THIS MONTH'S PROJECTS

LEVEL RATING SYSTEM

To find out the level of difficulty for each of these projects, turn to our ratings for the answers.

●○○○ . . . . Beginner Level
●●○○ . . . . Intermediate Level
●●●○ . . . . Advanced Level
●●●● . . . . Professional Level

DOUBLE FEATURE: TWO BAROMETRIC PRESSURE SYSTEMS IN ONE

I planned on building a wireless station this month but, due to several requests, I decided to show you how to build a couple of different standalone barometric pressure systems.

First, we will build a graphical plotting display that will keep track of nearly 20 days of hourly readings and give you a 12 hour weather prediction. The second system will actually speak the current pressure and inside temperature when you push a button.

The heart of both systems is the SCP1000 pressure sensor. The SCP1000 is an absolute pressure sensor that requires no calibration and will give up to nine readings a second. The sensor is the most accurate and stable pressure sensor I have ever used. The resolution is so fine that it can register a change in pressure when the sensor has been raised a foot or so off the table.

Normally, the SCP1000 would be difficult for the hobbyist to work with due to its small size and form factor, but a company called Sparkfun Electronics has provided a special breakout board shown in Figure 1. It has the sensor installed and a 0.1" header pad. The SCP1000 sensor itself comes in both SPI and I^2C versions, but I found the SPI the easiest to interface. This is the sensor used on the Sparkfun breakout board.

The SCP1000 is a 3V device and needs a voltage of 2.4 to 3.3 volts to operate. This also means the interface leads cannot exceed the supply voltage. We will be using a 5V system so we will need to create a 5V interface for the sensor shown in Schematic 1.

To wire this interface, you have a couple of options. First, you can use some sort of protoboard like the Schmartboard shown in Figure 2. To make things even easier, Kronos Robotics (my company) has a 3.3V to 5V interface kit. The kit has an application note that gives you step-by-step instructions on how to build the small interface board shown in Figure 3.

FIGURE 1

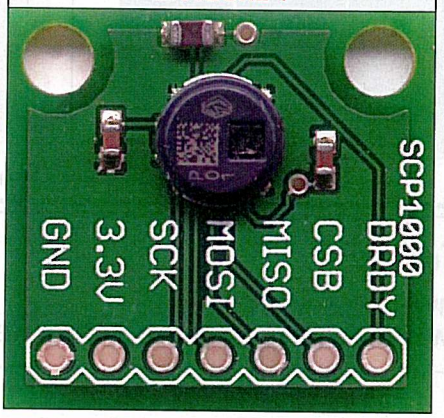
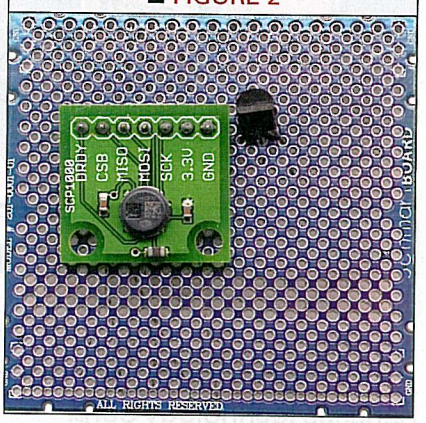


FIGURE 2



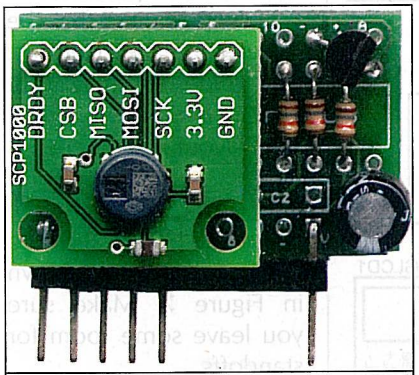


FIGURE 3

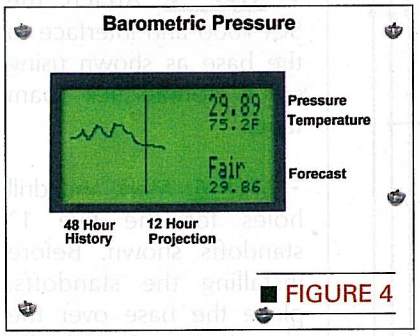
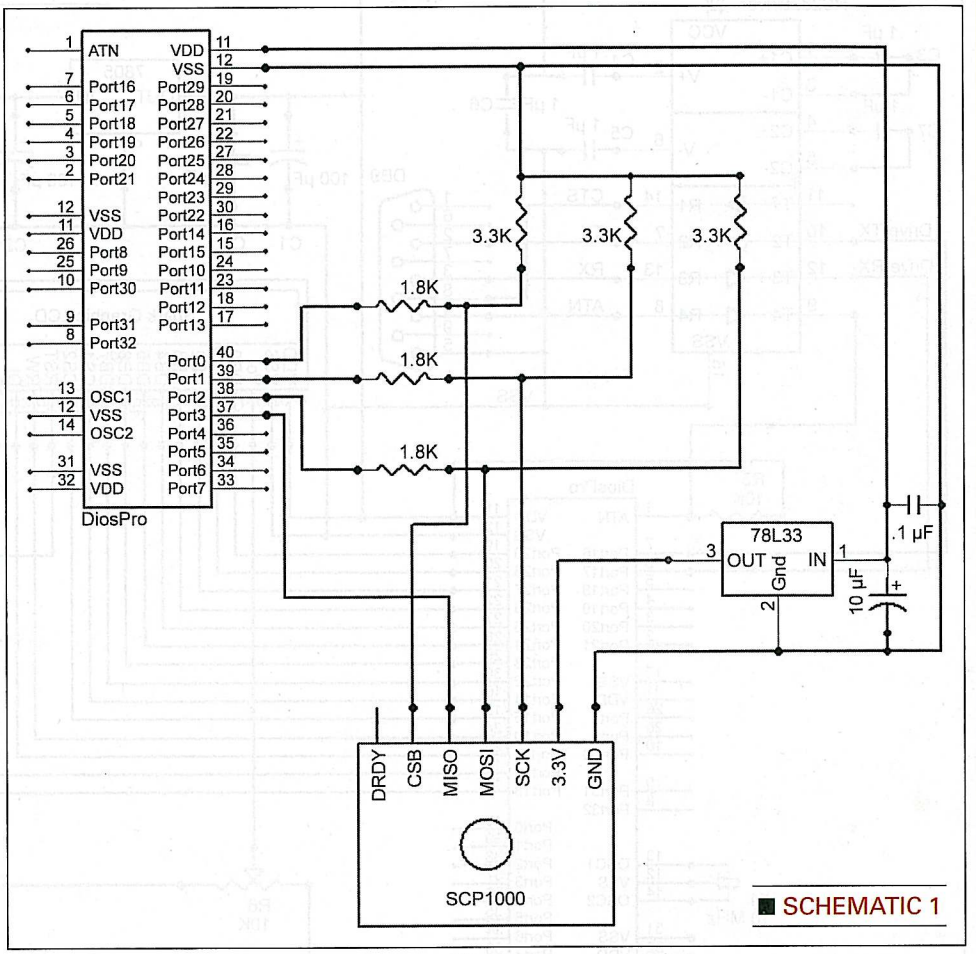


FIGURE 4



SCHEMATIC 1

To use the board, you attach one of the included headers to the SCP1000 breakout board and plug it into the interface board. You will find an application note at **www.kronosrobotics.com/Projects/SCP1000.shtml**.

Barometric Pressure Plotter

To plot and display the barometric pressure, we will be using another Sparkfun product — a graphic LCD display for under $20. This is a 128 x 64 (B&W) display with backlight (shown in Figure 5). Note that I added the 20-pin header.

The LCD uses a parallel interface and is connected to the DiosPro microcontroller as shown in Schematic 2. While this may seem a bit complicated, to make things real simple, Kronos Robotics has a carrier board called the "Dios Universal LCD Carrier" shown in Figure 6. We will be using this carrier with a DiosPro to collect and store data from the SCP1000 sensor. This data will be analyzed and then plotted on the graphic LCD.

Plotter Construction

If you use the Universal Carrier and the SCP1000 interface shown, then the display assembly is quite simple to build.

- ***STEP 1:*** Assemble the Dios Universal LCD carrier. Place the I/O, Keypad, UART, and SW headers on the top of the board and bend them slightly so they will not interfere with the LCD when installed.

- ***STEP 2:*** Assemble the interface board for the SCP1000 using the application note found here: **www.kronosrobotics.com/Projects/SCP10**

FIGURE 5

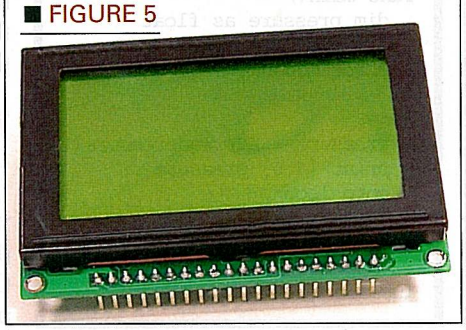
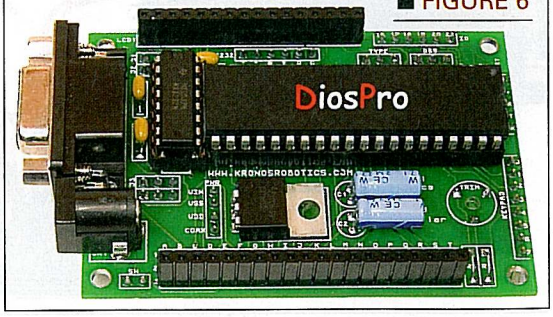
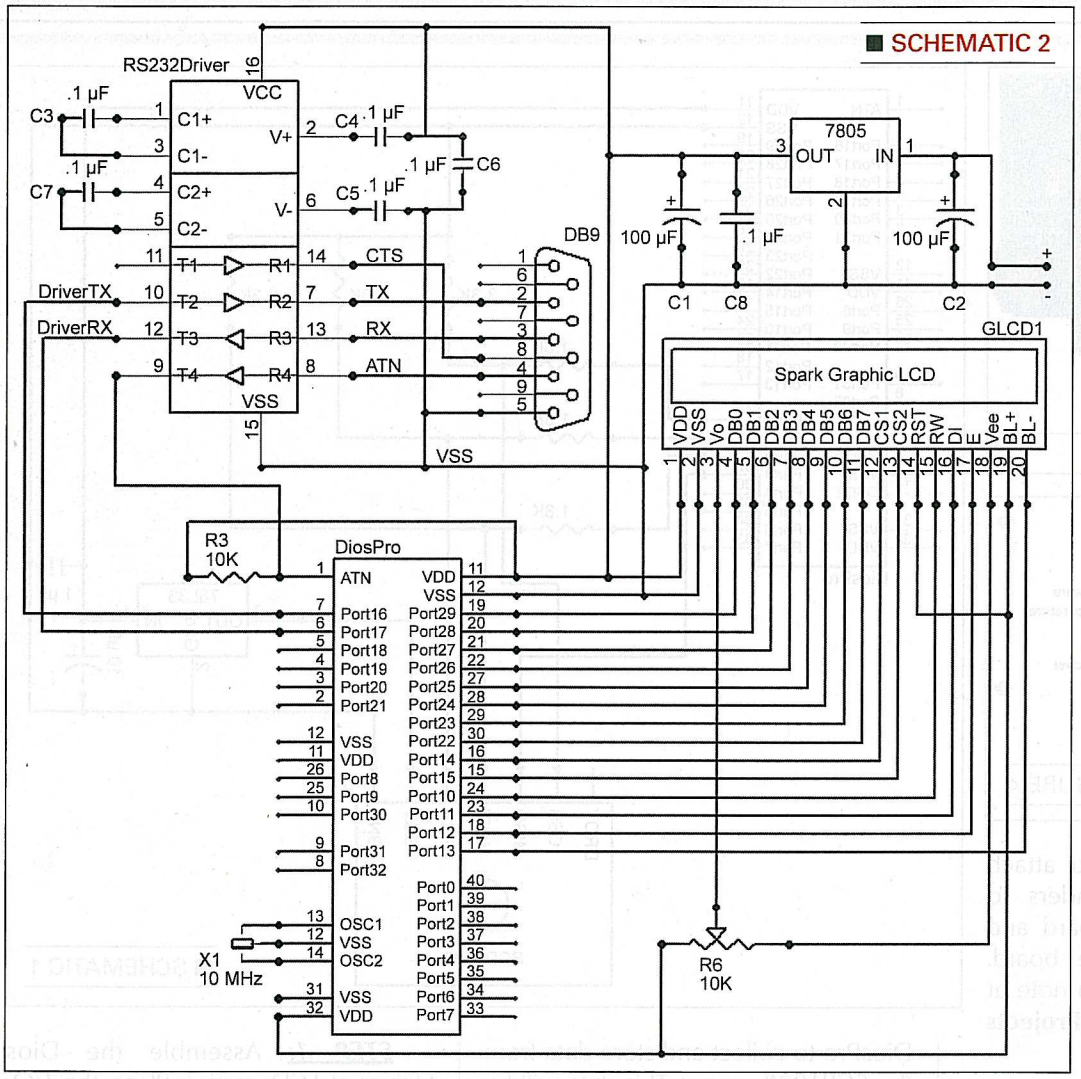


FIGURE 6





00.shtml.

Then install the seven-pin header on the top of the board and bend the pins as shown in Figure 7.

• ***STEP 3:*** Cut two pieces of compressed PVC to 4.5" x 3.5" to be used as the base and top. You can use acrylic or just about any material. Acrylic can be purchased at most home centers. The nice thing about using acrylic is that you don't need to cut a hole for the display since it's transparent.

Keep in mind that the dimensions are given just as a guide. You may want to use a different size if your layout is different.

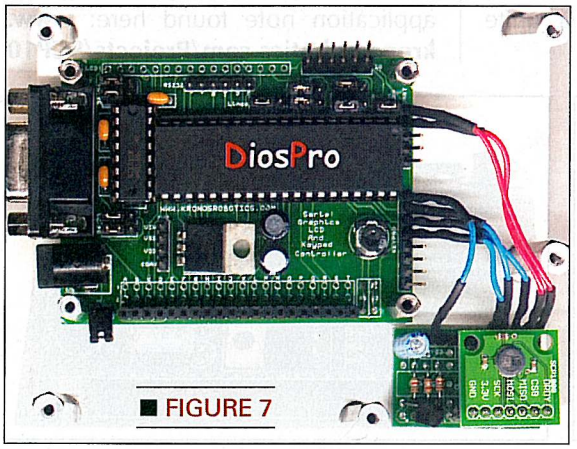
• ***STEP 4:*** Attach the Universal LCD Carrier to the lower base as shown in Figure 7. Make sure you leave some room for standoffs.

• ***STEP 5:*** Attach the SCP1000 and interface to the base as shown using some double-stick foam tape.

• ***STEP 6:*** Mark and drill holes for the five 1" standoffs shown. Before installing the standoffs, place the base over the top piece and mark the holes so you can duplicate them for the standoffs.

• ***STEP 7:*** Using five jumpers, connect the sensor interface to the Universal LCD Carrier. I prefer using the Schmartboard jumpers. In this case, I used the blue 2" and red 3" jumpers.

• ***STEP 8:*** Attach the LCD to the Universal Carrier then attach the top cover to the standoffs. If you are not using transparent material, you will have to add a cutout for the display. To mark this, I attached the cover, then with a fine point marker, I noted the position of the display on the underside of the board.



■ FIGURE 7

Program 1

```
'SCP1000-D01 Test
func main()
  dim pressure as float
  SCPinitD01(0,1,2,3)
  SCPsetmodeHR()

loop:
  pressure = SCPreadpressure()
  print {7.0} pressure
  goto loop

endfunc

include \lib\SCP1000.lib
```

Testing the Display

To program the DiosPro, you need to install the free compiler and connect the Universal Carrier to your PC with a nine-pin serial cable and plug a DC

power source into the 2.1 coax. Refer to the carrier instruction manual for more details.

To test the interface to the SCP1000, load and run Program 1. This will display the pressure reading in Pascal units on debug terminal.

I have also included a second program called Program2.txt that will display the temperature.

The program called bargraph1.txt is the main plotting program. While the program is a bit lengthy to present here, I will describe a few of the details that you may want to change in the program.

Station Pressure

At the very beginning of the program is a statement where I assign a constant called offset. This is a value that will allow you to change the absolute pressure value to the station pressure. This value is added to the absolute value. You should change this value so that your pressure gauge reads the same as your local forecast.

Default EEPROM

The first portion of the plot is the last 48 hour history. This data is saved into the first 96 EEPROM locations. When you program the DiosPro, the data statements at the beginning of the program initialize these locations. You can comment these statements out if you want to experiment with your exiting data so that they won't get cleared.

Data Dump

Once you have programmed the DiosPro and have verified that it is working properly, you can configure the program port to connect to the onboard hardware UART by setting the DB9 jumpers. The small bit of code following the loop label tests for a character value of 65 at each pass. If this value is received, the complete 20 day history will be sent out the UART at 9600 baud. The data is dumped with the low order byte first, then the high order byte. Each byte pair represents one hour of data with a total of 460 hours. The first pair is the current hour index at the time of the dump.

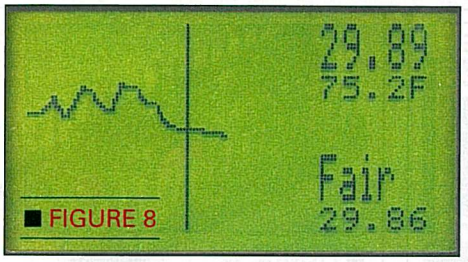


FIGURE 8

The Display

The display has a vertical line near the center of the display shown in Figure 8. The plot to the left of this line represents the last 50 hours of pressure readings. The plot to the right of this line is a 12 hour projection. On the upper righthand side of the display is the current barometric pressure and inside temperature. On the lower righthand side is the 12 hour forecast and pressure.

How Well Does it Work?

I have been using this pressure sensor for the last couple months and it works pretty well. It has accurately predicted the weather for most changes in my area. For instance, while I was writing the display indicated that a change in the weather was going to occur in the next 12 hours. My grass was getting tall and if it were to rain I would have to wait another week to cut it, so I stopped everything and took out my mower. Sure enough, about 13 hours later it started to rain.

I prefer the SCP1000 pressure sensor over the 1-Wire sensors that I have used, as it seems to give a more reliable reading and does not require 14V to operate.

Going Further

All that is needed to make this display wireless is to add a Zigbee unit. I will look into doing just that when I build the wireless station next month.

I had experimented with the SCP1000 for use in other projects like my R/C helicopter and found that in order to get the high resolution required, you need to read the SCP1000 about once every 1.8 seconds. This is too long for this type of application, however, I have read where others have taken two SCP1000s and alternated the readings in order to double the sample rate. This also gives you some redundancy in case of a failure.

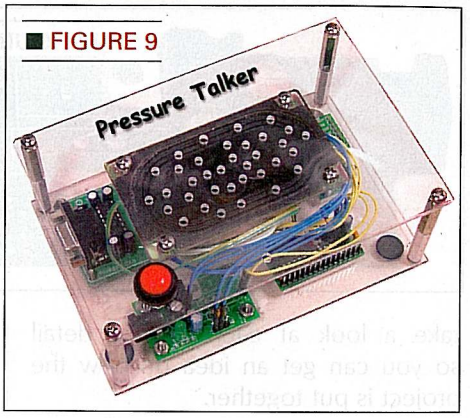


FIGURE 9

Talking Barometric Pressure

Now it's time to take a totally different approach and build a talking barometer. I wanted to keep things as simple as possible, so we will take the modular approach on this project. I will take various boards that are readily available, both assembled and in kit form, and create the talking barometer shown in Figure 9.

Pressure Talker Construction

If you can solder, then you can build this project. Even if you can't, you can probably get someone to assemble the various modules. Let's

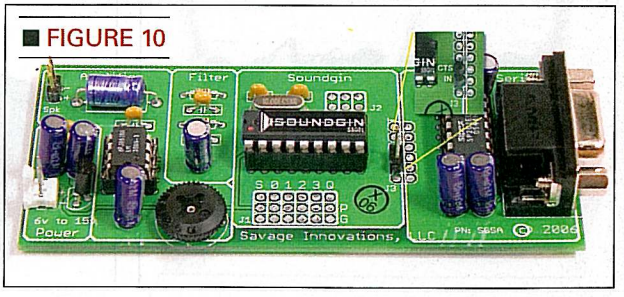
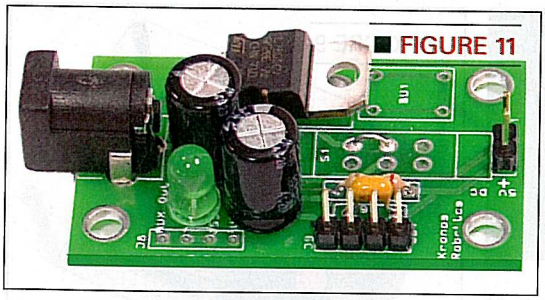
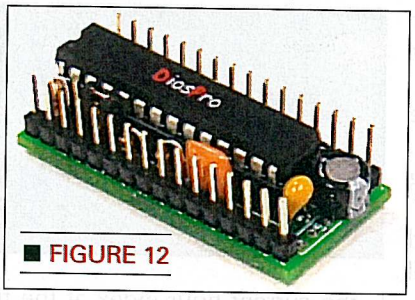


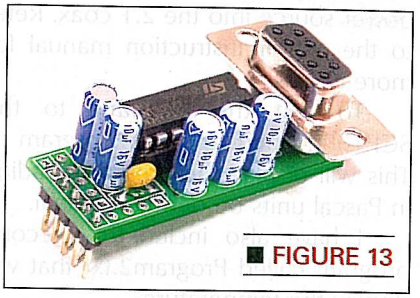
FIGURE 10



■ FIGURE 11



■ FIGURE 12



■ FIGURE 13

take a look at each one in detail so you can get an idea on how the project is put together.

Sound Module

I will use the Soundgin sound chip from Savage Innovations to create speech. The Soundgin chip is capable of creating music, sound effects, and a set of phonetic variations called Allophones. The chip itself requires a filter and amplifier so you need a development board. Savage Innovations makes such a board and it includes everything you need to create speech and other sound effects. You will need to add a two-pin header to the pads shown in Figure 10. Unfortunately, the pads are all filled with solder so you will have to remove it in order to install the header. Refer to the small inset as it shows pin placement on the board.

Regulator Module

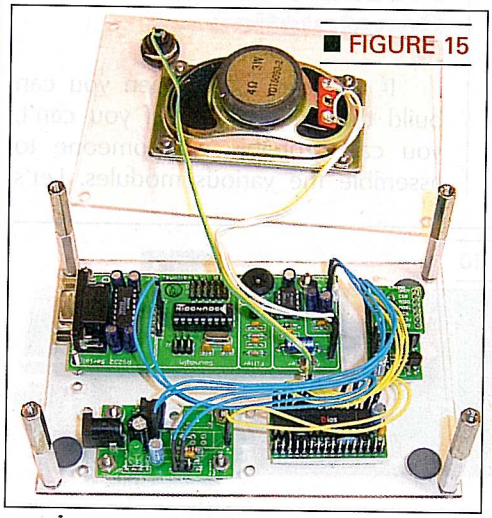
The module shown in Figure 11 is a simple and inexpensive 5V regulator that can be purchased in kit form, or assembled from Kronos Robotics. The board also has a header to give you access to the Vin power directly from the coax. We need this to power the sound module. The voltage regulator requires the use of an AC adapter with an output of 6.5 to 14 VDC. This is applied to the 2.1 coax connector, center positive. You can purchase one of these from RadioShack. I have also provided a Jameco part number in the Parts List.

When you assemble this board, you have the option to install the two-pin headers on the bottom for use with a breadboard. For our application, install them on the top of the board so that we can use wire jumpers to connect to the other modules.

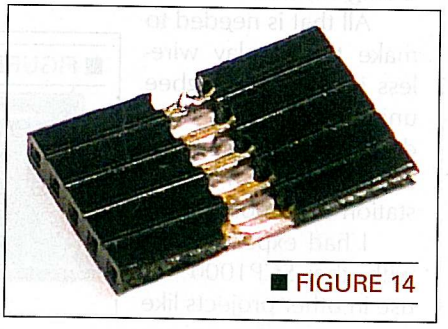
Dios Carrier 1 Module

The Dios Carrier 1 module shown in Figure 12 is a small carrier that accepts the DiosPro 28-pin chip. It is also available in both kit and assembled form from Kronos Robotics.

As mentioned, the 12-pin headers need to be installed on the top of the board as shown in Figure 12 so that we can use wire jumpers. This board does not include the PC interface needed to program the chip like the Universal LCD carrier does, so we need to use an EZRS232 module.



■ FIGURE 15



■ FIGURE 14

EZRS232 Module

In order to program a DiosPro used in a Carrier 1, you need an EZRS232 module. Again, this is a very inexpensive module available from Kronos Robotics in both kit and assembled form (Figure 13).

To plug the EZRS232 driver directly into the Carrier 1 board, you need to create a small five-pin female-to-female header. This can be done by soldering two five-pin headers together as shown in Figure 14. You can also use five jumper wires to connect the EZRS232 driver to the carrier. The EZRS232 driver gets its power from the carrier board.

SCP1000 Interface Module

I have already gone over the SCP1000 interface module available from Kronos Robotics (shown in Figure 3). This module will allow you to plug the Sparkfun SCP1000 directly into our 5V system.

Project Assembly

All the modules are connected together using Schmartboard jumpers shown in Figure 15. The complete diagram is shown in Schematic 3.

- ***STEP 1:*** Let's start assembly by cutting the base and top pieces. I used acrylic here so others could see the inside of the project, but any material can be used. I used 4" x 8" pieces, but they could be larger if you need the space.

  Once cut, place 5/32" holes into each corner of the base and top for the standoffs.

- ***STEP 2:*** Place your speaker in the center of the top piece and mark the mounting holes. Trace the speaker, as

well, to give us an outline so you can drill a series of holes for the sound.

• ***STEP 3:*** Drill the sound holes as shown back in Figure 9. You also need to drill the hole for the button. For the Jameco button listed in the Parts List, you'll need a 1/2" hole. Once all the holes are drilled, install the speaker and button. If your speaker does not have mounting holes, use hot glue to attach it to the top by running a bead around the speaker. For better adhesion, score a couple lines around the speaker.

• ***STEP 4:*** You will need to attach two wires to the speaker and two wires to the button as shown in Figure 14. Attach a two-pin female header to the opposite end of each of these. These headers will be used to connect to the modules.

• ***STEP 5:*** Attach the modules in the positions shown in Figure 14. The easiest way to mount the boards is with double stick foam tape. If you want to, you can drill holes in the Soundgin board and use machine screws to mount it.

• ***STEP 6:*** Attach a 1" MF standoff to a 1" FF standoff, then attach it to the base with a #6 machine screw. Do this with each of the four corners on the base.

• ***STEP 7:*** I used Schmartboard 5" and 7" jumpers to attach the modules. Refer to the boards, as well as the module documentation, for the actual locations.

Yellow 5" Jumpers

- Regulator + (header closest to the Carrier 1) to Carrier 1 + (closest to cap)
- Regulator - (header closest to the Carrier 1) to Carrier 1 - (closest to cap)
- Carrier 1 Port0 to SCP1000 CSB
- Carrier 1 Port1 to SCP1000 CK
- Carrier 1 Port2 to SCP1000 MOSI Carrier 1 Port3 to SCP1000 MISO

Blue 7" Jumpers

- Regulator + (header closest to coax) to SCP1000 5V
- Regulator - (header closest to coax) to SCP1000 Gnd
- Regulator Vin (four-pin header) to Soundgin Power +
- Regulator Vss (four-pin header) to Soundgin Power -
- Carrier 1 Port9 to Soundgin IN (added header)
- Carrier 1 Port11 to Soundgin CTS (added header)

Speaker Header to Soundgin SPK header

Button Header to Carrier 1 Port4 and Port5

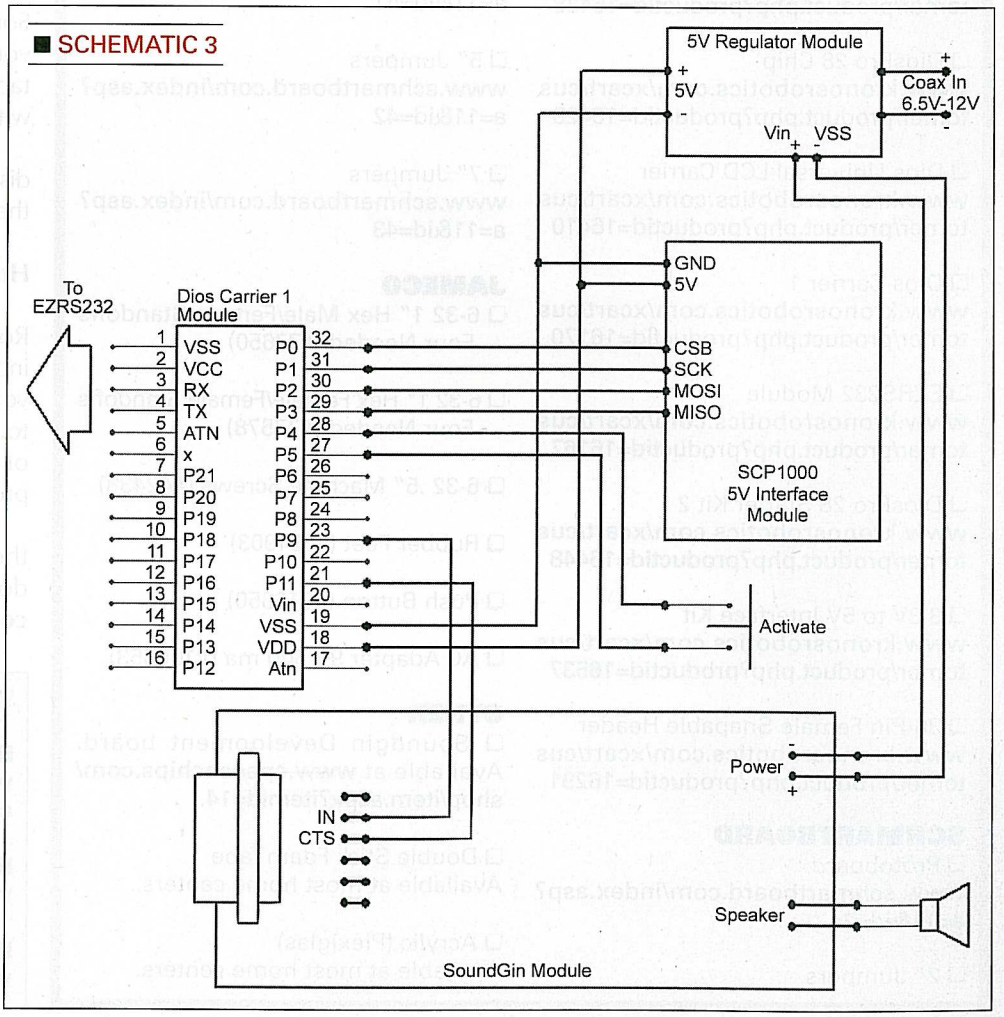
• ***STEP 8:*** Attach the top base to the four standoffs with four #6 machine screws. You will want to attach some rubber feet to the bottom base so that the machine screws don't scratch anything. You can purchase these at most home centers. There is a part number listed for a set from Jameco.

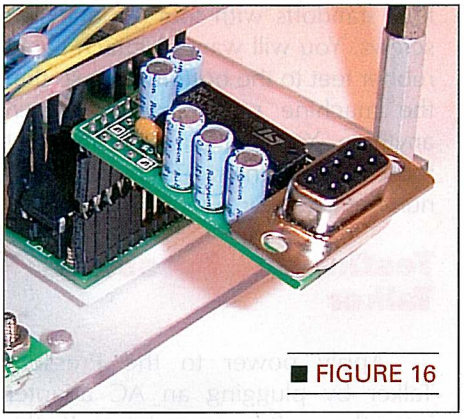
Testing the Pressure Talker

Apply power to the Pressure Talker by plugging an AC adapter into the voltage regulator. If the green LED does not light or is dim, then remove the power and recheck your wiring.

Plug the EZRS232 into the Carrier 1 by using a five-pin double header as shown in Figure 16. Install the free Dios compiler and connect the PC to the EZRS232 using a nine-pin straight cable. The DiosPro chip is already

■ SCHEMATIC 3





■ FIGURE 16

programmed with a test program, so once you run the compiler and power-up the Pressure Talker, it should display test data in the debug terminal.

• TEST 1: First, you need to test the pressure sensor by loading and programming the same Program 1. This will display pressure data in the debug window.

• TEST 2: Next, load up the program called PressureTalket1.txt. This program will say the word "Barometer" when started and will say it when the button is pressed.

Main Program

Load the program called Pressure Talker2.txt and program it into the DiosPro chip. Once programmed, the Pressure Talker will speak the indoor temperature and pressure each time the button is pressed. The pressure and temperature is constantly displayed in the debug window.

As in the previous project, at the very beginning of the program is a statement where I assign a constant called offset. This is a value that will allow you to change the absolute pressure value to the station pressure. This value is added to the absolute value. You should change this value so that your pressure gauge reads the same as your local forecast.

The Soundgin chip uses Allophones to define the speech. Each word that the Pressure Talker speaks is made up of these Allophones. They are defined as table entries at the beginning of the program. You may change these if you like. You can download special software from the Soundgin website and that will help you define these. Just make sure the table entry starts with a label and ends with a 0.

Use the SGplay command to display your new word by passing the label.

How Well Does it Work?

I recently took the Soundgin to RobotFest. About 50% of the individuals could not understand the voice. It does take a little getting used to. However, I will say that a couple of visually impaired individuals had no problems with the speech.

Updates and source code for these projects are available for download at **www.kronosrobotics.com/Projects/pressure.shtml**. NV

PARTS LIST

KRONOS ROBOTICS

❑ DiosPro 40 chip
www.kronosrobotics.com/xcart/customer/product.php?productid=16428

❑ DiosPro 28 Chip
www.kronosrobotics.com/xcart/customer/product.php?productid=16429

❑ Dios Universal LCD Carrier
www.kronosrobotics.com/xcart/customer/product.php?productid=16410

❑ Dios Carrier 1
www.kronosrobotics.com/xcart/customer/product.php?productid=16170

❑ EZRS232 Module
www.kronosrobotics.com/xcart/customer/product.php?productid=16167

❑ DiosPro 28 Starter Kit 2
www.kronosrobotics.com/xcart/customer/product.php?productid=16448

❑ 3.3V to 5V Interface Kit
www.kronosrobotics.com/xcart/customer/product.php?productid=16537

❑ 36-Pin Female Snapable Header
www.kronosrobotics.com/xcart/customer/product.php?productid=16291

SCHMARTBOARD

❑ Protoboard
www.schmartboard.com/index.asp?a=11&id=24

❑ 2" Jumpers
www.schmartboard.com/index.asp?a=11&id=112

❑ 3" Jumpers
www.schmartboard.com/index.asp?a=11&id=41

❑ 5" Jumpers
www.schmartboard.com/index.asp?a=11&id=42

❑ 7" Jumpers
www.schmartboard.com/index.asp?a=11&id=43

JAMECO

❑ 6-32 1" Hex Male/Female Standoffs — Four Needed (#77650)

❑ 6-32 1" Hex Female/Female Standoffs — Four Needed (#77578)

❑ 6-32 .5" Machine Screws (#42438)

❑ Rubber Feet (#651903)

❑ Push Button (#616850)

❑ AC Adapter 9V 500 ma (#100853)

OTHER

❑ Soundgin Development board. Available at **www.speechchips.com/shop/item.aspx?itemid=14.**

❑ Double Stick Foam Tape
Available at most home centers.

❑ Acrylic (Plexiglas)
Available at most home centers.

WEB LINKS

■ Kronos Robotics
www.kronosrobotics.com/xcart/customer/home.php

■ SchmartBoard
www.schmartboard.com/mscva

■ Jameco
www.jameco.com