Using the IR Sensor System with the VEX Cortex

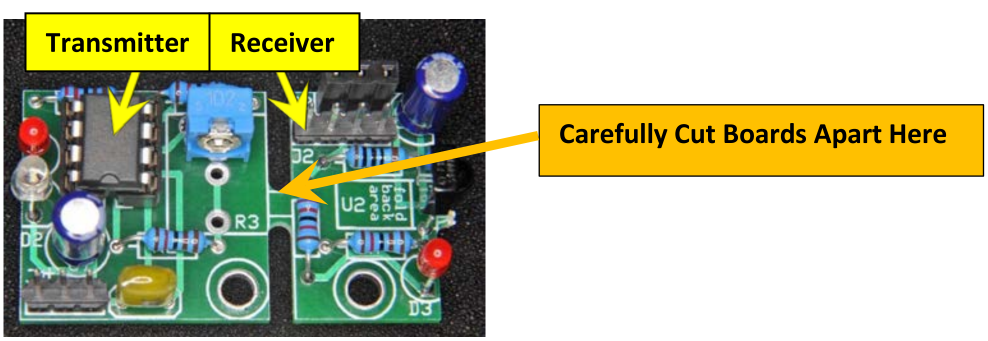
GOAL To conduct simple sensitivity tests and attach the BEST IR Sensor System to a VEX Cortex Brain.

Materials Needed

- BEST IR Sensor System (Assembled PCBs)
- Two #4 screws (if mounting to a robot)
- VEX Cortex Brain and battery
- 3-wire cables
- Additional 3-wire cable (If choosing Step3-Option 2)
- 4-wire cable (If choosing Step 3- Option 3)
- Oscilloscope or frequency counter (optional)

This document will walk you through testing and debugging the BEST IR Sensor System's sensitivity and two different options for attaching to a VEX Cortex Brain.

Step 1: Separate the PC Boards



In the kit the two PC boards are manufactured as a single board to make it a little larger for easier assembly. After soldering on all the components, the two boards will be cut apart. If you have not already done so, cut the board as described in the final step of the assembly instructions.

The board that emits the IR light is the transmitter and is the larger of the two boards. The IR emitting diode is the clear one, located on one edge of the board. Note that as supplied it emits light in a direction perpendicular to the plane of the board. If you want it to emit light in the plane of the board, **carefully** bend the two leads 90 degrees, the IR diode will extend beyond the edge of the board.

The board that receives the IR light is the receiver and is the smaller of the two boards. The IR detecting IC is the dark, 3-legged device with a bump on its side, located on one edge of the board. The bump is a lens so the IR should shine on the bump. Note that as supplied it is “looking” in the plane of the board. If you want it to “look” in a direction perpendicular to the plane of the board **carefully** bend the IC back so the bump is away from the board. There is a clear area on the board that it should fit into without hitting anything.

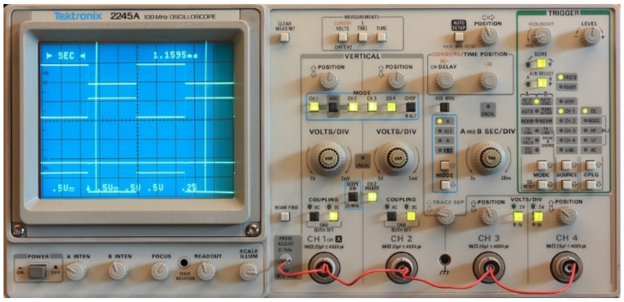
Note that the leads on the IR emitting diode and IR detecting IC can likely only be bent one time without breaking the leads so you should think carefully before making your decisions on orientation.

The two large holes, one on each board, are for mounting and are sized for a #4 screw.

Step 2: Adjust the System’s Sensitivity

For help in debugging, the transmitter board has a red LED that will be on when the IR emitter is on. The receiver board has a red LED that will be on when an IR signal is detected. Due to the limited power available, the receiver LED is fairly dim.

The IR signal is modulated at 38KHz to reduce sensitivity to other light sources – the detector is tuned to 38KHz and is less sensitive to constant light or other modulation frequencies. For maximum sensitivity tune the transmitter board to 38KHz. You can adjust the modulation frequency to adjust the system’s sensitivity by turning the white potentiometer on the transmitter board. There are a couple ways to make this adjustment.



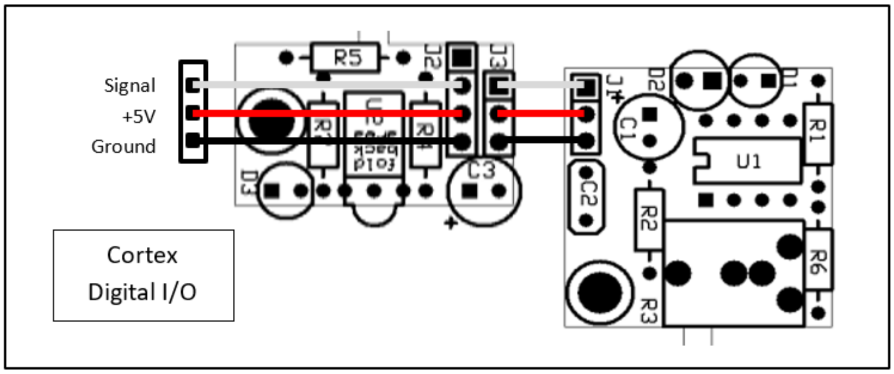
If you have an oscilloscope or frequency counter connect it to pin 7 of the 8-pin DIP to monitor the frequency. If you lack that equipment, don't panic. You can do a pretty good job without any special test equipment. Separate the transmitter and receiver until the receiver just barely detects the IR signal. Now when you adjust the potentiometer you should find that the receiver detects the IR signal somewhere in the middle of its range of travel but does not detect the IR signal at either the counter-clockwise extreme or the clockwise extreme. Find the two spots of the potentiometer's travel where it switches between detecting and not detecting and set the potentiometer halfway between these two points.

You may not want maximum sensitivity. Perhaps your transmitter and receiver are very close together, counting wheel spokes as they go past, for example. Frustratingly, it doesn't work reliably because the receiver is also picking up reflections from other parts of your robot. In this case you can "detune" the transmitter so that your receiver can detect the strong signal from the transmitter but not the weaker reflections. Flat black paint may also be your friend here. ☺

Before you Begin: The final step has three options, organized by difficulty. Read over each section before deciding which option is right for your team.

Step 3-Option 1: Using a Single Cortex Digital I/O Port

This option uses a daisy chain to connect the IR sensor system to a single Cortex port. The IR emitter runs continuously. It is suitable for a "beam break" application.



Option 1, Daisy Chain Hookup

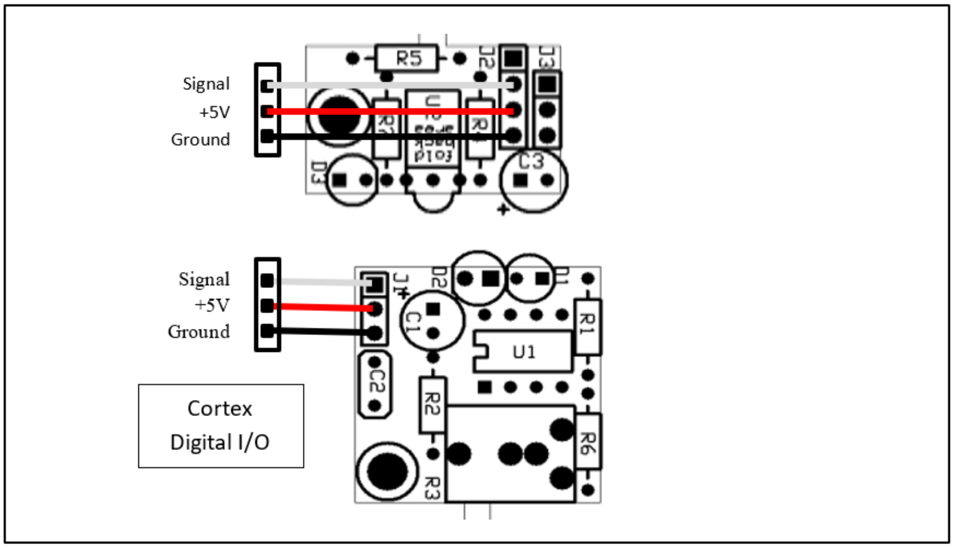
Using a 3-wire cable, connect the Cortex to the receiver board. The male end of the cable plugs into one of the "DIGITAL INPUTS/OUTPUTS" of the Cortex, black wire in the ground hole, red wire in the +5V hole, and white wire in the signal hole. (See the VEX Cortex Pin-out at the end of this document). The female end of this cable plugs onto 3 of the 4 pins on the receiver board, black wire on the pin closest to the dark, 3-pin IR detector IC. The 4th pin on the receiver board, the pin closest to the edge of the board, remains unconnected.

Using another 3-wire cable, connect the receiver board to the transmitter board. The male end of the cable plugs into the 3-pin socket on the receiver board, black wire in the same end as that of the first cable. Note that the cable's connector will be offset 0.1" from the first cable's connector. The female end of this cable plugs onto the 3-pin connector on the transmitter board, black wire closest to the bright yellow capacitor, away from the corner of the board.

Set the Cortex digital I/O port as an input. The "signal" line will be pulled low when the IR emitter on the transmitter board (or IR from any source) shines on the IR receiver on the receiver board.

Step 3-Option 2: Using Separate Cortex Digital I/O Ports

This option uses two Cortex ports. The IR emitter can be controlled. It is suitable for simple signaling (on/off or pulse width), sending binary data by "bit banging" (software UART), or a "beam break" application.



Option 2, Separate Connections Hookup

Using a 3-wire cable, connect the Cortex to the receiver board. The male end of the cable plugs into one of the "DIGITAL INPUTS/OUTPUTS" of the Cortex, black wire in the ground hole, red wire in the +5V hole, and white wire in the signal hole. (See the VEX Cortex Pin-out at the end of this document). The female end of this cable plugs onto 3 of the 4 pins on the receiver board, black wire on the pin closest to the dark, 3-pin IR detector IC. The 4th pin on the receiver board, the pin closest to the edge of the board, remains unconnected.

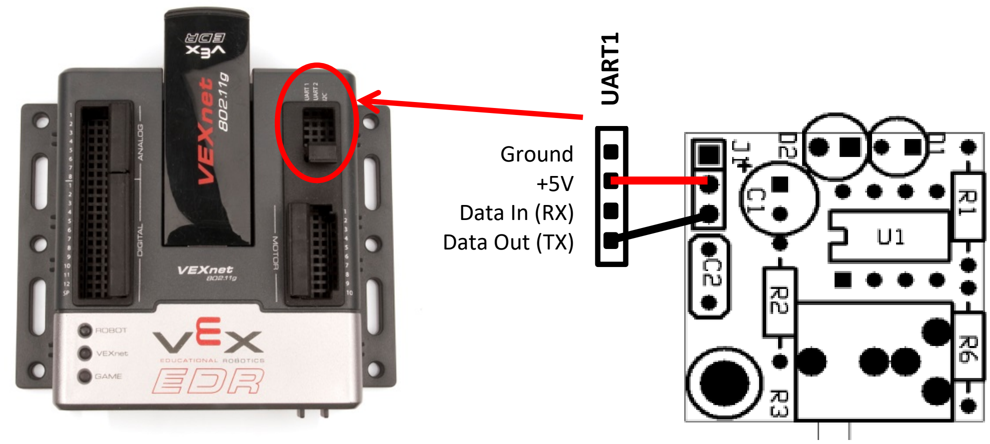
Using another 3-wire cable, connect the Cortex to the transmitter board. The male end of the cable plugs into a different one of the "DIGITAL INPUTS/OUTPUTS" of the Cortex, black wire in the ground hole, red wire in the +5V hole, and white wire in the signal hole. The female end of this cable plugs onto the 3-pin connector on the transmitter board, black wire closest to the bright yellow capacitor.

Set the Cortex digital I/O port connected to the receiver board as an input and the port connected to the transmitter board as an output. The IR emitter can be controlled by the Cortex output, high=on and low=off. The Cortex input will be pulled low whenever IR (from any source) is detected.

Step 3-Option 3A: Using a Cortex UART port (Transmit Only)

This option uses one of the Cortex UART ports. The "Data Out (TX)" line controls the IR transmitter board; the IR receiver board is not used. It is suitable for sending binary data or ASCII characters from your robot to the field. Your robot will not be able to receive data. The game "Made 2 Order" uses this connection.

Given the 38KHz modulation of the IR diode, we want to keep the data rate as low as we can. The lowest rate to which the Cortex can easily be set is 600 baud, so that is what we will use. You will have to set this in your Cortex program.



VEX Cortex

BEST IR Transmitter

Yes, the Cortex TX pin drives the BEST IR transmitter ground pin. The IR transmitter board acts as a single component that emits IR pulses as long as Cortex Data Out pin is pulled low.

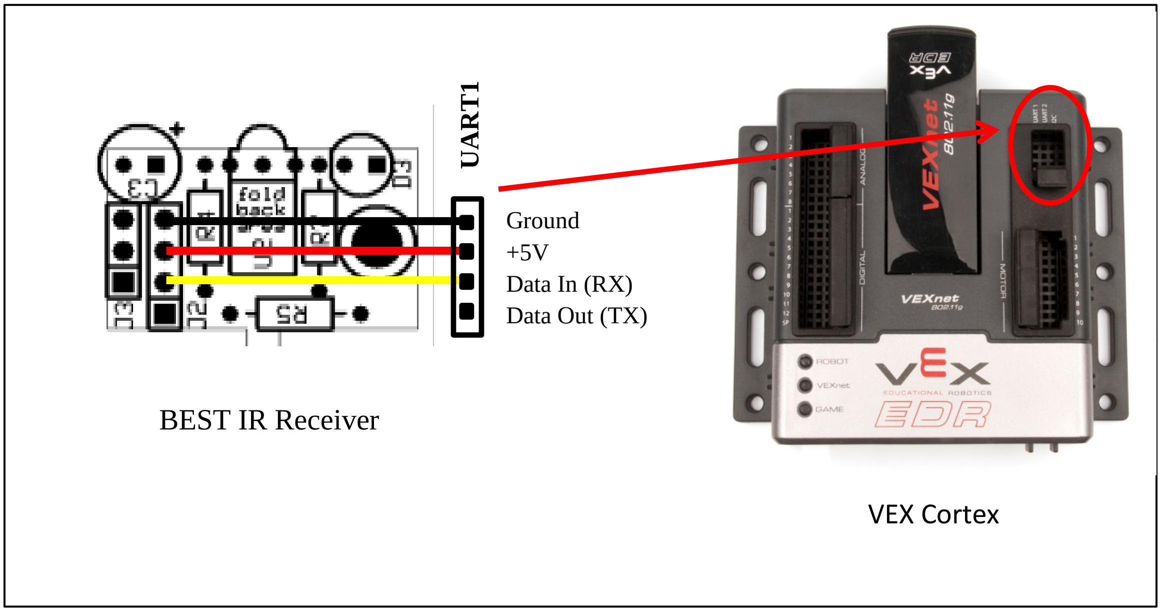
When building the BEST IR transmitter board make sure C1 is *not* installed. If by some chance it is installed, remove it by heating the pads on the backside of the board and pulling C1 out. (In this application C1 is not needed, and leaving it in causes large current spikes on the Cortex TX pin.)

For this hookup you will need a couple single M-F cables. Connect as shown in the diagram above.

Step 3-Option 3B: Using a Cortex UART port (Receive Only)

This option uses one of the Cortex UART ports. The "Data In (RX)" line receives the data that the IR receiver board detects; the IR transmitter board is not used. It is suitable for receiving binary data or ASCII characters from the field. Your robot will not be able to transmit data. This option has not been used in any games yet.

Given the 38KHz modulation of the IR diode, we want to keep the data rate as low as we can. The lowest rate to which the Cortex can easily be set is 600 baud, so that is what we will use. You will have to set this in your Cortex program.

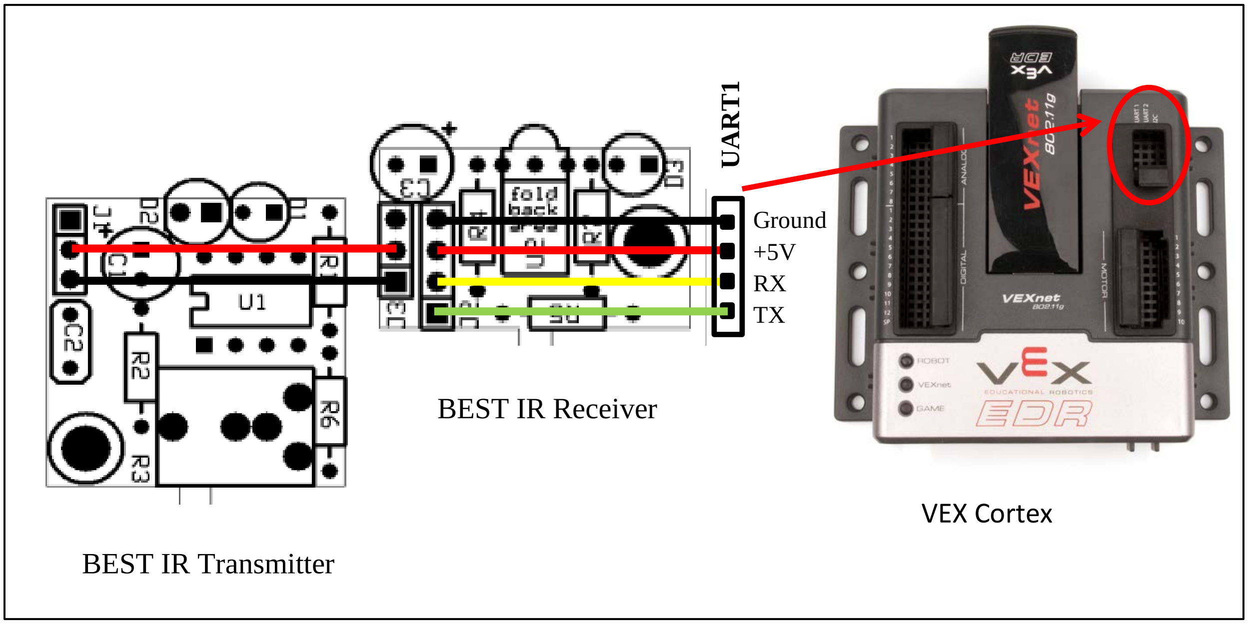


Using a 3-wire M-F servo extension cable, connect the receiver board to the Cortex as shown above. Make sure you don't end up with a twist in the cable.

Step 3-Option 3C: Using a Cortex UART port (Transmit & Receive)

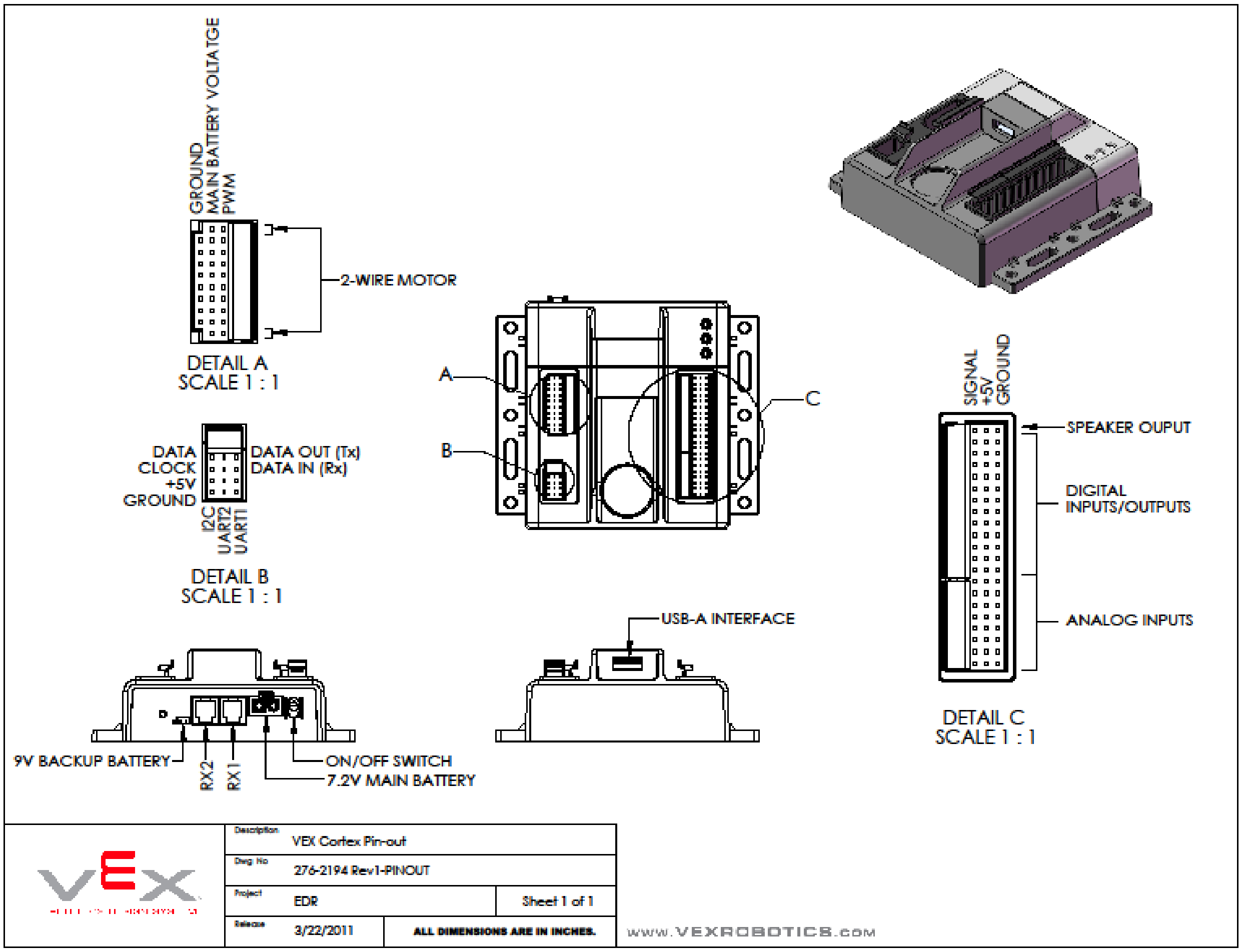
This option uses one of the Cortex UART ports. The "Data Out (TX)" line controls the transmitter board, the "Data In (RX)" line is driven by the output of the receiver board. It is suitable for sending binary data or ASCII characters from one robot to another, from a robot to the field, or from the field to a robot. This option has not been used in any games yet.

Given the 38KHz modulation of the IR diode, we want to keep the data rate as low as we can. The lowest rate to which the Cortex can easily be set is 600 baud, so that is what we will use. You will have to set this in your Cortex program.



Ideally this hookup would use a 4-wire M-F cable and a 2-wire M-F cable. Since we don't have those in the kit you will need to improvise. A servo extension cable plus a single wire will work to connect the receiver to the Cortex. The IR transmitter connects to the IR receiver using only 2 wires, so *don't* use a servo extension cable for this connection.

After configuring the UART and pointing the IR emitter at the IR detector you should be able to send data from the UART's data out pin to the UART's data in pin – the "Hello, world!" test.



Cortex Connector Pin-out

Testing/Verifying IR Sensor Communications

To test proper communications with the BEST IR Sensor, follow these guidelines:

First, make sure that you have testing the standard IR transmitter/receiver operation by following the instructions in the "Testing and Adjustment of IR Sensor System" document. This will confirm basic operation of the pair; that the transmitter is transmitting and the receiver is receiving as expected.

To verify correct operations of data communications between the transmitter and receiver, create a program to send a data via your IR transmitter connected to the UART port of your Cortex microcontroller as described earlier in this document. Use one of the options below to verify the receiver side of the communication.

Option 1: Use an Oscilloscope to see the received data

Point the transmitter at the receiver and look at the receiver's serial output data (pin x) on an oscilloscope. You should be able to see the serial data toggling high and low. Reading left to right you can see the "value" of each bit of the serial transmission. There is one start bit (high value), 8 data bits and one stop bit (high value). See the example below:

Option 2: Use another microcontroller, such as an Arduino for receiving data

Connect your IR receiver board to another microcontroller such as an Arduino. Use the example receiver code described in the "BEST IR Code Viewer" document to receive data and print it to a screen. Connect Arduino to your PC running the Arduino Integrated Development Environment (IDE). You can now send codes from your IR transmitter to this receiver and have the received codes displayed on the computer screen.

Option 3: Implement transmitter to receiver loopback using Cortex

Similarly, to option 2, implement both the transmitter and receiver code on your Cortex using one of the available programming options (Simulink, easyC, RobotC, PROS). Connect both the IR transmitter and IR receiver to the Cortex UART port using the instructions in THIS document. Point the IR transmitter at the IR receiver and send a byte of data. Your receiver code can detect the code and take action accordingly. If your Cortex is connected to your PC via USB, you can often display results to a console window on the PC (this depends on the programming option you are using).