Embedded System Design with ARM
Prof. Indranil Sengupta
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

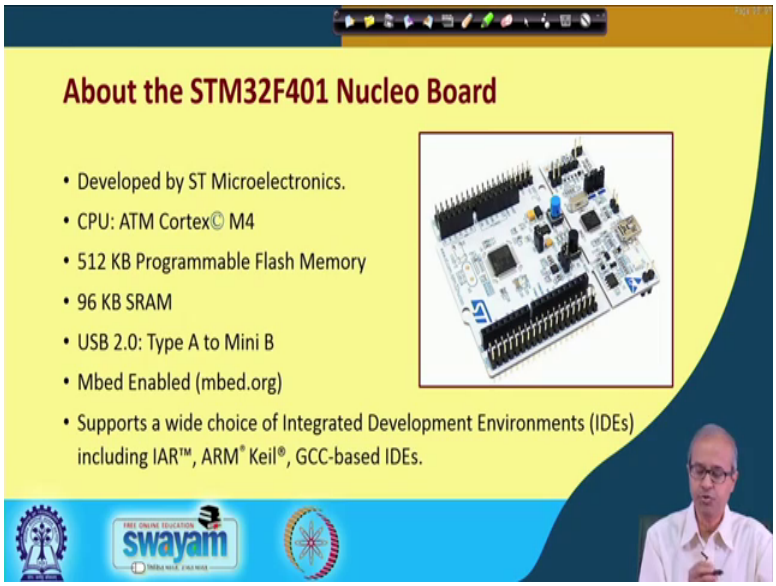
Lecture – 10
About the STM32F401 Nucleo Board

So, in this lecture we shall be introducing you to one very popular microcontroller development board which is used in embedded system design. The name of the board is STM32F401. This is the title of this lecture, About the STM32F401 Nuclear Board.

(Refer Slide Time: 00:42)

Now, in this lecture we shall be telling you some notable features about the board. So, what are the connectors and pins which are available and some of the specific details. Subsequently when you do the experiment when you see the demonstrations you will be able to appreciate exactly how we are doing the things out there, fine.

(Refer Slide Time: 01:10)



So, this STM32F401 is a board you can see the picture here. This is manufactured by a company called ST microelectronics. This board as you can see contains a lot of small components. Here you can see a larger IC chip here, this is the actual microcontroller sitting inside and there are many other accessories.

So, you see the main features which are mentioned here this board is manufactured by a company ST microelectronics. They do not manufacture the chip, they manufacture the board, and the CPU that is use this middle chip that I am showing the central one, this is ARM Cortex M4. This is the ARM processor which is used and the name of the processor is ARM Cortex M4, right.

Now, the feature of this ARM Cortex M4 board is inside it there is 512 kilobytes of program memory. Now, here they have used flash memory as the program memory they have not used a ROM because programming a ROM is quite troublesome. Flash memory you can program it on the fly, you can download it, you can stored in flash. And flash is a non-volatile memory, even you switch off the power the program will not disappear, and there is also some RAM – random access memory which you can use to store data. So, there is 96 kilobytes of that.

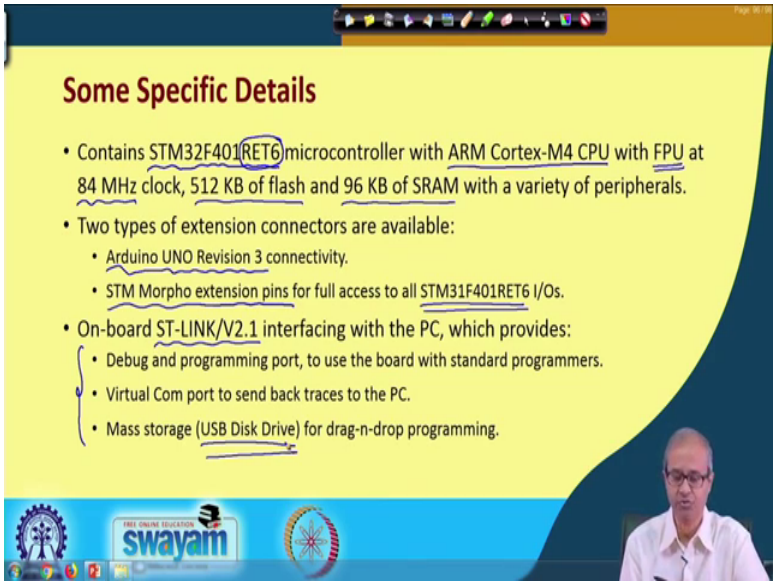
There is also an USB interface it is out here through which you can connect this board to other devices, like your PC or desktop your laptop you can connect to it, and this board is embed enabled. Well, we shall be seen these things in more detail later. Well, embed is

like a platform if a board is embed enabled then all the tools that are available under embed you can use along with this board.

Well, embed is a set of tools which are primarily meant for the developers of this embedded system boards and this IoT internet of things kind of applications and they have developed a lot of utilities which make the process of development very easy, ok.. And, both there are some features which supports a wide choice of integrated development environments which are called ids.

There are several software based interfaces through which you can access these boards, you can program these boards, use this board and so on. This we shall see later because in our experiments we shall be using this embed dot org these kinds of tools. There we shall show how to interface, how to use and how to program this ARM device which is inside this board, fine.

(Refer Slide Time: 04:42)



Now, some specific details some more details are mentioned here. Now, in this board as I mentioned we have STM32F401 microcontroller there is a version RET6. This contains ARM Cortex-M4 CPU with lot of other things. There is a FPU which is running at 84 megahertz clock functional processing unit, 512 kilobytes of flash I already talked about 96 kilo byte of SRAM and a lot of IO ports. And other peripherals they are provided which makes embedded development very easy.

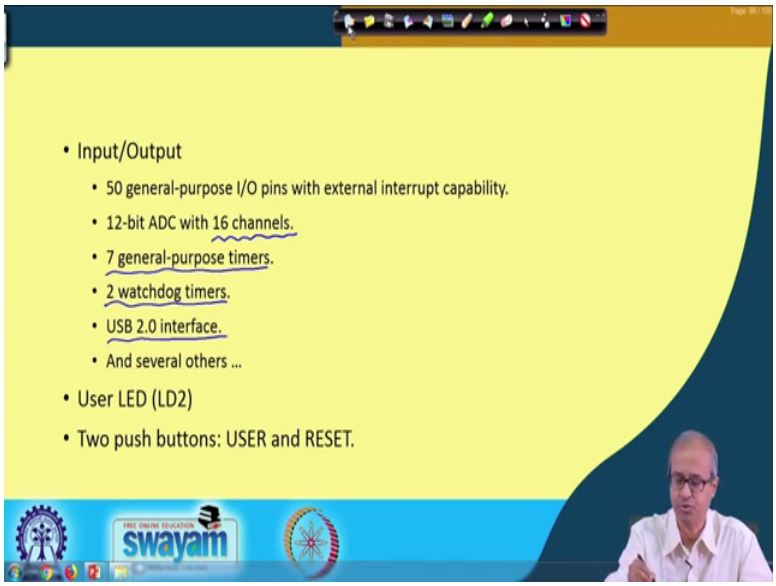
And, this we shall be showing there are two types of extension connectors that are available as part of the board. One is Arduino UNO Revision 3 compatible. You see Arduino is a very popular platform for developing small embedded system design today. You say Arduino is a very popular processor which is computationally very simple not as powerful as ARM. But, because Arduino is so popular when you are developing something around an ARM processor, well, it may so happen there may be some other subsystems with which you have to make connections which are based on Arduino. In order to just enable that this board also provides an Arduino some kind of connector I shall show you.

Now, in addition to that all the signals that are provided by this board, this STM this is called STM Morpho extension pins they provide you full access to all the input output pins of this STM microcontroller that is sitting inside the board, but and this is something which would have which is already mentioned that inside the board. There are some inbuilt interfaces which are provided that allows you to interface with several standard integrated development environments using which you can run some debuggers from outside, if your board is not working properly you can write, you can test or diagnose the board using those debuggers.

You can use a virtual serial port virtual com port through which some of the data you can send back to your PC and displayed on your PC screen. And, just like whenever you mount something on a PC or laptop you see that device mounted as a drive this nuclear board also when you connect it to a PC through this USB board which is provide USB connector which is provided this board also gets mounted as a USB disk drive and when you program this board you simply drag and drop that program into that disk drive icon that program will automatically get downloaded you see.

This driver is automatically installed as part of the environment. So, it becomes very easy for the developer, ok. All these things we shall see later when you show you the demonstration.

(Refer Slide Time: 08:25)



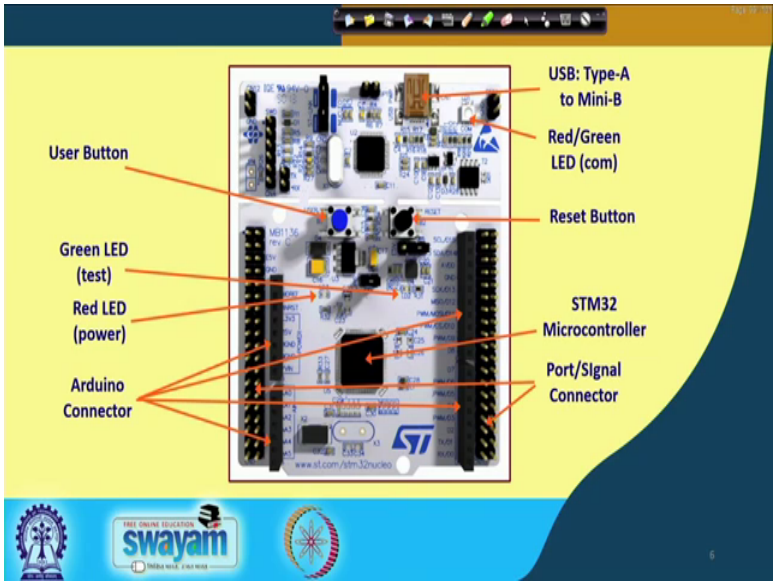
Now, regarding the external interfaces a very quick summary is shown here. Regarding input output ports there are fairly large number of IO ports that are provided, 50 general-purpose IO pins are provided. Now, one good thing is that all these IO pins can also be used to connect with external interrupt sources. There are some constraints we shall be discussing this later, but the thing is that if we want to implement an application where some external devices sending an interrupt you can connect it to any of these 50 pins.

This is unlike some older processors let us say the processor 8085 which many of you may be knowing this 8085 had some dedicated interrupt pins there are some pins called trap RST 7.5, 6.5, 5.5 and INTR there are five interrupt lines which are dedicated. So, when you wanted to connect an interrupt device interrupting device you have to connect to one of these five devices five lines. But, here you can connect anywhere some pin you can use as a general purpose IO you can also use as a destination of an interrupt you can connect an interrupt line to that also, right.

And, another thing is that there is a built in analog to digital converter. You see AD converters are very important for embedded system design because many of the external parameters that you want to sense they are analog in nature. They are continuously varying parameters which can be translated to voltages. So, in order to interface them if we have an ad converter on board it becomes very convenient.

So, inside the board there is a 12-bit AD converter and you can connect up to 16 input devices to it 16 channels are supported, right. There are several timers counter timers provided on board. There are 7 of them, two of them are watchdog timers, 2 watchdog timers are provided and I mentioned there is a USB interface, USB 2.0 version interface. This is primarily used to connect with a host system like a PC or a laptop, and there are many others, there are some LEDs, some push button switches, these are also there as part of the port.

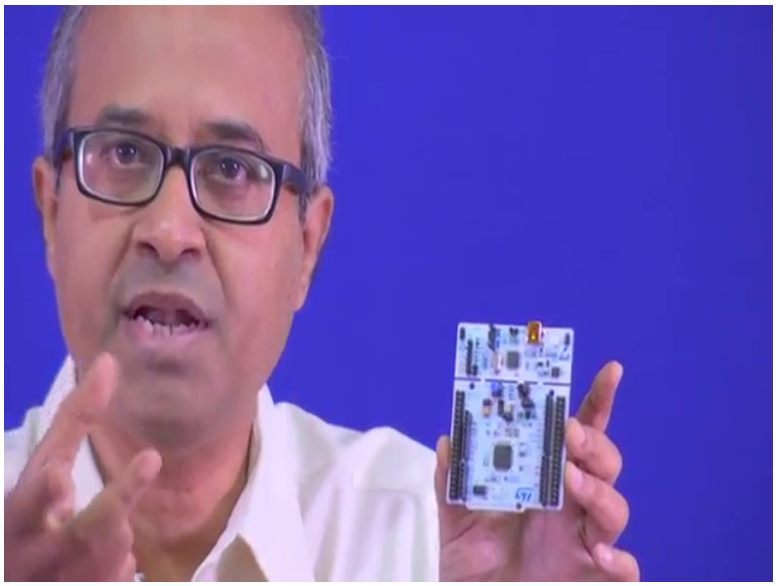
(Refer Slide Time: 11:13)



Now, here you have a picture of the board. Now, let me show you this board also I have this board with me.

(Refer Time: 11:25).

(Refer Slide Time: 11:55)



So, you can see this board. This is that nuclear board which we are talking about. So, if you can see the two sides of it, there are so many connectors you can see. There is some black smaller sized connectors and at the internal part this you can see, these are the Arduino compatible connectors. So, if you look at an Arduino board they will have a connector with an exactly shape like this you can connect a cable from there directly to this. But, if you see the other pins available on the two sides you see there are so many pins available you can see this pins. This pins provide you with the signals that the internal microcontroller on board is generating. So, all the signals you can you can access either from these detailed signal pins or a small subset of that is available over these Arduino interface pins, you can also avail it from there.

Now, you can see there is a USB interface out here at top from here you can connect a USB cable, ok, this you can connect with the PC this one. And you can see the microcontroller chip sitting here, this is your microcontroller and there are small other accessories. Now, let us see through this slide some of the features here what these different subsystems contain, alright.

So, now if you see here this is the same board that I had shown. Now, in this board you see this black area this black connectors these are the Arduino connectors that we were talking about, right. Now, in this board you will see that some labels are given you can see this A5, A4, A3, A2, A1, A0 these are the analog input ports; that means, AD

converter I told you there is a inbuilt AD converter you can connect an analog input directly to these ports. There are some voltage sources available 5 volts, 3.3 volts ground these are available and different input output port pins are available here. Some of them are data pins digital data pins, some of them are pulse width modulation pins; these we shall see later that what these are.

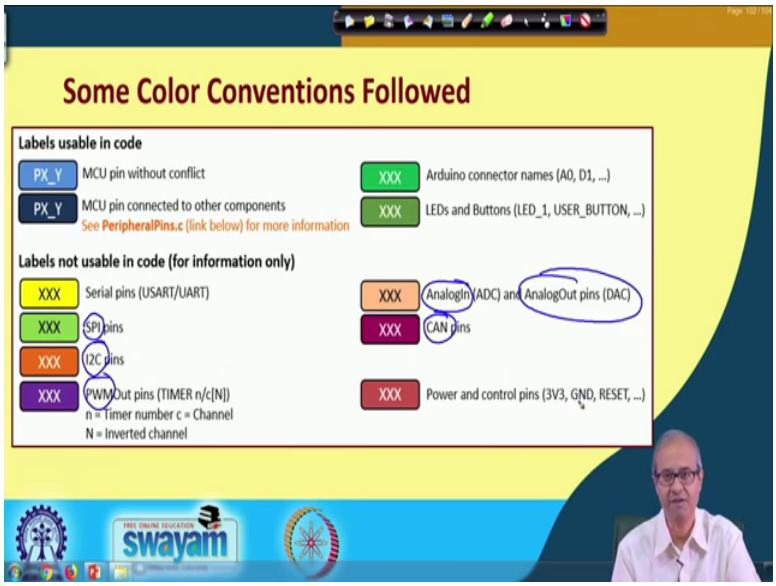
But, over this, is a this STM 32 microcontroller pins you can have access to all the detail pins because this pins are ports signal connected that had come in directly from the STM 32 microcontroller. So, all the port pins are available here. There are three ports A, B, C they are 16-bit ports and there are many other signal lines, ok. These will see.

An addition you can see on top I told you here we have the USB interface from here you can connect the USB cable you can connect it to the PC, there is a reset button sitting here you can reset the machine anywhere. And for indication there are some small LEDs connected in board one LED is here, one LED is here one LEDs here and there are some also push button switches.

So, other than the reset button there is another push switch here which you can use in a program. So, can write a small program and take input from there now the thing is that whenever you are using this board to develop an embedded application you will have to make some connections with this board. So, you must understand minimally that what the signal lines are here available over these connectors, because you will have to make all your connections through these connectors. Well, having them connected to their Arduino pins will be easier because, most of the time you will be needing only a few connections, but beyond that you may have to have access to the detailed connector pins which are available here and here, right.

So, let us see those pins quickly into have some idea that that what kind of pins are there.

(Refer Slide Time: 16:47)

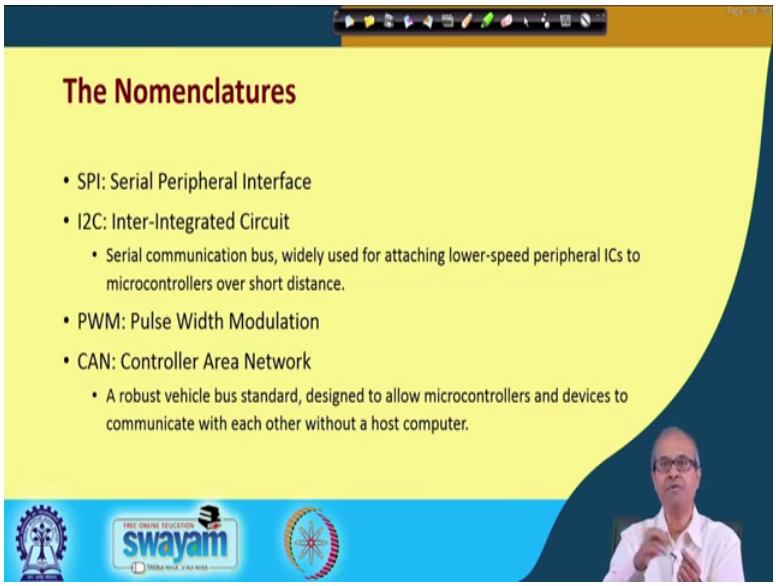


But, before that there are some color conventions which are followed in those pin descriptions. See the various colors you can see they indicate different-different categories of pins. Like this blue color pins indicate the micro controller unit pins without conflict; without conflict means they are available to the user, they are not connected anywhere else. But, this gray color ones are these are some pins connected to other components. Maybe you have seen in the board there are some LEDs some switches, so many things are there maybe some of this pins are internally connected to those devices. So, when you use them from outside you should be careful.

And, in addition this yellow ones refer to some serial pins which are connected to USART – universal serial a synchronous receiver transmitter for serial communication. Then there are some this I shall talk about what these are there are some pins which are called SPI pins, there is some I2C pins, there is some PWM pins, and this Arduino connector pins I have already mentioned. There are some which are Arduino connectors, some LED and buttons are there, there are some analog input pins, some analog output pins. Of course in this board there are no analog output pins only analog input pins are there and there is something called CAN; CAN pins.

So, I shall be seeing what this acronyms mean SPI, I2C, CAN and of course, there are some power pins 3.3 volt, 5 volts ground etcetera.

(Refer Slide Time: 18:50)



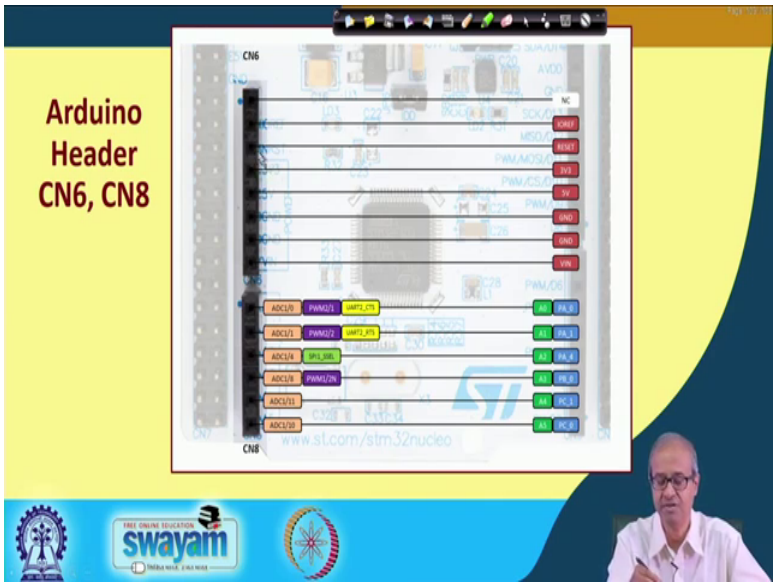
But, first let us look at the nomenclature. This SPI stands for Serial Peripheral Interface. So, you can use this pins to develop some serial communication application whenever you want, ok.

Now, I2C is a standard bus, this is the full form for Inter-Integrated Circuit; I square C actually that twice I2C it is called. This is a serial communication bus which is very frequently used to connect some low speed devices to a microcontroller of course, over very short distances. So, for embedded system application this I2C interface can be very useful, if we have a device which is I2C compatible then you have to have this I2C connection.

PWM stands for Pulse Width Modulation, this we shall be demonstrating. Pulse width modulation is a very convenient way of controlling external devices. And CAN is the short form for Controller Area Network. This is also an emerging standard, this was developed primarily for vehicles. For installing IO a for means incorporating IoT and this embedded system features in automobiles this controller area network was developed and this CAN ports whatever is available here those are compatible with that standard.

So, if we have a device or if you want to use it as part of a CAN you will have to use those signal lines, right.

(Refer Slide Time: 20:40)

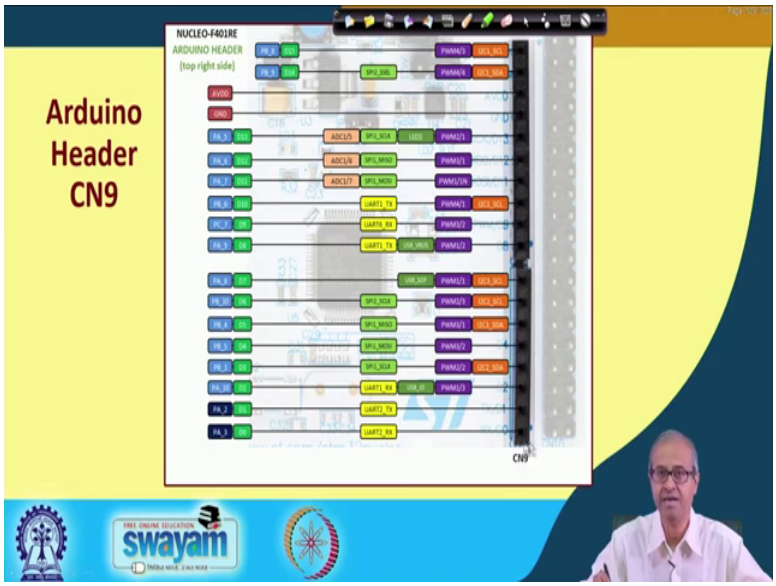


Now, let us have a very quick look at the different connectors that have already seen in the board. Many of them are very small you may not be able to read very clearly, but I am just basically telling you. On the left side of the board these are the Arduino connectors the lower one is called CN8, the upper one is called CN6 connector.

So, in CN8 you will see there are the analog input ports or provided six of them; sorry six of them are provided some of this pins are multifunctional. They can either be used as an analog pin or they can be used as a PWM pin or they can be used as a UART pin. So, as I told you in ARM the philosophy was that they do not use dedicated pins for different functions, same pin you can use for several different applications, right. So, here it is mentioned like this. And these are actually connected to these ports port A0, port A1, port A4, PB 0, PB 1, PC 0. So, some port lines are connected to these. So, you also know what these pins mean in terms of the ports.

Similarly, the upper line they are primarily power supply lines, 5 volt, ground, ground 3.3 volt, reset and so on, right. These are the Arduino connector first set.

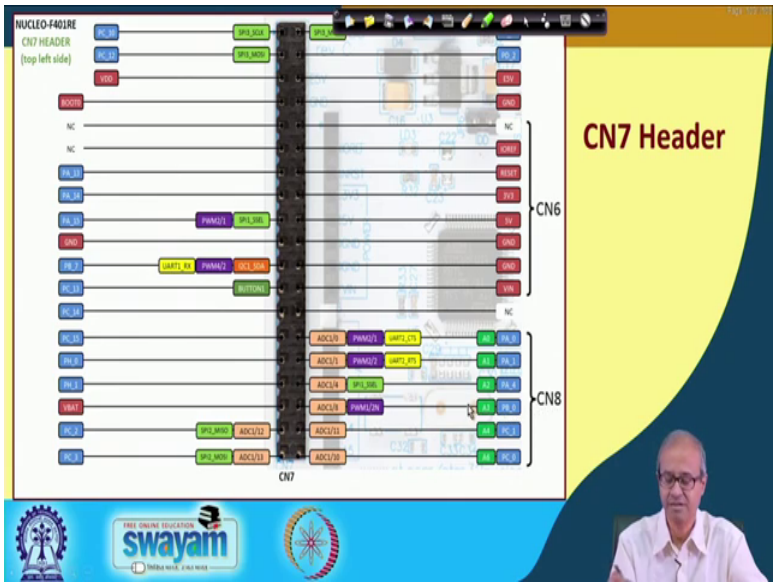
(Refer Slide Time: 22:25)



And, the second set of Arduino connectors that appears on the right side that is like this is called CN9 connector that is this is a larger sized connector. This contains many signal as you can see so many pulse width modulated ports PWM you can see there are. So, many I2C ports also some of them you can use, these are USB, UART pins, these are some additional AD converter, these are some is analog input pins and these are some port pins. So, you can also use them as some IO port pins.

So, you see these are the pins which you will be actual be using when you are developing an application; either from the outside you will be connecting it to a digital input or an output or an analog input or a PWM output these are the kind of interfaces that you will be doing mostly. So, we shall be utilizing these details because, you must be knowing which pin is what, because when you write a program you will have to specify I want to write something to port number 5. So, you must know which pin is port number 5, so that you connect the wire to the correct point, right ok.

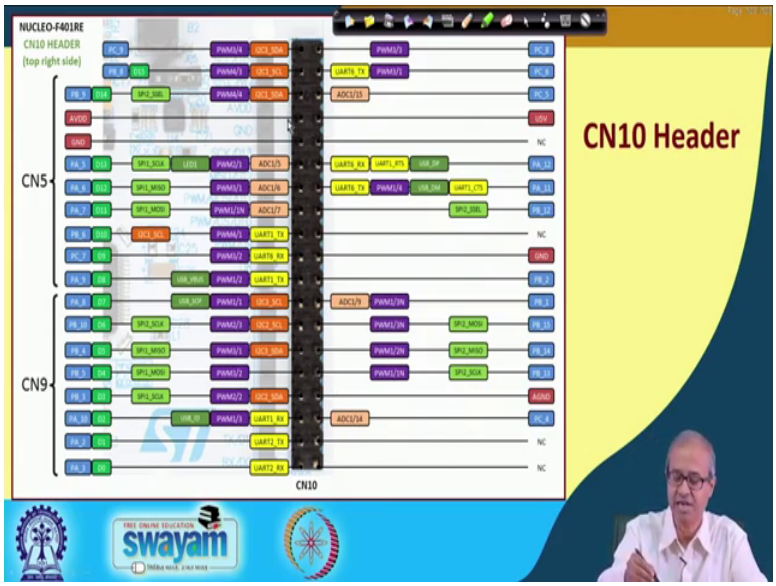
(Refer Slide Time: 23:39)



Now, these are the detailed pins that are coming from the microcontroller sitting inside the board. On the left side this is called CN7, this is the big connector those pins you have seen, and here the signals are all mentioned all port Port C, Port A. You see, Port A, Port B, Port C there are three 16-bit ports which are available in the microcontroller. So, all those 48 pins are available over these connectors.

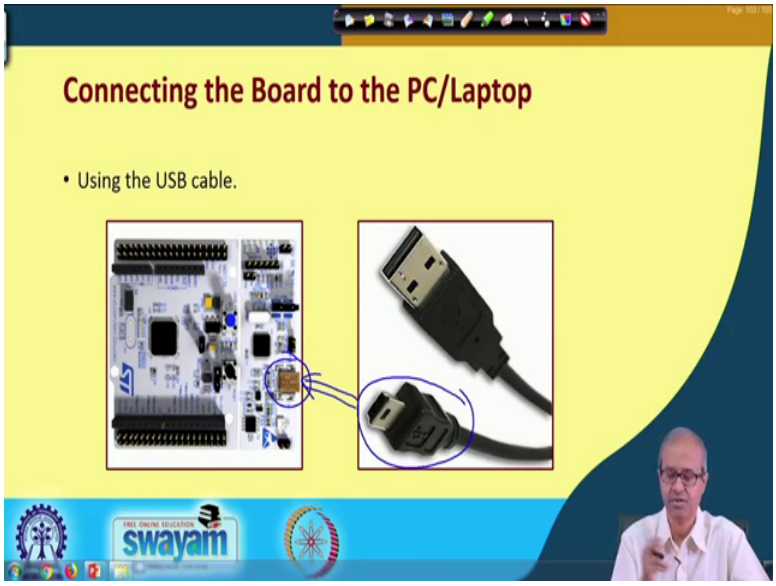
In addition there are so many other thing like you see the power supply some a part of these pins are the CN6 Arduino connector. So, a part of this pins at this CN8 Arduino connector, but in addition you have many other pins. So, I am not going into detail of this, you can see them yourself.

(Refer Slide Time: 24:36)



Similarly, on the on the other side right side there is another port this is CN10. So, here also you have the other port. So, many Port B, Port A, PWM ports, ADC ports, SPI, I2C and so many ports are there, all are all are available. So, whatever you need to use and connect you can use from this connectors. And, this connectors are available on the internet also, you can search with the name of this board you will get all the details, right. So, these you should give handy whenever you are doing the experiment.

(Refer Slide Time: 25:13)



And, just was saying that this was the board and this was the connector USB connector or you need a USB cable like this the shorter side this will be connected here and the other side will be a standard type C connector this will go into your PC or laptop. This is how we shall be doing the experiment. In the PC, we shall be developing the application, we shall be writing the program, we shall be compiling the code and with this board we shall be interfacing with all the devices, sensors, actuators everything. And, after writing the code we shall be downloading the code on the board and well, whatever you have downloaded it, it would start running, ok. This is how we shall be showing you how the things work.

So, with this we come to the end of this very short introduction of this nuclueo board. But, one thing I would strongly suggest well, we shall be telling you something as part of this course, we shall be showing you some demonstrations. But, if you are really interested to learn this embedded system design in a hands on way which is the best way to learn we strongly recommend that you should procure some of these boards yourself and some of the experiments that we shall be showing or demonstrating you should actually do it yourself in your place it may be your home, in your lab, in your office wherever you are, you should do it yourself and you should get a direct feeling of what is it to develop an application and build an embedded system.

So, we shall be continuing with our discussion. In our next lecture, we shall be talking about some of the IO port detail of this board that we have just now seen, specifically we shall be talking about the pulse width modulation ports, the interrupt lines and so on. And subsequent to that we shall be looking at the analog inputs, outputs and some other applications and demonstrations.

Thank you.