

The screenshot displays the Intelitek easyC for Vex controller interface. On the left is a component palette with categories like Inputs, Outputs, Program Flow, and RC Control. The main workspace shows a block-based flowchart for a program named 'BUMPERTEST.BDS'. The flowchart starts with 'I/O' and 'Variables' blocks, followed by 'BEGIN', two comment blocks, a 'WHILE' loop block, and an 'END' block. The code block within the while loop contains: bumper = GetDigitalInput [ 6 ]; and PrintToScreen [ "Bumper Switch". The right pane shows the corresponding C code:

```

1 #include "UserAPI.h"
2
3 int loop = 1;
4 int bumper;
5
6 void main ( void )
7 {
8     //Connect Bumper Switch to DI 6
9     //Switch Open = 1, Switch Closed = 0
10    while ( loop == 1 )
11    {
12        bumper = GetDigitalInput [ 6 ] ;
13        PrintToScreen ( "Bumper Switch = %d\n" , (int)bu
14    }
15 }

```

At the bottom, a status bar shows: Executing: "C:\Program Files\Intelitek\easyC\mcc18\bin\mplink.exe" /I"C:\Program Files\Intelitek\easyC\mcc18\lib" "C:\Program Files\Intelitek\easyC\WEX\Linker\18f8520user.lkr" "C:\Program Files\Intelitek\easyC\mcc18\bin\mplink.exe" BUILD SUCCEEDED: 30-07-2005 14:32:47. The bottom status bar also indicates: Ready, NUM Program size: 17808 Bytes, Line 1 of 15, 2:32 PM.

April 12, 2007

# Programming with easyC and WPILib

## Speakers

- ▶ Adam Martin

  - Software Engineer - intelitek
  - FRC and FVC team 40 team #9

- ▶ Brad Miller

  - WPI Robotic Resource Center
  - FRC team #190 and FVC team #37

## Why Use easyC and WPILib

- ▶ It makes programming easier – reduces the “barrier to entry” (marketing-eese for gives people access with less experience)
- ▶ easyC is unified environment with all aspects of FRC & FVC programming tied together in a single package.
- ▶ Your team focuses on your problem – not embedded processor programming
- ▶ You don’t have to reinvent the wheel – the base code is already there
- ▶ easyC Gives a common base between FRC and Vex platforms

## What You're Going to See Today

- ▶ Show how to set up and test motors
- ▶ How to write operator control programs
- ▶ How to write autonomous programs
- ▶ How to use the hard-to-use sensors
  - *encoder, gyro, CMU Camera, Interrupt Watcher*
- ▶ Learn more about using WPILib outside of the easyC environment
- ▶ Do all this right before your very eyes
  - ▶ Nothing up our sleeves
  - ▶ No pre-canned programs

## Audience Poll

- ▶ How many are interesting in learning more about programming for the Vex Robotics System?
- ▶ How many are interesting in learning more about programming for FRC?
- ▶ How many are novice programmers?
- ▶ How many are experienced programmers?

## Good News!

- ▶ This conference is applicable for everyone.

## Who is intelitek?

- ▶ Intelitek is a world-leading developer, producer and supplier of industrial CNC machines and technology training solutions.
- ▶ Our educational solutions are the ideal choice for Pre-Engineering programs, Automation Programs and Middle School and High School Technology Programs.
- ▶ Our educational and industrial product line covers subjects such as CAD, CAM, CNC, robotics, machine vision, FMS, CIM, hydraulics, pneumatics, PLC, sensors, process control and data acquisition. We also offer e-learning solutions designed to prepare students for careers in technologically advanced business environments.
- ▶ For over 20 years we have provided comprehensive solutions for training in engineering, mechatronics, automated manufacturing and industrial technologies.

## Who is intelitek? (continued)

- ▶ More than 10,500 CNC and 12,500 robots installed worldwide
  - ▶ PLTW
  - ▶ Skills USA
- ▶ More than 1,100 CNC machines to industrial accounts
  - ▶ World leader in industrial Benchtop VMC's
- ▶ 250 CIM installations worldwide
- ▶ More than 500 Labs worldwide
- ▶ Over **\$250 Million installed** in North America



# What is easyC?

Intelitek easyC for Vex controller

File Edit View Options Build & Download Window Help

Inputs

- Bumper Switch
- Light Sensor
- Limit Switch
- Line Follower
- Optical Shaft Encoder
- Ultrasonic Sensor

Outputs

- Motor Module
- Servo Module
- Digital Output

Program Flow

- IF - Else
- IF
- While Loop
- For Loop
- Timer
- Wait
- Assignment
- Print To Screen
- Comment
- User Code

RC Control

- Arcade - 2 motor
- Arcade - 4 motor
- Tank - 2 motor
- Tank - 4 motor
- Motor Module Rx
- Servo Module Rx
- Rx Input
- PWM Control

Function Blocks Project

BUMPERTEST.BDS

```

void main ( void )
{
  //Connect Bumper Switch to DI 6
  //Switch Open = 1, Switch Closed = 0
  while ( loop == 1 )
  {
    bumper = GetDigitalInput ( 6 );
    PrintToScreen ( "Bumper Switch
  }
}
    
```

1 #include "UserAPI.h"  
 2  
 3 int loop = 1;  
 4 int bumper;  
 5  
 6 void main ( void )  
 7 {  
 8 //Connect Bumper Switch to DI 6  
 9 //Switch Open = 1, Switch Closed = 0  
 10 while ( loop == 1 )  
 11 {  
 12 bumper = GetDigitalInput ( 6 );  
 13 PrintToScreen ( "Bumper Switch = %d\n" , (int)bu  
 14 }  
 15 }

Executing: "C:\Program Files\Intelitek\easyC\mcc18\bin\mmlink.exe" /I"C:\Program Files\Intelitek\easyC\mcc18\lib" "C:\Program Files\Intelitek\easyC\WEX\Linker\18f8520user.lkr" "C:\Program Files\Intelitek\easyC\WEX\Linker\18f8520user.lkr"  
 BUILD SUCCEEDED: 30-07-2005 14:32:47

Build Output

Ready NUM Program size: 17808 Bytes Line 1 of 15 2:32 PM

## easyC Development Timeline

- ▶ January 2005 – intelitek introduced to Vex
- ▶ April 2005 – Beta version of easyC demonstrated to RadioShack at FIRST Championship
- ▶ July 2005 - easyC version 1.0 developed for RadioShack for the Vex robotics system.
- ▶ September 2005 - Version 2.0 for Vex released for the 2006 FVC challenge
- ▶ Version 2.5 for FRC released for 2006 FRC kickoff.
- ▶ easyC Pro for FRC and Vex released for 2007 FRC kickoff.

# Robot Configuration

**Controller Configuration**

**v5 E X**  
ROBOTICS DESIGN SYSTEM

**INTERRUPTS**

#	Description
6	
5	
4	
3	
2	
1	Encoder

INTERRUPTS

**ANALOG / DIGITAL**

#	Description
16	
15	
14	
13	
12	
11	
10	
9	
8	
7	
6	
5	
4	
3	
2	
1	Gyro

**MOTORS**

#	Description
8	
7	
6	
5	
4	
3	
2	Right Motor
1	Left Motor

MOTORS

ANALOG / DIGITAL

Restore Defaults   Show Controller   OK   Cancel   Help

Left-Click to set Digital I/O  
Right-Click to set Analog Input

# On-line Window

**On-Line**

In order to use the On-Line control, the On-Line code must be downloaded to the Vex controller.

Download On-Line Code     Verify & Download

**Motors:**

#	Value	Set Group	Invert Direction
#1:	127	None	N/A
#2:	127	None	N/A
#3:	127	None	N/A
#4:	127	None	N/A
#5:	127	None	N/A
#6:	127	None	N/A
#7:	127	None	N/A
#8:	127	None	N/A

**I/O:**

Port:	Value:	Type:
#1:	<input type="checkbox"/>	Analog Input
#2:	<input type="checkbox"/>	Analog Input
#3:	<input type="checkbox"/>	Analog Input
#4:	<input type="checkbox"/>	Analog Input
#5:	<input type="checkbox"/>	Digital Input
#6:	<input type="checkbox"/>	Digital Input
#7:	<input type="checkbox"/>	Digital Input
#8:	<input type="checkbox"/>	Digital Input
#9:	<input type="checkbox"/>	Digital Input
#10:	<input type="checkbox"/>	Digital Input
#11:	<input type="checkbox"/>	Digital Output
#12:	<input type="checkbox"/>	Digital Output
#13:	<input type="checkbox"/>	Digital Output
#14:	<input type="checkbox"/>	Digital Output
#15:	<input type="checkbox"/>	Digital Output
#16:	<input type="checkbox"/>	Digital Output

**Interrupts:**

#	Description
1	Encoder
2	
3	
4	
5	
6	

Reset All

Close    Help    Save Descriptions

**INTERRUPTS**

#	Description
1	Encoder
2	
3	
4	
5	
6	

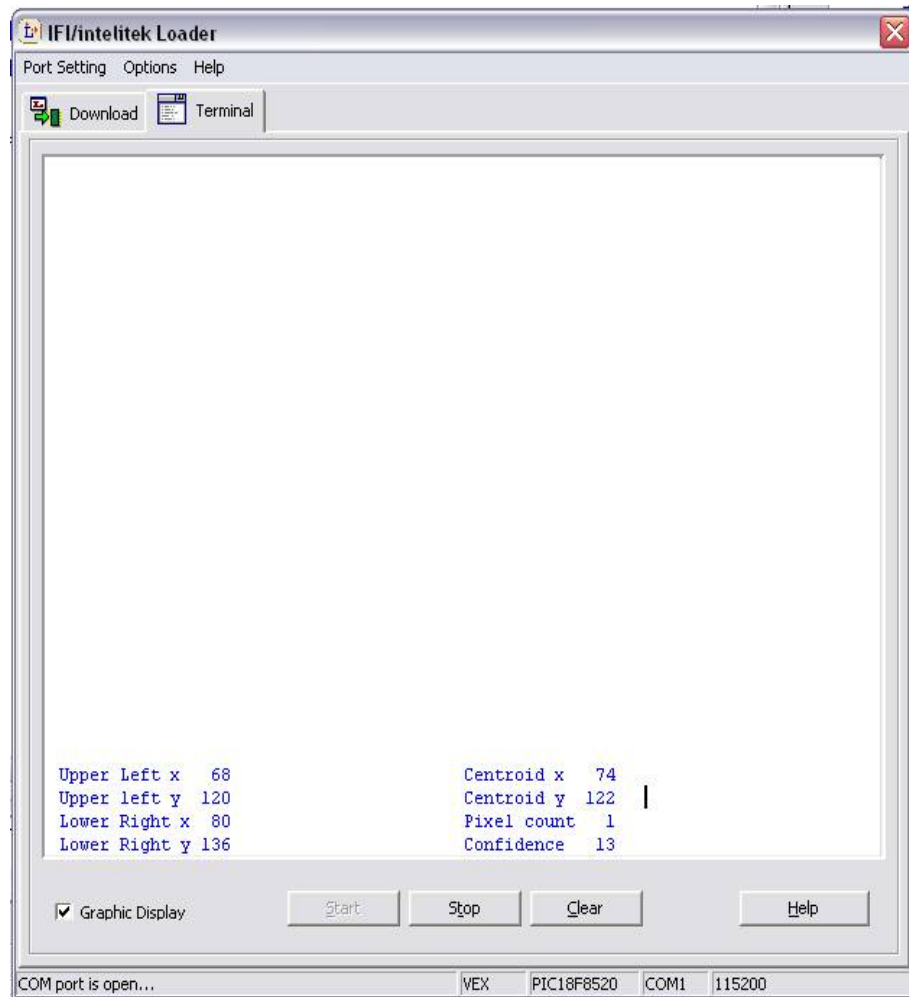
**MOTORS**

#	Description
1	Left Motor
2	Right Motor
3	
4	
5	
6	
7	
8	

**ANALOG / DIGITAL**

#	Description
1	Gyro
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
16	

# Graphics Display Window



## easyC Syntax Editor

- ▶ Intelitek

# Competition Project with Operator Control

- ▶ Intelitek

## Driving the Robot Autonomously

- ▶ Intelitek



## Using the Encoder

- ▶ Intelitek

## Driving Straight with the Gyro

- ▶ Intelitek

## CMU Camera

- ▶ Intelitek

## What's WPILib

- ▶ WPILib is the code that runs in the robot controller.

## WPILib Additions

- ▶ Additional devices
  - ▶ Serial ports, compass, and more
- ▶ Drive functions built-in
- ▶ Better timer support
  - ▶ More timer options
  - ▶ Timer interrupt routines
  - ▶ Repetitive timer notifications
- ▶ Extensibility
  - ▶ You can add your own devices
  - ▶ Exchange devices from others
- ▶ Upgrading without source code merging

## Summary

- ▶ Use easyC and WPILib to make your team more productive
  - ▶ Anyone can program the robot
  - ▶ You spend your time on your robot, not integrating code
  - ▶ Makes Vex a perfect learning tool for FRC Teams