Unit1:

InternetofthingsandcharacteristicsofIOTandprotocolsusedinLinkLayer

A. InternetofThings(loT)comprisesthingsthathaveuniqueidentitiesandareconnectedtotheInternet.

ThecharacteristicsofIoT

1. Self-Configuring
2. DynamicandSelf-Adapting
3. InteroperableCommunicationProtocols
4. UniqueIdentify

IntegratedintoinformationNetwork

IoTprotocolsusedin LinkLayerare

1.802.3-
Ethernet2.802.1
1-WiFi
3.802.16-WiMax
4.802.15.4-LR_WPAN2G/3G/4G

VariousenablingtechnologiesandapplicationsofIOT

IoTisenabledbyseveraltechnologiesincludingWirelessSensorNetworks,CloudComputing,BigDataAnalytics,EmbeddedSystems,Security Protocolsandarchitectures,CommunicationProtocols,WebServices,Mobileinternet and semanticsearchengines.

1) WirelessSensorNetwork(WSN): Comprisesofdistributeddeviceswithsensorswhichareusedtomonitortheenvironmentalandphysicalconditions.ZigBeeisoneofthemostpopularwirelesstechnologiesusedbyWSNs.

WSNsusedinIoTsystemsaredescribedasfollows:

- WeatherMonitoringSystem:inwhichnodescollecttemp,humidityandotherda ta, whichisaggregatedandanalyzed.
- Indoorairqualitymonitoringsystems: tocollect dataontheindoorairqualityandconcentrationofvariousgases.
- SoilMoistureMonitoringSystems:tomonitorsoilmoistureatvariouslocations.
- SurveillanceSystems:useWSNsforcollectingsurveillancedata(motiondatadetectio n).
- SmartGrids:useWSNsfor monitoringgridsatvariouspoints.

2) **CloudComputing:**Servicesareofferedtousersindifferentforms.
   - Infrastructure-as-a-service(IaaS):provides users the ability to provision computingandstorageresources.Theseresourcesareprovidedtotheusersasavirtualmac hineinstancesandvirtualstorage.
   - Platform-as-a-Service(PaaS):providesuserstheability todevelopanddeployapplication in cloud using the development tools,APIs, software libraries andservicesprovidedbythecloudserviceprovider.
   - Software-as-a-Service(SaaS): provides the user a complete software application ortheuserinterfacetotheapplicationitself.

3) **BigDataAnalytics:**Someexamplesofbig datageneratedbyIoTare
   - SensordatageneratedbyIoTsystems.
   - Machinesensor datacollected fromsensors established in industrialand energysystems.
   - HealthandfitnessdatageneratedIoTdevices.
   - DatageneratedbyIoTsystemsfor locationandtrackingvehicles.
   - Datageneratedbyretailinventorymonitoringsystems.

4) **CommunicationProtocols:**formtheback-boneofIoTsystemsandenablenetworkconnectivityandcouplingtoapplications.
   - Allowdevicestoexchangedataovernetwork.
   - Definetheexchangeformats,dataencodingaddressingschemesfordeviceandrouting ofpacketsfromsourcetodestination.
   - Itincludessequencecontrol,flowcontrolandretransmissionoflostpackets.

5) **EmbeddedSystems:**isacomputersystemthathascomputerhardwareandsoftwareembeddedto

perform specifictasks.EmbeddedSystemrangefromlowcostminiaturized devicessuchasdigitalwatchestodevicessuchasdigitalcameras, POSterminals, vendingmachines,appliances etc.,

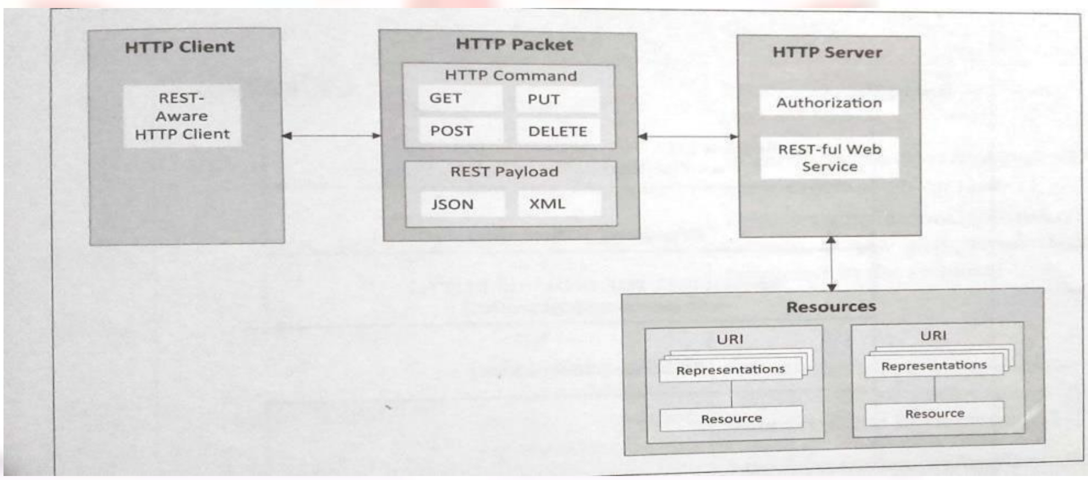
CommunicationAPIsofIOT

IoTCommunicationAPIs:

a) **RESTbasedcommunicationAPIs(Request-ResponseBasedModel)**
b) **WebSocketbasedCommunicationAPIs(ExclusivePairBasedModel)**

a) **REST based communication APIs:** Representational State Transfer(REST) is a set ofarchitecturalprinciples bywhichwecandesignwebservicesandwebAPIsthatfocusonasystem'sresourcesand haveresourcestates areaddressedandtransferred.

The REST architectural constraints: Fig. shows communication between client server withRESTAPIs.



Client-Server:
Theprinciplebehindclient-serverconstraintistheseparation of concerns.Separationallows clientandservertobeindependentlydeveloped andupdated.

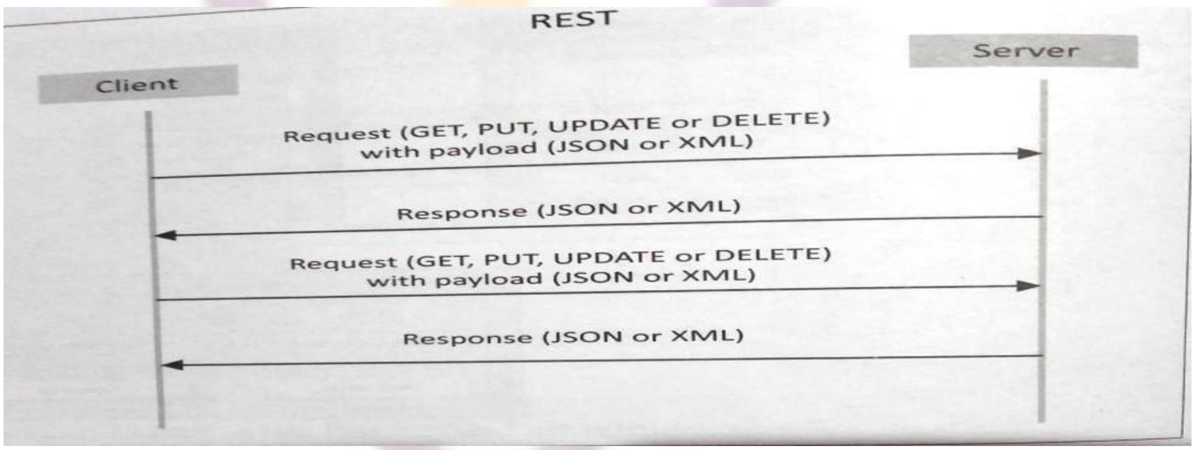
Stateless:
Eachrequestfromclienttoservermustcontainalltheinfo.Necessarytounderstandtherequest,andcannottakeadvantageofanystoredcontextontheserver.

Cache-able:
Cache constraint requires that the data within a response to a request be implicitly orexplicitly labeled as cache-able or non-cacheable. If a response is cache-able, then a clientcacheisgiventherighttoreusethatresponsedataforlater,equivalentrequests.

LayeredSystem:
constraints the behavior of components such that each component cannot see beyond theimmediatelayerwithwhichtheyareinteracting.

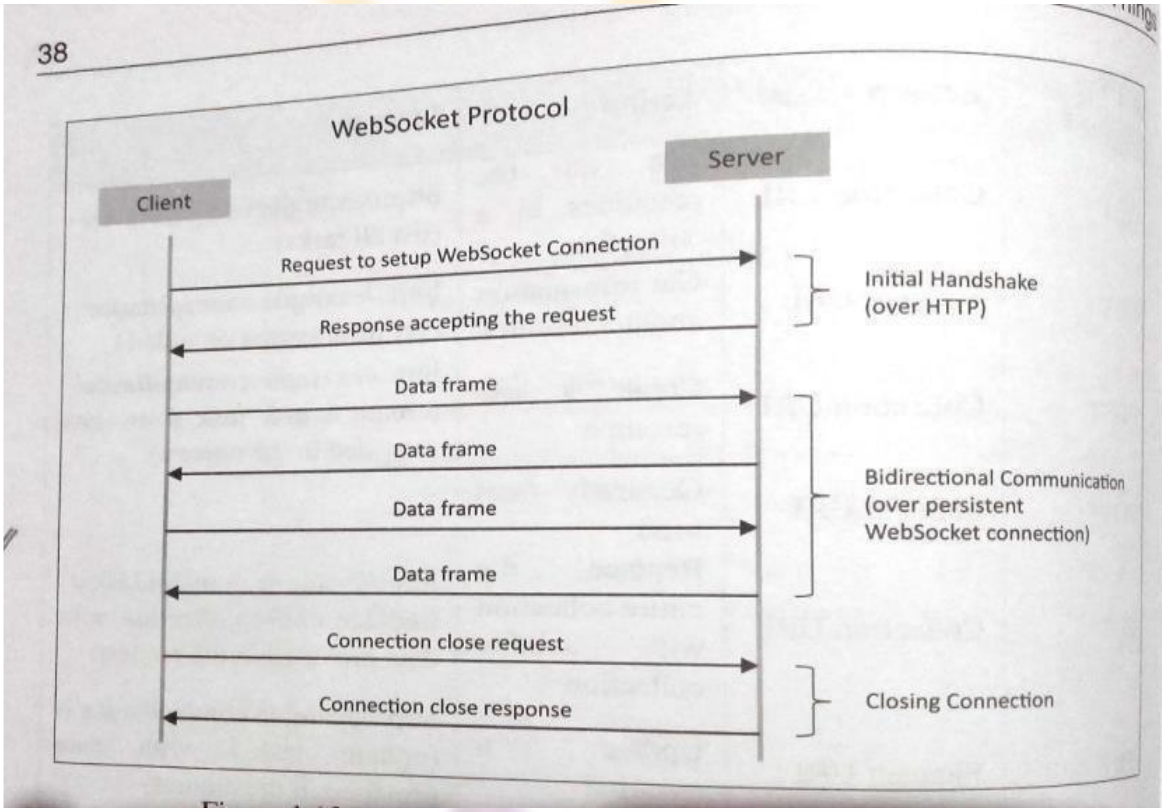
UserInterface:
constraintrequiresthatthemethodofcommunication betweenaclientandaservermustbeuniform.
CodeonDemand:
Serverscanprovideexecutablecodeorscriptsforclientstoexecuteintheircontext.Thisconstraintisthe onlyonethatis optional.

Request-ResponsemodelusedbyREST:



RESTful webservice is a collection of resources which are represented by URIs. RESTfulweb API has a base URI(e.g: http://example.com/api/tasks/). The clients and requests tothese URIs using the methods defined by the HTTP protocol(e.g: GET, PUT, POST orDELETE).ARESTfulwebservicecansupportvariousinternetmediatypes.

b) WebSocket Based Communication APIs: WebSocket APIs allow bi-directional, fullduplex communication between clients and servers. WebSocket APIs follow the exclusivepaircommunicationmodel.



PhysicalDesignOfIoT

1) ThingsinIoT:

The things in IoT refers to IoT devices which have unique identities and perform remotesensing, actuating and monitoring capabilities. IoT devices can exchange dat with otherconnected devices applications. It collects data from other devices and process data eitherlocallyorremotely.

An IoT device may consist of several interfaces for communication to other devices bothwiredandwireless.Theseincludes

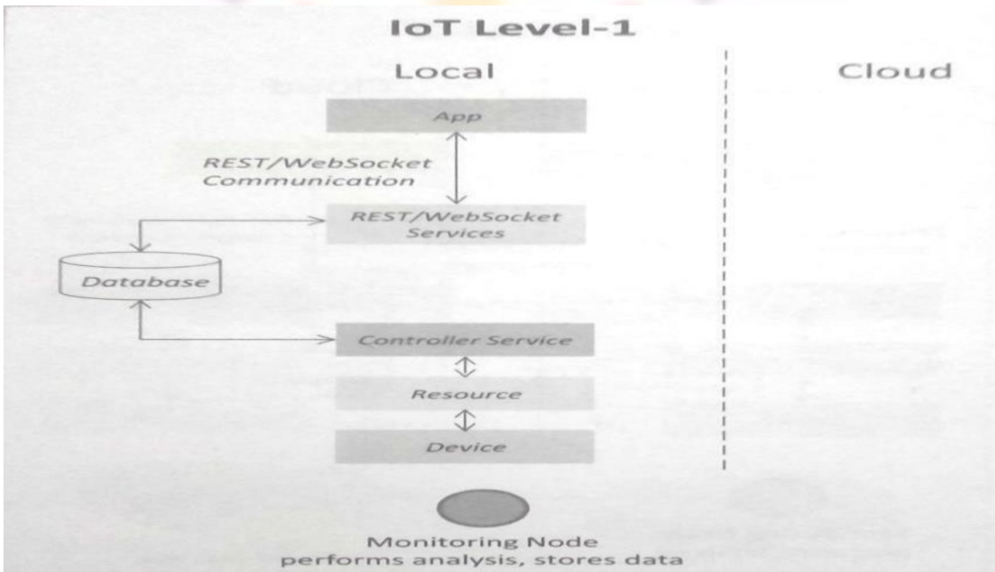
1. I/Ointerfacesforsensors,
2. Interfacesforinternetconnectivity
3. memoryandstorageinterfaces
4. audio/videointerfaces.

1) IoTProtocols:

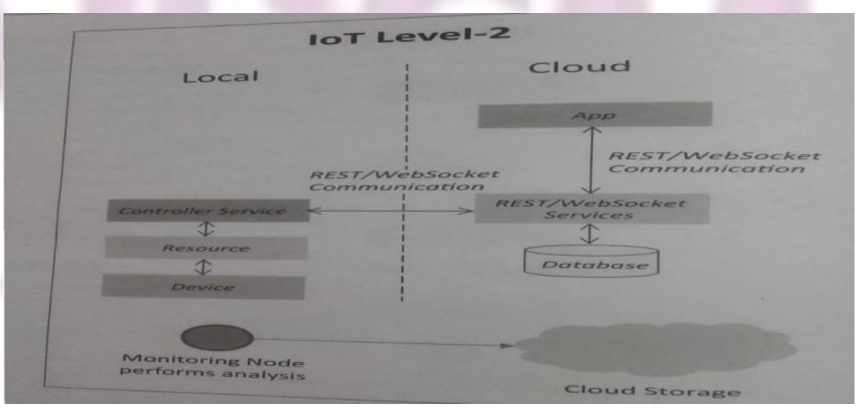
a) LinkLayer: Protocolsdeterminehowdataisphysicallysentoverthenetwork'sphysicallayer or medium. Local network connect to which host is attached. Hosts on the same linkexchange data packets over the link layer using link layer protocols. Link layer determineshow packets are coded and signaled by the h/w device over the medium to which the hostisattached.

5. **IOTlevelswithneatdiagramAns:**

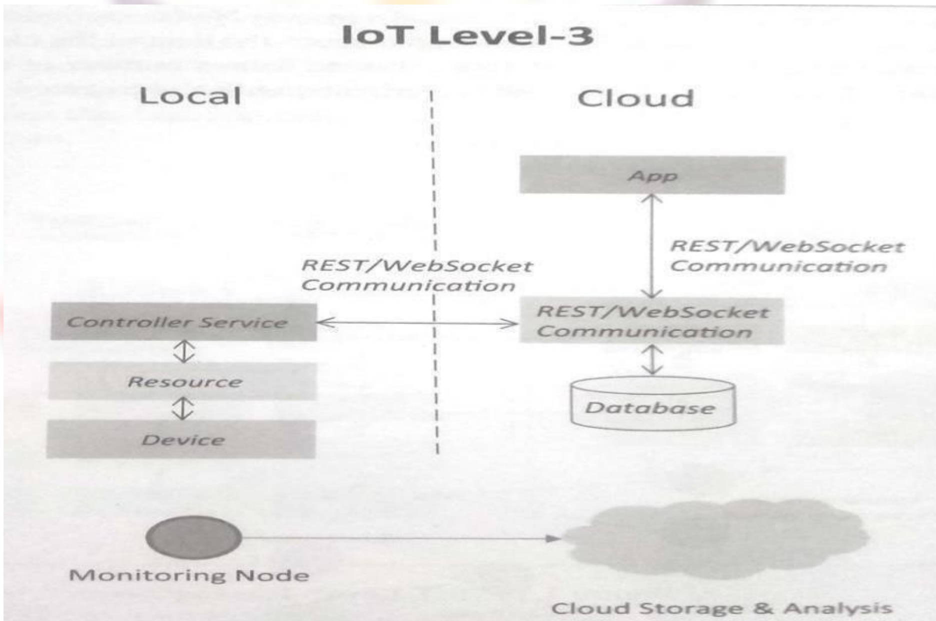
A. **IoT Level 1:** System has a single node that performs sensing and/or actuation, stores data,performs analysis and host the application as shown in fig. Suitable for modeling low costandlowcomplexitysolutionswherethedatainvolvedisnotbigandanalysisrequirementarenotc omputationallyintensive.Ane.g.,ofIoTLevel1isHomeautomation.



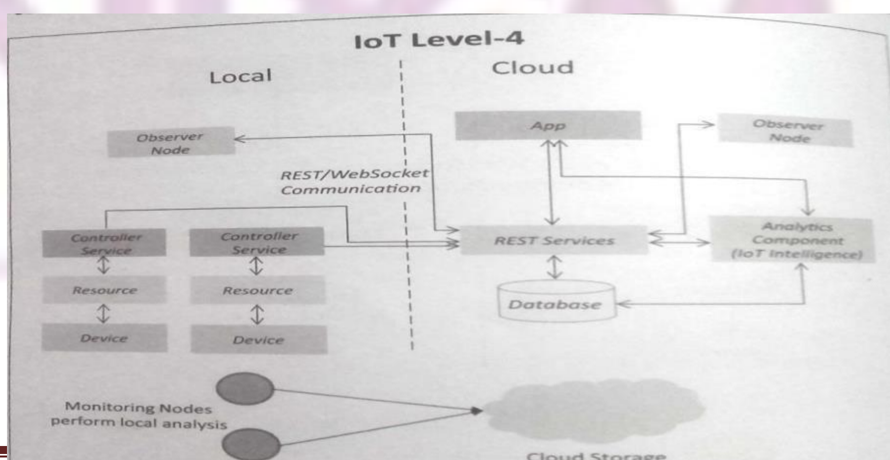
IoT Level2: has a single node that performs sensing and/or actuating and local analysis asshowninfig.Dataisstoredincloudandapplicationisusuallycloudbased.Level2IoTsystemsare suitable for solutions where data are involved is big, however, the primary analysisrequirement is not computationally intensive and can be done locally itself. An e,g.**,** of Level2IoTsystemforSmartIrrigation.



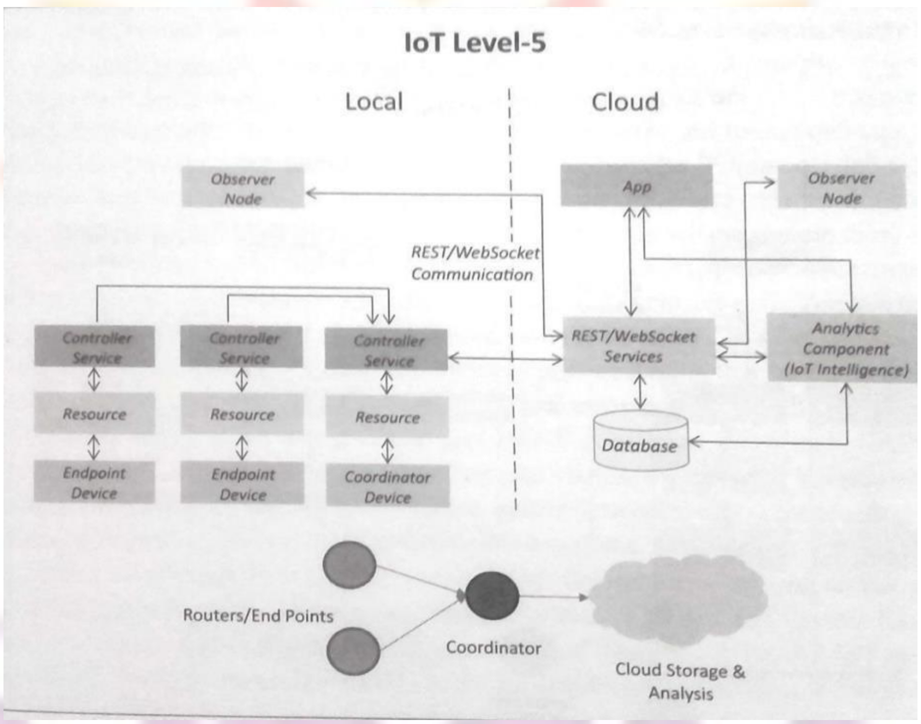
IoT Level3: system has a single node. Data is stored and analyzed in the cloud application iscloud based as shown in fig. Level3 IoT systems are suitable for solutions where the datainvolved is big and analysis requirements are computationally intensive. An example of IoTlevel3systemfortrackingpackagehandling.



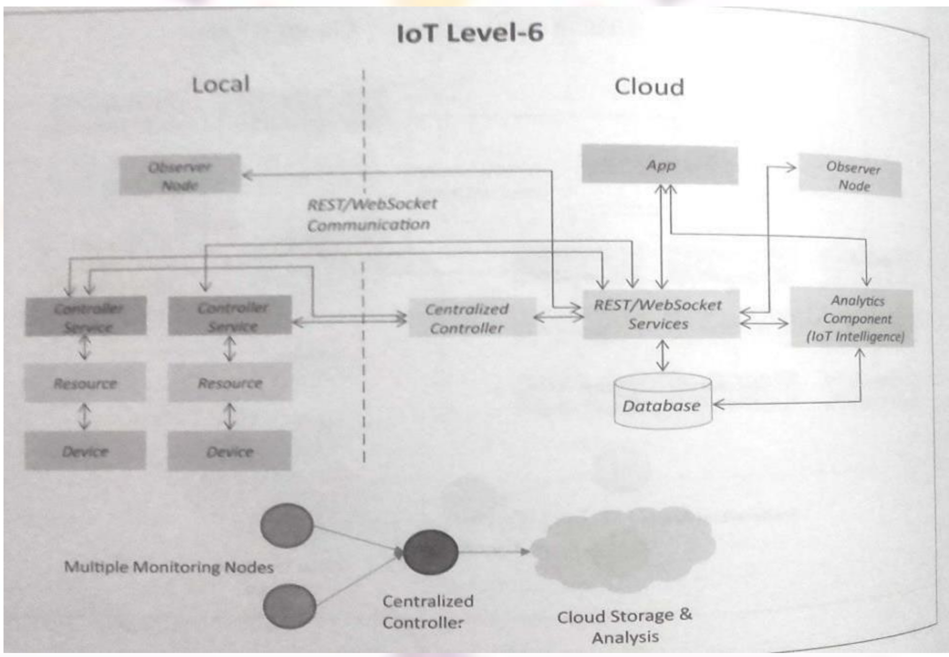
IoTLevel4:
Systemhasmultiplenodesthatperformlocalanalysis.Dataisstoredinthecloudandapplication iscloudbasedasshowninfig.Level4containslocalandcloudbasedobservernodeswhichcansubscri betoandreceiveinformationcollectedinthecloudfromIoTdevices.Anexampleofa Level4IoT systemforNoiseMonitoring.



IoT Level5: System has multiple end nodes and one coordinator node as shown in fig.The endnodes that perform sensing and/or actuation. Coordinator node collects data from the end nodesand sends to the cloud. Data is stored and analyzed in the cloud and application is cloud based.Level5 IoT systems are suitable for solution based on wireless sensor network, in which datainvolved is big and analysis requirements are computationally intensive. An example of Level5systemforForestFireDetection.



IoTLevel6: Systemhasmultipleindependentendnodesthatperformsensingand/oractuationandsensed datatothecloud.Dataisstoredinthecloudandapplicationiscloudbasedasshowninfig.Theanalyticscom ponentanalysesthedataandstorestheresultinthecloudatabase.Theresultsarevisualizedwithcloudbas edapplication.Thecentralizedcontrollerisawareofthestatusofallthe end nodes and sends control commands to nodes. An example of a Level6 IoT system forWeatherMonitoringSystem.

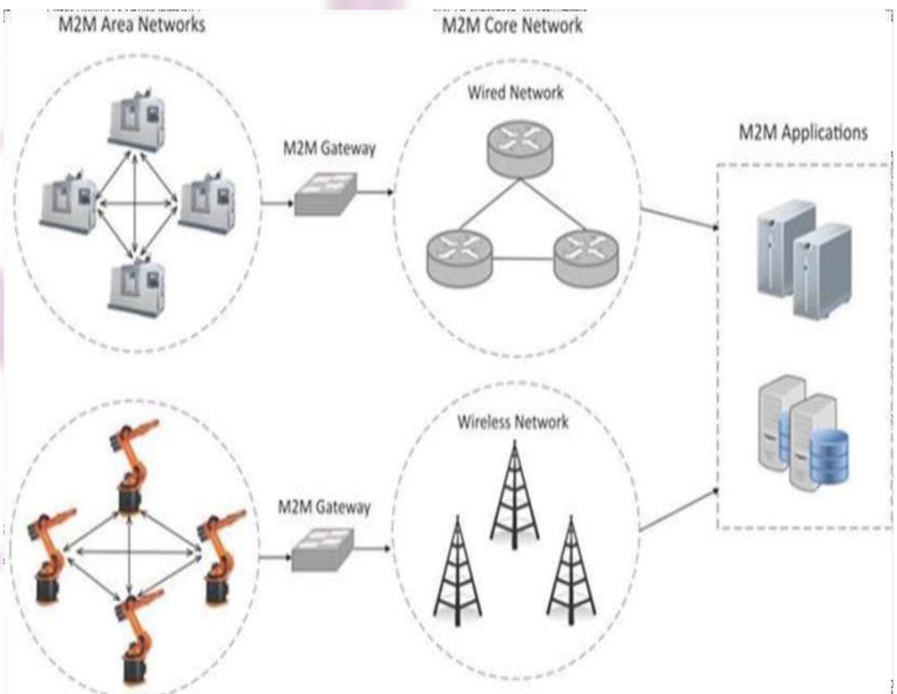


UNIT-2

1. **M2MArchitecture**
Machine-to-Machine(M2M)referstonetworkingofmachines(or devices) forthepurposeofremotemonitoringandcontrolanddataexchange.
   - Termwhichisoftensynonymouswith IoTisMachine-to-Machine(M2M).
   - IoTandM2Mareoftenusedinterchangeably.

Fig.Showstheend-to-endarchitectureofM2Msystemscomprisesof M2Mareanetworks,communicationnetworksandapplicationfomain.

- An M2M area network comprises of machines(or M2M nodes) whiach haveembedded network modules for sensing, actuation and communicating variouscommunictionprotocolscanbeusedforM2MLANsuchasZigBee,Bluetooth,M-bus, Wireless M-Bus etc., These protocols provide connectivity betweenM2MnodeswithinanM2M areanetwork.
- The communication network provides connectivity to remote M2M area networks. Thecommunication network provides connectivity to remote M2M area network. The communicationnetworkcanuseeitherwiredorwirelessnetwork(IPbased).WhiletheM2Marenetworks useeitherproperietoraryornon-IPbaedcommunicationprotocols,thecommunicationnetworkusesIP-basednetwork. Since non-IP based protocols are used within M2M area network, the M2M nodes withinonenetworkcannotcommunicate withnodesinanexternalnetwork.
- TоenablethecommunicationbetweenremoteM2Marenetwork,M2Mgatewaysare used.



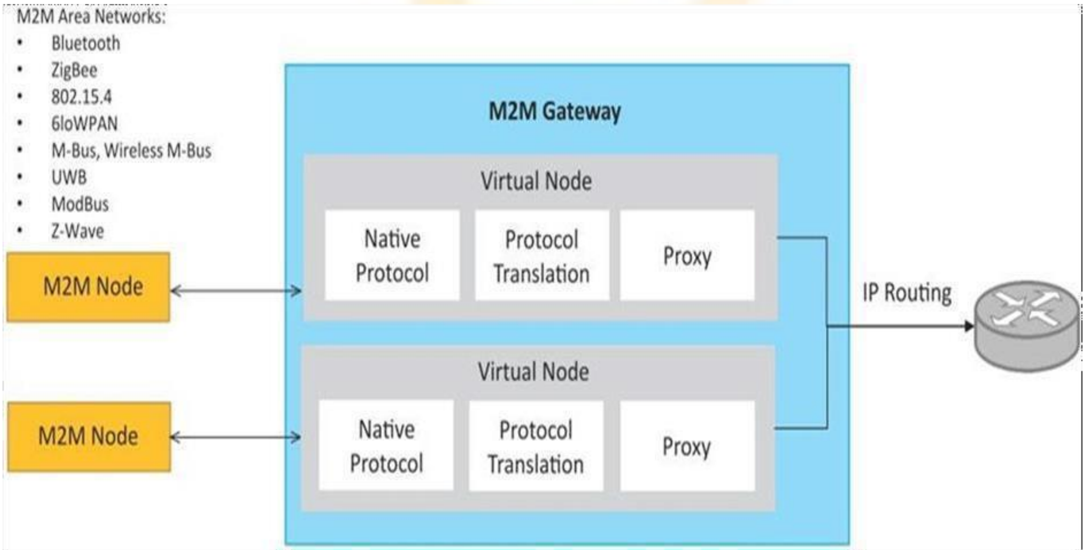


Fig. Shows a block diagram of an M2Mgateway.

The communication between M2M nodes and the M2M gateway is based on the communicationprotocols which are naive to the M2M are network. M2M gateway performs protocol translationsto enable Ip-connectivity for M2M are networks. M2M gateway acts as a proxy performingtranslations from/to native protocols to/from Internet Protocol(IP). With an M2M gateway, eachmode inanM2MareanetworkappearsasavirtualizednodeforexternalM2Mareanetworks.

2. **SDNarchitectureKey**

elementsofSDN:

1) **Centralized Network Controller :** With decoupled control and data planes and centralizednetworkcontroller,thenetworkadministratorscanrapidlyconfigurethe network.

2) **ProgrammableOpenAPIs**

SDNarchitecturesupportsprogrammableopenAPIsforinterfacebetweentheSDNapplicationandcontrollayers(Northboundinterface).

3) **StandardCommunicationInterface(OpenFlow)**

SDNarchitectureusesastandardcommunicationinterfacebetweenthecontrolandinfrastructurelayers(Southboundinterface).OpenFlow,whichisdefinedbytheOpenNetworking Foundation (ONF) is the broadly accepted SDN protocol for the Southboundinterface.

3. NFVarchitecture

NetworkFunctionVirtualization (NFV)

- NetworkFunction Virtualization (NFV)is atechnology thatleveragesvirtualization toconsolidate the heterogeneous network devices onto industry standard high volume servers,switchesandstorage.
- NFV is complementary to SDN as NFV can provide the infrastructure on whichSDN canrun.

Keyelements ofNFV:NFVArchitectureVirtualizedNetworkFunction(VNF):

VNFisasoftwareimplementationofanetworkfunctionwhichiscapableofrunningovertheNFVInfrastructure(NFVI).

1) **NFVInfrastructure(NFVI):**

NFVIincludescompute,networkandstorageresourcesthatarevirtualized.

2) **NFVManagementandOrchestration:**

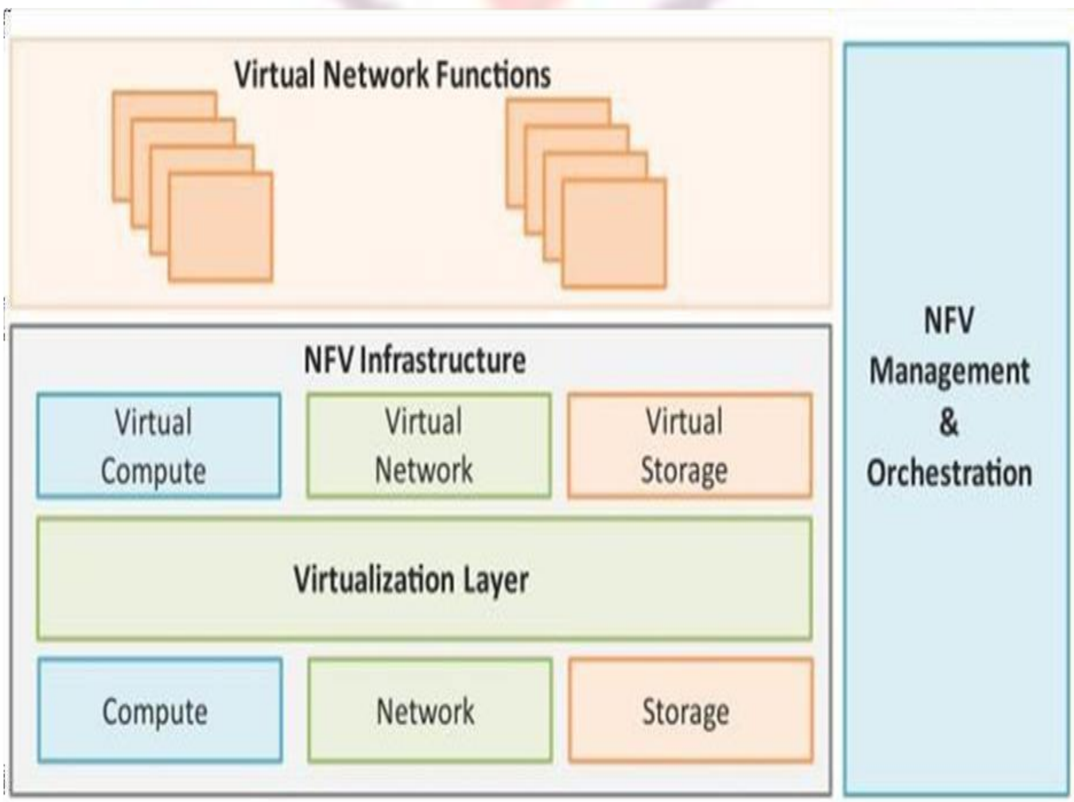
NFV Management and Orchestration focuses on all virtualization-specific management tasksand covers the orchestration and life-cycle management of physical and/or software resourcesthat supporttheinfrastructurevirtualization,andthelife-cyclemanagementofVNFs.

Need for IoT Systems Management

Managingmultipledeviceswithinasinglesystemrequiresadvancedmanagementcapabilities.

1) **Automating Configuration :** IoT system management capabilities can help in automating thesystemconfiguration.
2) **Monitoring Operational & Statistical Data :** Management systems can help in monitoringopeartional and statistical data of a system. This data canbe used for fault diagnosis orprognosis.
3) **Improved Reliability:** A management system that allows validating the system configurationsbeforetheyareput intoeffectcanhelpin improvingthesystemreliability.
4) **System Wide Configurations :** For IoT systems that consists of multiple devices or nodes,ensuringsystemwideconfigurationcanbecriticalforthecorrectfunctioningof thesystem.
5) **MultipleSystemConfigurations:**Forsomesystemsitmaybedesirable tohavemultiplevalidconfigurationswhichareappliedatdifferenttimesorincertainconditions.

Retrieving &Reusing Configurations : Managementsystems which havethe capability ofretrieving configurations from devices can help in reusing the configurations for other devices ofthesametype.



4. IOTsystemManagementwithNETCONFIG–YANG

YANGisadatamodelinglanguageusedtomodelconfigurationandstatedatamanupulatedbytheNETCONFprotocol.
The generic approach of IoT device management weith NETCONF-YANG. Roles ofvariouscomponentsare:

1) ManagementSystem
2) ManagementAPI
3) TransactionManager
4) RollbackManager
5) DataModelManager
6) ConfigurationValidator
7) ConfigurationDatabase
8) ConfigurationAPI
9) DataProviderAPI

1) **ManagementSystem:** Theoperatorusesamanagementsystemtosend NETCONF messagestoconfigure the IoT device and receives state information and notifications from the device asNETCONFmessages.
2) **ManagementAPI:** allowsmanagementapplicationtostartNETCONFsessions.
3) **TransactionManager:** executesalltheNETCONFtransactionsandensuresthatACIDpropertiesholdtrueforthetrasactions.
4) **RollbackManager:** isresponsibleforgeneratingallthetransactionsnecessarytorollbackacurrentconfigurationtoitsoriginalstate.
5) **DataModelManager:** KeepstrackofalltheYANGdatamodelsandthecorrespondingmanagedobjects. Alsokeepstrackoftheapplications whichprovidedataforeachpartofadatam,odel.
6) **ConfigurationValidator:** checksiftheresultingconfigurationafterapplyingatransactionwouldbeavalidconfiguration.
7) **ConfigurationDatabase:** containsbothconfigurationandoperastionaldata.
8) **Configuration API :** Using the configuration API the application on the IoT device can be readconfiguration data from the configuration datastore and write opeartional data to the opearationaldatastore.
9) **Data Provider API:** Applications on the IoT device can register for callbacks for various eventsusingtheDataProviderAPI.ThroughtheDataProviderAPI,theapplicationscanreport statisticsandopeartionaldata.

5. LimitationsofSNMP

SNMPisstatelessinnatureandeachSNMPrequestcontainsalltheinformationtoprocesstherequest. Theapplicationneeds tobeintelligenttomanagethe device.

1. SNMPisaconnectionlessprotocolwhichusesUDPasthetransportprotocol,makingitunreliableasthere wasnosupportforacknowledgementofrequests.
2. MIBsoftenlackwritableobjectswithoutwhichdeviceconfigurationisnotpossibleusingSNMP.
3. ItisdifficulttodifferentiatebetweenconfigurationandstatedatainMIBs.
4. Retrievingthecurrentconfiguration fromadevicecanbedifficult withSNMP. Earlier versionsofSNMPdidnothave strongsecurityfeatures.

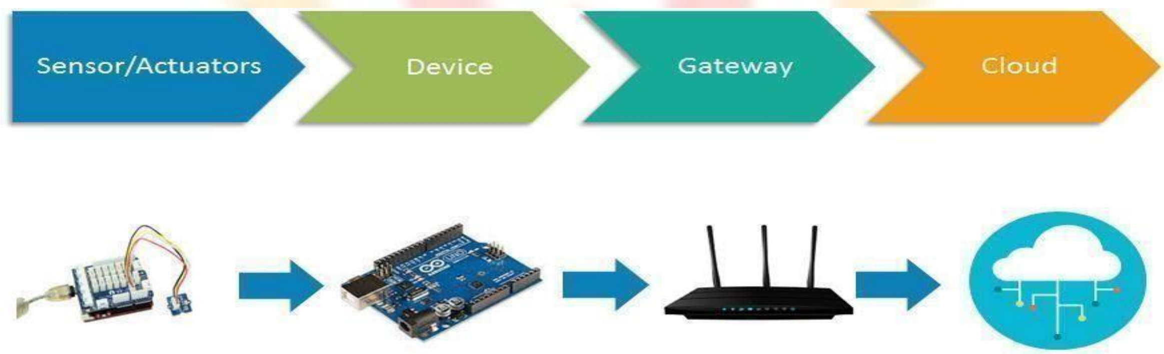
UNIT–3

1. IOTarchitecture

Stateoftheart

IoT architecture varies from solution to solution, based on the type of solution whichwe intend to build. IoT as a technology majorly consists of four main components, overwhichanarchitectureisframed.

1) Sensors
2) Devices
3) Gateway
4) Cloud



Stages of IoT Architecture

Stage1: Sensors/actuators

Sensorscollectdatafromtheenvironmentorobject undermeasurementandturnitintousefuldata.Think of the specialized structures in your cell phone that detect the directional pull of gravityand the phone's relative position to the “thing” we call the earth and convert it into data thatyourphonecanusetoorientthedevice.

Actuators can also intervene to change the physical conditions that generate the data. An actuatormight,forexample,shutoffapowersupply,adjustanairflowvalve,ormovearoboticgripperi nanassemblyprocess.

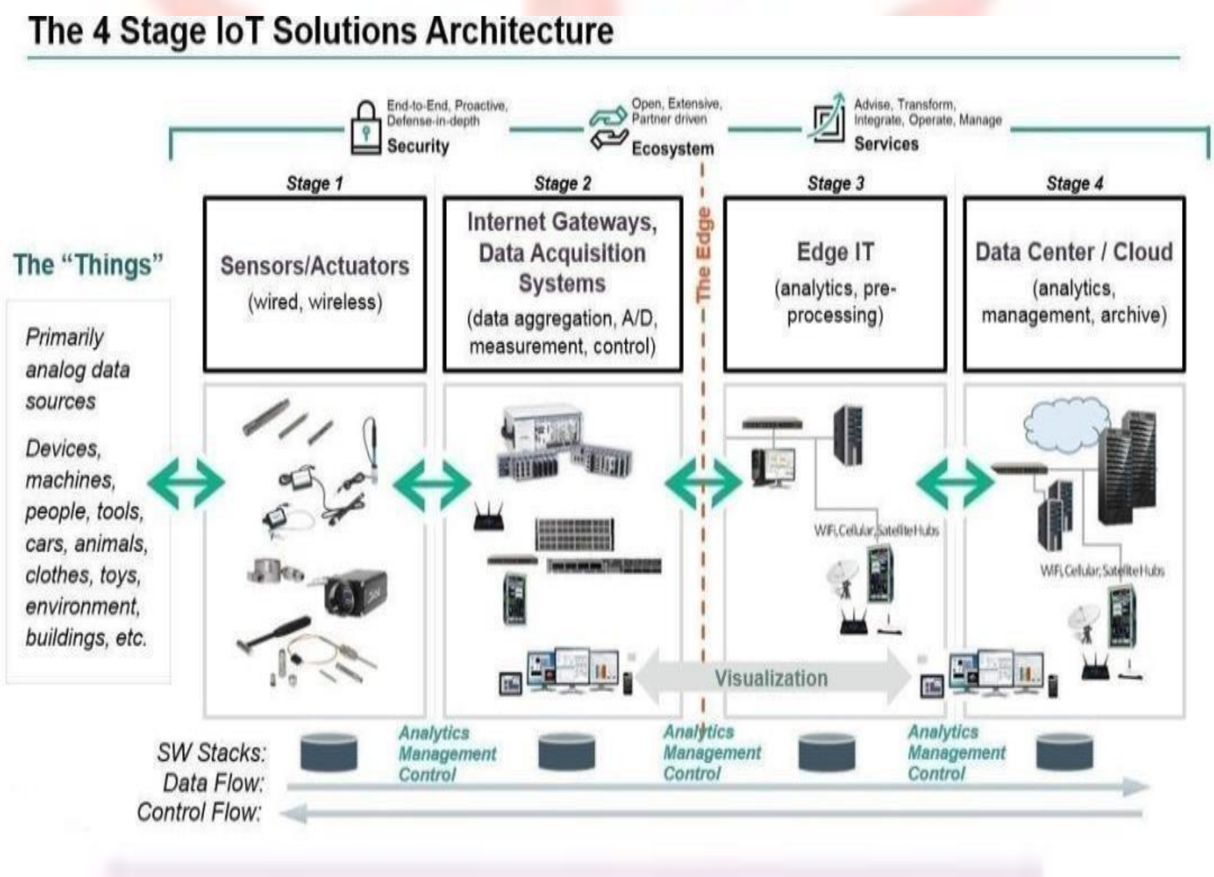
The sensing/actuating stage covers everything from legacy industrial devices to robotic camerasystems, water level detectors, air quality sensors, accelerometers, and heart rate monitors. Andthe scope of the IoT is expanding rapidly, thanks in part to low-power wireless sensor networktechnologiesandPoweroverEthernet,whichenabledevicesonawiredLANtooperatewithou ttheneedforanA/C powersource.

Stage 2:-

TheInternet gateway

Thedatafromthesensorsstartsinanalogform.Thatdataneedstobeaggregatedandconvertedinto digital streams for further processing downstream. Data acquisition systems (DAS) performthesedataaggregationandconversionfunctions.TheDASconnectstothesensornetwork,aggregates outputs, and performs the analog-to-digital conversion. The Internet gateway receivesthe aggregated and digitized data and routes it over Wi-Fi, wired LANs, or the Internet, to Stage 3systems for further processing. Stage 2 systems often sit in close proximity to the sensors andactuators.

For example, a pump might contain a half-dozen sensors and actuators that feed data into a dataaggregationdevicethatalsodigitizesthedata.Thisdevicemightbephysicallyattachedtothepump.An adjacent gateway device or server would then process the data and forward it to the Stage 3 orStage4systems.Intelligentgatewayscanbuildonadditional,basicgatewayfunctionalitybyaddingsuch capabilities as analytics, malware protection, and data management services. These systemsenabletheanalysisofdata streamsinrealtime.



Stage3:-EdgeIT

Once IoT data has been digitized and aggregated, it's ready to cross into the realm of IT. However,the data may require further processing before it enters the data center. This is where edge ITsystems,whichperformmoreanalysis,comeintoplay.EdgeITprocessingsystemsmaybelocatedinremoteofficesorotheredgelocations,but generallythesesitinthefacilityorlocationwherethesensors reside closer to the sensors, such as in a wiring closet. Because IoT data can easily eat upnetwork bandwidth and swamp your data center resources, it's best to have systems at the edgecapable of performing analytics as a way to lessen the burden on core IT infrastructure. You'd alsoface security concerns, storage issues, and delays processing the data. With a staged approach, youcan preprocess the data, generate meaningful results, and pass only those on. For example, ratherthanpassingonrawvibrationdataforthepumps,youcould aggregateandconvertthedata,analyze it,andsendonlyprojectionsasto wheneachdevicewillfailorneedservice.

Stage 4:-

Thedata centerandcloud

Data that needs more in-depth processing, and where feedback doesn't have to be immediate, getsforwarded to physical data center or cloud-based systems, where more powerful IT systems cananalyze, manage, and securely store the data. It takes longer to get results when you wait until datareaches Stage 4, but you can execute a more in-depth analysis, as well as combine your sensor datawith data from other sources for deeperinsights. Stage 4 processingmay take placeon-premises,in the cloud, or in a hybrid cloud system, but the type of processing executed in this stage remainsthesame,regardlessoftheplatform.

2. DatatypesandListDatatypes

Every value in Python has a datatype. Since everything is an object in Python programming, datatypesareactuallyclasses andvariables areinstance(object)ofthese classes.

TherearevariousdatatypesinPython.Someoftheimportanttypesarelistedbelow.

PythonNumbers

Integers,floatingpointnumbersandcomplexnumbersfallsunderPythonnumberscategory.Theyaredefinedasint,floatandcomplexclassinPython.Wecanusethetype() functiontoknowwhichclassavariableoravaluebelongstoand theisinstance()functiontocheckifanobjectbelongstoaparticularclass.

Script.py

1.a=5
2.print(a, "is of type",
type(a))3.a=2.0
4.print(a,"isoftype",type(a))5.a
=1+2j
6. print(a,"iscomplexnumber?",isinstance(1+2j,complex))

Integers can be of any length, it is only limited by the memory available. A floating point numberisaccurateupto15decimalplaces.Integerandfloatingpointsareseparated bydecimalpoints.1isinteger, 1.0 is floating point number. Complex numbers are written in the form, x + yj, where x istherealpartandyis theimaginarypart.Herearesome examples.

>>> a= 1234567890123456789

>>>a12345

67890

123456789

>>>b=0.1234567890123456789

>>>b0.123

45678

901234568

>>>c=1+2j

>>>c(1+2j)

PythonList

List isanorderedsequenceofitems.ItisoneofthemostuseddatatypeinPythonandisveryflexible.All the items in a list do not need to be of the same type. Declaring a list is pretty straight forward.Itemsseparatedbycommas areenclosed withinbrackets[].

CourseName: INTERNET OFTHINGS
Year/Sem:IV-I

>>> a =[1,2.2,
'python']
Wecanusetheslicingoperator[]toextractanitemorarangeofitems fromalist.Indexstarts form0inPython.

Script.py

```
1.a= [5,10,15,20,25,30,35,40]
2.#a[2] =15
3.print("a[2]=",a[2])
4.#a[0:3]=[5,10,15]
5. print("a[0:3]=",a[0:3])
6.#a[5:]=[30,35,40]
7. print("a[5:]=",a[5:])
```

Listsare mutable,meaning;valueofelementsofalistcanbealtered.

```
>>> a=[1,2,3]
>>>a[2]=4
```

PythonTuple

Tupleisanorderedsequencesofitemssameaslist.Theonlydifferenceisthattuplesareimmutable.Tuplesoncecreatedcannotbemodified.Tuplesareusedtowrite-protectdataandareusuallyfasterthan list as it cannot change dynamically. It is defined within parentheses () where items areseparatedbycommas.

```
>>>t=(5,'program',1+3j)
```

PythonStrings

String is sequence of Unicode characters. We can use single quotes or double quotes to representstrings.Multi-line stringscanbedenotedusingtriplequotes,'''or""".

```
>>>s="Thisisastring"
>>>s='''amultiline
```

Likelistandtuple, slicingoperator[]canbeusedwithstring. Stringsareimmutable.
Script.py

```
a={5,2,3,1,4}
#printingsetvariableprint("a=",a)#data
typeofvariableaprint(type(a))
```

We can perform set operations like union, intersection on two sets. Set have unique values. Theyeliminateduplicates.Since,setareunorderedcollection,indexing hasnomeaning.Hencetheslicingoperator [] does not work. It is generally used when we have a huge amount of data. Dictionariesareoptimizedforretrievingdata.Wemustknowthekeytoretrievethevalue.InPython,dictionariesare defined within braces {} with each item beinga pairin theform key:value. Key and valuecanbeofanytype.

```
>>>d={1:'value','key':2}

>>>type(d)

<class'dict'>
```

Weusekeytoretrievetherespective value.But nottheotherwayaround.

Script.py

```
d =
{1:'value','key':2}print(type(d))print("d[1]
=",d[1]);
print("d['key']=
",d['key']);#Generateserrorprint("d[2]=",d[2]);
```

PythondatastructuresPythonif...elseStatement
EveryvalueinPythonhasadatatype.SinceeverythingisanobjectinPythonprogramming,datatypesareactually classes and variables are instance (object) of these classes. Decision making is required when wewanttoexecutea codeonlyifacertainconditionissatisfied.

Theif…elif…elsestatement

isusedinPythonfordecisionmaking.PythonifStatementSyntax

```
If
testexpression
:
statement(s)
```

Here,theprogramevaluates thetestexpressionandwillexecutestatement(s)onlyifthetextexpressionisTrue. If the text expression is False, the statement(s) is not executed. In Python, the body of the if statement isindicatedbytheindentation.
Bodystartswithanindentationandthefirstunindentedlinemarkstheend.Pythoninterpretsnon-zerovaluesasTrue.Noneand0areinterpretedasFalse.

PythonifStatementFlowchart

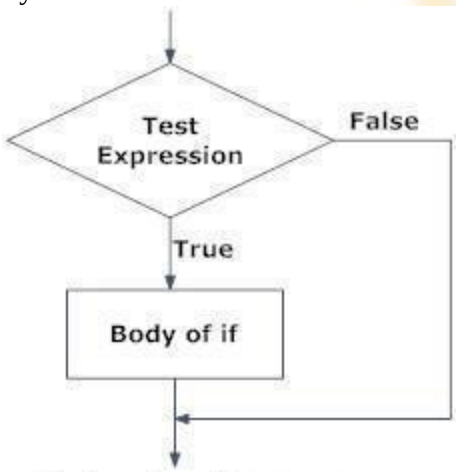


Fig: Operation of if statement

Example: PythonifStatement

#Ifthenumberis positive,weprintanappropriatemessagenum=
3ifnum>0:
print(num,"isa positive

number.") print("This
isalwaysprinted.")num = -1 if
num> 0: print(num, "is a
positivenumber.")print("Thisisalsoalwaysprinted.")

Whenyou runtheprogram,theoutputwillbe:
3 is apositivenumberThisis always
printedThisisalsoalwaysprinted.

In the above example, num> 0 is the test expression. The body of if is executed only if this evaluates to True.Whenvariablenumisequalto3,testexpressionistrueandbodyinsidebodyofifisexecuted.Ifvariablenumis equal to -1, test expression is false and body inside body of if is skipped. The print() statement falls outsideofthe ifblock(unindented).Hence,itisexecutedregardlessofthetestexpression.

Pythonif...elseStatementSyntaxif
test
expression:
Bodyofifels
e:
Bodyofelse

Theif..elsestatementevaluatestestexpressionand will executebody ofifonlywhentest conditionisTrue.IftheconditionisFalse,bodyofelseisexecuted.Indentationisusedtoseparatetheblocks.

Pythonif..elseFlowchart

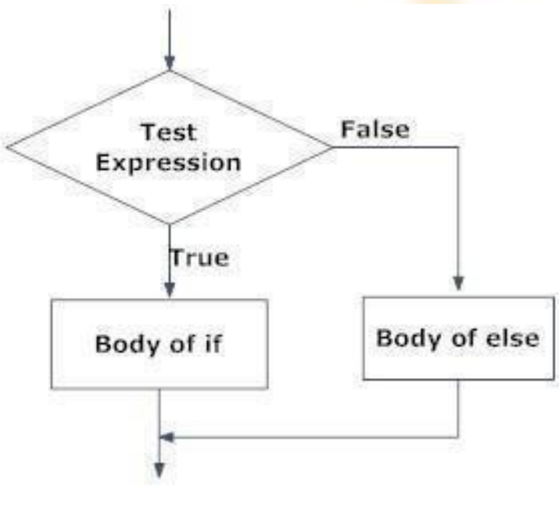


Fig: Operation of if...else statement

Exampleofif...else

#Programchecks ifthenumberispositiveornegative#Anddisplaysanappropriatemessagenum=3
#Trythesetwovariationsaswell.#num=-
5#num=0
ifnum>= 0:print("Positiv
eorZero")else:
print("Negativenumber")
Inthe aboveexample,when numisequalto3,the testexpressionistrueand bodyofifisexecutedand bodyofelseisskipped.

Ifnumisequal to-5,thetestexpression isfalseandbodyofelseisexecutedandbodyofifisskipped.

Ifnumisequalto0,thetestexpression
istrueandbodyofifisexecutedandbodyofelseisskipped.Pythonif...elif...elseStatementSyntaxif testexpression:
Bodyofif
eliftestexpression:

Bodyofelifelse:
Bodyofelse

The elif is short for else if. Itallows ustocheckformultipleexpressions.Ifthe condition for if is False,itchecksttheconditionofthenextelifblockandso on.IfalltheconditionsareFalse,body ofelseisexecuted.Onlyoneblockamongtheseveralif...elif...elseblocksisexecutedaccordingtothecondition.Theifb lockcanhave onlyoneelseblock.Butitcanhavemultipleelifblocks.

PythonFileMethodsFiles
File is a named location on disk to store related information. It is used to permanently store data in a non-volatile memory (e.g. hard disk). Since, random access memory (RAM) is volatile which loses its data whencomputeristurnedoff,weusefilesforfutureuseofthedata.Whenwewanttoreadfromorwritetoafileweneed to open it first. When we are done, it needs to be closed, so that resources that are tied with the file arefreed.Hence,inPython,afileoperationtakesplaceinthe followingorder.

Openafile
Readorwrite(performoperation)Cl
osethefile

Howtoopena file?

Python has a built-in function open() to open a file. This function returns a file object, also called a handle, asitisusedtoreadormodifythefileaccordingly.

```
>>>f=open("test.txt") #openfileincurrentdirectory
>>>f=open("C:/Python33/README.txt")#specifyingfullpath
```

We can specify the mode while opening a file. In mode, we specify whether we want to read 'r', write 'w' orappend 'a' to the file. We also specify if we want to open the file in text mode or binary mode. The default isreadingintextmode.Inthismode,weget stringswhenreadingfromthefile.Ontheotherhand,binarymodereturnsbytesandthisisthemodetobeused whendealingwithnon-textfileslikeimage or exefiles.

PythonFileModes

Mode	**Description**
'r'	Openafileforreading.(default)
'w'	Openafileforwriting.Createsanewfileifitdoesnotexistortruncatesthefileifitexists.
'x'	Openafilefor exclusivecreation.Ifthefilealreadyexists,theoperation fails.
'a'	Openforappendingattheendofthefilewithouttruncatingit. Createsanewfileifitdoesnotexist.
't'	Openintextmode.(default)

'b'	Openinbinarymode.
'+'	Openafileforupdating(readingandwriting)

```
f=open("test.txt")
#equivalent     to'r'     or'rt'
f=open("test.txt",'w')#writeintextmode
f=open("img.bmp",'r+b')#readandwrite inbinarymode
```

Unlikeotherlanguages,thecharacter'a'doesnotimplythenumber97untilitisencodedusingASCII(orotherequivalent encodings). Moreover, the default encoding is platform dependent. In windows, it is 'cp1252' but'utf-8' in Linux. So, we must not also rely on the default encoding or else our code will behave differently indifferent platforms. Hence, when working with files in text mode, it is highly recommended to specify theencodingtype.

```
f=open("test.txt",mode='r',encoding='utf-8')
```

PythonProgramwithRaspberryPIwithfocusofinterfacingexternalgadgets,controllingoutput,readinginputfrompins.

LightanLEDthroughPythonprogram.

Solution:

OneofthebiggestsellingpointsoftheRaspberryPiisitsGPIO,orGeneralPurposeInput/Output ports.Theyare the little pins sticking out of the circuit board and allow you to plug various devices into your RaspberryPi.Withalittleprogramming,youcanthencontrol themordetectwhattheyaredoing.

Inthis tutorialIamgoingtoshowyouhowtolightanLED.InadditiontoyourRaspberryPirunningRaspbian,whatyouwillneedis:

ABreadboard

AnLED

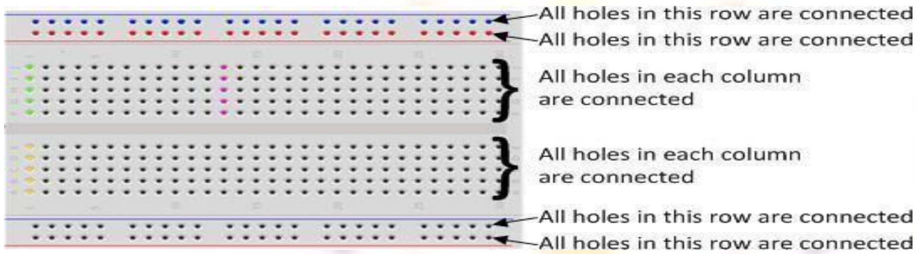
A 330 ohm

resistorTheBreadb

oard:

he breadboard is a way of connecting electronic components to each other without having to solderthem together. They are often used to test a circuit design before creating a Printed Circuit Board(PCB).

Theholesonthebreadboardareconnected inapattern.

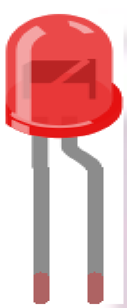


WiththebreadboardintheCamJamEduKit,thetoprow ofholesareallconnectedtogether–markedwithreddots. And so are the second row of holes – marked with blue dots. The same goes for the two rows of holes atthebottomofthebreadboard.
Inthemiddle,thecolumnsofwiresareconnectedtogetherwithabreakinthemiddle.So,forexample,allthegreenholes markedareconnectedtogether,buttheyarenotconnectedtotheyellowholes, northepurpleones.Therefore, any wire you poke into the green holes will be connected to other wires poked into the other greenholes.

TheLED:

Whenyoupickup theLED, youwillnoticethatone legislongerthantheother.Thelongerleg(knownasthe‘anode’), is always connected to the positive supply of the circuit. The shorter leg (known as the ‘cathode’) isconnectedtothenegativesideof thepowersupply,knownas‘ground’.

LEDswillonlyworkifpowerissuppliedthecorrectwayround(i.e.ifthe‘polarity’iscorrect).You willnotbreak the LEDs if you connect them the wrong way round – they will just not light. If you find that they donotlightinyourcircuit,itmaybebecausetheyhavebeenconnectedthewrongwayround.



LEDstandsforLightEmittingDiode,andglowswhenelectricityispassedthroughit.

TheResistor:

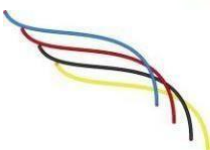
You must ALWAYS use resistors to connect LEDs up to the GPIO pins of the Raspberry Pi. TheRaspberry Pi can only supply a small current (about 60mA). The LEDs will want to draw more,and if allowed to they will burn out the Raspberry Pi. Therefore putting the resistors in the circuitwillensurethatonlythissmallcurrentwillflowandtheRaspberryPiwillnotbedamaged.

Resistorsareawayoflimitingtheamountofelectricitygoingthroughacircuit;specifically, theylimittheamountof ‘current’ thatisallowedtoflow.The measureofresistanceiscalledthe Ohm(Ω),andthelargertheresistance,themoreitlimitsthecurrent.Thevalueofaresistorismarkedwithcolouredbandsalongthelengthoftheresistorbody.
Youwillbeusinga330Ωresistor.Youcanidentifythe330Ωresistorsbythecolourbandsalongthebody.Thecolourcodingwilldependonhowmanybandsareontheresistorssupplied:

Iftherearefourcolourbands,theywillbeOrange,Orange,Brown,andthenGold.
Ifthereare fivebands,thenthecolourswillbeOrange,Orange,Black,Black,Brown.
Itdoesnotmatterwhichwayroundyouconnecttheresistors.Currentflowsinbothwaysthroughthem.

JumperWires:

Jumper wires are used on breadboards to ‘jump’ from one connection to another. The ones you will be usingin this circuit have different connectors on each end. The end with the ‘pin’ will go into the Breadboard. Theendwiththepieceof plasticwithaholeinitwillgoontotheRaspberryPi’sGPIOpins.



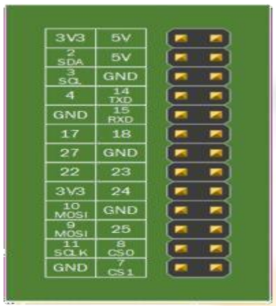
TheRaspberry Pi'sGPIOPins:

GPIOstandsfor **GeneralPurposeInputOutput**.ItisawaytheRaspberryPicancontroland monitor the outside world by being connected to electronic circuits. The Raspberry Pi is able to controlLEDs, turning them on or off, or motors, or many other things. It is also able to detect whether a switch hasbeen pressed, or temperature, or light. In the CamJamEduKit you will learn to control LEDs and a buzzer,and detect when a button has been pressed. The diagram below left shows the pin layout for a Raspberry PiModels A and B (Rev 2 - the original Rev 1 Pi is slightly different), looking at the Raspberry Pi with the pinsin the top right corner. The new 40 pin Raspberry Pi’s shares exactly the same layout of pins for the top 13rowsof GPIOpins.

BuildingtheCircuit:

Thecircuitconsistsofapowersupply(theRaspberryPi),anLEDthatlights whenthepowerisapplied,and

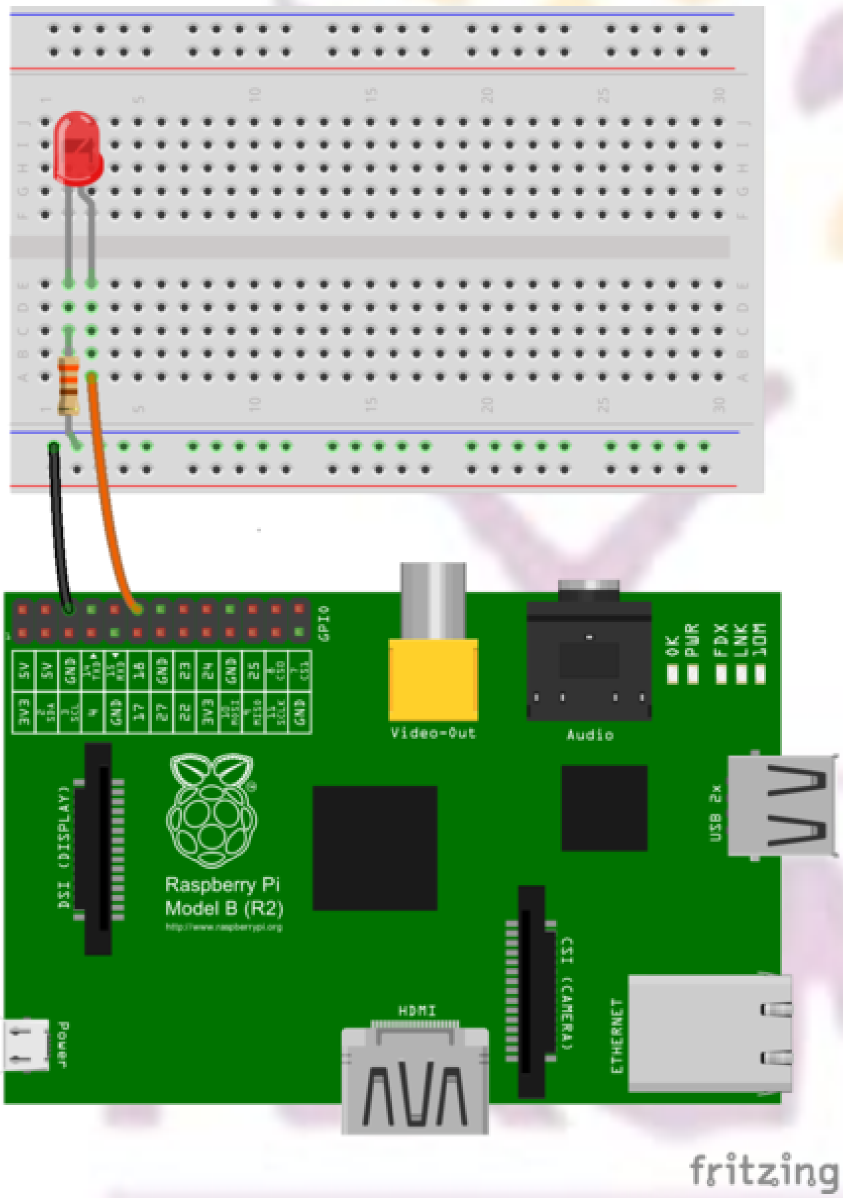
Models A & B



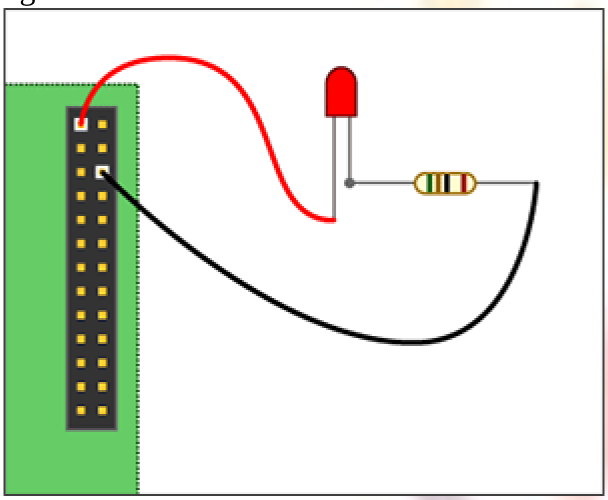
Models A+, B+ & Pi2

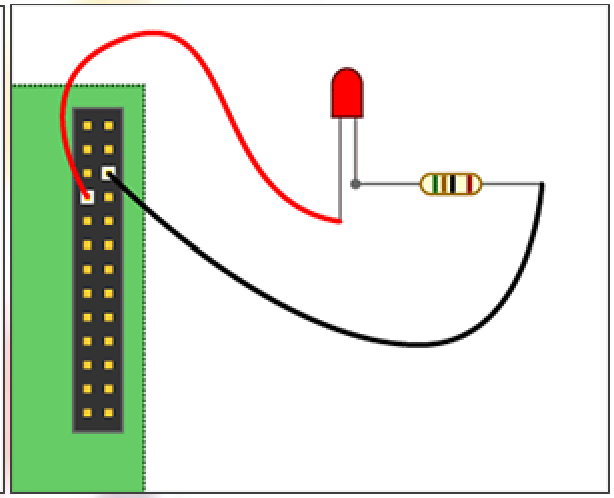
aresistortolimitthecurrentthatcanflowthroughthecircuit.

You will be using one of the ‘ground’ (GND) pins to act like the ‘negative’ or 0 volt ends of a battery. The‘positive’ end of the battery will be provided by a GPIO pin. Here we will be using pin 18. When they are‘takenhigh’,whichmeansitoutputs3.3volts,theLEDwilllight.Nowtakealookatthecircuitdiagrambelow.You shouldturnyourRaspberryPiofforthenextbit,justincaseyouaccidentallyshortsomethingout.



Useoneofthejumperwiresto connectagroundpintothe rail,markedwithblue,onthe breadboard.The female end goes on the Raspberry Pi's pin, and the male end goes into a hole on the breadboard.Then connect the resistor from the same row on the breadboard to a column on the breadboard, asshownabove.
Next,pushtheLEDslegsinto thebreadboard,withthelongleg(withthekink)ontheright.
Lastly,completethecircuitbyconnectingpin18totherighthandlegoftheLED.Thisisshownherewiththeorangewire.





The Code:

YouarenowreadytowritesomecodetoswitchtheLEDon.TurnonyourRaspberryPiandopentheterminalwindow.
Createanewtextfile“LED.py”bytypingthefollowing:
nano LED.py
Type inthefollowingcode:
import RPi.GPIO as GPIOimport
timeGPIO.setmode(GPIO.BCM)GPIO.setwarnings(False)GPIO.setup(18,GPIO.OUT)

print"LEDon"GPIO.output(18,GPIO.HIGH)ti
me.sleep(1)print"LED
off"GPIO.output(18,GPIO.LOW)
Onceyouhavetypedallthecodeandcheckedit,saveandexitthetexteditorwith“Ctrl+x”then“y”then“enter”.

RunningtheCode:

Torunthiscodetype:
sudo python LED.py
You willseetheLEDturnonforasecondandthenturnoff.

Ifyourcodedoes notrunandanerrorisreported,editthecodeagainusingnanoLED.py.

UNIT–4

Introduction to Raspberry
PiRaspberry Pi GPIO Pin
DescriptionBasic building
blocksofIOTDeviceRaspberryPiinter
faces
OtherIOTdevices

Introduction toRaspberryPiRaspberryPi
RaspberryPiisalow-costmini-computerwiththephysicalsizeof acreditcard.RaspberryPi
runs various flavors of Linux and can perform almost all tasks that a normal desktop computer can do.Raspberry Pi also allows interfacing sensors and actuators throughthe general purpose I/O pins. SinceRaspberryPirunsLinuxoperatingsystem,itsupportsPython"outofthebox".RaspberryPiisalow-costmini-computer with the physical size of a credit card. Raspberry Pi runs various flavors of Linux and can performalmost all tasks that a normal desktop computer can do. Raspberry Pi also allows interfacing sensors andactuators through the general purpose I/O pins. Since Raspberry Pi runs Linux operating system, it supportsPython"outofthebox".

RaspberryPi

LinuxonRaspberry Pi

Raspbian: Raspbian Linux is a Debian Wheezy port optimized for Raspberry Pi.Arch:ArchisanArchLinuxportforAMDdevices.
Pidora:PidoraLinuxisaFedoraLinux optimizedforRaspberryPi.RaspBMC:RaspBMCisanXBMCmedia-centerdistributionforRaspberryPi.
OpenELEC:OpenELECisafastanduser-friendlyXBMCmedia-centerdistribution.RISCOS:RISCOSisa veryfastandcompactoperatingsystem.

CourseName:INTERNET OFTHINGS **CourseCode**:**CS724PE**
Year/Sem:IV-I Regulation:R18

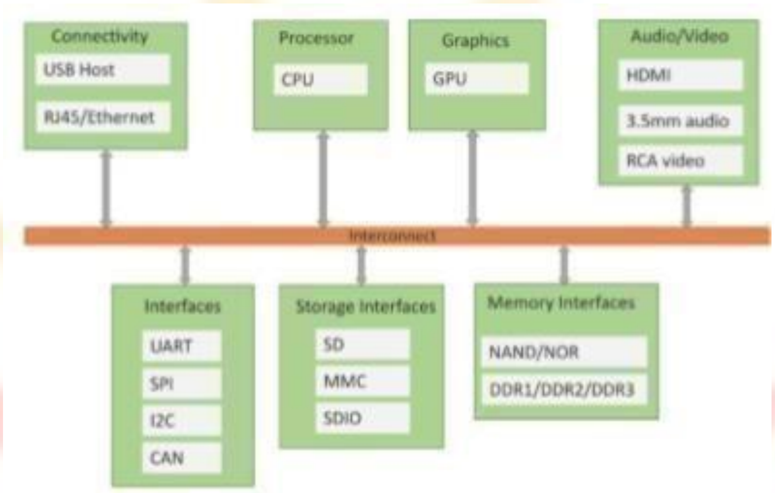
RaspberryPiGPIOPinDescriptionRaspberryPiGPIO

Basicbuildingblocks ofIOTDevice

Basicbuilding blocksofanIoTDevice

Sensing: Sensorscanbeeitheron-board theIoTdeviceorattachedtothedevice.
Actuation: IoT devices can have various types of actuators attached that allow taking actions uponthephysicalentitiesinthevicinityofthedevice.
Communication: Communicationmodulesareresponsibleforsendingcollecteddatatootherdevicesor cloud-based servers/storage and receiving data from other devices and commands from remoteapplications.

Analysis & Processing : Analysis and processing modules are responsible for making sense of the collected data.



BlockdiagramofanIoTDeviceR

aspberryPiinterfaces

Serial:TheserialinterfaceonRaspberryPihasreceive(Rx)andtransmit(Tx)pinsforcommunicationwithserialperipherals.
SPI: Serial PeripheralInterface(SPI)isasynchronousserial dataprotocolusedforcommunicatingwithoneormoreperipheraldevices.
I2C: TheI2Cinterfacepinson RaspberryPiallowyoutoconnecthardwaremodules.I2Cinterfaceallowssynchronousdatatransferwith justtwo pins-SDA(dataline)andSCL(clockline)

OtherIOTdevicespc
DuinoBeagleBone
BlackCubieboard

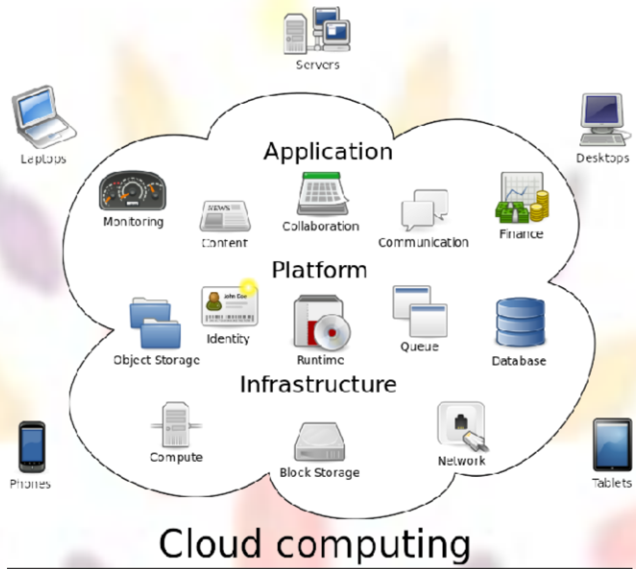
UNIT–5

INTRODUCTIONTOCLOUDCOMPUTINGC
LOUD STORAGEAPI'S
WAMPFOR
IOTPYTHONPACKA
GES
PYTHONWEBAPPLICATIONFRAMEWORK–
DIJANGOCASESTUDYINIOT-SMARTCITIES

INTRODUCTIONTOCLOUDCOMPUTING

The Internet of Things (IoT) involves the internet-connected devices we use to perform the processes andservices that support our way of life. Another component set to help IoT succeed is cloud computing, whichactsasasortoffrontend.Cloudcomputingisanincreasinglypopular servicethatofferesseveraladvantagestoIOT,andisbasedonthe conceptofallowinguserstoperformnormalcomputingtasksusingservicesdeliveredentirely over the internet. A worker may need to finish a major project that must be submitted to a manager, butperhaps they encounter problems with memory or space constraints on their computing device. Memory andspace constraints can be minimized if an application is instead hosted on the internet. The worker can use acloud computing service to finish their work because the data is managed remotely by a server. Anotherexample: you have a problem with your mobile device and you need to reformat it or reinstall the operatingsystem. You can use Google Photos to upload your photos to internet-based storage. After the reformat orreinstall, you can then either move the photos back to you device or you can view the photos on your devicefromthe internetwhenyouwant.Concept

In truth, cloud computing and IoT are tightly coupled. The growth of IoT and the rapid development ofassociated technologies create a widespread connection of "things." This has lead to the production of largeamounts of data, which needs to be stored, processed and accessed. Cloud computing as a paradigm for bigdata storage and analytics. While IoT is exciting on its own, the real innovation will come from combining itwith cloud computing. The combination of cloud computing and IoT will enable new monitoring services andpowerful processing of sensory data streams. For example, sensory data can be uploaded and stored with cloudcomputing,laterto be usedintelligentlyforsmartmonitoring and actuationwithothersmartdevices.Ultimately, the goal is to be able to transform data to insight and drive productive, cost-effective action fromthose insights. The cloud effectively serves as the brain to improved decision-making and optimized internet-based interactions.However, when IoT meets cloud, new challenges arise. There is an urgent need for novelnetwork architectures that seamlessly integrate them. The critical concerns during integration are quality ofservice (QoS) and quality of experience (QoE), as well as data security, privacy and reliability. The virtualinfrastructureforpracticalmobilecomputingandinterfacingincludesintegratingapplications,storagedevices ,monitoring devices, visualization platforms, analytics tools and client delivery. Cloud computing offers apractical utility-based model that will enable businesses and users to access applications on demand anytimeandfromanywhere.



Servicemodels

Service delivery in cloud computing comprises three different service models: software as a service (SaaS),platformasa service(PaaS),andinfrastructureasa service(IaaS).

Software as a service (SaaS) provides applications to the cloud's end user that are mainly accessed via a webportal or service-oriented architecture-based web service technology. These services can be seen as ASP(applicationserviceprovider)ontheapplicationlayer.Usually,aspecificcompanythatusestheservicewouldrun, maintainandgivesupportsothatitcanbereliablyusedoveralongperiodoftime.
Platform as aservice (PaaS) consists of the actual environment fordeveloping and provisioning cloudapplications.Themainusersofthislayeraredevelopersthatwanttodevelopandrunacloudapplication foraparticular purpose. A proprietary language was supported and provided by the platform (a set of importantbasicservices)toeasecommunication, monitoring,billingandotheraspectssuchasstartupaswellastoensurean application's scalability and flexibility. Limitations regarding the programming languages supported, theprogrammingmodel,theabilitytoaccessresources, andthelong-termpersistencearepossibledisadvantages.

Infrastructure as a service (IaaS) provides the necessary hardware and software upon which a customer canbuildacustomizedcomputingenvironment.Computingresources,datastorageresourcesandthecommunications channel are linked togetherwith these essential ITresources to ensure the stability ofapplicationsbeingusedonthecloud.Thosestackmodelscanbereferredtoasthemediumfor IoT,beingusedand conveyed by the users in different methods for the greatest chance of interoperability. This includesconnectingcars,wearables,TVs,smartphones,fitness equipment,robots,ATMs,andvendingmachinesaswellastheverticalapplications,securityandprofessionalservices, andanalyticsplatforms thatcomewiththem.

CLOUDSTORAGEAPI'S

A cloud storage API is an application program interface that connects a locally-based application to a cloud-based storage system, so that a user can send data to it and ccess and work with data stored in it. To theapplication,the cloud storage system isjustanothertargetdevice,like tape ordisk-based storage.Anapplicationprograminterface(API)iscodethatallowstwosoftwareprogramstocommunicatewith eachother.The API defines the correct way for a developer to write a program that requests services from an operatingsystem (OS) or other application. APIs are implemented by function calls composed of verbs and nouns. Therequiredsyntaxisdescribedinthe documentationoftheapplicationbeingcalled.

HowAPIs work

APIsaremadeup oftworelatedelements.Thefirstis aspecificationthatdescribeshow information is exchanged between programs, done in the form of a request for processing and a return of thenecessary data. The second is a software interface written to that specification and published in some way foruse.Thesoftwarethatwantstoaccess thefeaturesandcapabilitiesoftheAPIissaidtocallit,andthesoftwarethatcreatestheAPIissaidtopublishit.

CloudModelsarereliedonCommunication API

CommunicationAPIfacilitatedatatransfer,controlinformation transferfrom applicationtocloud,oneservicetoanother

ItalsoexistintheformofCommunicationProtocolsItsup ports RPC,PUBSUBandWAMP

Eg.PopularAPIisRESTfulAPI(communicationincloudmodel)Djangow ebframeworkisusedtoimplementCommunicationAPI

WAMPFORIOT

WebApplicationMessagingProtocol(WAMP)isasub-protocolofWebsocketwhichprovidespublish-subscribeandremoteprocedurecall(RPC) messagingpatterns.

WAMP

Transport: Transportischannelthat connectstwopeers.

Session: Session isaconversationbetweentwopeersthat runsoveratransport.

Client: Clientsarepeersthatcanhaveoneor moreroles.Inpublish-subscribemodelclientcanhavefollowingroles:
Publisher: Publisher publishes events (including payload) to the topic maintained by the broker.Subscriber:Subscribersubscribestothetopicsandreceivestheeventsincludingthepayload.
InRPCmodelclientcanhavefollowingroles:–

Caller: Caller issues calls to the remote procedures along with call arguments. – Callee: Calleeexecutes the procedures to which the calls are issued by the caller and returns the results back to thecaller. • Router: Routers are peers that perform generic call and event routing. In publish-subscribemodelRouterhas theroleofaBroker:–Broker:Brokeracts asarouter and

routesmessagespublishedto atopicto allsubscriberssubscribedtothetopic.InRPCmodelRouterhastherole ofaBroker:–
Dealer:DealeractsarouterandroutesRPCcallsfromtheCallertotheCalleeandroutes resultsfromCallee toCaller.
ApplicationCode:ApplicationcoderunsontheClients(Publisher,Subscriber,CalleeorCaller).

PYTHONPACKAGESJSON

JavaScript Object Notation (JSON) is an easy to read and write data-interchange format. JSON is used as analternativetoXMLandiseasyformachinestoparseandgenerate.
JSON is built on two structures: a collection of name–value pairs (e.g., a Python dictionary) and ordered listsofvalues(e.g.,aPythonlist).

XML

XML (Extensible Markup Language) is a data format for structured document interchange. The Pythonminidom library provides a minimal implementation of the Document Object Model interface and has an APIsimilartothatinotherlanguages.

HTTPLib&URLLib

HTTPLib2andURLLib2arePythonlibrariesusedinnetwork/internetprogramming.SMTPLib

Simple Mail Transfer Protocol (SMTP) is a protocol which handles sending email and routing email betweenmail servers. The Python SMTPLib module provides an SMTP client session object that can be used to sendemail.

NumPy

NumPy is a package for scientific computing in Python. NumPy provides support for large multi-dimensionalarraysandmatrices.

Scikit-learn

Scikit-learnisanopensourcemachinelearninglibraryforPythonthatprovidesimplementationsofvariousmachinelearningalgorithmsforclassification,clustering,regressionanddimensionreductionproblems.

PYTHONWEBAPPLICATIONFRAMEWORK–DIJANGO

DjangoisanopensourcewebapplicationframeworkfordevelopingwebapplicationsinPython.A web application framework in general is a collection of solutions, packages and best practicesthatallowsdevelopmentof webapplicationsanddynamicwebsites.

Thus, web applications built with Django can work with different data bases without requiringany Code changes.
Withthisflexibilityinwebapplicationdesigncombinedwiththepowerfulcapabilitiesofthe Pythonlanguageand thePythonecosystem,Djangoisbestsuitedforcloud applications.
Djangoconsistsofan object-relationalmapper,aweb templatingsystemandaregular-expression-basedURLdispatcher.

DjangousesaModel–Template–View(MTV)framework.

Model

The model acts as a definition of some stored data and handles the interactions with the database. Ina web application, the data can be stored in a relational database, non-relational database, an XMLfile, etc. A Django model is a Python class that outlines the variables and methods for a particulartypeofdata.

Template

In a typical Djangoweb application, the template is simply an HTML page with afew extraplaceholders. Django's template language can be used to create various forms of text files (XML,email,CSS,Javascript,CSV,etc.).

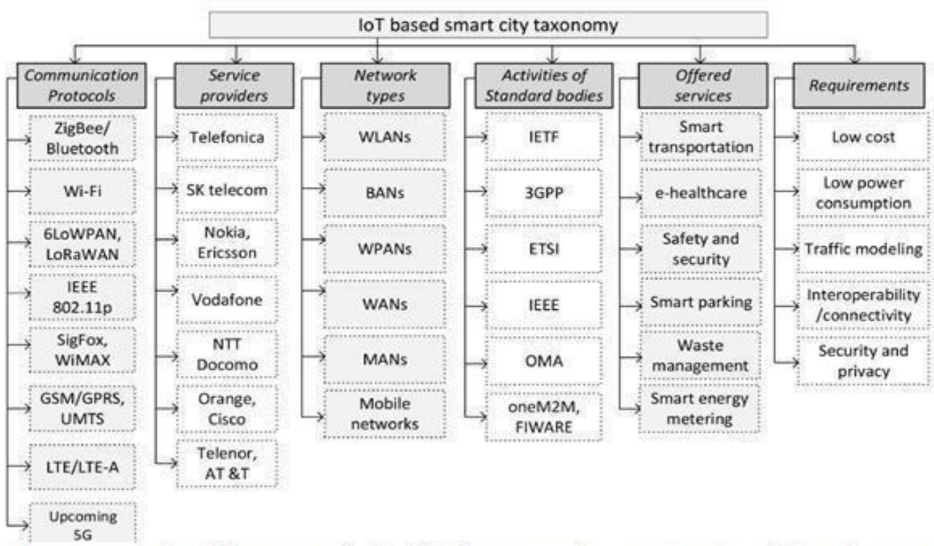
View

Theviewtiesthemodelto thetemplate.Theview iswhereyouwritethecodethatactuallygeneratesthewebpages.Viewdetermineswhatdataistobedisplayed, retrievesthedatafromthedatabaseandpassesthedatatothetemplate.

CaseStudyinIoT:SmartCities

The Internet-of-Things (IoT) is the novel cutting-edge technology which proffers to connect plethora of digitaldevices endowed with several sensing, actuation and computing capabilities with the Internet, thus offersmanifold new services in the context of a smart city. The appealing IoT services and big data analytics areenabling smartcityinitiativesall overtheworld.These servicesare transforming citiesbyimprovinginfrastructure,transportationsystems,reducedtrafficcongestion,wastemanagementandthequalityofhumanlife.Inthispaper,wedeviseataxonomytobestbringforthagenericoverviewofIoTparadigmforsmartcities,integrated information and communication technologies (ICT), network types, possible opportunities andmajor requirements. Moreover, an overview of the up-to-date efforts from standard bodies is presented. Later,we give an overview of existing open source IoT platforms for realizing smart city applications followed byseveralexemplarycasestudies.Inaddition,wesummarizethelatestsynergiesandinitiativesworldwidetakento promote IoT in the context of smart cities. Finally, we highlight several challenges in order to give futureresearchdirections.

An illustration of IoT based smart city



A representation of IoT-based smart city taxonomy

This section presents a taxonomy of IoT based smart cities which categorizes the literature on the basis ofexisting communication protocols, major service providers, network types, standardization efforts, offeredservices,andcrucialrequirements.

CommunicationProtocols

IoTbased smartcity realization significantly relieson numerousshortand wide range communicationprotocols to transportdatabetween devices and backend servers.Mostprominentshortrange wirelesstechnologies include Zig-Bee, Bluetooth, Wi-Fi, Wireless Metropolitan Area Network (WiMAX) and IEEE802.11p which are primarily used in smart metering, e-healthcare and vehicular communication. Wide rangetechnologies such as Global System for Mobile communication (GSM) and GPRS, Long-Term Evolution(LTE), LTE- Advanced are commonly utilized in ITS such as vehicle-to infrastructure (V2I), mobile e-healthcare,smartgridandinfotainmentservices.Additionally,LTE-MisconsideredasanevolutionforcellularIoT (C-IoT). In Release 13, 3GPP plans to further improve coverage, battery lifetime as well as devicecomplexity[7].Besideswell-knownexistingprotocols,LoRaalliancestandardizestheLoRaWANprotocoltosupport smart city applications to primarily ensure interoperability between several operators. Moreover,SIGFOX is an ultra narrowband radio technology with full star-based infrastructure offers a high scalableglobal network for realizing smart city applications with extremely low power consumption. A comparativesummary2ofthe majorcommunicationprotocols.

ServiceProviders

Pike Research on smart cities estimated this market will grow to hundreds of billion dollars by 2020, with anannual growth of nearly 16 billion. IoT is recognized as a potential source to increase revenue of serviceproviders. Thus, well-known worldwide service providers have already started exploring this novel cuttingedge communication paradigm. Major service providers include Telefonica, SK telecom, Nokia, Ericsson,Vodafone,NTTDocomo,Orange,TelenorgroupandAT&T whichoffervarietyofservicesandplatforms forsmartcityapplicationssuchasITSandlogistics,smartmetering,homeautomationande-healthcare.

NetworkTypes

IoT based smart city applications rely on numerous network topologies to accomplish a fully autonomousenvironment. The capillary IoT networks offer services over a short range. Examples include wireless localarea networks(WLANs),BANsandwirelesspersonalarea networks (WPANs).Theapplicationareasincludeindoor e-healthcare services, home automation, street lighting. On the other hand, applications such as ITS,mobile e-healthcare and waste management use wide area networks (WANs), metropolitan area networks(MANs), and mobile communication networks. The above networks pose distinct features in terms of data,size,coverage,latencyrequirements,andcapacity.