Part b) Next, build the external circuit on the breadboard as described in Figure 12.1. Connect the input to the positive logic switch and connect the output to the headphones. DO NOT FORGET THE RESISTOR between the microcontroller and the headphones. Be sure to connect one side of the headphones to ground. *Do not place or remove wires on the protoboard while the power is on.* Since the sound is caused by the oscillations, it doesn't matter which side of the headphones go to the microcontroller and which side goes to ground. Figure 12.5 shows four stereo jacks. The jack is used to connect the headphones to the circuit. Most jacks have three pins. Luckily for us, sound will be created if we connect any two of these three pins to the circuit, in either direction.

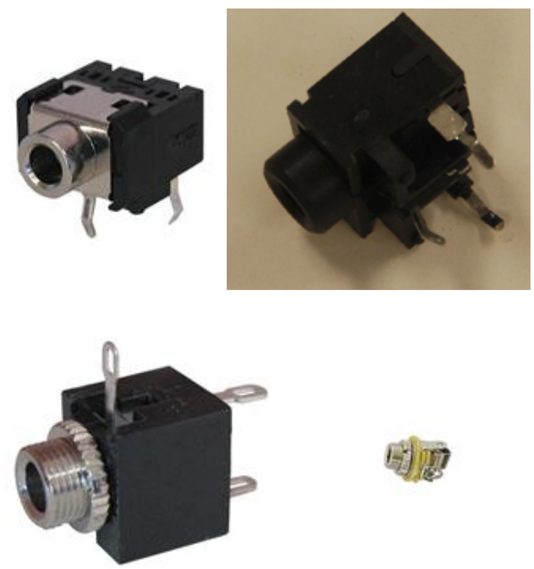

Figure 12.5. You can use either a stereo or mono audio jack.

Figure 12.6 shows one way to connect the headphone jack to the microcontroller. This figure shows which pin I connected to ground and which pin I connected to the R = 1 kΩ resistor. This jack will plug into the breadboard, but there are many other ways to build the connection.

Help

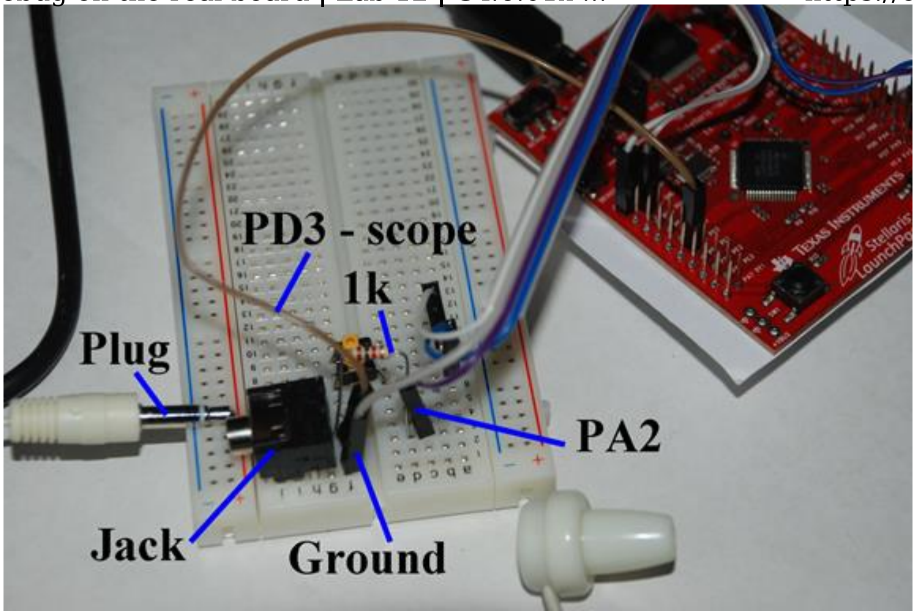

Figure 12.6. Lab 12 built with an audio jack that plugs into the breadboard.

If your audio jack does not plug into your breadboard and you have access to a soldering iron you can solder solid wires to the audio jack. The solid wires are the same type of wires used to build circuits on the breadboard. This means the other end of the wire will plug into the breadboard.

Figure 12.7 shows a system without an audio jack. This last method can be used if your audio jack does not plug into your breadboard and you do not have access to a soldering iron. It is not very elegant, but if you strip off about 3 cm off one end of a solid wire, you can twist the stripped portion of the wire around the audio plug of the headphones. Be careful not to short the two wires together.

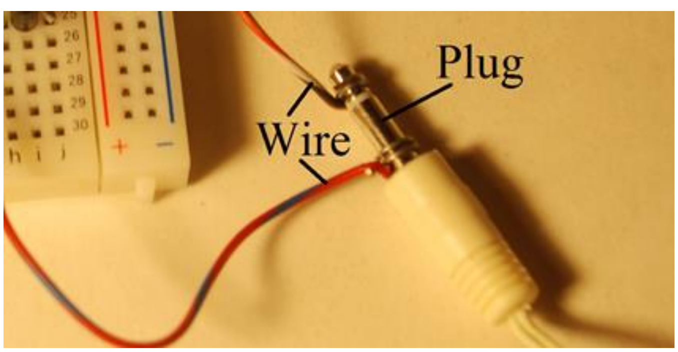

Figure 12.7. A simple way to connect the headphones to the circuit.

Debug your combined hardware/software system on the actual LaunchPad.To use the TExaS oscilloscope you need these things

1) Connect PD3 to the place you wish to measure;

2) You must be running Lab 12 (or Lab 13) project that calls **TExaS_Init** with the **ScopeOn** parameter;

3) Open the **TExaSdisplay** application. Within TExaSdisplay, set the COM port to match the board , open the COM port, and select Scope mode.

4) The TM4C123 must be running for data to be streamed from PD3, sampled at 10 kHz by ADC1, and passed through

UART0 to the PC.

When using the scope, your software cannot use or change the mode on PD3, ADC1, Timer4, and UART0. Since the scope runs on your TM4C123 microcontroller as a Timer4 interrupt service routine, you will need to leave interrupts enabled for the scope to operate. As far as scopes are concerned this one has very limited specifications (but the price is right):

1) Range 0 to 3V
2) Precision 8 bits
3) Sampling rate 10 kHz (zooming time does not change sampling rate)
4) Memory buffer 4096 samples (400ms)
5) Rising edge trigger
6) Measures average, peak-to-peak, period and frequency

DEMONSTRATION OF SCOPE ON REAL BOARD

DR. RAMESH YERRABALLI: In this video, we'll

show you a working solution for Lab 12 and its interaction

with the TExaSdisplay software.

Two things to note are PD3 is going to be our scope input, which

is similar to the previous labs where used it to measure voltage;

and the other thing here is the TExaS_Init routine that we call.

Make sure that we have the ScopeOn parameter pass to it.

DR. JONATHAN VALVANO: And the way it works

is PD3 is sampled by the A-to-D converter 10,000 times per second.

And TExaS will then stream that data out the UART, across the USB cable,

and into the TExaSdisplay application shown here.

DR. RAMESH YERRABALLI: So we will now show you how it works.

Jon is going to open the TExaSdisplay software.

And he's going click the scope input.

And right now it's showing you what it's collecting as data.

I'm going to in to interact with my solution here.

There's a single button.

A press of this button should result in a tone of 440 Hertz.

So I'm going to press it.

And I released it.

And it's currently playing a note.

And the interesting thing to note in the display

is the measurement data at the bottom.

DR. JONATHAN VALVANO: OK.

You can see that it is executing a square wave at 440 Hertz.

There's a couple of things we can do with the display.

We can push F6, function F6, and change the time scale

so that we see more data.

Or we can push F7 and see less data.

So we can zoom in and zoom out in time.

DR. RAMESH YERRABALLI: So one of the other things you can do in this lab

is test the fact that when I switch-- press the button again

it should turn off.

So I pressed it, and released it, and it's turned off.

The last thing you can do in this lab--

DR. JONATHAN VALVANO: Yeah, Professor Yerraballi,

what happens when I plug in the headphones?

DR. RAMESH YERRABALLI: So if you plug in the headphones--

DR. JONATHAN VALVANO: All right, turn it on.

DR. RAMESH YERRABALLI: And first I'm going to turn it on.

DR. JONATHAN VALVANO: All right, here we go, plugging in the headphones,

we'll see what happens.

Where'd it go?

DR. RAMESH YERRABALLI: So we notice that the signal has dropped.

It should be present.

And we see that it's attenuated.

It's almost scaled down significantly.

That's because we're measuring in load conditions, and not without load.

DR. JONATHAN VALVANO: That means when it's time to grade your Lab 12,

don't plug in the headphones.

All right, now you try it.

Help

The real-board grader is similar to the simulation grader:
0) Initialization tests will check for one input, one output, and SysTick interrupts;
1) Initially the switch is off, and it makes sure the output is low;
2) You press the switch, and it makes sure the output toggles 880 times per sec (440 Hz);
3) You release the switch, and it makes sure the output continues to toggle;
4) You press the switch again, and it makes sure the output goes low (no toggling);
5) You release the switch, and it makes the output remains low.

During the real-board grading you will have to push the external switch, as directed by the **ActionMsg** field, so that all cases are tested. This grader will arm edge-triggered interrupts on your output pin. So, when your software makes the output go low to high, the grader ISR is run. This ISR will make a very accurate measurement of the actual rate at which you are toggling. Your grader may fail if the switch bounces a lot; so touch and release the switch gently during grading.

REAL BOARD GRADER

DR. JONATHAN VALVANO: All right, let's

Lab12D Real board grader

YouTube

show you how to get a grade for Lab 12

in the real board.

We begin in edX by taking-- Control-C, Copy-- this four digit number.

We go over to Keil.

Since we're going to debug on the real board,

we make sure TExaS is the operating system.

Over in the Debug tag, we make sure that the Stellaris is the proper debugger.

That's the real board debugger.

Next we build, like we always do.

And then we download, because we're using the real board.

And now we're going to debug on the real board.

There we go.

Help

1:48 / 1:48

1.0x

About (https://www.edx.org/about-us) Jobs (https://www.edx.org/jobs)
Press (https://www.edx.org/press) FAQ (https://www.edx.org/student-faq)
Contact (https://www.edx.org/contact)

EdX is a non-profit created by founding partners Harvard and MIT whose mission is to bring the best of higher education to students of all ages anywhere in the world, wherever there is Internet access. EdX's free online MOOCs are interactive and subjects include computer science, public health, and artificial intelligence.

(http://www.meetup.com/edX-Global-Community/)

(http://www.facebook.com/EdxOnline)

(https://twitter.com/edXOnline)

(https://plus.google.com/108235383044095082735/posts)

(http://youtube.com/user/edxonline)

Terms of Service and Honor Code -
Privacy Policy (https://www.edx.org/edx-privacy-policy)