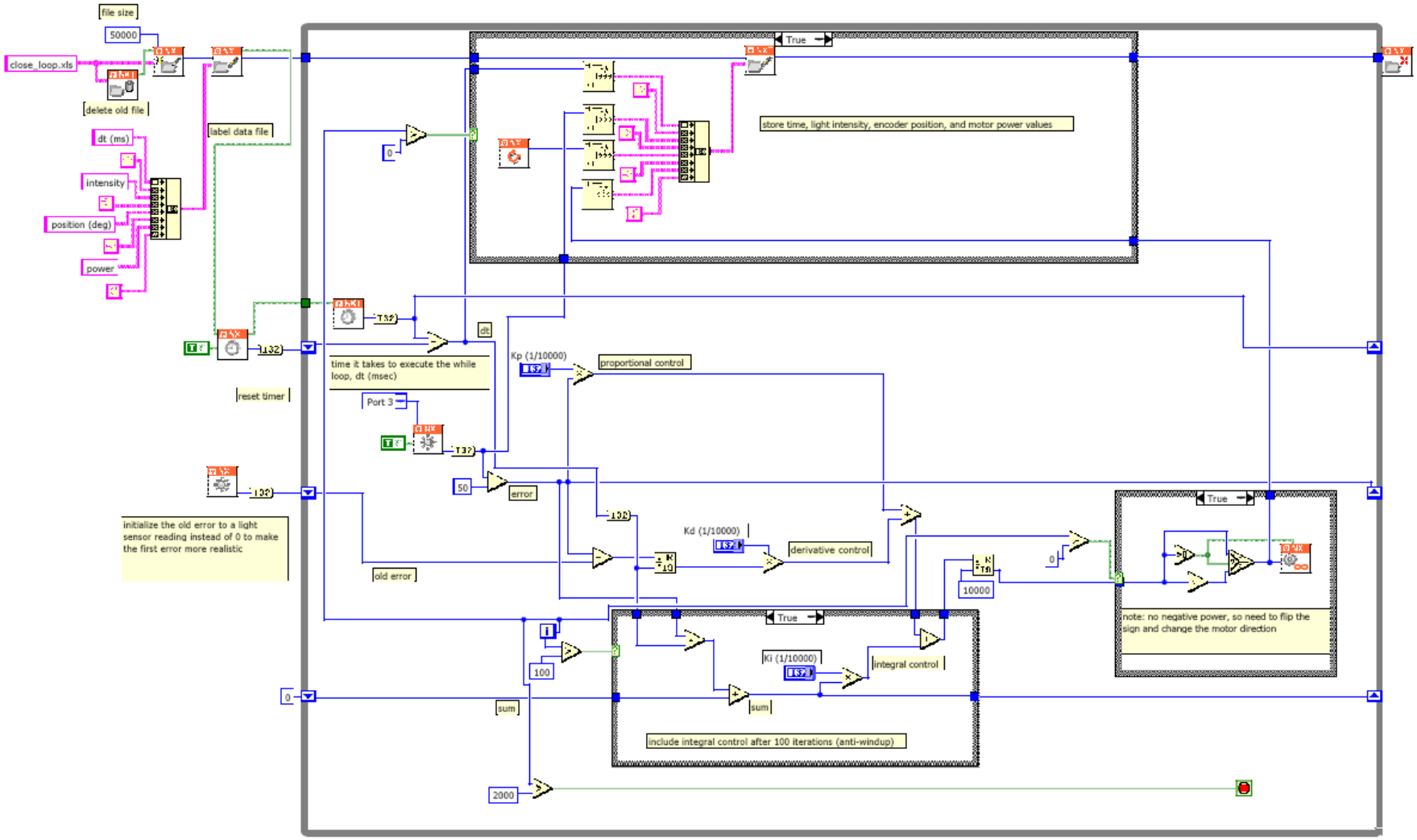
Charger, USB connection, battery pack.

Various lego parts.

Lego Mindstorms ®

Many other sensors and parts are available and interface cleanly.

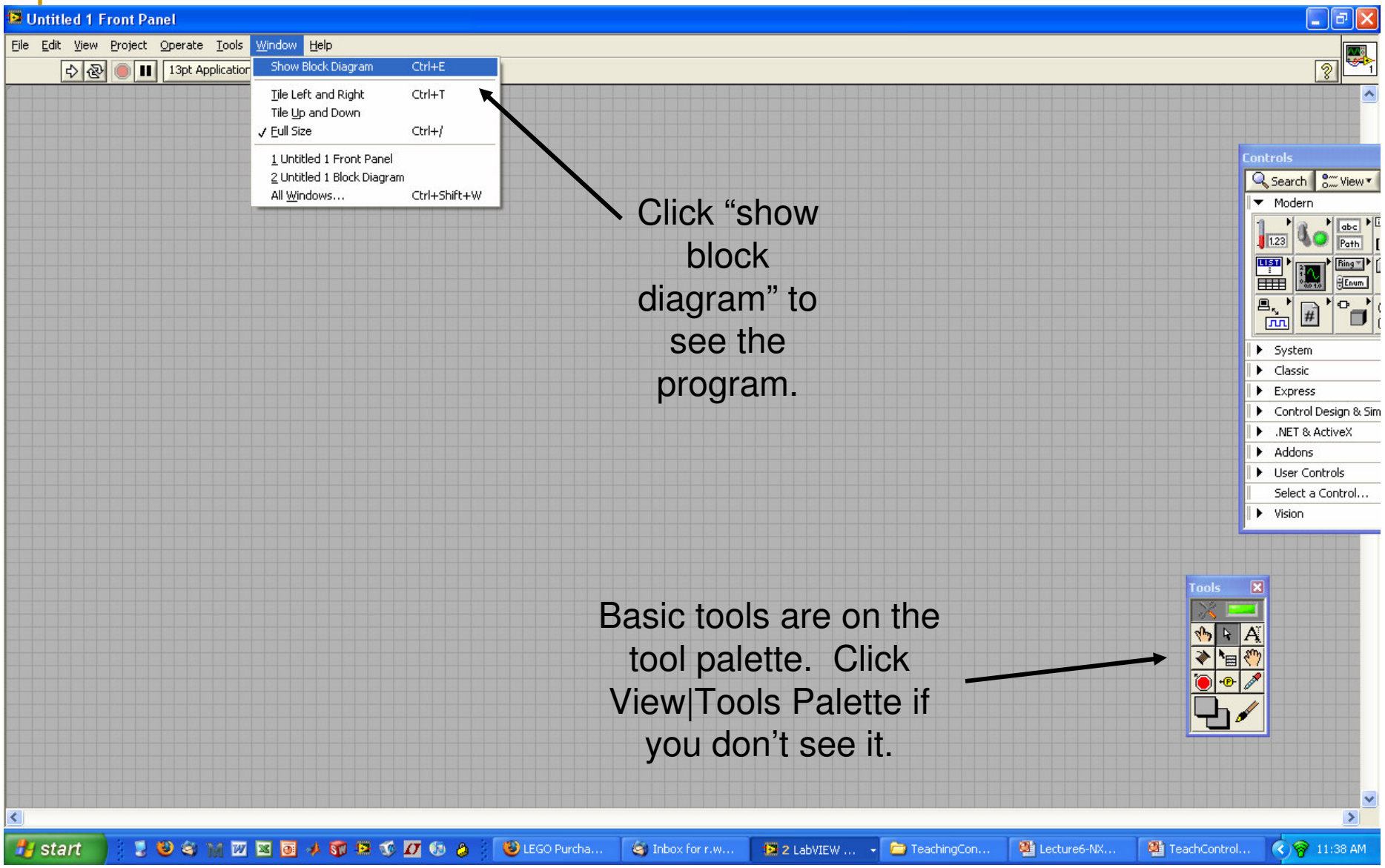
Example – LabView (PID with Antiwindup)

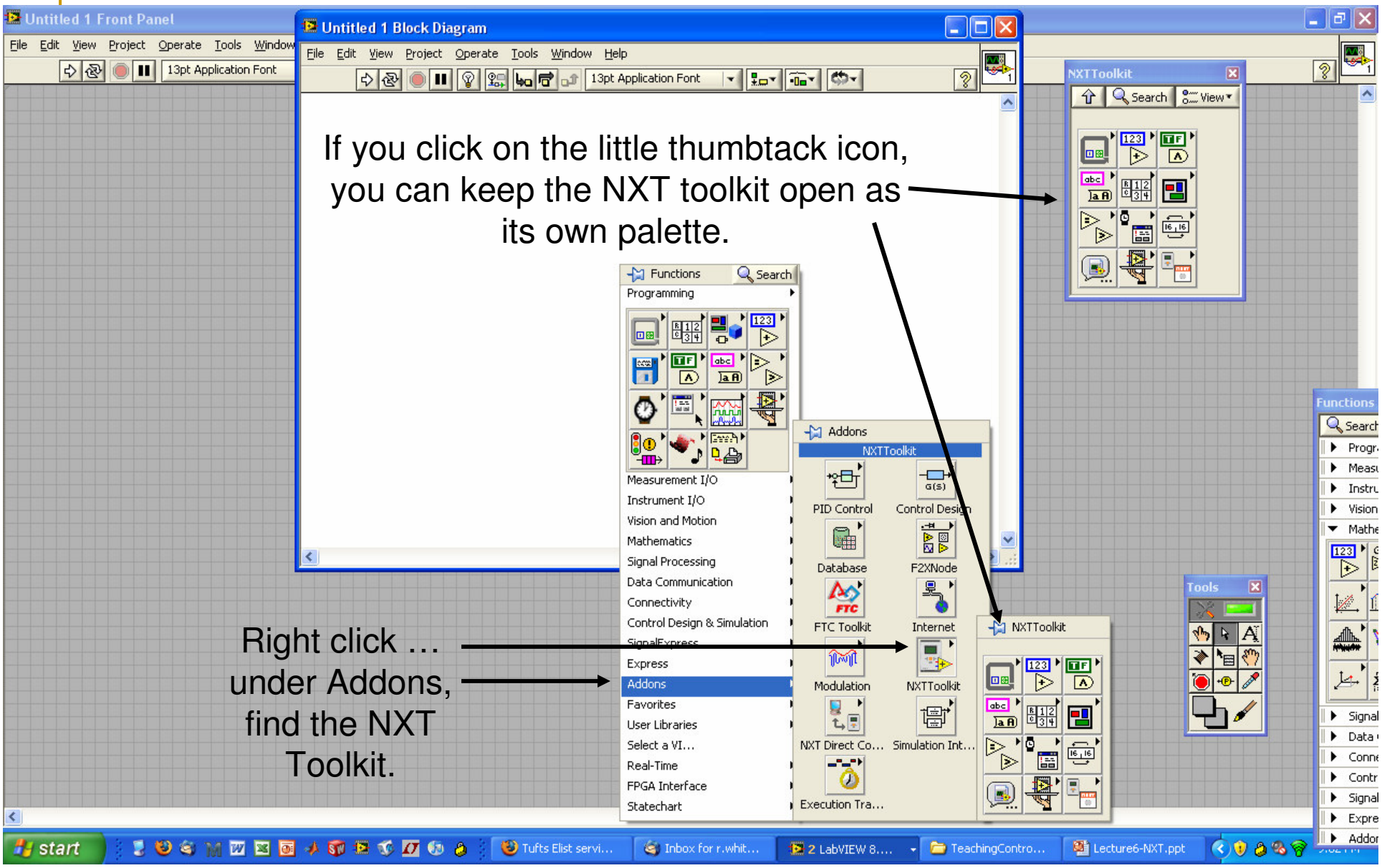


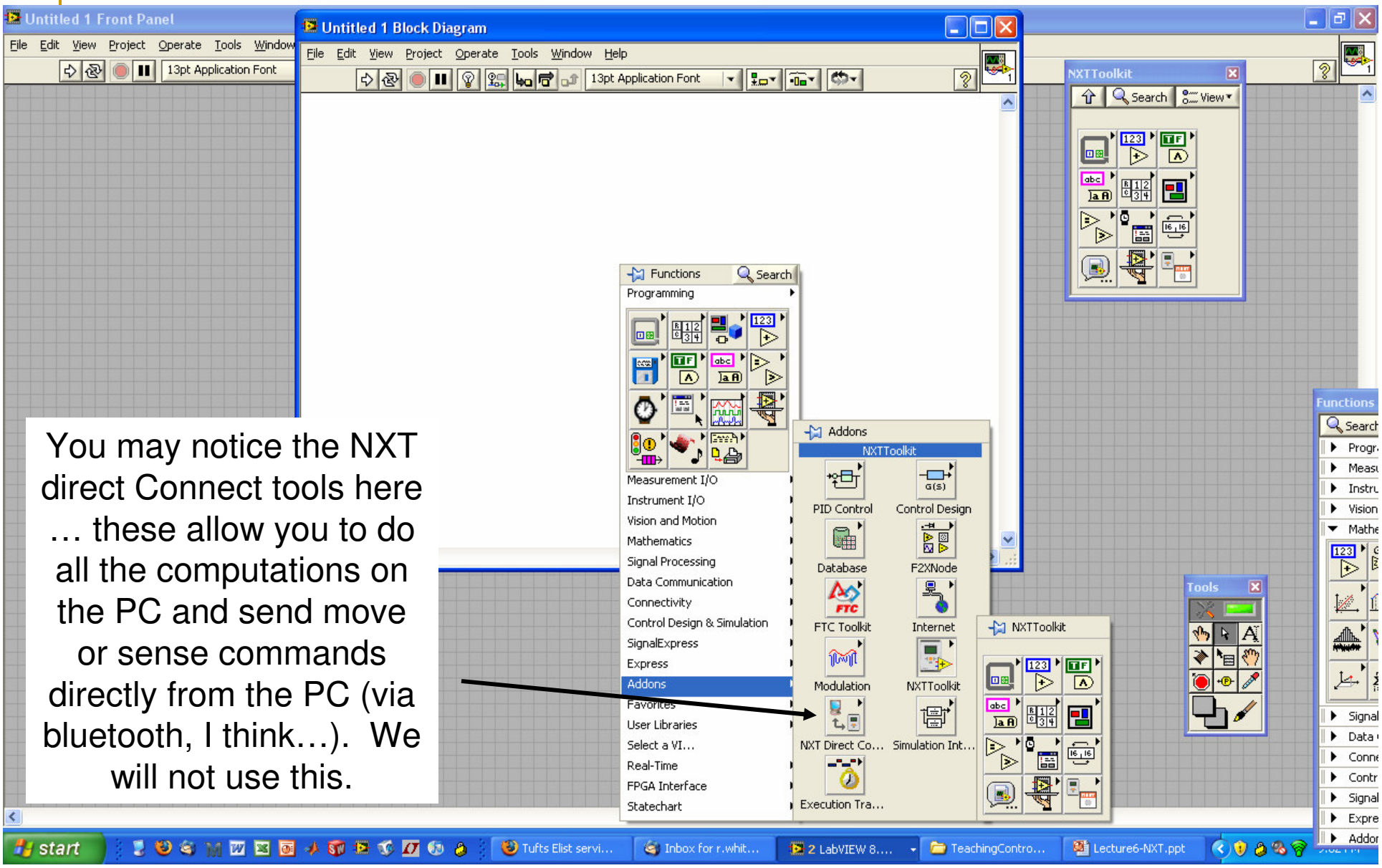
Example – RobotC (PID with Antiwindup)

```
17
18  const float kSetPoint = 30;  //setpoint (Light)
19  const int kDt= 1;  //Time step per loop (ms)
20  const int kTotalTime=2; //Total time to run for (seconds)
21  compileConst kNeedBytes = 10000; //8 byte header plus 3 bytes
22  const int kLogEvery=1; //log data every kLogEvery times throu
23  const int kAntiWindup=1.2; //Antiwindup... don't start integr
24
25  int index;
26  int nPower;
27  int fError;
28  int fLight = 0;
29  int fOldLight = 0;
30  int nCurrentTime = 0;
31  float fIntegrator = 0;
32  float fDeriv;
33
34  task main()
35  {
36
37    SetSensorType(S1, sensorLightActive);//Set up sensor 1 as a
38    CreateDatalog(kNeedBytes);//Set up datalogging
39    motor[motorA] = 0;
40
41    wait1Msec(100);
42
43    ClearTimer(T1); //Set T1 to zero
44
45    for (index = 0; nCurrentTime <= kTotalTime*1000; index++)
46    {
47
48      fOldLight=fLight;  //Remember sensor value from last time
49      fLight=(float) SensorValue[S1];
50
51      fError=kSetPoint-fLight;    //Compute current error
52
53      //Add to current integrator value
54      if (nCurrentTime>kAntiWindup)
55        fIntegrator+=fError*kDt*0.001;
56
```

```
57      if (index>0)
58        fDeriv=(fLight-fOldLight)/(kDt*0.001);
59      else
60        fDeriv=0;
61
62      //Compute power to send out (PID):
63      nPower=(int) fP*fError+fI*fIntegrator+fD*fDeriv;
64
65      //Saturation on Motor (probably not needed):
66
67      if (nPower>100)
68        nPower=100;
69      else  if (nPower<-100)
70        nPower=-100;
71
72      //Drive out motor power:
73      motor[motorA] = nPower;
74
75      nCurrentTime=time1[T1];
76
77      //Log data:
78      if (index%kLogEvery==0) {
79        AddToDatalog(nCurrentTime);
80        AddToDatalog(fLight);
81        AddToDatalog(nPower);
82      }
83
84      //Delay to get to kDt:
85      if ((kDt*(index+1)-nCurrentTime)>0)
86        wait1Msec(kDt*(index+1)-nCurrentTime);
87
88      nCurrentTime=time1[T1];
89
90    } //End control loop
91
92    //Turn off motor and save data to file
93    motor[motorA] = 0;
94    SaveNxtDatalog();
95
96  }//End program
```

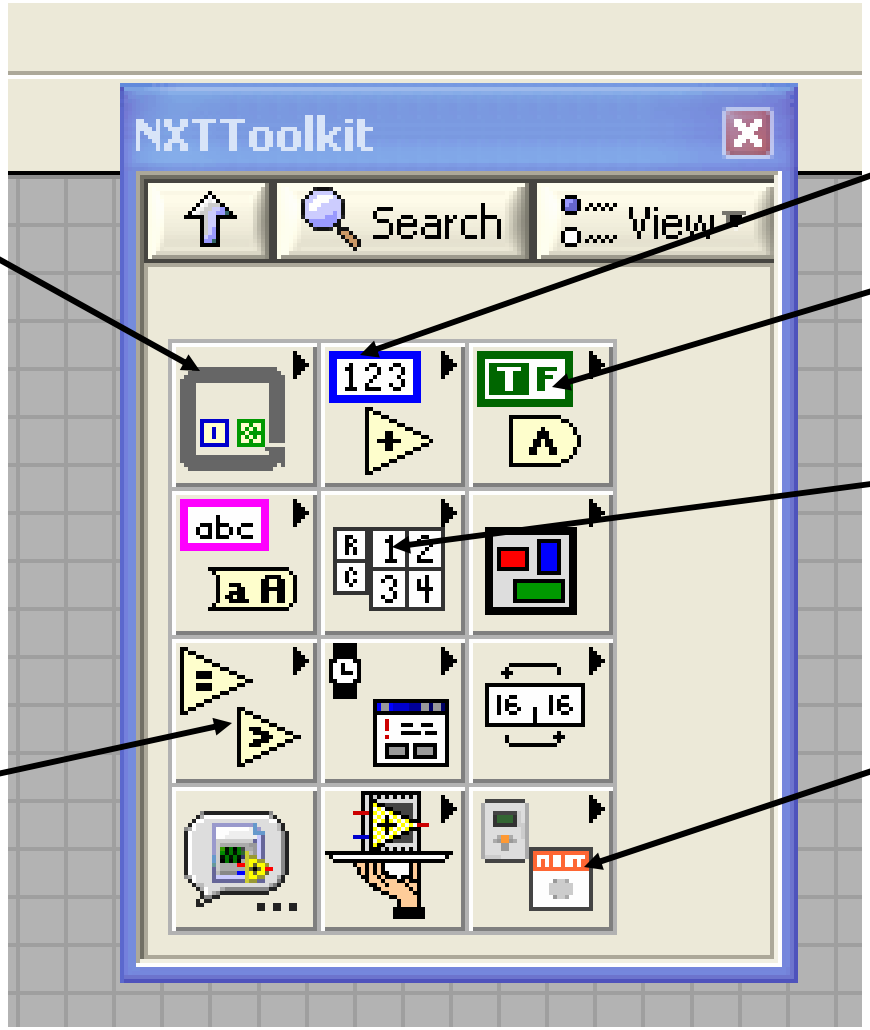

Untitled 1 Front Panel
File Edit View Project Operate Tools Window Help
13pt Application
Show Block Diagram Ctrl+E
Tile Left and Right Ctrl+T
Tile Up and Down
Full Size Ctrl+/
1 Untitled 1 Front Panel
2 Untitled 1 Block Diagram
All Windows... Ctrl+Shift+W
Click "show block diagram" to see the program.
Controls
Search View
Modern
System
Classic
Express
Control Design & Sim
.NET & ActiveX
Addons
User Controls
Select a Control...
Vision
Basic tools are on the tool palette. Click View|Tools Palette if you don't see it.
Tools
start
LEGO Purcha...
Inbox for r.w...
2 LabVIEW ...
TeachingCon...
Lecture6-NX...
TeachControl...
11:38 AM


Untitled 1 Front Panel
Untitled 1 Block Diagram
If you click on the little thumbtack icon, you can keep the NXT toolkit open as its own palette.
Right click … under Addons, find the NXT Toolkit.
NXTToolkit
Search
View
Functions
Programming
Measurement I/O
Instrument I/O
Vision and Motion
Mathematics
Signal Processing
Data Communication
Connectivity
Control Design & Simulation
SignalExpress
Express
Addons
Favorites
User Libraries
Select a VI...
Real-Time
FPGA Interface
Statechart
PID Control
Control Design
Database
F2XNode
FTC Toolkit
Internet
Modulation
NXTToolkit
NXT Direct Co...
Simulation Int...
Execution Tra...
Tools
start
Tufts Elist servi...
Inbox for r.whit...
2 LabVIEW 8....
TeachingContro...
Lecture6-NXT.ppt



You may notice the NXT direct Connect tools here … these allow you to do all the computations on the PC and send move or sense commands directly from the PC (via bluetooth, I think…). We will not use this.

Here is the NXT toolkit itself. These are the **only** commands you can use if you want the program to compile and download onto the NXT.



Loops and case structures

Math and logic

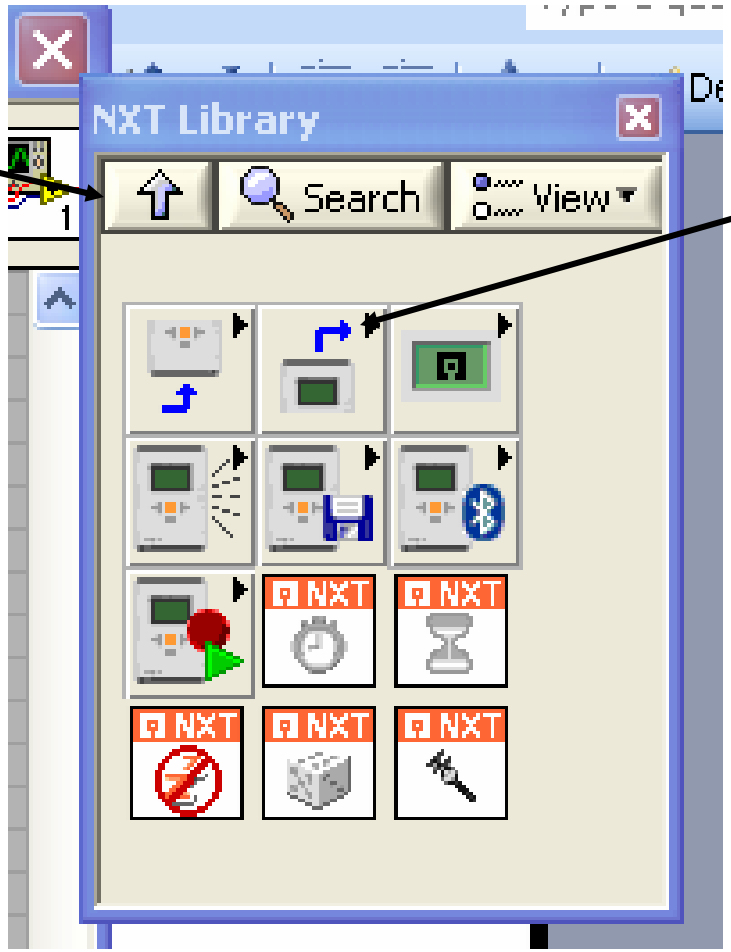
arrays

logicals

"NXT Library": Motor, sensor, onboard timer, NXT display, bluetooth, and NXT filesystem

Here is the “NXT Library” part of the NXT Toolkit

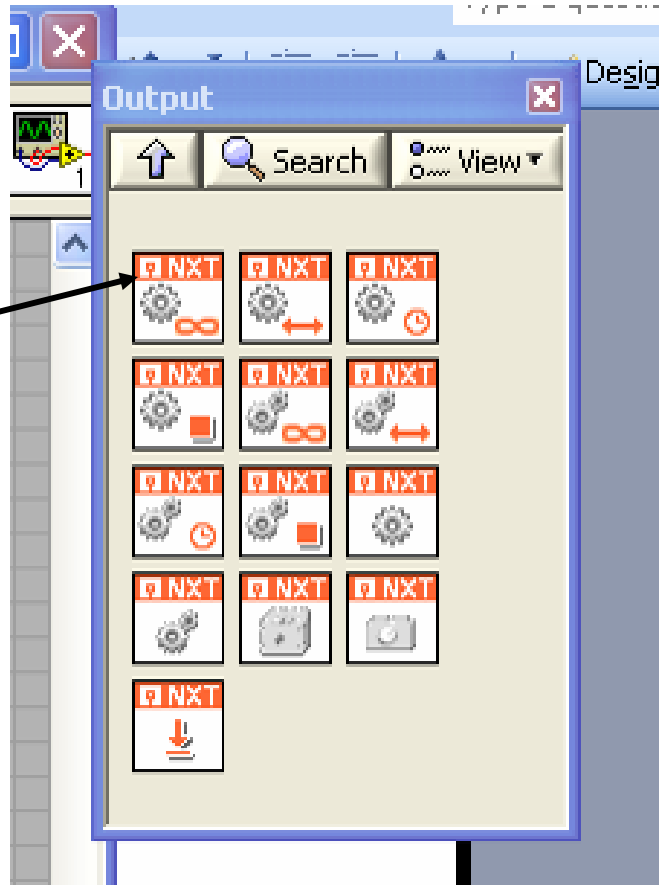
Click this arrow to go back up to the higher level palette



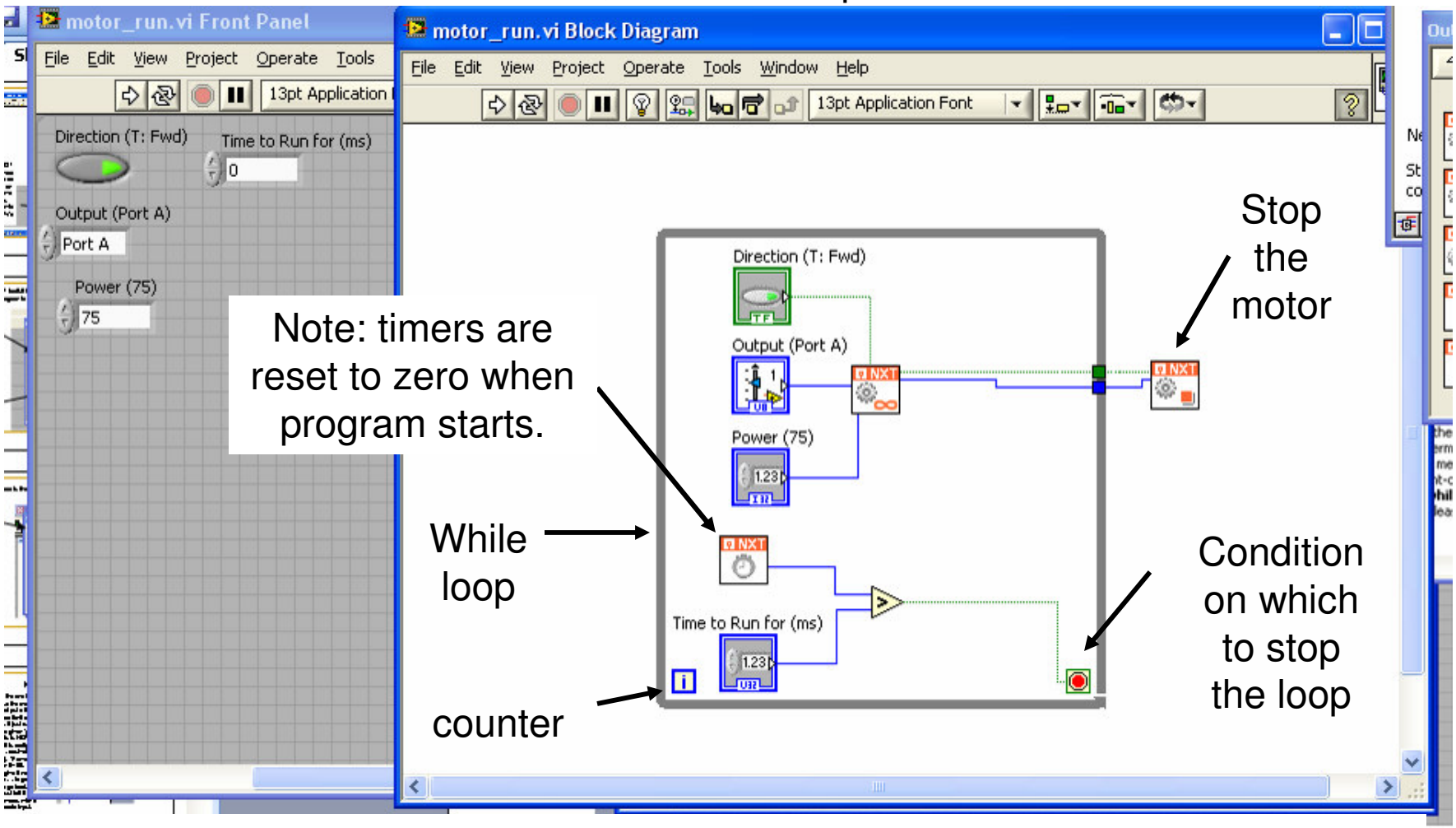
Let’s take a quick look at the motor commands

Motor commands

This is the one I want you to use the most … it does not put a time or distance limit on the motor … it just runs the motor at a given "power" level (this is NOT really power – it is really PWM % on … equivalent to voltage). "Power" can be from 0-100. You cannot give it negative power … to get the motor to go backwards you must tell it to go backwards with a separate input.

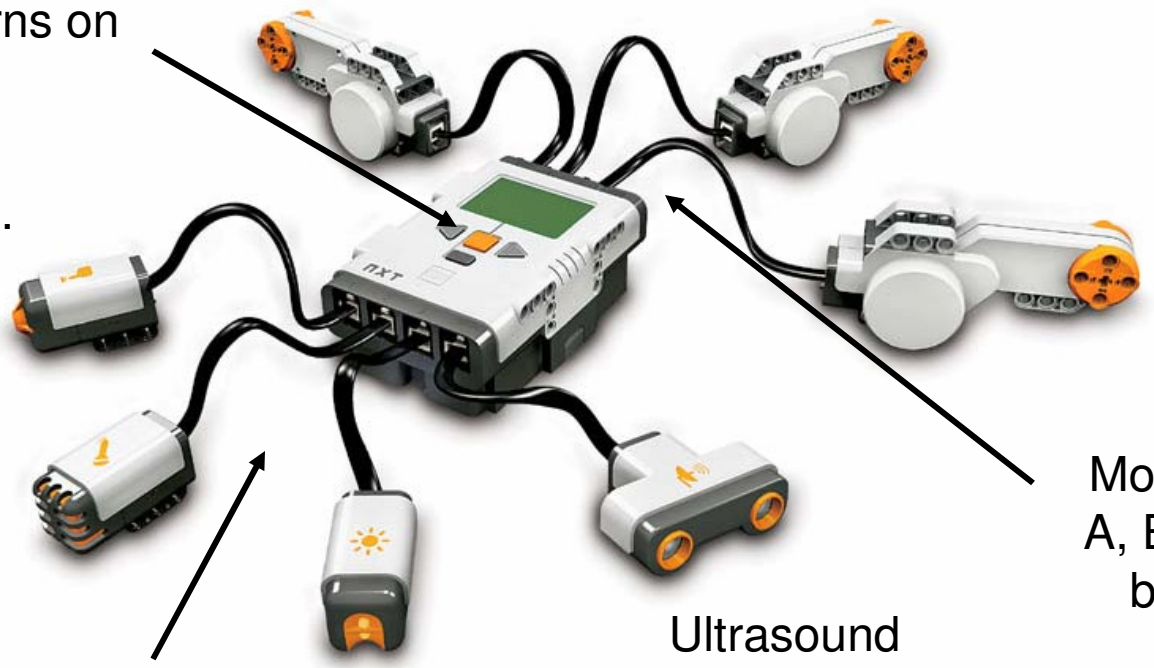


Here is a program that sets a motor output power level, waits for a certain time, and then stops the motor.

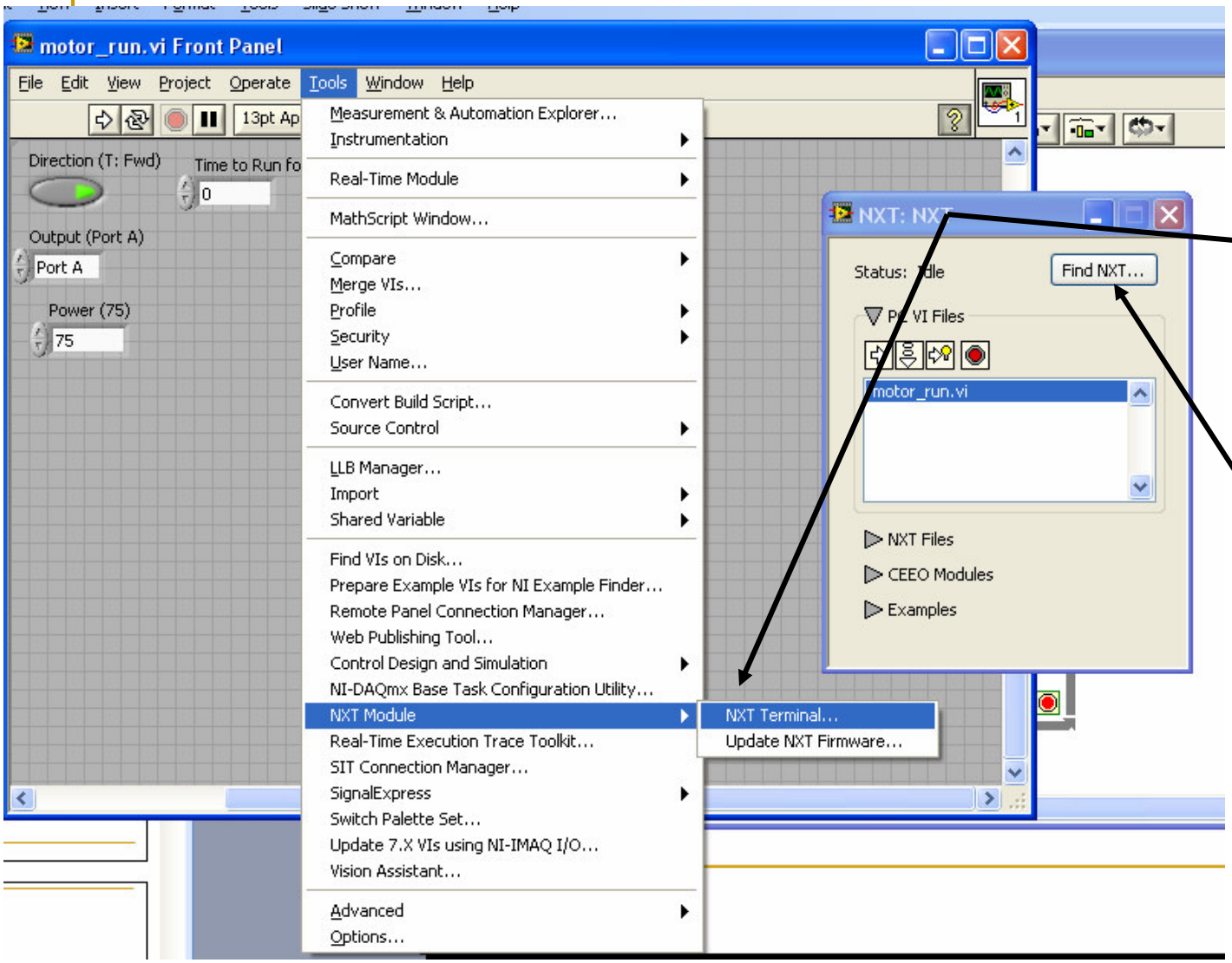


The value of front panel controls are taken at compile time. The order of operation of blocks is not well defined unless the output of one block is an input to another block … in that case, the block with the input must wait for the block with the output to finish before it will execute.

How to hook things up.



Compiling and downloading to the NXT.

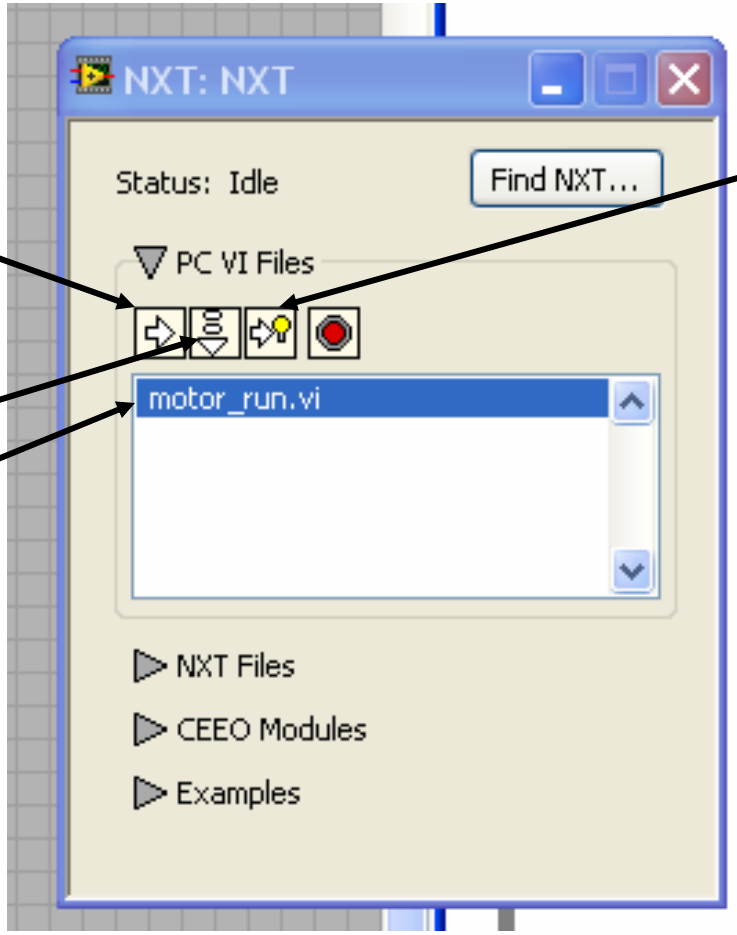


1. Plug in the USB cable and turn on the NXT.
2. Run “NXT Terminal” under Tools|NXT Module
3. Click “Find NXT…”… you should see your NXT including battery level

Compiling and downloading to the NXT.

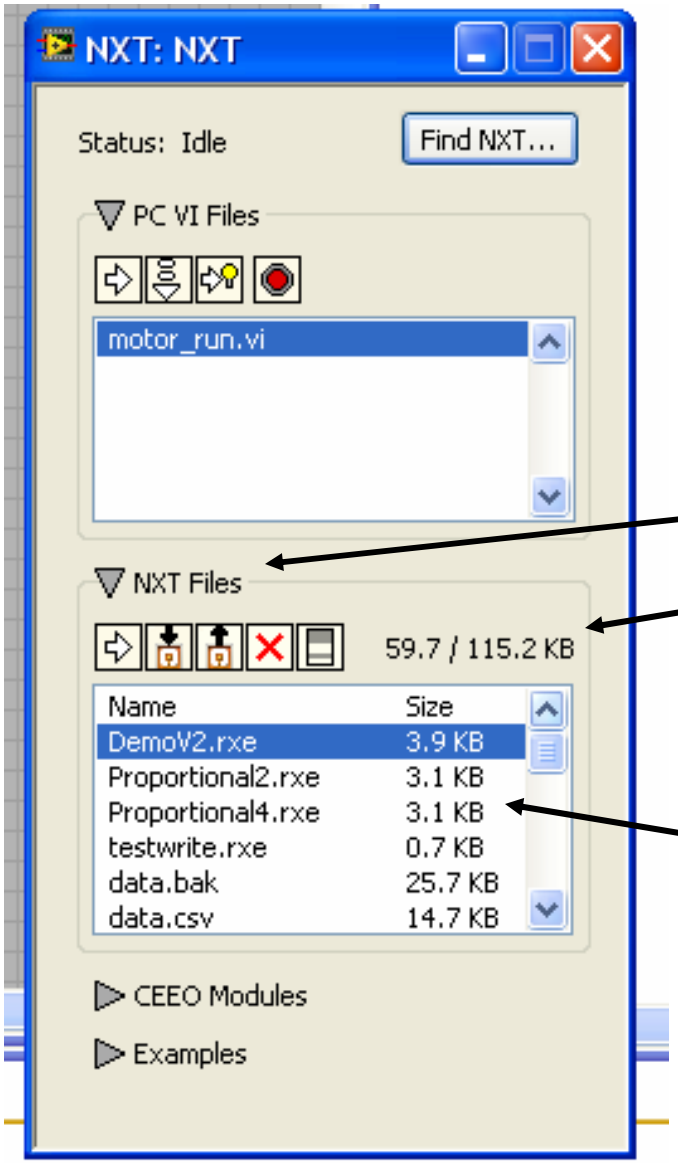
This arrow will compile, download, and immediately run the program.

This arrow will compile the selected program, download to the NXT.



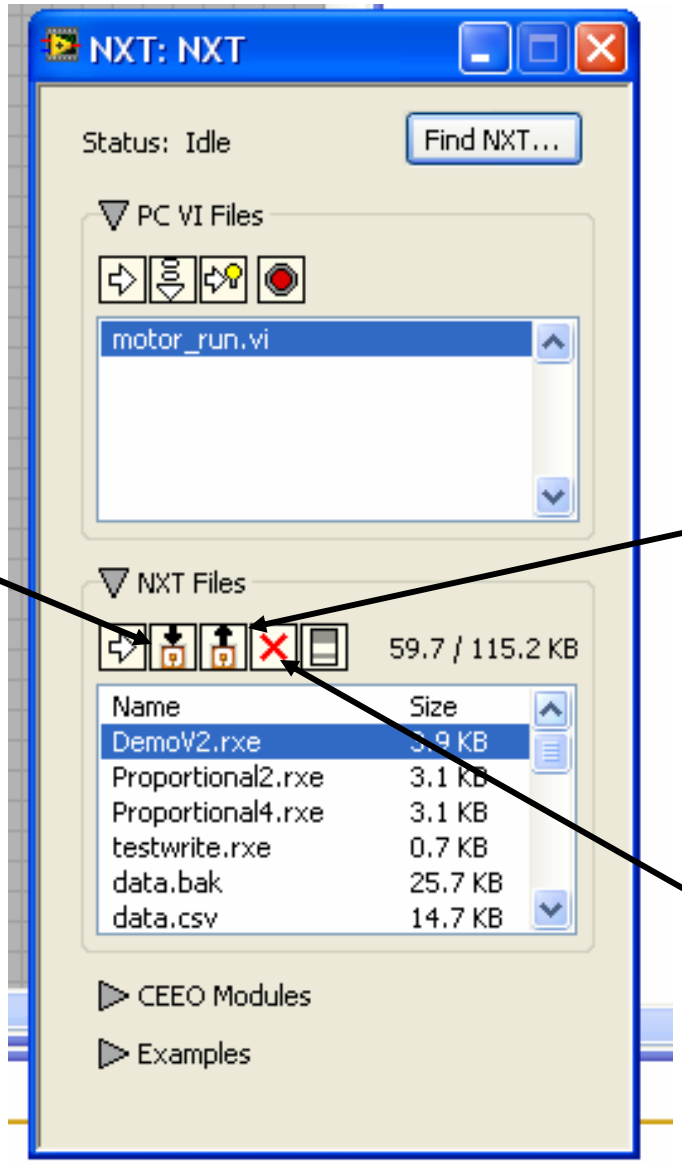
This arrow will run the program in debug mode … the NXT should remain connected to the computer, and you can send data back to the Labview front panel. Your program will run more slowly.

Once a program is downloaded to the NXT, it will appear in the list of programs accessible on the NXT front panel, and you can run it from the NXT itself without the computer connected.



If you open up the “NXT files” section, you will see all the files stored on the NXT. Don’t run out of space.

.rxe files are programs you have written and downloaded. .rtm files are examples, they can be deleted. .rso files are sound files, they can be deleted. .ric files are icon graphics (leave these alone). Do not delete NVConfig.sys



You can download files from the PC to the NXT

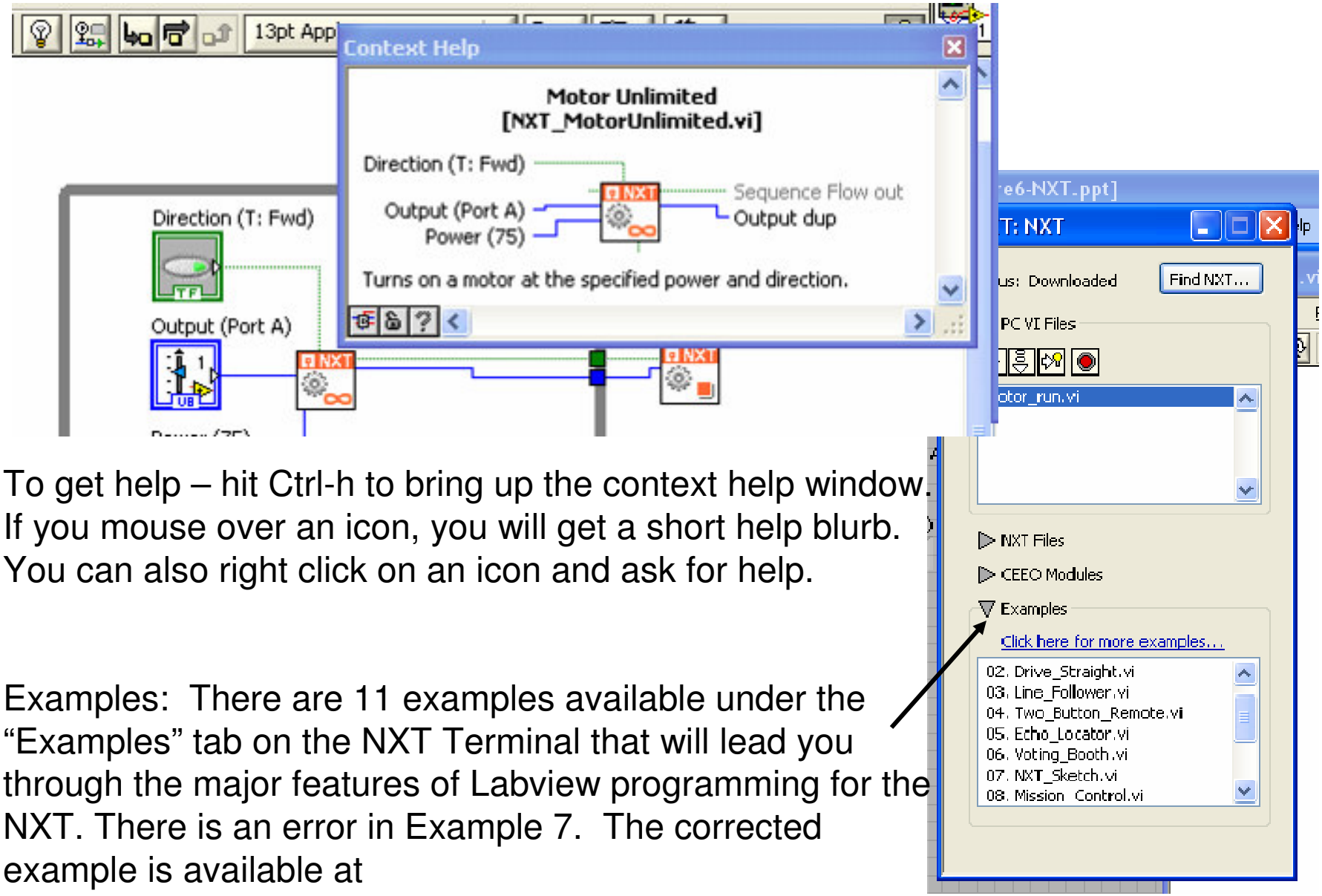
You can upload files from the NXT to the PC

Delete files on the NXT

There are file read/write icons inside the “NXT Library” which you can use for writing and reading data to datafiles on the NXT. Files on the NXT will stay when it is turned off.

In addition to the files stored on the NXT, you can (and should!) store the source code for your program as a Labview "VI" (a .vi file) on your PC.

To turn off the NXT, keep pushing this gray button until it says “Turn Off?”. Select the check mark icon and press the Orange button.



To get help – hit Ctrl-h to bring up the context help window. If you mouse over an icon, you will get a short help blurb. You can also right click on an icon and ask for help.

Examples: There are 11 examples available under the "Examples" tab on the NXT Terminal that will lead you through the major features of Labview programming for the NXT. There is an error in Example 7. The corrected example is available at

http://www.legoengineering.com/component/content/article/105

The first time you try to connect to or download a program, the software may tell you the NXT does not have the most recent firmware. There will be an option to download the most recent firmware. Do so. You should be running LEGO MINDSTORMS NXT FIRMWARE v1.21

OK – go! Make your NXT do something... Move a motor, read a sensor, write sensor values to a file, send info back to the NXT display or to the Labview front panel in debug mode, switch motor directions based on a sensor value … whatever … get that NXT working!!!