

cps-16

Tue, 8/2 5:36AM 19:36

SUMMARY KEYWORDS

adafruit, whippersnapper, projects, python, devices, newsletter, internet, board, written, data, circuit, sensors, iot, io, programming, people, component, firmware, code, called

SPEAKERS

Paul Cutler



Paul Cutler 00:00

Welcome to the circuit Python Show. I'm your host Paul Cutler. This episode I'm joined by Brett rebel. Brent is an engineer at Adafruit Industries and a graduate student at the University of Massachusetts start with, he works on all things embedded, but especially devices which connect to the internet. Brent has worked for Adafruit for five years first and the Adafruit factory within the PCB fabrication department as an intern and currently as an engineer. Brent, welcome to the show. Thanks for having me. Let's start at the beginning. What is your origin story? How did you first get involved in computing and electronics?



00:33

Those are two completely separate questions for me, because they take different forms. Computing, I got my first computer, I'm not very old. I'm 26. So like, I've listened to previous back stories of people. And they're like, Oh, I started on a mainframe. But I started on a Dell dimension. With a Pentium four, I upgraded the GPU as much as I could really. And then for programming, I got into it via scripting for Flash games. So I basically wanted to cheat at them. But I also didn't realize I was learning as I was cheating, I use the scripting engine called Cheat Engine. And you make these things called trainers for games where you like, manipulate the memory of certain things. So like, if you want unlimited health, you would essentially like freeze the memory address on the health bar, and then it wouldn't move. Or if you wanted to increase the value, you would like increase the memory, before I knew how to even program really like how to find the things I'm looking for within like an application. And then I got into programming and more so in late junior high to high school, and then more and more and more in college and got into Python as well via scripting things I wanted to do. And then I had familiarity with it when it came time to actually apply it to work.



Paul Cutler 01:42

So you're currently working at Adafruit on Internet of Things. What is IoT and the Internet of Things mean to you?



01:50

So to me, like the IoT is a way to link physical objects to the internet. So physical objects containing electronics in them. So either some type of instrumentation, like they'll have sensors that will report data on the physical world, like weather station, or they could also report data on the device internally selected devices, battery, or actuator. So things like motors or LEDs that interact with things in the physical environment. And then those things like the things on the internet, are networks and like the Internet is a network. So they send and receive data, either to or from the internet itself, or an application on the internet, or gathering data from other devices or they talk to other devices. So it's best described as a network rather than the internet because it contains like kind of nodes and graphs between devices and things that makes a



Paul Cutler 02:46

lot of sense. Adafruit is known for a lot of those products, but they also have an editor and other things service kind of like what you were touching on just now called Adafruit. Io Tell me about that



02:55

Adafruit. Io is a platform. I started working with it in 2018. But it's older than that. It's a service and a platform that is used to simplify connecting things to the internet. So specifically, Adafruit looked at existing solutions when it was made, and decided they'll write their own that has a lot of simplicity in what it does. And then they also wanted privacy, and user controllable privacy, which is something also important in the internet of things. So it's a platform that you can send data to the Adafruit IO Server, and it can be displayed on charts, you can interact with the data with dashboards, you can click a button and your device will do something. But you can also send values from your device to the service to have a graph to have it stored. And you can download the data you can destroy the data. It's your data, and it's your devices pretty much.



Paul Cutler 03:52

I've seen some really neat learn guides that show how to use Adafruit IO, I think you might have actually written a few of them. Are there any that come to mind that people should check out or any user created projects using Adafruit IO that you've seen that kind of highlight what it's capable of



04:06

the COVID lock down period was definitely or shut down however you want to view it was co2 monitoring was really hot during that time. And then any type of gardening project like Adafruit sells these capacitive soil sensors. So using those to monitor plants, and people were starting

to like, grow stuff at home and like kind of explore that they had a backyard. And those guys are always interesting, especially due to like anything if this physical process is interesting to me, because a lot of like the physical world is modeled better with the Internet of Things.

P

Paul Cutler 04:40

And the Adafruit I O is has a free tier is that right?



04:43

It does. There's two tiers, there's a free tier and it's like 30 data points a month and you can have like on our services. You can have two devices connected through this whippersnapper firmware that we'll probably talk about. And but you can only have Send feeds and then for \$10 a month, you can have more data points per minute. You have unlimited devices, unlimited dashboards, the only thing that's Unlimited is how often you can send data and receive data with the service.

P

Paul Cutler 05:11

I think that maxes out at about 60 requests a minute. Yeah,



05:15

it maxes out at 60. And then you can actually boost that value per month via like we have a system called boosts. So it doesn't get you up to the data rate of something commercial like AWS IoT, or Azure IoT, which are really meant for commercial applications of IoT projects that have fleets of sensors, right, like Adafruit. Io is meant as an experimentation platform, not as a commercial platform.

P

Paul Cutler 05:40

And if someone is looking to get into Azure IoT, Liz Clarke just published a guide a couple of weeks ago on how to do that as well.



05:46

Yeah, it's a great guide. I wrote the library like originally for work and then Microsoft came along and like did a really good pass over it and sifted and Liz's like, taken that and ran with it even further. And I'm happy to see that.

P

Paul Cutler 06:01

That's awesome. You mentioned whippersnapper, it's a new firmware. Yeah. For someone

who's never heard of what are snapper? What is it?



06:08

It's an application written for Arduino. It contains basically everything you need for your board to communicate with the Internet. And specifically, it communicates with our Internet of Things platform called Adafruit IO. And it allows your hardware to be controlled from the internet, and configured from the internet and sent values to the internet. And this is all no code. No programming involved and visual on a website.



Paul Cutler 06:31

For the no code solution. How do you define no code? What are some of those building blocks someone might see on a web page? Can you walk me through that and help our listeners understand?



06:40

Yeah, sure. So there's a few different types of like solutions for programming, right. So there's straight programming, which is text based, and you type into a computer, and it's either compiled or interpreted. So like Python, or C, in that reverse order, and then that code produces an application. There's another way to produce code, which is no code, which is starting to gain a little bit of popularity, there was a few articles around the times, and no code. The gist of why it's interesting to me at least is it increases the amount of people who can make and develop software applications. So these are people who never learned programming but want to build something. And when I was teaching a class A while ago, at New York City resistor, about circuit Python, actually, somebody working there wanted to create a monitor for their, they were a construction worker, and they wanted to create a monitor for their construction site to track I don't remember, but they wanted to track something. But the intricacies around doing an Internet of Things project versus doing a regular embedded projects were too much. So this is something for scientists, students, people who are wanting to build something, but may not have the knowledge base, like the Internet of Things requires a huge knowledge base, right? Like you need to know web development and some level, you need to know how you're going to send the data to the internet, you need to know what type of data you want to send. You need to know what you're actually sending. And you need to be able to display this on the web somehow. So like that is a huge swath of like computer electrical engineering and computer science. And it's difficult. So this like no code solution through whippersnapper and Adafruit. Io is designed to be super simple for people who couldn't build these types of projects before were stopped on these types of projects, because the level of complexity was too much.



Paul Cutler 08:33

Is there a hope that they might transition into something like circuit Python or learn programming from them?



08:39

Yeah, I hope so. Back when circuit Python was first introduced to me, one of the like, I guess pitches for circuit Python was that a student would be able to purchase a board and maybe use make code and then they'd be be be able to like graduate I guess, so to speak to circuitpython, then they would be able to use the same board and graduate to see. So this is once again, like using the same board, the same development board that that are comfortable with using and then they can move to a more like if they want to add features to it, or low power modes to it or other like they want to calculate a rolling average on a sensor measurement that they can't do and whippersnapper. They can move this project to circuit Python, but at least they proved it out. And they started to gain familiarity with the hardware and what it's capable of. Right.



Paul Cutler 09:27

So the whippersnapper firmware runs on about a dozen boards right now. I think Wi Fi is probably one of the requirements. I had an ESP s two lying around so I hooked it up and it's pretty neat. You have the ability to update the firmware right from the webpage. And then you can add, I think they're called actions or components you need to add a component first. So tell me about how you once you've got the board and the firmware installed, how do you add some of those components and then how can you add them to a dashboard on Adafruit IO



09:59

Yeah, so once you have your board set up, generally when people build projects, they'll have something in mind that they want to build. So hopefully like, or they'll have some parts leftover from a previous project, you essentially can add components visually by clicking. And these components need to be physically plugged into your board. And if they're physically plugged into the board, you can click on add a new component from the website, and it'll drop down this whole parts list of all the parts that are compatible with your boards. And we have like, nice 3d renders of everything. And we also have pictures of them. And while some things are similar, like you and I both know that a button and a switch are both digital input components, they will send a boolean value one or zero if they're open or closed. The target audience for whippersnapper might not necessarily know that and like an absolute absolute beginner, like we're talking about somebody who has the park on their desk and is looking for what to hook it up to, we'll look at the part picture and say this is what it is. So it kind of like looks like a video game inventory. And it's supposed to, you're supposed to visually identify the part with what's on the project. So you click that, and you click Create component. And what happens when you click that button that says Create component is the website over Wi Fi sends a command over a protocol called MQ TT, and it tells the board basically, hey, I would like to create an LED on pin 13. And the device will execute a command and create that pin. And then the dashboard will show that pin has been created. And you can interact with it. You can turn it on or off. Or you can read data from it. If it's an input pin, I guess



Paul Cutler 11:42

whippersnapper is currently in beta. Are there any big gotchas that people should be looking out for if they try and test it?



11:49

Every little piece of feedback, especially when beta like as far as user interactions, buttons being in the wrong place, software, crashing software doing unexpected things, typical beta things I'm interested in, there's nothing that at this point, there's nothing that's super limited, like there's some peripherals like that are in the pipeline, like one wire for the Dallas somebody temperature sensors and then NeoPixel dot store API's servo API, a display API. So there's things that are like still being worked on. But generally the service itself is like the features are there for somebody to start working and experimenting with it, and just improve over time.



Paul Cutler 12:27

Where do you see some of those improvements going next?



12:31

Sometime in the spring, I wrote a few guides, and I wrote a guide for adding a new board to upper snapper, and then adding new components. And I saw community contribution go by earlier. So like a lot of the model for this is definitely taken from the circuit Python projects where we have this open source firmware for circuit Python, and parts of the web application they can interact with. So each component that you'll see in the component pickers actually json file, and they're data driven files, which means that if you go into the repository and you add a new JSON file, it will actually add a new component to the website. So this is a way of opening Adafruit IO up to user contributions that aren't impacting like the court, like the application is quite big. But these are the types of contributions that would be beneficial for whippersnapper. So I'd like to see more community contributions as we start to head out of beta increased sensor support as we head out, maybe every sensor Adafruit makes will be pretty cool. And externally submitted sensors like from companies that aren't Adafruit, and boards that are from companies that aren't. And I also want to do low power mode sometime this summer. I think that's very crucial to IoT projects. And I think that over the last two years like whippersnappers, a pretty cool interface for configuring a device and then sending it into low power mode. And that's kind of what I want to do.



Paul Cutler 13:58

Internet of Things. Projects are really hard and finishing projects is really hard. What advice would you give someone that's starting or trying to finish an IoT type project?



14:09

I mean, the simple answer is get paid. Write like the more pressing answers like how do I finish a project and the stick with it. And this is like, crucial to finishing an Internet of Things project because there's so many different layers associated with these projects. It's not a typical electronics project, where you'll revise a PCB, but you'll maybe revise the software as well, that

runs on the device and you'll maybe revise the interface on the web that or the mobile app that the device connects to. So getting it to a good enough point, I think is important and then everything can be made faster like especially even as we worked on whippersnapper like it originally wasn't as quick as it currently is. It takes time to revise maybe you want to use a faster communication protocol. Maybe you want to change into the communication protocol, maybe you decide Wi Fi doesn't reach long enough and you want to use a different transport. Getting the project complete is important. And then revising it after I've always felt is the way to go for finishing projects. I

P

Paul Cutler 15:13

couldn't agree more. When we were preparing for the podcast, it turns out that we both have a love of newsletters, and I'll make sure to put some of these in the show notes. But I wanted to ask you for just a couple of your favorite newsletters and they can be on any topic.



15:28

I'll give you three, I won't plug the one I work for until the fourth one, that'll be the secret one. So I hit open on three newsletters, basically, every week, I was trying to think of like three I continuously hit open on one of them is the prepared if you're an engineer, I can't recommend that enough. It's written by multiple authors and usually changes every week or every few weeks. So like a civil engineer and architect people who are really into bicycles, people are into environmental city design, and has like News Analysis vectors up videos about that sector manufacturing, right. So like, videos have this really cool manufacturing process I've never seen before or like a guy making something, or underclassmen making something out of material I've never heard of before. That one's really good. That's the prepared. I like a newsletter by Alison Roman. So recipe newsletter, she gives you a story and a recipe every week. I recently got a larger kitchen. That's not also in my living room. So I've been cooking a little bit more and enjoying her recipes and like, oh, I need to actually get some equipment to cook with.

P

Paul Cutler 16:28

I definitely need to link to that newsletter. Yeah, it's good.



16:31

She has a really good burger recipe. Even if you don't like burgers, they're pretty tasty. And then the last one is Blackbird spy plane, which is a newsletter. It's written by this guy, Jonah Weiner, who said in California, so clothing, it's about design. And I like it because his voice is an author's like, super jarring. But it's really fun to read. And he kind of like uses this weird mix capitalization style. And it feels like you're being yelled at but like in the military, and it has this weird like mix of like technical lingo within the design world like aesthetic, and then also technical lingo in the military world like intelligence and reconnaissance but applied to something completely different. It's really fun to read.

P

Paul Cutler 17:11

And your fourth Adafruit newsletter would be?



17:14

Yeah, so the Adafruit newsletter, I'll plug it is I write their IoT newsletter comes out every single month, I basically gather information about IoT projects that either gets sent to me I pick up on Twitter, or I just pick up from being on internet, put it in Evernote and then convert it to Markdown and send it every month. I'll

P

Paul Cutler 17:33

share two quick newsletters and I'll link to these in the show notes as well. If you'd like to read and you enjoy sci fi and fantasy transfer orbit is a newsletter that's out there. And I forget the gentleman's name, but he does great reviews of both new books and some older books. From time to time. He's written for iOS nine and a couple other sci fi type websites and I really enjoy his reviews. He and I have very similar tastes. This is what I've learned. And it's always nice to find a reviewer that has similar tastes to you because then you know that you can trust them a little bit more as well. The next one speaking of similar taste, I've shared how much I love music on the podcast, but one is called indie mixtape was Steven heightened. Steven is a music critic, author and podcast host as well, who actually lives here in Minnesota like I do. But it's a comes out weekly on Monday on Fridays, I should say. And it just touches on some of the big new releases. There's usually an interview with an artist, but I'll put links to those in the show notes as well.



18:27

Sweet. I need more music to listen to. So that will work out very well.

P

Paul Cutler 18:32

Last question for you that I asked each guest you're about to start a new project. Which board are you going to reach for to prototype with.



18:39

Usually I have a project in mind for what I'm applying this to. I really liked the feather format. I think the feathers at this point, especially like the newer feathers, so the feather ESP 32 s three and the ESP 32 v2, like the ones with the charger and like there's so much circuitry built into them that it's hard not to reach for them and they have a pretty nice pinout I started with an Uno and the feather is pretty good for IoT projects. It has the battery, it so most of them now have Wi Fi. Probably a feather ESP 32 V two

P

Paul Cutler 19:14

and I have to ask is that whippersnapper compatible? Oh, you know it is? Of course it is Brent, thanks so much for being on the show. Thank you. Thank you for listening to the circuit Python Show. For show notes, transcripts and to support the show. Visit [circuit Python show.com](http://circuitpythonshow.com). Until next episode, stay positive