

Introduction to Internet of Things
Prof. Sudip Misra
Department of Computer Science & Engineering
Indian Institute of Technology, Kharagpur

Lecture - 22
Introduction to Arduino- I

So, in this lecture we are going to get some hands on of arduino programming, so this lecture and the next one as well. So, two lectures are completely devoted to arduino programming. So, in the first lecture, you are going to get some of the ideas about the basics of arduino and in the second one some of the examples about how the programming can be done with arduino. So, before we proceed further I would like to remind you a few things that arduino is very much popular. At present it is used in for different implementations of IoT throughout the world arduino devices are very much cheap, they are low resource consuming and that is why they are very much popular for use in implementation of internet of things.

So, in the first module, we have seen different things we have understood the concepts of internet of things, basic concepts the overall philosophy of internet of things. We have also seen that there are different types of sensors, different types of sensing possible, different types of actuators that principles behind different types of sensing, different types of actuation. We have seen that there are different types of networks that are possible for use for adoption for use in IoT, different types of communication devices standards can also be used for communicating in internet of things.

So, having understood those how can we use these concepts for building a real internet of things may be in a smart phone scenario at home to improve some of the you know daily tasks that we do at home. Or in a smart homes smart cities scenario like in a smart hospital smart you know smart transportation connected vehicles and so on. So, of are all these we need to take help of different IoT devices. And one of the very popular once is arduino. So, arduino if you have to use for the building of internet of things, you have to you have to buy this which are very cheap and then you have to program these and this is what I am going to teach you in this particular course.

So, with me actually I have Mr. Anandroop Mukharjee who is going take over and Mr. Mukharjee is going to take you through the hands on of arduino programming through the

from the starting from the basics to the little bit moderately advance concepts of arduino programming.

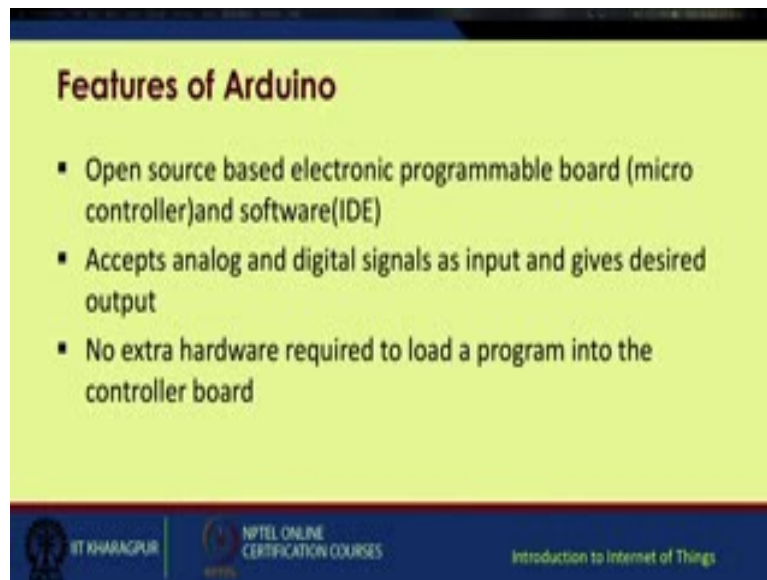
(Refer Slide Time: 03:33)



So, I would like to show you first how a arduino device looks like. So, this is the arduino UNO. So, arduino UNO, arduino has different variants they all have different you know differences in specifications and so on. So, this is the arduino UNO. And this is this device that has to be program, this has to be programmed. As you can see over here this is very small in size and it can be very much integrated with this internet of things you know when you are trying to implement internet of things it can be implemented on top. So, this device actually you know the difference sensors that you have you know learnt in this particular course.

So, these different sensors can be fitted to this device, the different actuators can be fitted to this device, and this sensors the different sensors and the different actuators in after fitting the data that is received from the sensors. These can be sent through the communication unit in this which Anandroop is going to talk about, and how this data can be disseminated and can be sent for further analytics further storage and so on.

(Refer Slide Time: 05:08)



Hello, I am Anandhroop Mukharjee. I am the TF for this course. I will be taking you through the basic features of arduino in this lecture. So, to start off with as you already have heard arduino has become very popular nowadays. So, first of all the main reasons is it is an open source programmable board with a built in microcontroller and the software IDE. And this software IDE will help you change the behavior of the microcontroller according to your needs. So, it accepts analog as well as digital signals which can be given as inputs and it will give outputs which are mainly digital. So, no extra hardware is required to load a program into the controller board. So, for the people who have work with 8051 series microcontrollers, 8085 microprocessors, they must have remembered that you needed an extra programmer to actually program the processor board and they are well lots of interfacing ICs and all those things are not required with the arduino base systems.

(Refer Slide Time: 06:02)

Types of Arduino Board

- Arduino boards based on ATMEGA328 microcontroller
- Arduino boards based on ATMEGA32u4 microcontroller
- Arduino boards based on ATMEGA2560 microcontroller
- Arduino boards based on AT91SAM3X8E microcontroller

IT KHARAGPUR | NPTEL ONLINE CERTIFICATION COURSES | Introduction to Internet of Things

So, to start off with there are few basic variations of the arduino boards they have there are ATMEGA328 base microcontroller, they have ATMEGA32u4 four series microcontrollers, they have ATMEGA2560 series microcontrollers and then there are ATMEGA91SAM3X8E series microcontrollers. So, these are some of the core microcontrollers as you can see if you focus on this IC this arduino board, so we are using a UNO board to give a demonstration. So, this is the IC chip. And basically all the other once are either the voltage converters or interfacing ICs which are required for the input output functions with this ATMEGA series chip.

(Refer Slide Time: 06:57)

Arduino UNO

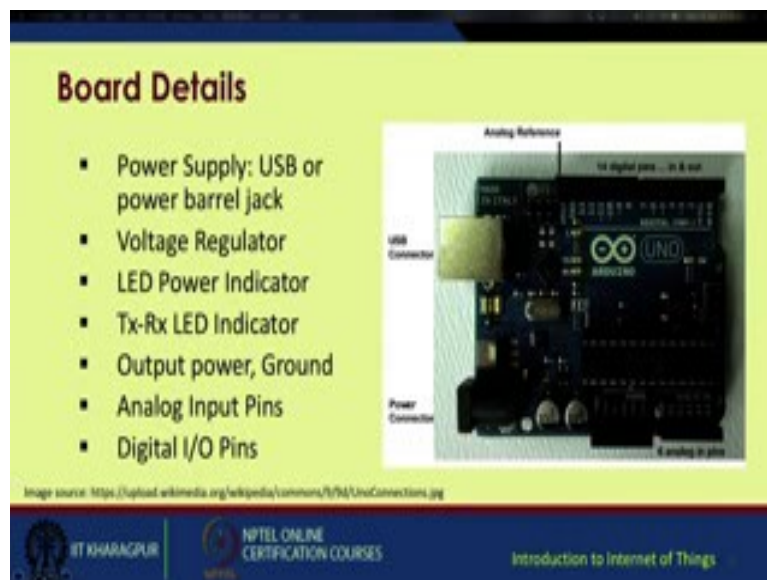
| Feature | Value |
|-------------------|--------------------|
| Operating Voltage | 5V |
| Clock Speed | 16MHz |
| Digital I/O | 14 |
| Analog Input | 6 |
| PWM | 6 |
| UART | 1 |
| Interface | USB via ATmega16U2 |



IT KHARAGPUR | NPTEL ONLINE CERTIFICATION COURSES | Introduction to Internet of Things

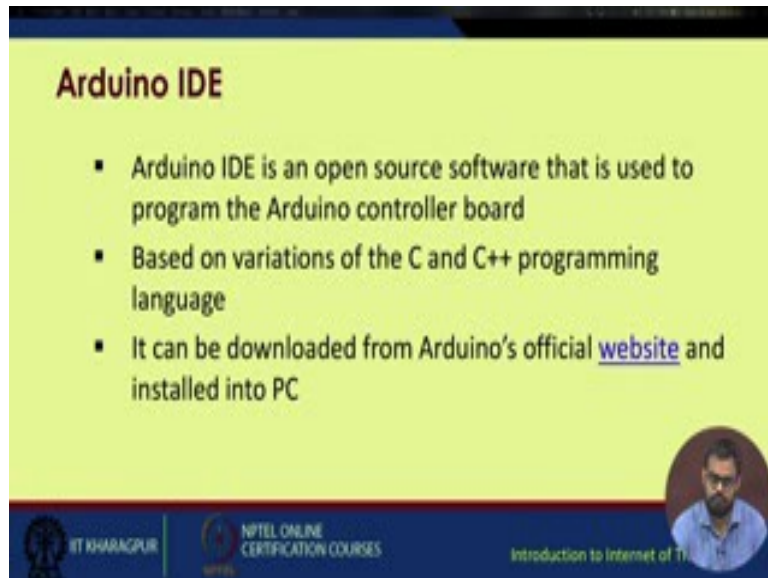
So, basic features of arduino UNO they operate at a voltage of five volt with the clock speed of 16 megahertz, and they have 14 normally arduino UNO has 14 digital input output pins, 6 analog input pins, 6 PWM pins, 1 UART that is universal asynchronous receiver and transmitter. And the interface is mainly via USB of ATMEGA16u2.

(Refer Slide Time: 07:26)



So, the board details are as you can see from this figure you have a USB connector to which you connector ATMEGA board to your PC it. The best thing about this is the interfacing is very easy you can connect your system arduino baseboard to a either a window base pc or a macintosh or Ubuntu or Linux base system. So, this is the USB connector then you have the power connector to power on the device in standalone mode; otherwise if you connected to a pc address power from the pc itself. And this is the analog reference pin you have 14 digital pins which can be used as input and output. As you can see starting from 0 to 13, these are the fourteen input output pins. And over here you have 6 analog pins A 0 to A 5 which can receive analog inputs and these are some of the power connectors you have 5 volt, 3.3 volt and ground connection and so on. So, these are just some of the basic components of the arduino.

(Refer Slide Time: 08:36)



Arduino IDE

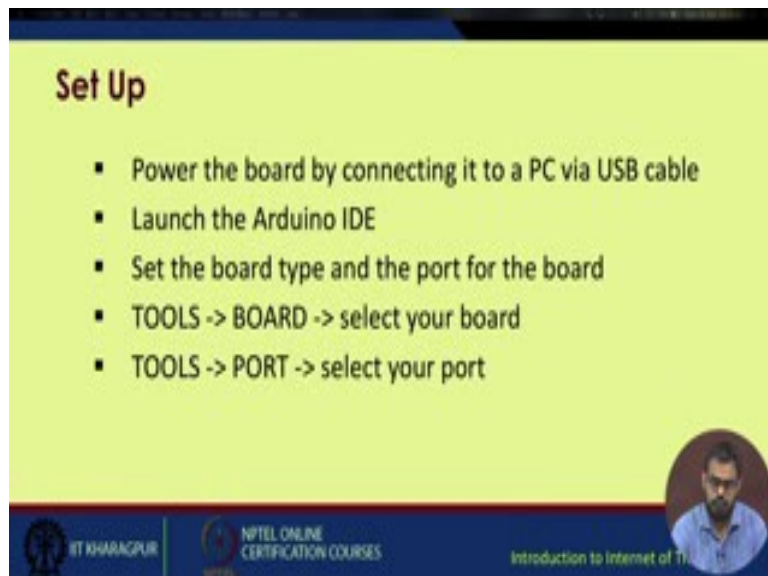
- Arduino IDE is an open source software that is used to program the Arduino controller board
- Based on variations of the C and C++ programming language
- It can be downloaded from Arduino's official [website](#) and installed into PC

IT KHARAGPUR | NPTEL ONLINE CERTIFICATION COURSES | Introduction to Internet of Things

The slide features a yellow background with a dark blue header and footer. A circular video feed of a man with a beard and glasses is positioned in the bottom right corner.

Now, the arduino IDE is basically an open source software, the arduino system itself is an open source system, the hardware's specifications are available you can in fact, if you have fabrication facilities you can actually fabricate your own arduino device. So, this arduino IDE is an open source software that is used to program the arduino board. So, it is based on the variations of C and C plus plus programming languages and it can be freely downloaded from the arduinos official website.

(Refer Slide Time: 09:10)



Set Up

- Power the board by connecting it to a PC via USB cable
- Launch the Arduino IDE
- Set the board type and the port for the board
- TOOLS -> BOARD -> select your board
- TOOLS -> PORT -> select your port

IT KHARAGPUR | NPTEL ONLINE CERTIFICATION COURSES | Introduction to Internet of Things

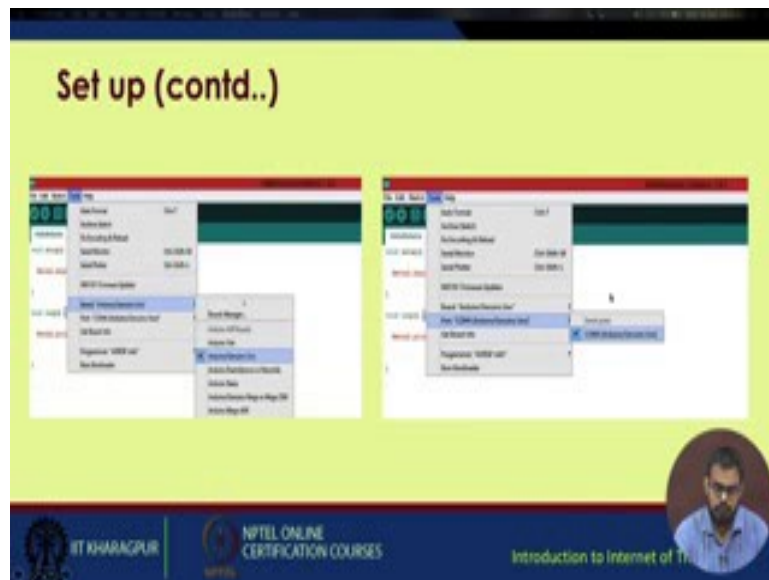
The slide features a yellow background with a dark blue header and footer. A circular video feed of a man with a beard and glasses is positioned in the bottom right corner.

So, the basic set up is the power from the board is derive as you can see the power from the board is derive from the PC using the USB. So, if you are initially for testing purposes you will be connecting it to the PC to upload your program and when you are running at in standalone mode when your program has been uploaded on this board you can run it from this power supply input. You plug in a DC adapter of 5 volts and it is going to work fine Then next will launch the IDE then select the board type.

So, let me show you, so over here I have my arduino IDE. As you can see so you have some basic functionalities; this is the code verification button; this is code upload button. Then you have the file menu you can create a new sketch, a sketch is actually the program you write for an arduino then you can open an existing program open recent programs and so on. You even have examples basic examples provided with the IDE which can work with various arduino base board.

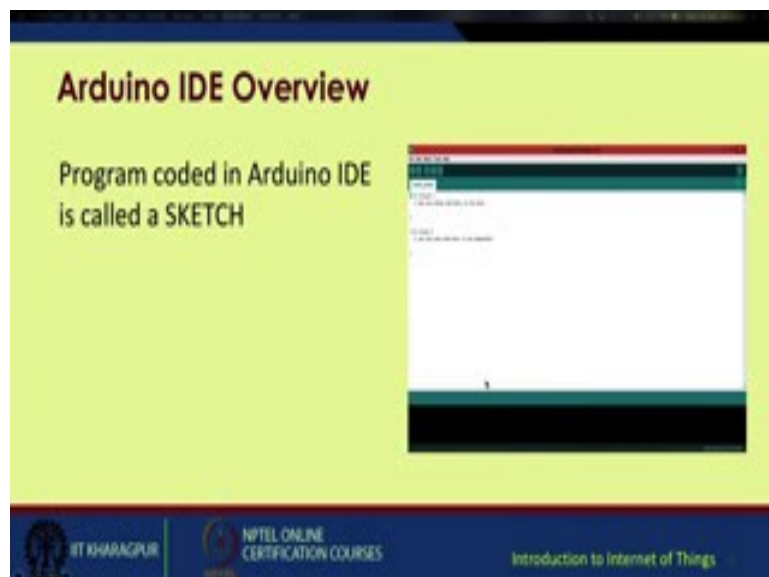
Now moving on to this sketch the most important part is this tool. Whenever you connect arduino board, so I have connected my arduino board to my PC. Now, from this tool these are the available boards you can see since I am using an arduino UNO it has been automatically selected, but in case it is not automatically selected you can choose the appropriate board arduino UNO. Now, for my MAC the port is already this USB modem 1421 you can see arduino UNO has already in selected. So, now, you are all set. One more important thing is over here you can see this button this is the serial monitor this is one of the good features are arduino that while executing serial programs, you do not need to have an external consoler that kind of software, you can just use the inbuilt serial monitor to view the program.

(Refer Slide Time: 11:55)



So, now the once the board and the ports have been appropriately selected as I have told you select the board then the corresponding port in your PC. For windows base systems it will be more or less direct it will show you a com base port it may be com 4, 10, 15 anything you choose it appropriately.

(Refer Slide Time: 12:12)




Then I have already shown you this. So, your arduino sketch as you remember the program which is written in to the arduino for written for the arduino is called a sketch. So, it consists mainly of two parts one is the setup and one is called the loop. Setup is analogous to for

normal mean C, C plus plus based program the main function you use it is analogous us to the setup function in arduino. And as the name suggest the loop function it is used for iteratively looping over instances.

(Refer Slide Time: 12:54)

Arduino IDE Overview (contd..)

- To create a new sketch
 - File -> New
- To open an existing sketch
 - File -> open ->
- There are some basic ready-to-use sketches available in the EXAMPLES section
- File -> Examples -> select any program





IT KHARAGPUR | NPTEL ONLINE CERTIFICATION COURSES | Introduction to Internet of Things

So, its more or less common from file you click on new it will open a new file and you can tryout various examples and sketches.

(Refer Slide Time: 13:06)

Arduino IDE Overview (contd..)

- Verify: Checks the code for compilation errors
- Upload: Uploads the final code to the controller board
- New: Creates a new blank sketch with basic structure
- Open: Opens an existing sketch
- Save: Saves the current sketch

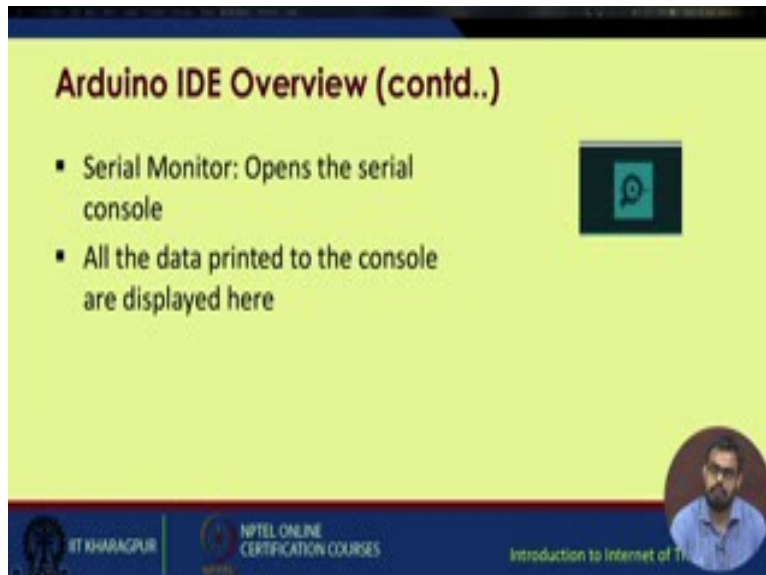


IT KHARAGPUR | NPTEL ONLINE CERTIFICATION COURSES | Introduction to Internet of Things

So, have also covered this one, this is the verify button. So, the main feature is prior to uploading your code if you have syntax errors or any such logical errors it will be caught

during verification, it will say your compilation has failed. Once you pass this verification check you can upload your code.

(Refer Slide Time: 13:37)



Arduino IDE Overview (contd..)

- Serial Monitor: Opens the serial console
- All the data printed to the console are displayed here

The slide features a small icon of the Serial Monitor window in the top right corner. At the bottom, there is a footer with the IIT Kharagpur logo, the text 'NPTEL ONLINE CERTIFICATION COURSES', and a circular portrait of a man. The text 'Introduction to Internet of Things' is partially visible at the bottom right.

Now, this we talked about this is the serial monitor whatever data is transmitted through the serial port is printed on the serial monitor.

(Refer Slide Time: 13:43)



Sketch Structure

- A sketch can be divided into two parts:
 - Setup ()
 - Loop()
- The function setup() is the point where the code starts, just like the main() function in C and C++
- I/O Variables, pin modes are initialized in the Setup() function
- Loop() function, as the name suggests, iterates the specified task in the program

The slide includes a screenshot of an Arduino sketch on the right side. The code shown is:

```
void setup() {  
  Serial.begin(9600);  
}  
  
void loop() {  
  Serial.println("Hello Arduino!");  
}
```

At the bottom, there is a footer with the IIT Kharagpur logo, the text 'NPTEL ONLINE CERTIFICATION COURSES', and a circular portrait of a man. The text 'Introduction to Internet of Things' is partially visible at the bottom right.

So, a sketch structure as I have told you it consists of two parts a setup part and the loop part. The function setup is the point where the arduino compiler actually starts the code. So, it is

just like analogous to the main function in C and C plus plus and various input output variables pin modes whether you need to as you remember you have 14 digital input output pins. So, you have to explicitly tell your system whether you want to use the pin in a read mode or the input mode or the output mode then the loop function is used for iteration.

So, in this example code you can see you just use a serial port the inbuilt serial port. So, we just write serial dot begin 9600 and 9600 is the baud rate. So, you can have various baud rates will come to that will cover that in the consecutive lectures you can have various baud rates for different systems, but 9600 is more or less the commonly use baud rate for most of the systems and within void loop you want to iteratively loop this hello arduino. So, this serial.println that is if you write serial.print it just prints the hello arduino string otherwise if you write println ln is actually new line. So, it will print hello arduino in the new line.

So, prior to this we will look at the sample code. So, as you can see in this hello arduino code within the word setup we have written serial dot begin 9600 and within void loop we have just written serial.println hello arduino. Now, prior to doing anything we verify the code as you can see it is compiling the sketch. If your code compilation is correct, so its correct just shows how much memory it is using in those. If you are in error, suppose I delete this semicolon. Now, again I verify the sketch, it will give an error. So, this is the good practice prior to uploading blindly, you just verify your code. Now, the code verification is successful, the ports have already been chosen, I upload my code. So, it is now compiling the sketch and it is uploading at to the arduino board.

Now, the code has been uploaded to the arduino board since the function of this program is to print hello arduino on the serial port iteratively will open the serial monitor. As you can see, it is printing hello arduino right. So, it is actually quite fast. We can actually modify it will put up function call delay let us say delay of 1 second. So, this thousand days actually the delay in milliseconds will put of a delay of 1 second. The code has been verified we upload it again. Now, again we open the serial monitor. Now, you can see now the delay has been in case. So, it prints after 1 second. So, I hope this was easy.

(Refer Slide Time: 18:17)

Supported Datatype

- Arduino supports the following data types-

| | |
|---------------|-------------------|
| Void | Long |
| Int | Char |
| Boolean | Unsigned char |
| Byte | Unsigned int |
| Word | Unsigned long |
| Float | Double |
| Array | String-char array |
| String-object | Short |



NPTEL ONLINE
CERTIFICATION COURSES

Introduction to Internet of Things



Now, let us move on to the next menu, ok. So, like other programs arduino also supports various data types you have void, int, Boolean, byte, word, float, array, string-object, long, char, unsigned, char, it is somewhat similar to your normal C programs. So, arduino has lots and lots of libraries since it is an open source platform collaboratively people also people and companies and organizations they upload their own arduino libraries



(Refer Slide Time: 18:37)

Arduino Function Libraries

- Input/Output Functions:
 - The arduino pins can be configured to act as input or output pins using the `pinMode()` function


```
Void setup ()  
{  
  pinMode (pin , mode);  
}
```

Pin- pin number on the Arduino board
Mode- INPUT/OUTPUT



NPTEL ONLINE
CERTIFICATION COURSES

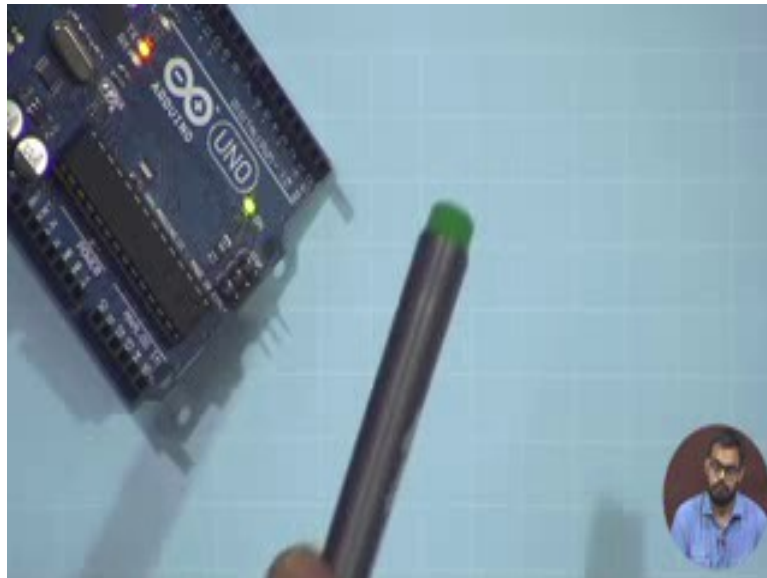
Introduction to Internet of Things



So, for most of the functions will obviously, get easy access to various libraries. So, as you already know the pins can be configured to act as input or output depending on your

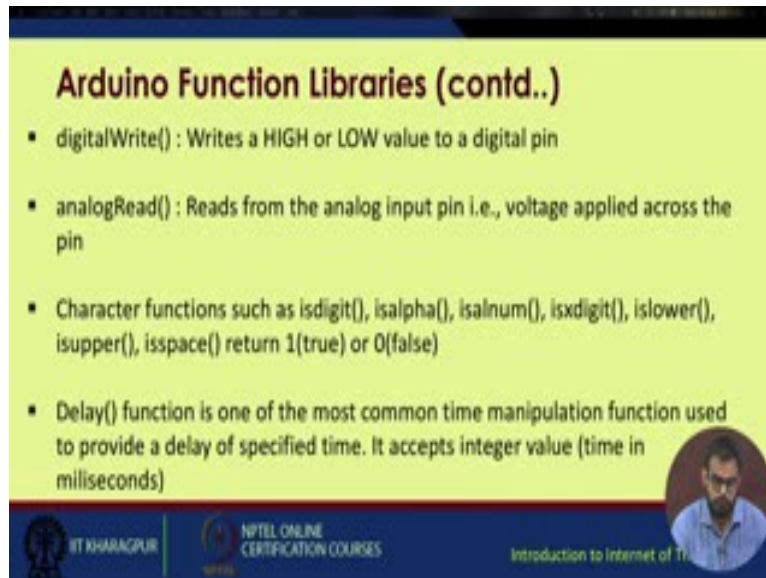
requirement. So, to do this, this function pin mode is used. So, you can see this syntax is `pinMode(pin , mode)`.

(Refer Slide Time: 19:23)



So, this pin is the number, the pin number actual pin number on the arduino board as you can if you focus on this board, you will see various pin numbers are written over here 1, 2, 3 since these are the digital pins it is also written digital. So, in the pin mode against the pin you just write the number of the pin. So, it is that simple and the mode you just write input or output if you want the pins to work in input mode like you are connecting various sensors to it which you will acquire sensor a inputs you put the pin in input mode. And if you want to actuate something maybe a light or LED or a motor, you put the pin in output mode.

(Refer Slide Time: 20:04)



Arduino Function Libraries (contd..)

- `digitalWrite()` : Writes a HIGH or LOW value to a digital pin
- `analogRead()` : Reads from the analog input pin i.e., voltage applied across the pin
- Character functions such as `isdigit()`, `isalpha()`, `isalnum()`, `isxdigit()`, `islower()`, `isupper()`, `isspace()` return 1(true) or 0(false)
- `Delay()` function is one of the most common time manipulation function used to provide a delay of specified time. It accepts integer value (time in milliseconds)


IIT KHARAGPUR | NPTEL ONLINE CERTIFICATION COURSES | Introduction to Internet of Things

So, various arduino function libraries you have digital write, so which writes high or low values to a digital pin. Whenever since you have to remember this thing mainly works on a binary logic you either have a high voltage or a low voltage corresponding to 1 or 0. So, your digital write it either writes a high or low value to the pin then analog read function, it reads an analog function analog input from the analog pins, you have six analog pins. Then character function you have various character functions to check whether it is a character or a digit. So, you can see the various functions `isdigit()`, `isalpha()`, `isalphanumeric()`, `isxdigit()`, `islower()` and various of the functions you can use to check whether the input is the character or a number or a alpha numeric combination and so on. So, this next one you have already covered this delay function. So, you will find it is one of the most versatile and commonly use functions in arduino. So, input is taken in milliseconds as you recall we put 1000 for the delay. So, 1000 milliseconds converts to 1 second.

(Refer Slide Time: 21:19)

Example- Blinking LED

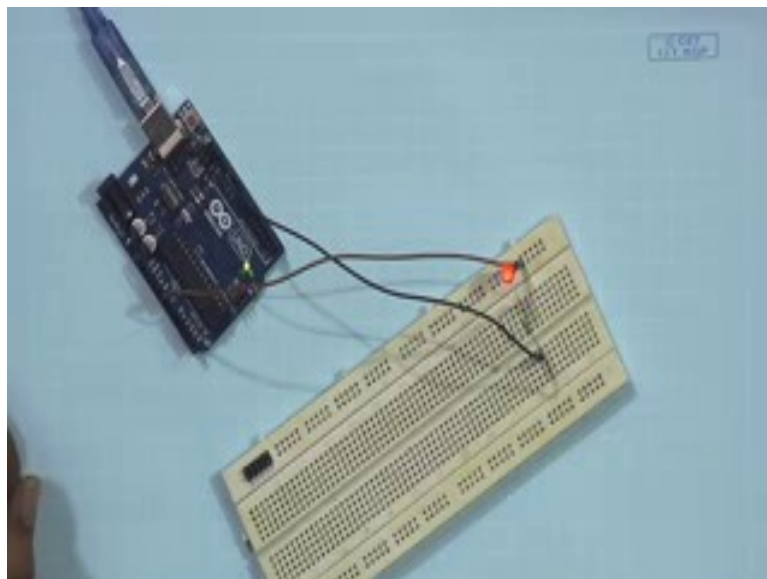
- Requirement:
 - Arduino controller board, USB connector, Bread board, LED, 1.4Kohm resistor, connecting wires, Arduino IDE
- Connect the LED to the Arduino using the Bread board and the connecting wires
- Connect the Arduino board to the PC using the USB connector
- Select the board type and port
- Write the sketch in the editor, verify and upload.



IT KHARAGPUR | NPTEL ONLINE CERTIFICATION COURSES | Introduction to Internet of Things

So, next one is will use a arduino and the normal bread board to switch on and off a LED.

(Refer Slide Time: 21:34)



Now, we have got a LED. We have got a arduino board as you remember these are just basics of electronics. So, for a LED the longer terminal is the positive one, whereas the shorter is the ground. So, we connect the longer terminal to one port, other to the ground. Across this LED we connect a 220 ohm resistance in series prior to taking input from the arduino board. So, this is LED blink.

So, if you concentrate on the setup part within the setup you see the pin mode selected is 12 and it has been selected as output. Basically translates to pin 12 will act as output. So, we will connect the LED to pin 12. Whereas, in the void loop you can see this digital write if you recall digital write either writes high or low values to a particular pin. So, digital write 12 high means you are writing high value to the digital pin to pin 12 and you gave a delay of for 1 second then again you put the pin 12 to low then again a delay. So, this will calls the LED to blink on and off with 1 second delay each.


Now if you come back to the processor board, so we have connected the LED and the resister on the bread board from one side of the bread board we connect a jumper cable to pin 12 one the arduino and for the other since this side was the negative side or the ground we connected to a ground. So, we simply connect this to ground pin. So, again refreshing we have just created the simple circuit. We verify our board is arduino UNO, the port is already selected, we verify our code. Now, the code has been verified we upload our code. Just as soon as the code is uploaded, you will see the LED starts blinking with a delay of 1 second right.

Now for easier checks, you can always use pin 13, pin 13 is by default for arduino UNO at least is connected to the inbuilt LED. So, I mainly check validity of a code based on pin 13, you would not require external interfaces then. So, if you focus on the arduino board this small LED over here, this is the inbuilt LED or the indicator. Now, uploaded my code, the same function you saw on the external LED is being applied on this code, I will just change the pin number from 12 to 13.

(Refer Slide Time: 25:21)

Example- Blink (contd..)

Connect the positive terminal of the LED to digital pin 12 and the negative terminal to the ground pin (GND) of Arduino Board




IT KHARAGPUR | NPTEL ONLINE CERTIFICATION COURSES | Introduction to Internet of Things

(Refer Slide Time: 25:28)

Example- Blink (contd..) image setup

```
void setup() {  
  pinMode(12, OUTPUT); // set the pin mode  
}  
void loop() {  
  digitalWrite(12, HIGH); // Turn on the LED  
  delay(1000);  
  digitalWrite(12, LOW); //Turn of the LED  
  delay(1000);  
}
```




IT KHARAGPUR | NPTEL ONLINE CERTIFICATION COURSES | Introduction to Internet of Things

So, we have covered these things, we have connected pin 12 and the other side to the ground. We uploaded the load the code and uploaded it.

(Refer Slide Time: 25:31)

Example- Blink (contd..)



```
void setup() {  
  pinMode(12, OUTPUT);  
}  
  
void loop() {  
  digitalWrite(12, HIGH);  
  delay(1000);  
  digitalWrite(12, LOW);  
  delay(1000);  
}
```

Set the pin mode as output which is connected to the led, pin 12 in this case.

Use digitalWrite() function to set the output as HIGH and LOW

Delay() function is used to specify the delay between HIGH-LOW transition of the output

IT KHARAGPUR | NPTEL ONLINE CERTIFICATION COURSES | Introduction to Internet of Things

So, ok this was it. So, in the next part, we will go into more details of arduino programming.

For now thank you.