Design for Internet of Things
Prof. T. V Prabhakar
Department of Electronic Systems Engineering
Indian Institute of Science, Bengaluru

Lecture - 27
Sensors

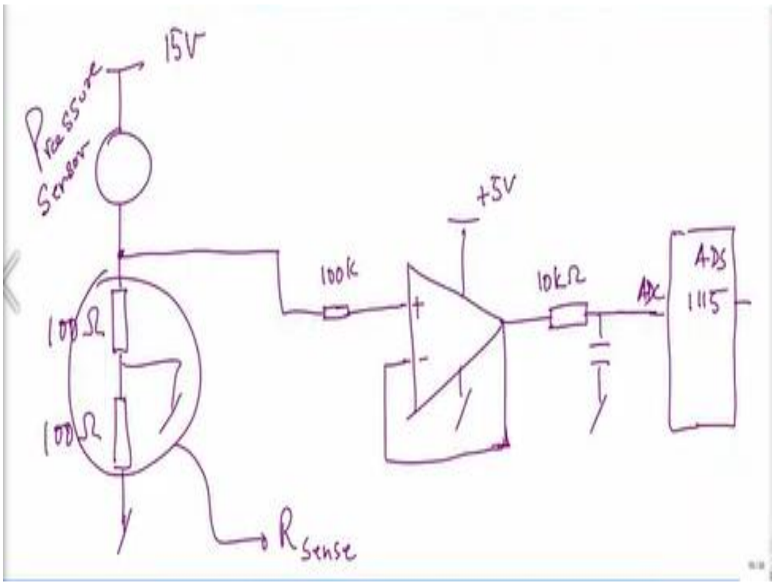
(Refer Slide Time: 00:27)

So, welcome back. You recall I showed you this pressure sensor and I had asked you also a question, what should be the V ref that you should set in order to make a proper measurement of this pressure sensor? I also mentioned to you that in the industrial world when you apply pressure sensors, many applications are there for pressure sensors and the normal way of measuring of, the measurement of pressure sensors happens to 4 to 20 milli ampere current loop.

This is the key. And this is actually giving you 4 milli amperes when the pressure is 0 bar and it will give you 20 milli ampere under full pressure. And I gave you the datasheet of this I explained this what is the maximum pressure that this pressure sensor can measure? So, that you already know. Now, it is not going to be enough if I just tell you that much. I must show you a demonstration of the 4 to 20 milli ampere current loop.

And explain to you and then point you to some literature as well which will allow you to build circuits of your own. So, for that what I did was, I will draw your attention to this circuit which I drew here.

(Refer Slide Time: 01:48)



Which is essentially what we have done on the board. You can see this is our familiar pressure sensor, the one I was holding in my hand, I have applied 15 volts to this pressure sensor, it has just two wires folks. It has just two wires you can see again hear that there are just two wires, one is you give it to 15 volts here, this is your output current which is coming. I am measuring this current through this resistor.

Because this is the R-sense, this is the R-sense of the pressure transducers and I will get some voltage already from here and now that voltage that you get here, for 20 milli amperes you will get a full voltage which should correspond to the maximum codeword of the ADC. Then you are using the full range of the ADC. When there is 0 bar, your ADC should read all zeros.

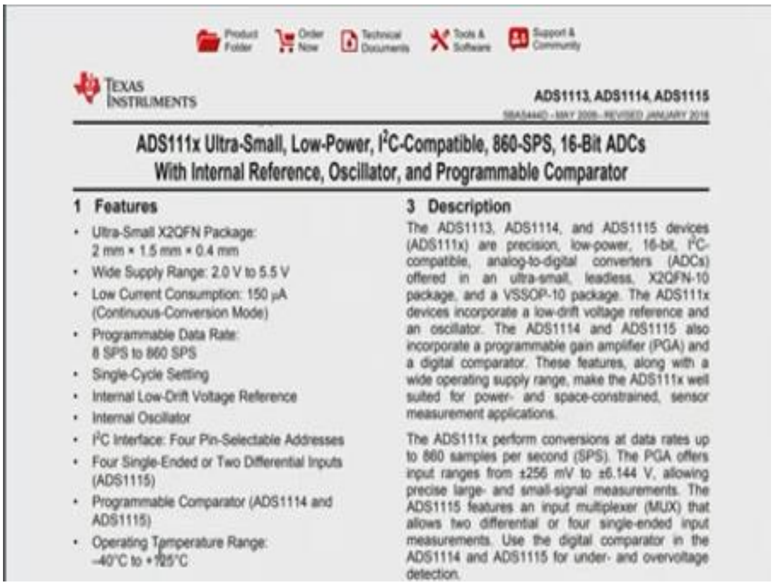
Code word should be 0. 000 let us say a three bit ADC and Code bit should be 111 when 20 milli amperes is flowing, then you know you have use the full range, that is how you should plan your sensor interface all the time to ADC. Otherwise, you will not get a good signal to noise ratio performance from the ADC. You know the basic expressions I do not go into that detail.

So, what I have done is we have put two resistors in series here 100 ohms and 100 ohms.

The centre of them is grounded and we are tapping the voltage here, and this voltage which we are tapping is essentially nothing but the current that is flowing out of the pressure sensor and that current is directly proportional to the pressure which the pressure sensor is actually seeing. That one we are using a simple voltage follower here with an Op-amp and this is using a microchip's MCP6V02.

And I will show you the schematic anyway it is out in the open source you can have a look at it also. And then you have a simple filter and then that is directly fed to the ADC of the ADS1115. Now why we chose ADS1115 and so on.

(Refer Slide Time: 04:11)



Product Folder | Order Now | Technical Documents | Tools & Software | Support & Community

TEXAS INSTRUMENTS

ADS1113, ADS1114, ADS1115

ADS111x Ultra-Small, Low-Power, I^2C-Compatible, 860-SPS, 16-Bit ADCs With Internal Reference, Oscillator, and Programmable Comparator

1 Features

- Ultra-Small X2QFN Package: 2 mm × 1.5 mm × 0.4 mm
- Wide Supply Range: 2.0 V to 5.5 V
- Low Current Consumption: 150 µA (Continuous-Conversion Mode)
- Programmable Data Rate: 8 SPS to 860 SPS
- Single-Cycle Setting
- Internal Low-Drift Voltage Reference
- Internal Oscillator
- I^2C Interface: Four Pin-Selectable Addresses
- Four Single-Ended or Two Differential Inputs (ADS1115)
- Programmable Comparator (ADS1114 and ADS1115)
- Operating Temperature Range: –40°C to +125°C

3 Description

The ADS1113, ADS1114, and ADS1115 devices (ADS111x) are precision, low-power, 16-bit, I^2C-compatible, analog-to-digital converters (ADCs) offered in an ultra-small, leadless, X2QFN-10 package, and a VSSOP-10 package. The ADS111x devices incorporate a low-drift voltage reference and an oscillator. The ADS1114 and ADS1115 also incorporate a programmable gain amplifier (PGA) and a digital comparator. These features, along with a wide operating supply range, make the ADS111x well suited for power- and space-constrained, sensor measurement applications.

The ADS111x perform conversions at data rates up to 860 samples per second (SPS). The PGA offers input ranges from ±256 mV to ±6.144 V, allowing precise large- and small-signal measurements. The ADS1115 features an input multiplexer (MUX) that allows two differential or four single-ended input measurements. Use the digital comparator in the ADS1114 and ADS1115 for under- and overvoltage detection.

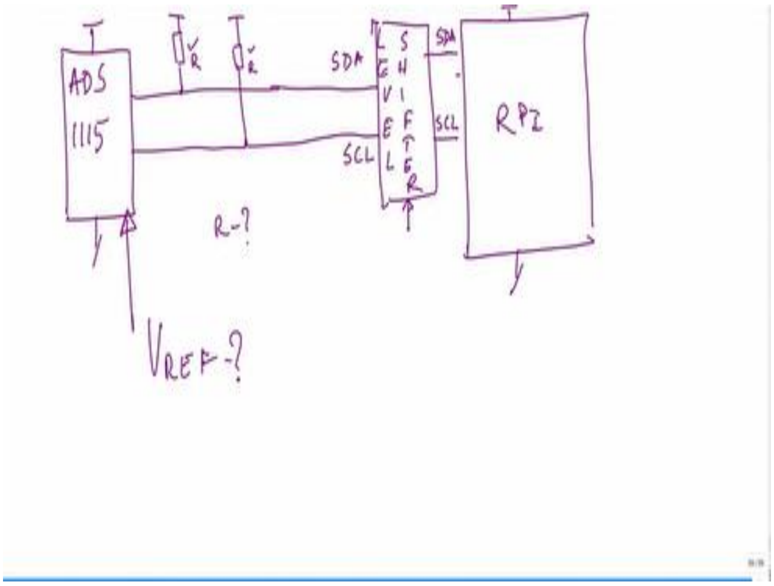
I will show you the data sheet. It provides you I2C interface to any controller. You can see it is I2C compatible. It offers you 860 SPS, each channel will now actually read 200 SPS roughly because it is, there are four channels you will get 200 SPS. So, if it is 200 SPS you are inputs signal should not be more than 20 SPS, this is the key point. So, this is how you would have want to choose this ADC.

Think about it this way folks. Input frequency can be about 80 SPS. Your system can be bought 80 SPS, so that if you do 10x it will become 800 SPS, so not more than 80 hertz signal for a single ADC channel. If it is, you are using all the four it is about 20 hertz channel or 20 SPS. So, SPS mean samples per second which is nothing but hertz, so 20 hertz. So, your input signal if it is not exceeding from 0 to about 15 to 20 hertz, low frequency four of them choose this chip that is all we are saying.

So, here you are, it is ultra small X2QFN, wide supply range, you can apply to 2 volts to 5.5, low current consumption- fantastically low, when it is in continuous conversion mode. Programmable rate 8 SPS to 860 SPS, single cycling, internal load drift voltage reference very important, we discussed a lot about the voltage references you can see that this chip has extremely low voltage reference internal oscillator very good.

It can work by itself it can generate the clock by itself and it can energize the sample and hold of the ADC all by itself using this clock which generates an I2C, 4 pin selectable addresses for single ended or two differential inputs that is fine and so on. So, now to see how to work with this ADS. Let me take you to just modify this part of the circuit. This is your digital output. So, I will say digital output.

(Refer Slide Time: 06:47)



When you say it is digital output you are going to get bits. So, here your ADS is there, 1115 and you get your digital output here. Now, how do you get your digital output onto a controller like Raspberry Pi for instance? You will use the I 2 C interface. Typically, I 2 C interface requires to pull up resistors collector outputs and there is a specification on what should be the value of this R values, what should be the R value?

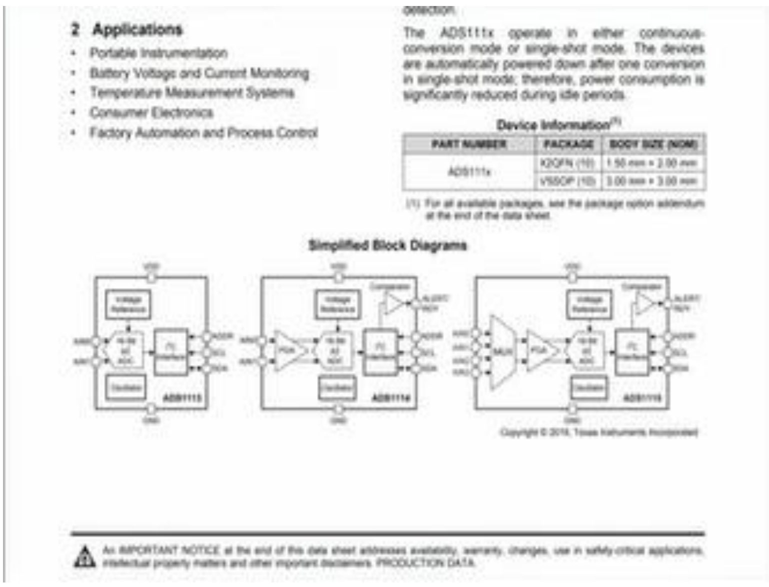
You have to look up the datasheet and apply them so that it will provide you a certain amount of bus capacitance across. One of them will be a SCL the other will be SDA. One is a SCL the other will be SDA, one is serial clock, the other is serial data. And you have to adhere to the specifications as laid out by ADS 1115 when you are interfacing to the Raspberry Pi. Hopefully if the operating voltages of both of them are matched, you do not need a level shifter between the two.

Otherwise, you may have to put a LEVEL SHIFTER.. This will give you proper level shifting that you can perhaps perform and this itself will have to ensure that it takes one voltage level on one side and then shifts it accordingly to the other side and then feed it to Raspberry Pi. Usually this can be handled need not be there if the voltage domains of both of them are almost the same.

You does not need to worry about, anyway this is a side information. So, now data is going into this system and you have to now adjust the digital data that is coming in will have to be taken up and for further processing. Now question is how do you configure the V REF of this? Because this is an ADC you have to configure its proper V REF for it. How do you configure this V REF? For that what you should do go back to the data sheet.

See folks, you had to keep referring to the data sheet as many times as possible, so that you understand it in a proper way on how to interface the pressure sensor.

(Refer Slide Time: 09:35)



So, let us go down this is not a successive approximation ADC, this is a 16 bit sigma delta ADC, this is very important. It generates a V REF which is internal to it, the internal voltage references there, which is what you would be using. However, you should scale it to proper V REF. Please note I mentioned that you must get its maximum output when it is at 20 milli amperes, and it should be minimum at 4 milli amperes the code word that is generated. So, how do you adjust that? That is the question.

(Video Starts: 10:13)

So, for that, let us go to this section on detailed description 9 and inside 9 there is this section called register map. So, go to register map it will take you directly there and then you start looking up this register map. And then what you do is you look up the two figures that you see here, there is a finger pointer 34 and in that figure number 34 all the details are mentioned. Configuration register figure number 36 you can pass through all the figures that you see.

In importantly look up the position related to 10 which is the PGA 2 is to 0 which essentially configures the programmable gain amplifier. So, now let us see what PGA 2 is to 0 is which is what is shown here. Now, you see folks you come directly to the point of saying that these bits set the full scale range of the programmable gain amplifier. These bits serve no function on any other chip but it is very specific to this particular chip.

If you write 000 then you are configuring the FSR to plus minus 6.144 volts and similarly 001 will correspond it to 4.096 volts and so on.

(Video Ends: 11:50)

From our experimentation we found that the maximum voltage that you get is 5 volts for 20 milli amperes. So, you can see you look here, you will see that you will lose out if you choose 001. But you will be able to capture well, if you choose 000 that means it is a little bit higher than what is expected your code words yet wont saturate, you are still within the range, you did not have used it fully I agree, but this is the closest you will get.

If you choose the other one you will actually lose the full range in fact, it will not deal saturate at 4096 and then you may actually lose the full 20 milliampere range. So, that is a little bit of a compromise folks, you cannot do anything. So, the point is you are to go and configure the ADS to use this programmable gain which will allow you to set it to the required full scale range.

Now, let us do a further demonstration to see if we are able to show you 4 milliamperes minimum and back to the same demonstration to see whether the for milliamperes is coming and how well it is translating to the actual value that you are looking at the pressure, you are interested in the pressure ultimately.

(Video Starts: 13:16)

One thing you have to note is how do I write the script and how do I see whether this value has been written correctly. So, let us see the Raspberry Pi screen, you will see this bit pattern which is 11000001. Those three zeros clearly indicate that you have chosen FSR which is equal to plus minus 6.144 which is essentially saying that this is what we have programmed and you must program this over I2C. So, that ADS gets configured in an appropriate manner.

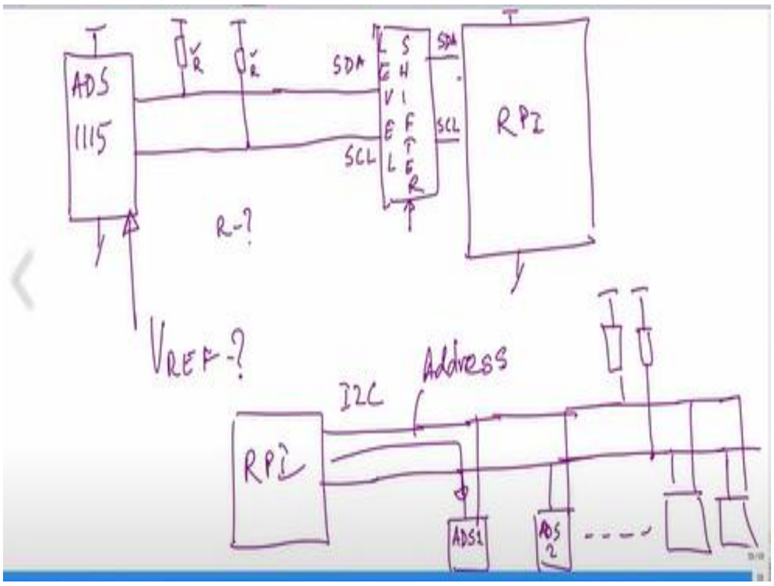
The script essentially which we see here can be scrolled. And you can see from top to bottom of the script. Essentially this is a Python script which is running all through and this is essentially one that we are used which is running on the Raspberry Pi. So, that it can configure the ADS chip accordingly. It can also acquire data from the ADS using the script essentially your read the I2C driver application is also integrated into this Python script.

So, it can read the data and just point out to the I2C part of the code. Yes, there you are read ADC int, channel 1 up to int ADS_F. And then you can see that set pointer to configure register. So, you have seen that 6.144 channel gain has been set and also the I2C part. Just highlight the I2C part of the code which will help us, there you are that is the portion of the I2C part.

You can see that the I2C part essentially is looking at supposing you have multiple ADSs on the bus. See I2C is a beauty because I2C allows you to if you go back to this picture.

(Video Ends: 15:18)

(Refer Slide Time: 15:19)



I2C says that, if this is the Raspberry Pi, R pi, and this is the I2C bus you can have – it is a multi drop bus, you can have one chip and you can continue to have one more chip and 2 points here. So, you can cascade as many of them. So, the command from here actually contains the address. Now what you see is this address. For example, x48, x49 x4B and x4A are four I2C ADS chips, which are connected onto this bus.

I will simply call it ADS 1, ADS 2 and up to ADS 4. This is the ADS 4, and you can have ADS 3. And you can also have ADS 4 not both will come from the same one, this will come from here. And I am assuming that there is a requirement for one pull up here. And there is one pull up requirement here. Just two resistors but you can support ADS 3 this is ADS 4. So, what are the addresses?

(Video Starts: 16:50)

How does the ADS chip know which one it is trying to pick? That is shown here 48, 49, 4B, 4A, are the four ADSs which are integrated. So, this Python script folks, is that complete system beautifully written by our colleagues Vasant, Prajwal and Abhishek so on. All of them have put efforts in trying to acquire data from the ADC chip by configuring the ADC reading the pressure sensors.

(Video Ends: 17:13)

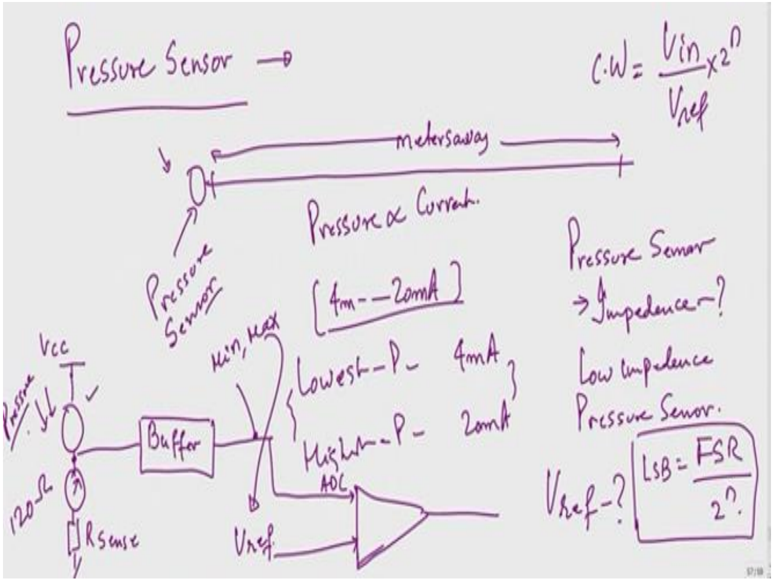
And also, this script can do other things, it can do actuation of you know the solenoid valve and all that but let us not get into that at the moment. So, it is this is essentially the way it is done. So, now what is pending is surely one part and that is am I getting at least close to four milliamperes when I have zero bar. When there is zero bar, I must get close to four milliamperes.

(Video Starts: 17:50)

For that what I will do, because I am not using a very expensive multimeter and that multimeter will start loading the system. I will run it first the reading will come here when I connect this probe like this. So, let us run it now and then let us see what the demonstration is. Let us start switch on the multimeter. This is the current meter which is connected in series folks, here is a riddle question.

(Video Ends: 18:17)

(Refer Slide Time: 18:18)



Let me put back this part two you were how do you think we are measuring the current? We are measuring the current in this by putting a ammeter here. Very simple folks, I hope you are following the logic. This is R sense value, this is the R sense and this is the current measurement. Now, let us see how this goes. Let us start the measurement.

(Video Starts: 18:48)

And let us see what is actually happening. So, you see now that it is actually measuring zero bar because there is no pressure that is coming up here. So, now let us switch on and see whether this value can change. So, let us see what actually it reads. You can see sometimes it reads higher about 7 milliamperes it reads about 6.7 and then even you switch off it goes to back to zero bar. So, it should go back to 4 milli amperes.

So, now you have 3.85 then it reads quite well. Now let me switch on again to show you that it is reading some pressure. Now how do you know what is the pressure? It is only giving you something close to 7 milliamperes, for that look up the data sheet. In order to figure out what is the average pressure that it is actually if it is measuring close to 6.5 or 7 milliamperes in the 4 to 20 milli ampere current loop.

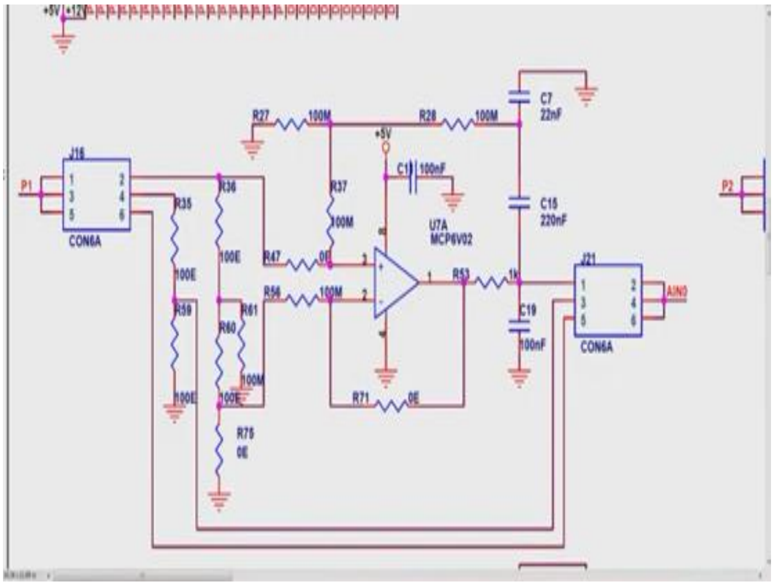
(Video Ends: 20:00)

So, now it is time to for you to design 4 to 20 milliampere current loop circuits by yourself. Let me point to one nice resource which you will be able to use, 4 to 20 milliampere transmitter circuits are already available as chips you can buy from TI, you can buy from so many other silicon vendors, receivers are also available, receiver circuits are also, chips are available. So, actually the 4 to 20 milliampere transmitter will actually have a very stable voltage reference.

Voltage power supply, and then from there it will derive the required current. And so that is the main heart of the 4 to 20 milliampere transmitter. So, then after you generate 4 to 20 milliampere, I mentioned to you that you can take either differential input, or you can take single ended using a single resistor you can take and use it through a buffer and all that. Where did we get all these circuits from?

Where do you think is our starting point for all of them? That is where I want to point you to look at our circuit here.

(Refer Slide Time: 21:10)



Which is shown here, you can see that this CON6A, is indeed the connector where you connect the pressure sensor. This is the same one that I showed you here 100k and there is a voltage follower and then there is some small filter and then it then goes to the ADS chip. So, where did we get all these ideas from?

(Video Starts: 21:31)

How do you design these ideas? So essentially, this is the application node that you may want to look up. Implementing for 4 to 20 milliampere sensor interface, it is easily available on the internet, you can pick up this and he gives a beautiful example of an oxygen sensor and how you have to use this oxygen sensor, gas sensor for checking for the safety of humans. And he describes the 4 to 20 milliamp output general characteristics.

Shows how a 20 million pair current loop is put in place, there is a transmitter, then there is a receiver and the current is flowing through this loop. If there is an interruption of the loop, you know that there is a problem with the sensor. That is the reason why they do not use zero milliampere if there is a problem in the loop, if it is chosen 0 to let us say 16 milliampere, you are in trouble because the receiver thinks that it is reading 0 milliampere.

But actually, there may be a problem with the wire, in which case also you will get 0. Therefore, it is always good to start with a nonzero value. That is why they choose 4 milliamperes. So, 4 milliampere mean 0 bar in our case, so always there will be a 4 milliampere loop, current loop that is running. And how to calculate the sense resistor is also shown here.

Then I showed you about the programmable gain amplifier settings for ADS which is scaling it accordingly. And then it is connected to the ADC. This is what I was telling you about adjusting the V ref essentially giving a gain to the voltage reference using a programmable gain amplifier. That will give you an FSR of 6.144 with this programmable gain amplifier, and then it is connected to the ADC and so on.

Please read this carefully. And you will understand all the bullets with respect to this 4 to 20 milliampere current loop configuration.

(Video Ends: 23:42)

That is all I have for the moment, folks. Hope you enjoyed the whole process of connecting a sensor to a system. And then how to configure the ADC in an appropriate manner. So, that you get the full range, full scale range of the sensor and its ability to improve the signal to noise ratio of the ADC itself because you are using the full range. Thank you very much.