Build The Ultimate Robot

Part 4

FireBot

By Michael Simpson

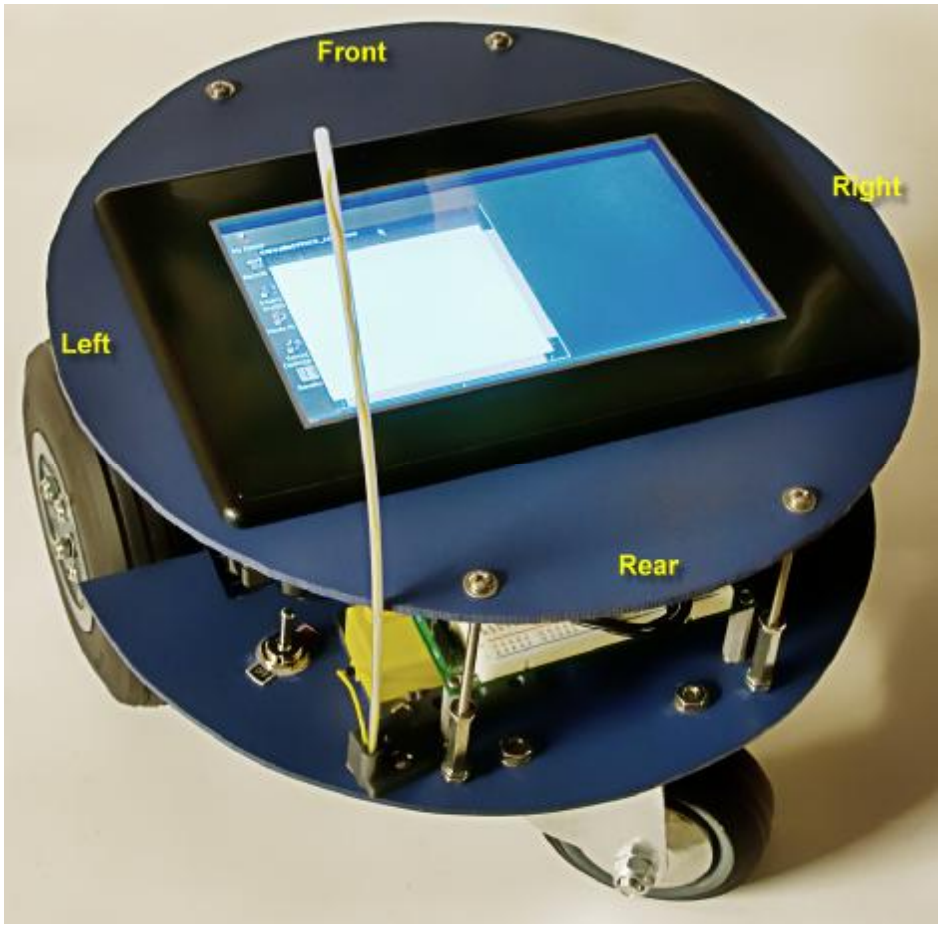
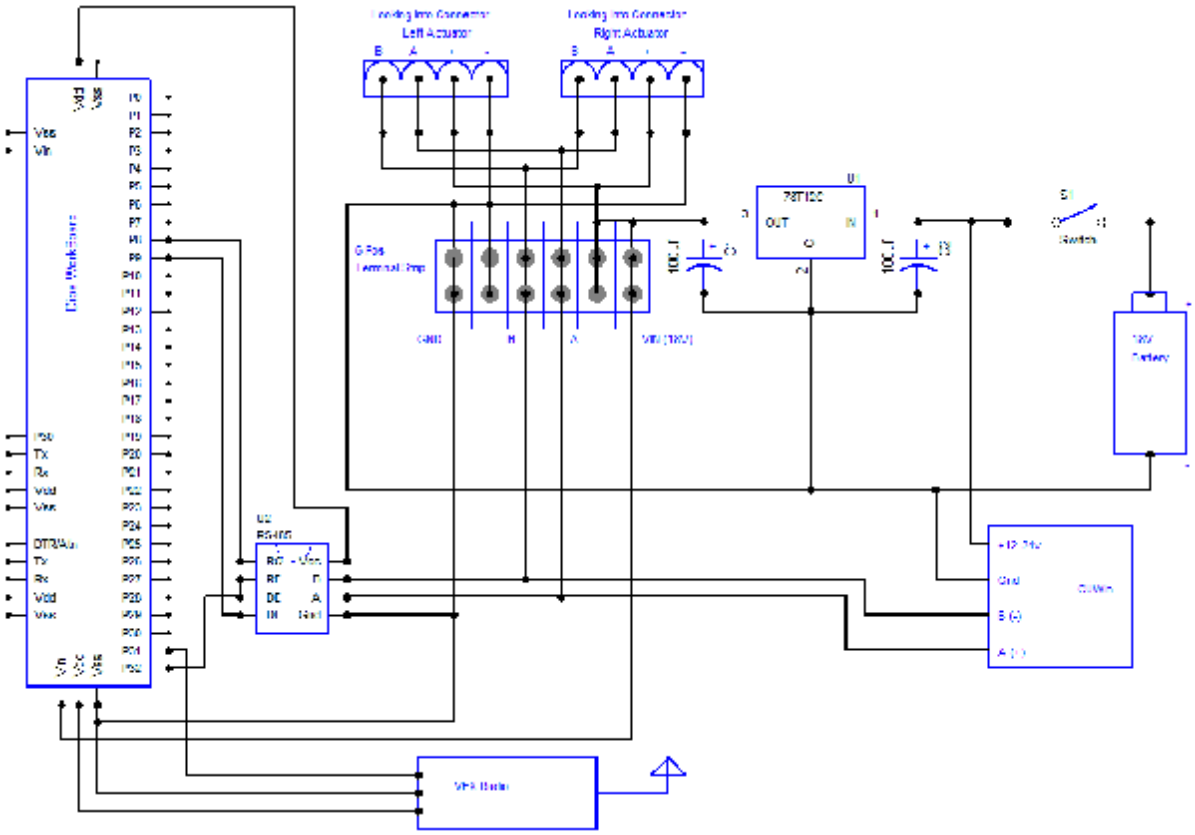


Figure 1

In the previous articles in this series we worked on both the Firebot and the Megabot. In this article we are going to finish up the Firebot shown in Figure 1. Before I get into

some of the construction techniques, I want to go over some of the components we will be using in this portion of the project.

Take a look at Schematic 1. While I will make every attempt to describe the actual connections in the Firebot, you may find it helpful to use the schematic for reference.



Schematic 1

The basic building blocks consist of the following:

- Ryobi 18v Li-Ion Battery Pack.
- Kronos Robotics DiosPro and Workboard.

- Vex Radio.
- CUWIN3500 CE based controller
- Firebot Base assembly completed in previous articles.

I will be going into detail on each one of the components as we assemble our Firebot. Let's start by looking at the power source.

Firebot Power Source

I looked at many different battery options when I started this project. One thing I decided early on was that I would be using some sort of Li-Ion battery. Li-Ion batteries give you the best power to weight ratio of any battery technology that is readily available to the public. For the Firebot I needed a 12-18v battery, but it had to be small enough to be contained within the base structure.

Last month when I was talking about tools I placed great emphasis on the portable Ryobi One+ system shown in Figure 2. I chose this system not only for its very reasonably priced tools, but also for its very efficient power system. After some extensive testing, the 18v Lion batteries have proven themselves as an excellent power source for robot projects.

Figure 2

The Ryobi 18v Li-Ion battery has a range of 16-20.5 volts and has a built in battery tester. For the larger RX-64 this isn't a problem, but is a little over the limit for the smaller RX-28 I am using for the Firebot. Looking at Schematic 1 you can see that I have added a simple 12v regulator circuit to drop the voltage down to more acceptable levels. In many of my early tests I did not use this regulator and while the RX-28's did not go up in smoke, they did get warm with normal use. For a faster Firebot leave off the regulator, but do so at your own risk. If you do decide to use a regulator I recommend the Jameco #871658. They are rated at 3A continuous and 5A peak. You will also need a TO220 heat sink, as they get very warm.

The greatest challenge in using the Ryobi battery is mounting it. The goal is to mount the battery so that it can be easily removed for charging or replacement. Fortunately, Ryobi has solved the mounting problem for us. They sell a very inexpensive flashlight shown in Figure 3. This little gem cost me $12.95 and worked out perfectly.

Figure 3

Remove the 6 screws and pull the flashlight apart and remove all the components. Just above the two center screw holes is a seam. This is where I cut the light in half. I used a band saw but just about any hand or hack saw would work as well. When finished, you should have two halves as shown in Figure 4.

Figure 4

Cut off the wires about an inch or so above the battery connector and insert it back into one of the halves, then using 5 screws reassemble the lower portion of the flashlight as shown in Figure 5. It's important that the red wire be located on the right side, as shown, so that it mates up with the positive terminal on the battery.

Figure 5

I then used the flashlight base to mark the Firebot base. I removed wheels, and using a jigsaw, cut out the mark. In order to hold the base in position you need to drill a small 5/64" hole in the base as shown in Figure 6. These will work as small pilot holes to attach some #4 machine screws. You will need about six holes around the base. Two in the front and two on each side. Insert the six screws, then slip the base into place.

Now you can mark the holes for six more on the underside of the base. With the battery holder in place insert the lower set of screws. The battery holder should now be held firmly in place and you can insert the battery as a test. A bead of hot glue on the seam will make the battery holder permanent, but I would hold off until you are in the very final steps of your project.

Figure 6

If you look at Schematic 1 you should notice that I use a central, six position barrier strip. I also wire the battery holder up to a switch. When completed it should look like Figure 7. With the battery mounted on the underside of the Firebot, we free up some space. We also lower the center of gravity, which makes the robot more stable.

Figure 7

At this point you should use a volt meter and verify that you have power on the two outer positions on the barrier strip when the switch is in the ON position.

!! Important !! This 18V battery has quite a bit of power. If shorted out it will melt the wires and could start a fire. While the battery its self has protective circuitry I would not trust it in the case of a direct short.

Add the 12 Volt Regulator

As I mentioned previously I will be adding a 12v regulator to drop the voltage from the 18v battery to safer levels for the RX-28 actuators. Looking at Schematic 1, you will see the 12v regulator is simply inserted in between the switch and the barrier strip positive power terminal. The negative side of the battery is connected to both the regulator and the barrier strip. Place a 100uf capacitor on both the input and output side of the regulator.

In my case, I modified a KronosRobotics 5v regulator and used a couple of standoffs to mount it on my base as shown in Figure 8. Another option would be to use a three position barrier strip to mount the regulator.

Figure 8

Connecting the RX-28 Actuators

When it comes to connecting the RS-28 actuators the common problem is that the included connectors are not long enough for most applications. Dynamixel knows this, as they included some connector components with each actuator. You will need to make a couple connectors in order to connect the actuators to the center barrier strip.

Using the provided terminals, attach Black, Red, Yellow, and White wires and insert them into the plug as shown in Figure 9. You can use a crimping tool or solder them to the terminals. Brief instructions are included in the RS-28 manual. I started with 12” of wire and cut to actual length as needed.

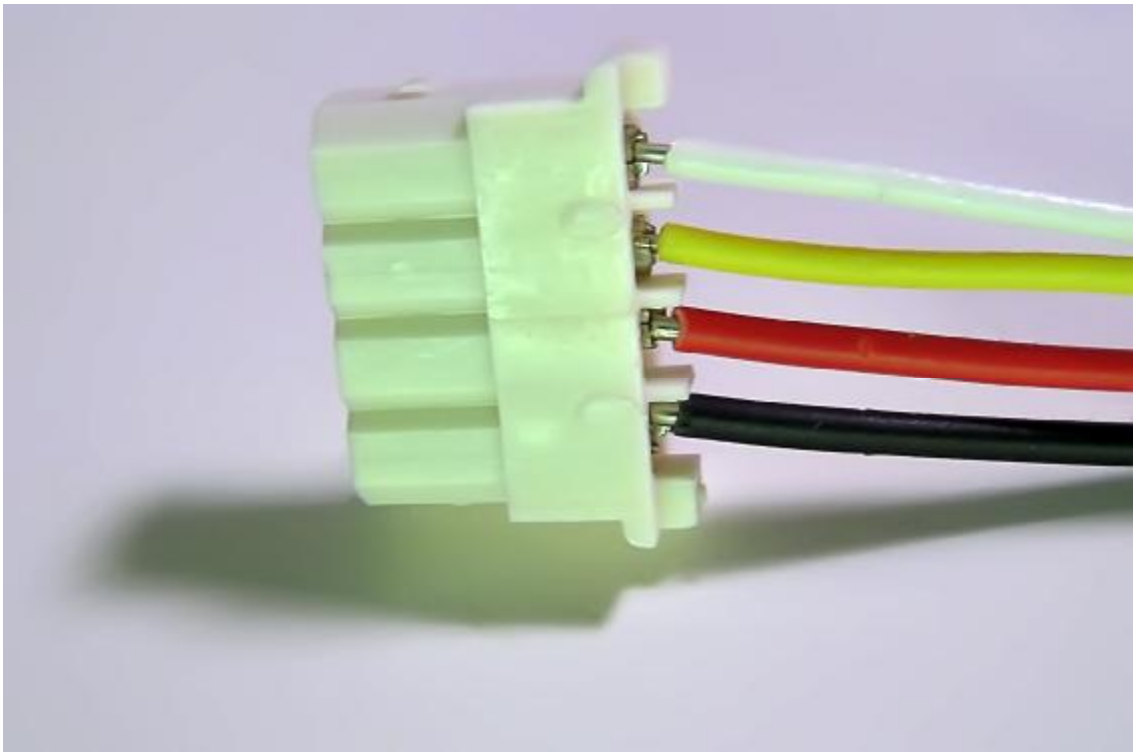


Figure 9

Plug the wire into the RX-28 actuator as shown in Figure 10. As you can see I drilled a hole and used a small wire clamp to hold the wires close to the base. This will help keep the wires from snagging on items.

Figure 10

Run the wires up underneath the barrier strip on the side nearest the battery holder. Drill some holes as needed. Twist the like colors together and connect them to the strip as shown in Figure 11. You will also need to connect a jumper between the red actuator terminal and the outside red power terminal. Do the same with the black actuator terminal and the outside black power terminal.

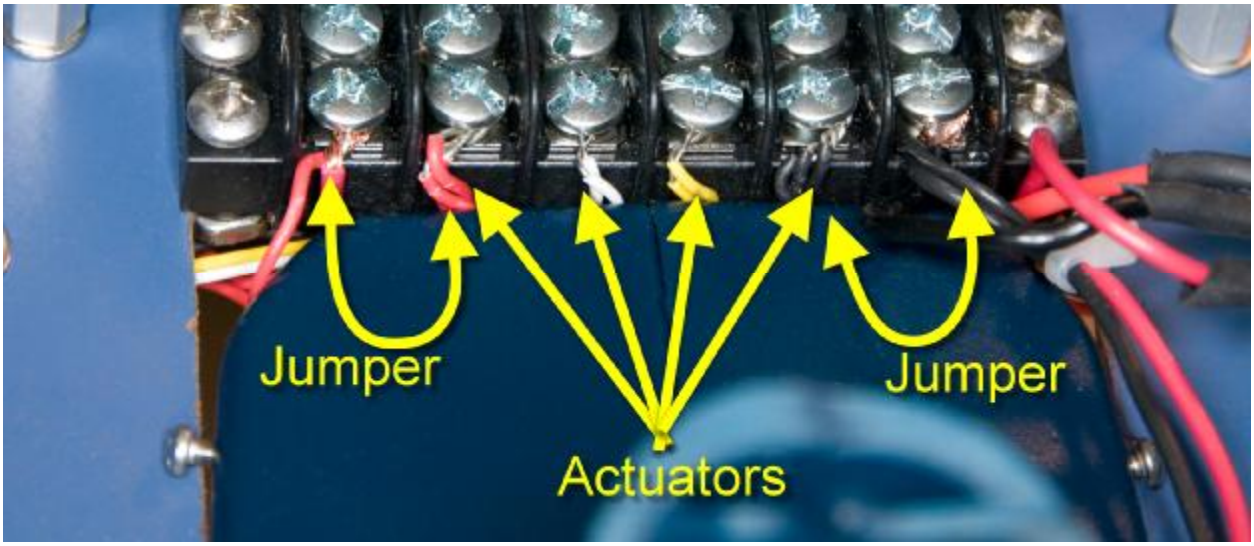


Figure 11

At this point you can apply power to the actuators by turning on the switch. When you do so you should see the LEDs on each actuator go on, then off. If they do not, remove the power and check your connections.

Microcontroller Hookup

Eventually we will be using a Windows CE computer to control our Firbot. You might ask, why even add a microcontroller to the project if we are going to use another

computer? Computers like PC's and CE devices are very powerful but they lack the low level control we need for various sensors we may want to use.

Eventually we are going to program the DiosPro in a sort of slave mode so that we can issue commands on the RS485 bus just like we do to the RX-28 actuator. This will allow us to connect whatever sensors we want to our DiosPro and read the results into the CE device, all with a single connection for both the sensors and actuators.

Dynamixel has seen fit to add an additional connector with each RX-64/28 actuator. This is a small male plug that you can attach wires to in order to create a receptacle that will allow you to use one of the included cables to connect your RX actuator to any application.

Attach four, 3" pieces of colored wires to one of the included plugs as shown in Figure 12. You will also need to cover the connections with some heat shrink.

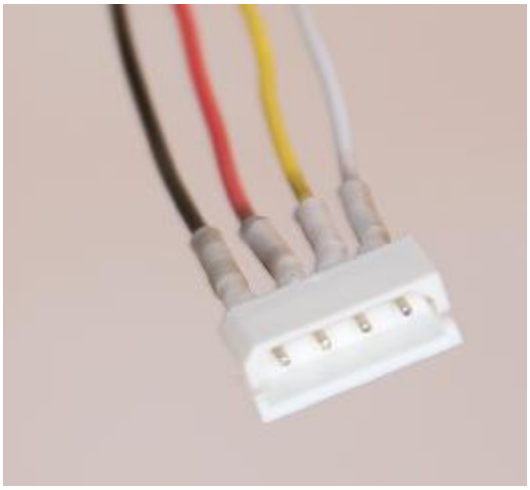


Figure 12

Connect the other end of the wires to the barrier strip as shown in Figure 13. The colors on this connector should match those used when you connected the actuator cables.

The Dios Workboard also needs two power connections. For this I used a couple ShmartBoard connectors that I always keep on hand. Remove one of the ends from each connector and attach to the barrier strip as shown in Figure 13. I used a green connector for my Vss (GND) connection, and a yellow connector for the Vin connection.

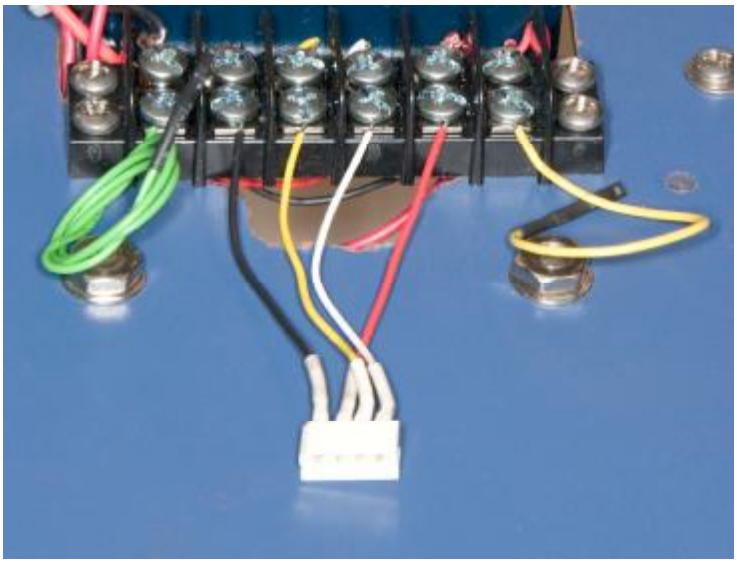


Figure 13

You will need 4 standoffs in order to connect the workboard to the base as shown in Figure 14. I used 1" standoffs here as it gave me room to rout cables.

I have added a chip to the DiosPro Workboard. This is an RS485 chip and you can find a detailed, step by step hookup sheet here:

http://www.kronosrobotics.com/Projects/RXhookupmain.shtml

This hookup will show you how to wire the RS485 chip to the workboard. It also shows you how to change the ID of a device using the USB2Dynamixel adapter. You will need to change the ID on the left actuator to #2 (see Figure 1 as a reference as to which actuator is the left actuator).

Once the board is mounted connect the main power terminal (Yellow wire) to the Vin pin on the workboard, and the GND power terminal (Green wire) to the Vss pin on the workboard.

Take one of the RX actuator cables and connect it to the plug you attached to the barrier strip. Take a 4-pin right angle header and attach it to the other end of the cable. Then plug that end of the cable into the workboard as shown in Figure 14.

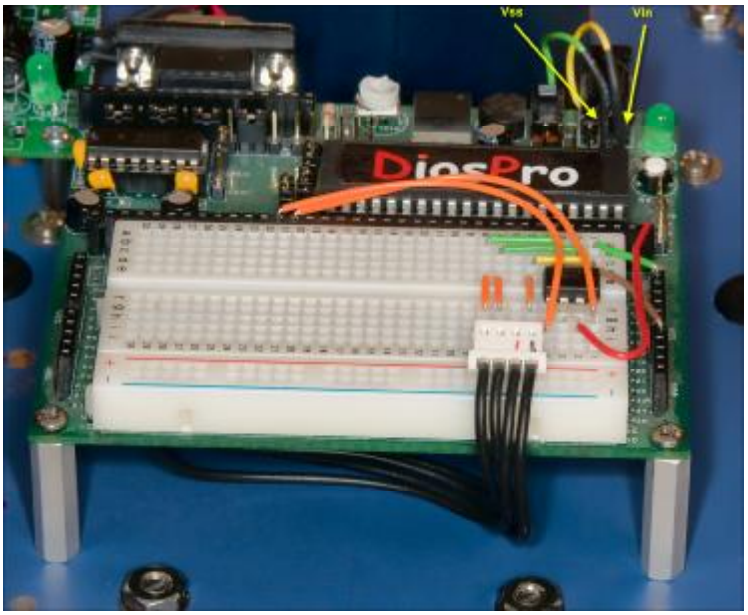


Figure 14

Actuator Interface Test

We can now test our hookup. When you turn on the power switch on the Firebot base you should see the LED on the workboard light up. If it does not, go back and check your connections.

Connect your PC to the program connector on the DiosWorkboard and load the program called “RXreadings.txt” into the Dios. This program takes four readings from the right side actuator (#1) and displays them. Feel free to change the device variable to 2 in order to check the left actuator.

Connect a Radio to Your Firebot

You may be asking why you should even mess with a radio if you plan on building a autonomous robot. It's simple. The radio will make it very easy to test your drive system and sensors, and a radio can also be used to take control of a runaway robot.

If you look at Schematic 1, I have connected a VEX radio to Port 31. I like the VEX radio for two reasons. First it's inexpensive. I have seen them online from $29-$49. The radio is also designed for use with a microcontroller. The DiosPro has a VEX library built-in and you can find a complete interface article here:

http://www.kronosrobotics.com/Projects/vexradio.shtml

Since I used 1" standoffs, I was able stick most of the VEX radio under the Dios Workboard as shown in Figure 15. The radio and antenna have mounts that make it easy to place in any location on your base. Connect the VEX power leads to the Vss and Vcc header connectors on the right side of the breadboard. Connect the VEX data lead to Port 31 on the workboard.

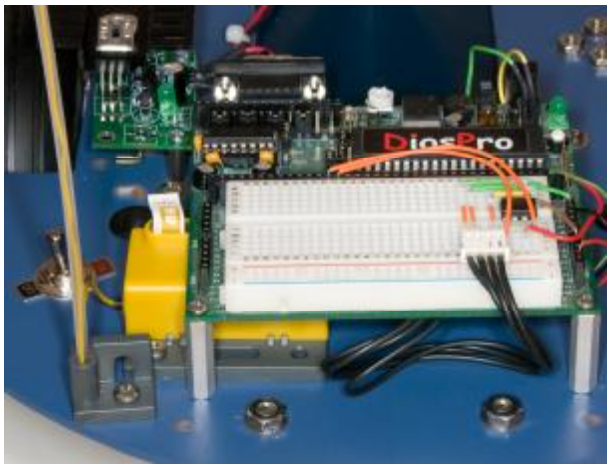


Figure 15

Load up program "VEX_Radioq.txt" program. This program will output the value of all 6 VEX channels along with a status. If the Status is 0 then the radio is off or out of range. If the value is 1 then the 6 data can be considered valid.

Before we move forward you need to build some sort of support for your Firebot. This will come in handy when you need to do tests with the robot located on your desktop. I have used everything from cardboard boxes to old flour tins to support my various robots while programming or testing. The stand shown in Figure 16 was built out of some scrap wood.

Figure 16

Ok you have your VEX radio connected and tested. You have your stand built. It's time for your Firebot to move. Load the program called "FireBotVex.txt" into the Firebot. When you turn on your VEX radio transmitter it will automatically calibrate the receiver and allow you to control your Firebot. I have set the program up to mix the channels 1 and 2 so that you have total proportional control over your robot.

After moving the robot around the house for a while I have to say this is one of the smoothest and quietest robots I have every built.

Add the Real Brain

It's time for our little Firebot to evolve. Let's add a real computer. In this case I am going to add a CE computer called the CUWIN3500. The CUWWIN3500 shown in Figure 17 is manufactured by a company called CMFILE Technology. The CUWIN3500 is designed for use of all kinds of applications, from industrial apps to public kiosks. It has a built-in touch screen but also supports an optional mouse and keyboard. While we won't be using the touch screen or putting the display to any real graphic use, they will come in hand for testing.

Figure 17

Since the CUWIN3500 is going to be our main controller we need to be able to control our two actuators as well as read our VEX radio and other sensors. To do this we will put the DiosPro in a slave mode and have it act like an actuator. This is a simple test, as the commands to do so have been built into the RX library. The hardware interface is identical, so only the DiosPro program needs to be changed.

The CUWIN3500 has an RS485 port with its own connector shown in Figure 18. Connect a Yellow wire to the + terminal and Brown wire to the – terminal on the green connector, as shown. You will also need two power leads. Connect a Red and Black wire to the + and – terminals as shown in Figure 18. You may want to add a wire clamp to act as strain relief. This will make removing the brain easier.

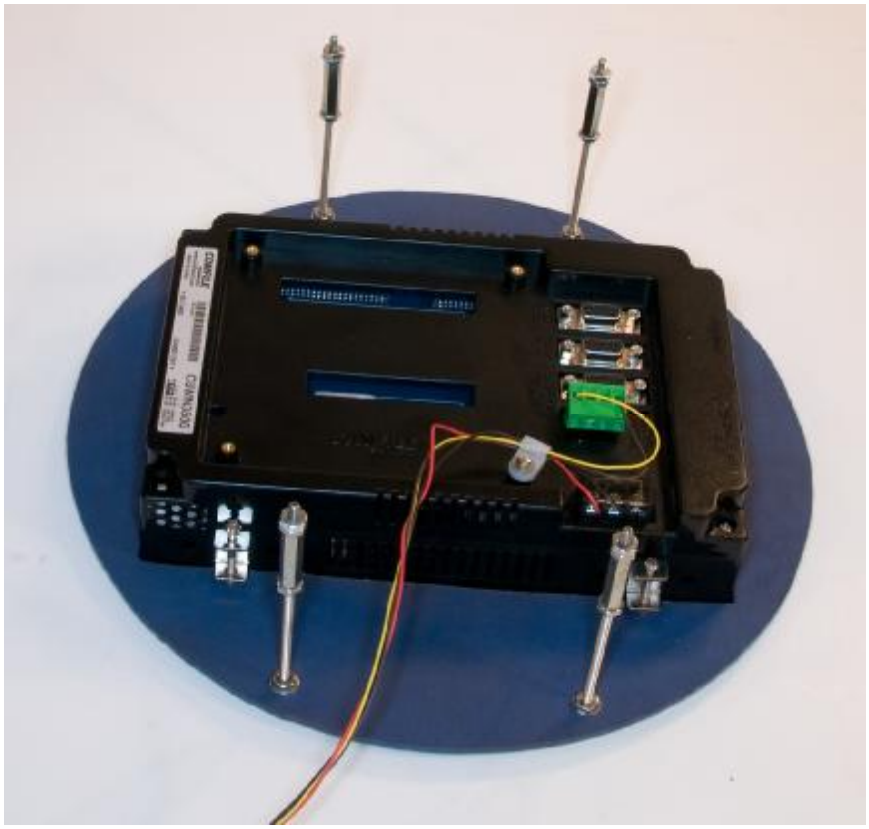


Figure 18

The CUWIN3500 comes with a template to help you create a cutout. I created a duplicate base (minus the wheel cutouts) and using the template, cut out the hole for the CUWIN3500. I used the included mounting brackets to lock the CUWIN3500 in place. I used a combination of standoffs and 2-1/2" machine screws to set the upper platform 3-3/8" from the lower platform. The actual height of the DiosPro Workboard will dictate how high the platform needs to be in order for the CUWIN3500 to clear the Workboard.

Since the CUWIN3500 can be powered from 12v-24v, you may connect to either side of the regulator. As shown in Schematic 1, I recommend the unregulated side. This puts less stress on the regulator. Connect the Yellow data wire to the data lead to pin 6 on the RS485 chip and the brown wire to pin 7.

When you turn your main power switch on, your CUWIN3500 should boot. If it does not, turn off the switch and check your connections. Once you know the CUWIN3500 is

powered properly load up the “RXSlavetest.txt” program and program the DiosPro. This program puts the DiosPro into slave mode.

Programming the Firebot Brain

I am going to use the Zeus as it’s an inexpensive development environment that runs on the PC and CE platforms. If you have programmed the DiosPro microcontroller you will find the language very similar, as the core engine shares some of the same code segments. The language itself utilizes a very rich set of Basic commands. Several libraries have also been included. The USB2AX library will allow you to communicate with several Dynamixel product lines including the AX, RX, DX, and EX series of actuators and sensors.

Let’s fire up the brain and load a program or two. There are a couple ways to program the CUWIN3500. One way is to use an SD card.

Load up the “RX_CUWin_test1.txt” source in ZeusPro. Create a CE5 Console application as shown in Figure 19.

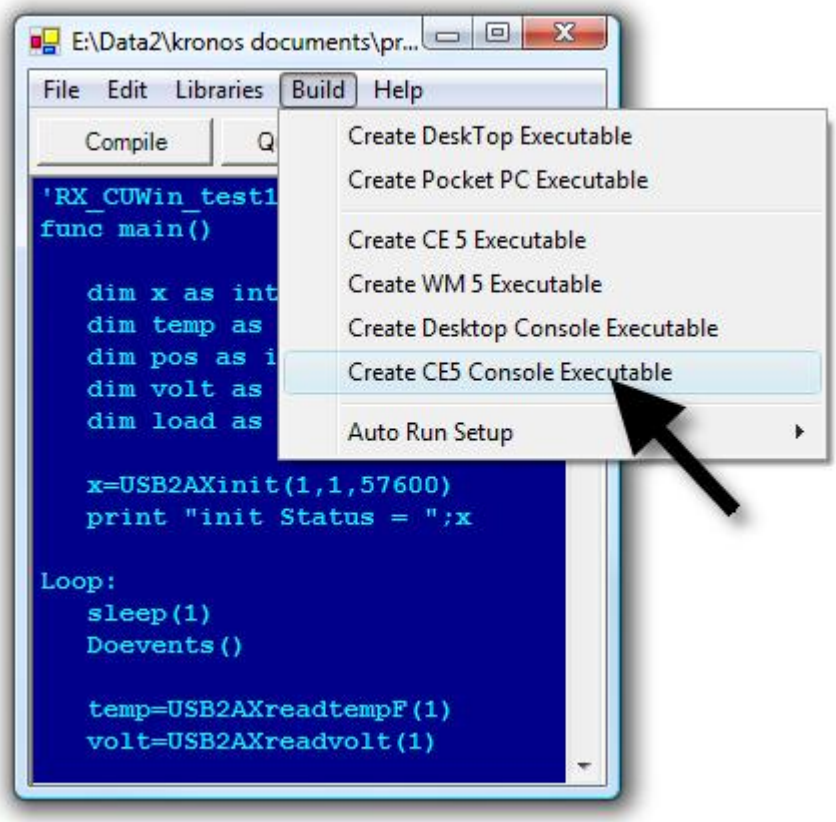


Figure 19

Copy that created executable called “RX_CUWin_test1_CE5C.exe” to an SD card. Once you have the CE executable on the SD card it’s a simple task to insert the SD card into the CUWIN3500 and execute the program.

While this works fine for single program tests, it can be time consuming when doing extensive development. There is a direct method that Microsoft has built into the PC-CE interface. It’s important that you have the latest version of “ActiveSync” when using XP and the latest version of” Windows Mobile Device Center” when running Vista.

Here is how it works: once you have the PC and CE device talking, load up ZeusPro, and under the “Build/Auto Run Setup” menu select the Auto Run option as shown in Figure 20.

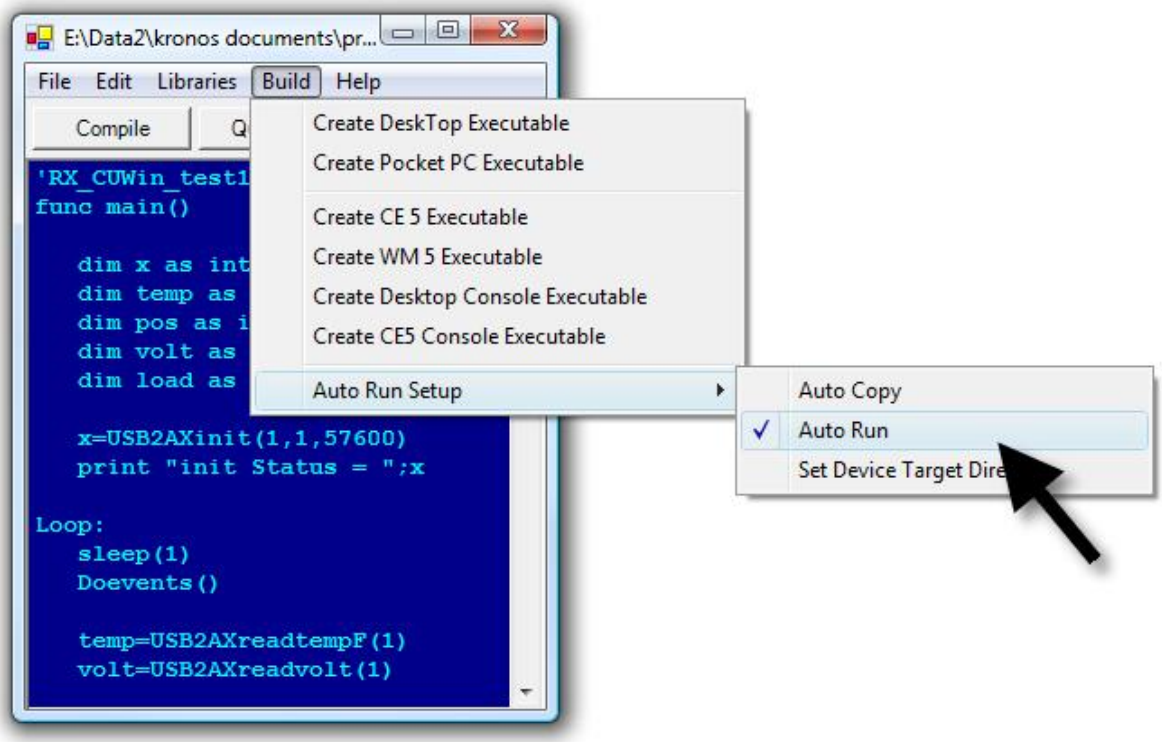


Figure 20

Now when you create the executable as you did in Figure 19 it will automatically be copied to the CUWIN3500 and started. How's that for single button programming? A couple things to keep in mind: first, you must stop the program if it's already running before you start it again. The default directory is the root directory. Anything copied to this area will be lost once the CUWIN3500 is rebooted. If you want to keep the program move it to the Flash Disk or a SD card. Second, keep the console display data to a minimum, as CE devices are not known for their high speed display systems.

The CE Programs

RX_CUWin_test1.txt

Displays 4 readings from actuator #1 and display them on the console.

RXDiosSlaveTest1.txt

Displays VEX radio data taken from the Slave DiosPro.

RXFireBotVEXCE.txt

This is the Firebot remote control program. Make sure the transmitter is on before starting the program.

I have included both the source and compiled versions of these programs. If you decide to use another CE device you will need to modify the programs and recompile them in order for them to work on your particular device.

Closing Thoughts

At first glance the RX-28 actuators may seem a bit expensive. However if you compare these with other midsized robot motors you find the cost comparable.

For instance the following set of components from Parallax will cost you $380

- Motor Mount and Wheel set #27971 $280
- 2, HB-25 Motor Controllers #29144 $100

2 RX-28s and wheels will cost you $432. The RX-28's have the added benefit of load and speed monitoring as well as temperature and voltage. On top of all that they can also be operated in a positional mode like a servo.

What's Next

I have not included any sensor interface for the Firebot, as I just don't have the space in this article for them. If you want to experiment yourself start with the DiosPro interface and get the sensor working standalone first, then migrate it first to the microcontroller-based control, then to the CE. Next month I will be working on the larger Megabot and many of the techniques and interfaces I use on the robot can be used on the Firebot.

Be sure to check out the KronosRobotics website for updates and source files to this project at:

http://www.kronosrobotics.com/Projects/megabot.shtml

Parts

The following are the components needed to build this portion of the project.

Jameco

www.jameco.com

- MC78T12CT 12V Regulator Chip - Jameco # 871658
- Right Angle Header – Jameco #103271
- Toggle Switch – Jameco #76241
- 6-Pos Barrier Strip – Jameco #231010
- TO220 HeatSink – Jameco # 1582663

ShmartBoard

ShmartBoard Jumpers, 5" Yellow
http://www.schmartboard.com/index.asp?a=11&id=42

ShmartBoard Jumpers, 12" Green
http://www.schmartboard.com/index.asp?a=11&page=a_products_accessories&id=122

COMFILE TECHNOLOGY

CUWIN3500
http://www.cubloc.com/product/05_01.php

Crustcrawler

RX-64
http://www.crustcrawler.com/motors/RX64/index.php?prod=67

RX-28
http://www.crustcrawler.com/motors/RX28/index.php?prod=66

Treaded Wheels
http://www.crustcrawler.com/products/rover/wheels.php?prod=28

KronosRobotics

DiosPro Chip
http://www.kronosrobotics.com/xcart/product.php?productid=16428

Dios Workboard
http://www.kronosrobotics.com/xcart/product.php?productid=16566

RS485 Chip
http://www.kronosrobotics.com/xcart/product.php?productid=16388

DiosPro Compiler
http://www.kronosrobotics.com/downloads/DiosSetup.exe

ZeusPro Development Environment
http://www.krmicros.com/Development/ZeusPro/ZeusPro.htm

Misc Items

Vex Radio
You will have to search the internet to locate a VEX radio. I recently purchased one from an online EBay store for $35

Ryobi P700 Flashlight

Available at most Home Depot home centers

Ryobi Lithium Ion Battery and Charger

Available online and at most Home Depot home centers