

Communications & Controls in IoT

EC5.204

Introductory Class

Instructors: Sachin Chaudhari and Aftab Hussain

Emails: firstname.lastname@iiit.ac.in

Jan. 05, 2023



**INTERNATIONAL INSTITUTE OF
INFORMATION TECHNOLOGY**

H Y D E R A B A D

Communications & Controls in IoT

EC5.204

Introductory Class

Instructors: Sachin Chaudhari and Aftab Hussain

Emails: firstname.lastname@iiit.ac.in

Jan. 05, 2023



**INTERNATIONAL INSTITUTE OF
INFORMATION TECHNOLOGY**

H Y D E R A B A D

Background: Sachin Chaudhari

- **Academics**
 - Associate Professor, SPCRC and SCRC, IIIT Hyderabad (Jul. 2021-onwards)
 - Asst Prof., SPCRC, IIIT Hyderabad (Jan. 2015-Jun. 2021)
 - Postdoc, Aalto University, Finland (2013-2014)
 - PhD, Aalto University, Finland (2007-2012)
 - M.E., IISc Bangalore, India (2002-2004)
 - B.E., VNIT, Nagpur (1998-2002)
- **Industry**
 - Senior Wireless Communication Engineer, Esqube Communications, Bangalore (an IISc start-up) (2004-2007)
- **Research Interests:**
 - Signal Processing and Machine Learning for Wireless Communication:
IoT for Smart Cities, 5G/6G, Satellite Communications
- **Research Projects on IoT**
 - **DST and PRIF:** IoT Enabled Smart Cities: Pollution Health and Governance
 - **CoE on IoT for Smart Cities:** Coordinator
 - **India's First *Living Lab* for Smart City Research**

Background: Aftab Hussain

- **Academics**
 - Asst Prof., IIIT Hyderabad (2018-present)
 - Postdoc Fellow, Harvard University (2016-2018)
 - MS+PhD, KAUST, Saudi Arabia (2011-2016)
 - B.Tech, IIT Roorkee (2005-2009)
- **Industry**
 - Design Engineer, Analog Devices India (2010-2011)
- **Research Interests:** Flexible electronics, sensor systems, smart cities, IoT
- **Research Projects**
 - Artificial muscles
 - Pressure sensor mat
 - 2-wheeler safety
 - Water sensors

Outline

- Course Intro
 - Introduction to IoT
 - Motivation and Importance
 - Few of the IoT activities at IIITH
- Course Administration
 - Syllabus
 - Resources
 - Evaluation
 - Tutorials

Introduction and Motivation

Internet of Things (IoT)

- [webopedia] The Internet of Things refers to the ever-growing network of **physical objects** that feature an **IP address** for internet connectivity, and the **communication** that occurs **between these objects** and other Internet-enabled devices and systems.

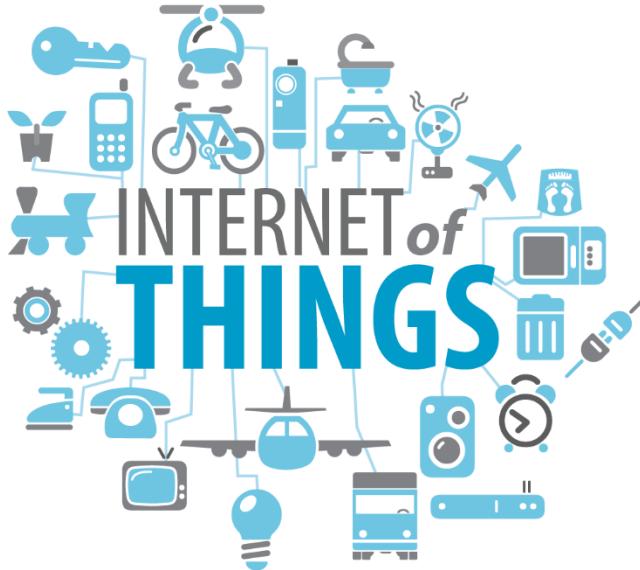


Image: <http://www.meccanismocomplesso.org/en/iot-internet-of-things/>

- IoT extends internet connectivity beyond traditional devices such as computer and smart-phones to a diverse range of devices such as thermostats, cars, lights, vending machine etc.

Differences between Computers and IoT Devices

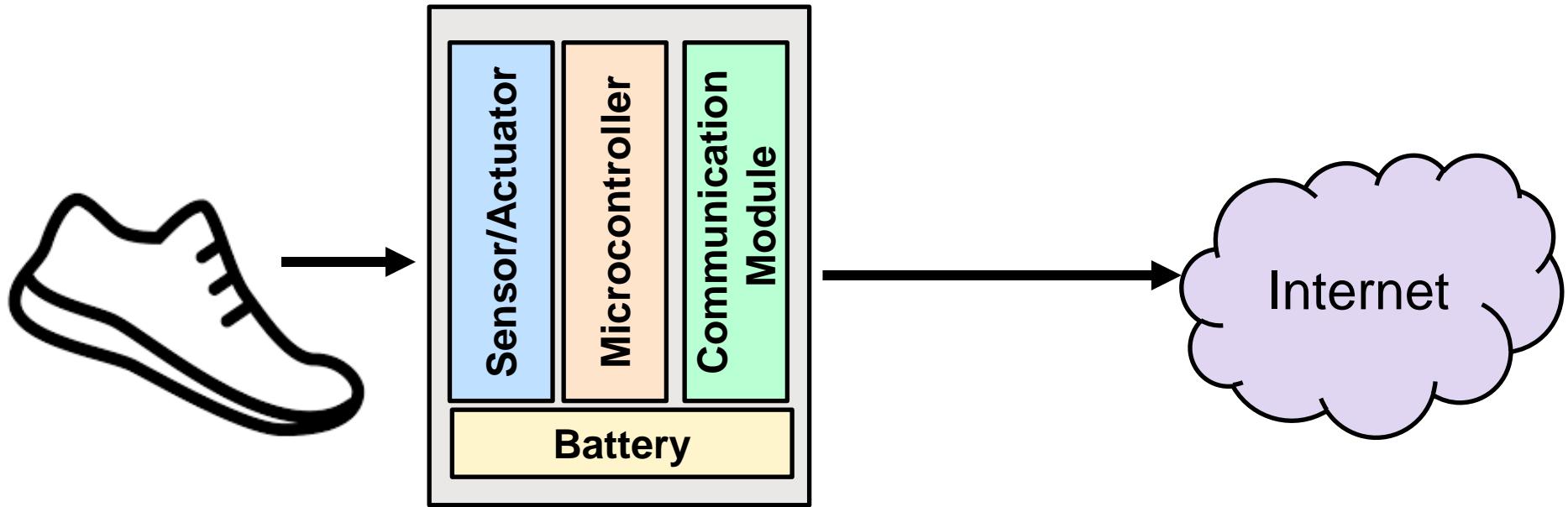
Computers

- Main task is to compute and run programs
- General purpose
- Significant resources
- Expensive
- Example: iPad

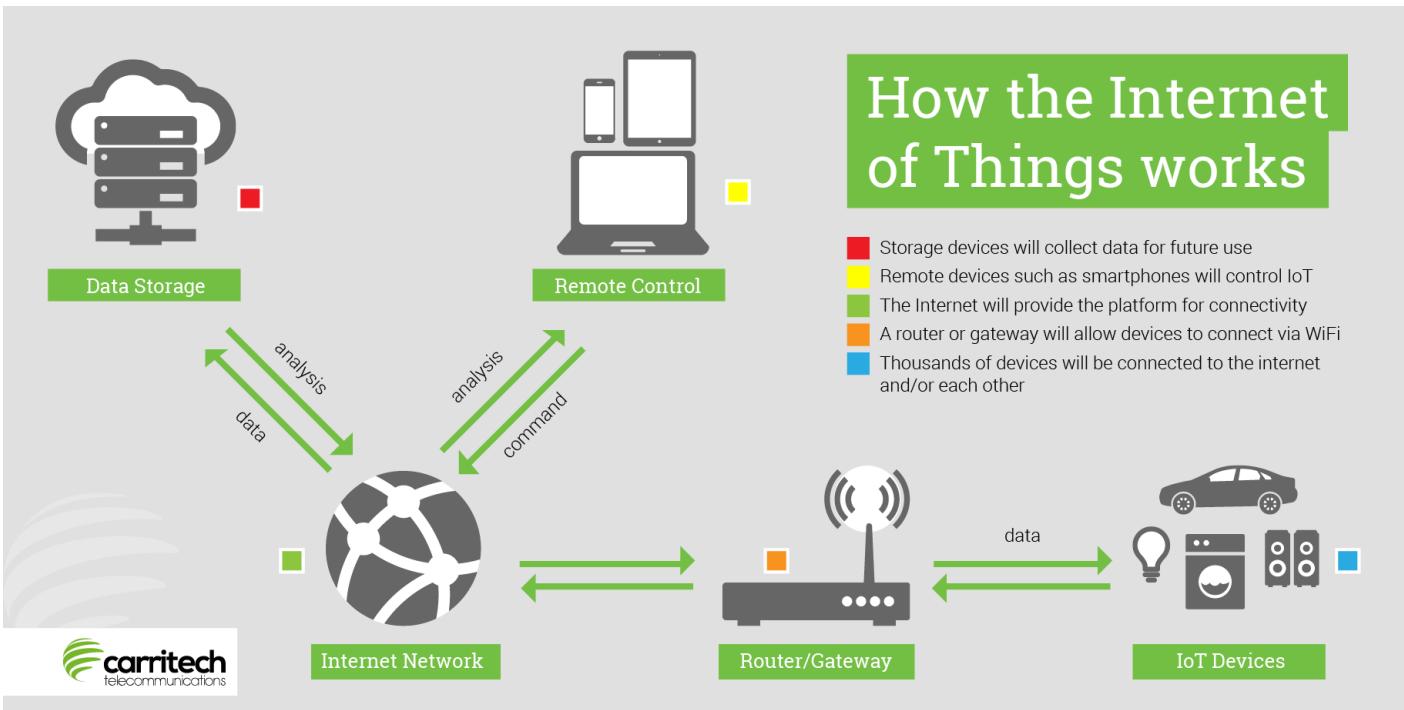
IoT Devices

- Main task is not computing
- Does specific application
- Limited resources
- Cheaper, efficient and faster for one (or very few things)
- Example: Washing Machine

How do you connect a thing to internet?



How does IoT work?



Picture Credit: <http://www.carritech.com/news/internet-of-things/>

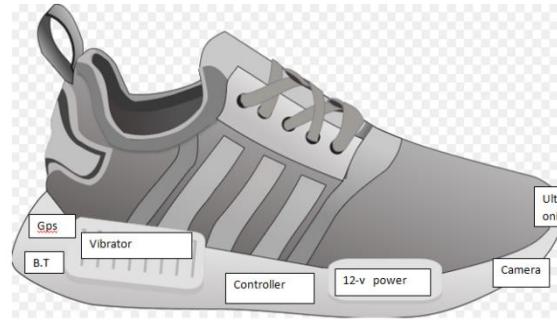
Why to connect a *thing* to internet?



Speaker+Mic

Smart Speakers :
Amazon Echo / Google Home

https://en.wikipedia.org/wiki/Smart_speaker



Shoes

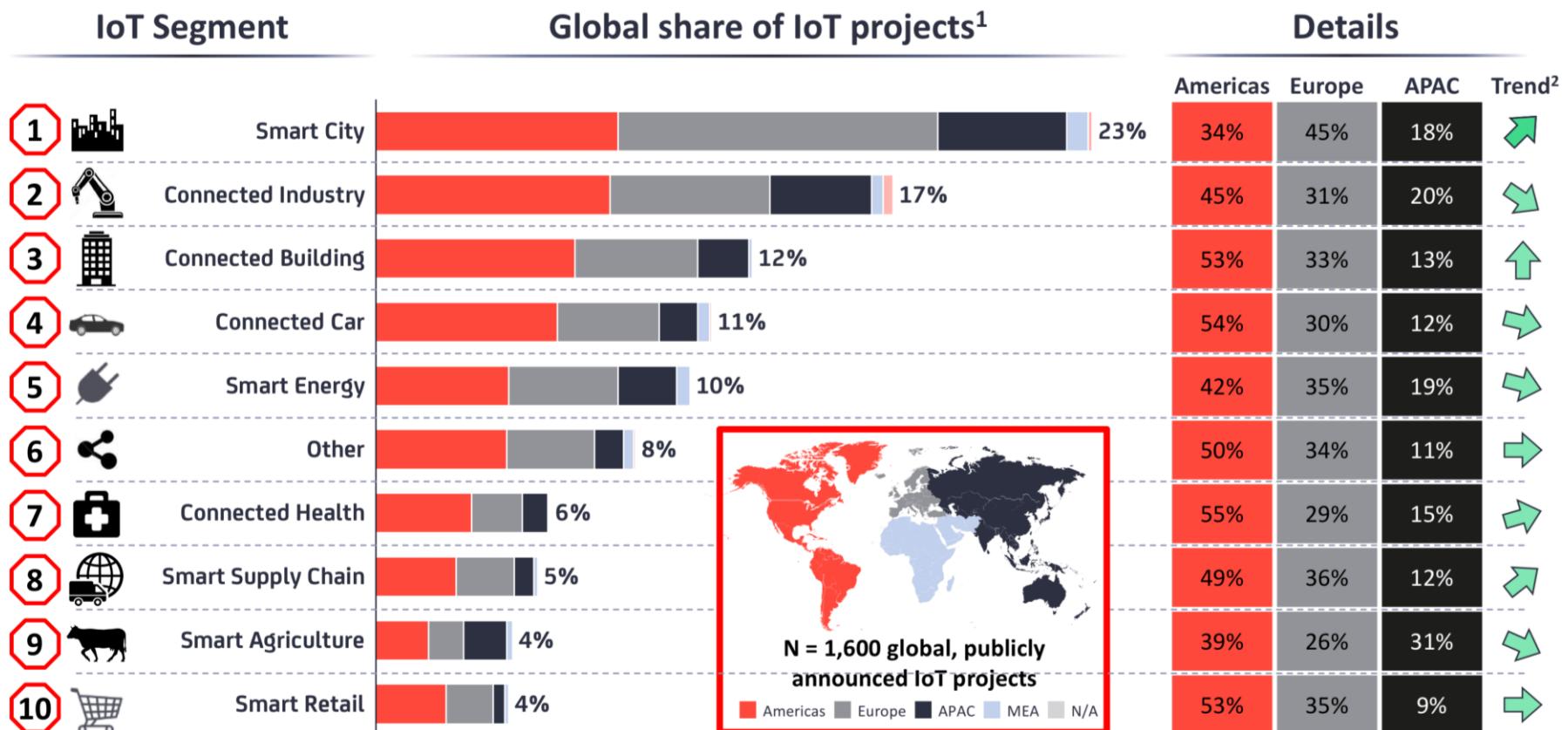
Smart Shoes

<https://innovate.mygov.in/innovation/smart-shoes-for-blind-person/>

Why IoT?

- Adds *smartness* to simple things
 - *To be smart, a thing doesn't need to have super storage or a supercomputer inside of it . All a thing has to do is connect to super storage or to a super computer*
 - <https://www.leverage.com/iot-ebook/what-is-iot>
- Consumers
 - Ease of access from anywhere and anytime
 - Efficient systems and reduced bills
- Companies
 - Real-time monitoring and response
 - Reduction in human errors
 - Increase in productivity
 - Predictive analysis

Applications of IoT



1.Based on 1,600 publicly known enterprise IoT projects (Not including consumer IoT projects e.g., Wearables, Smart Home). 2.Trend based on comparison with % of projects in the 2016 IoT Analytics Enterprise IoT Projects List. A downward arrow means the relative share of all projects has declined, not the overall number of projects 3. Not including Consumer Smart Home Solutions. **Source:** IoT Analytics 2018 Global overview of 1,600 enterprise IoT use cases (Jan 2018)

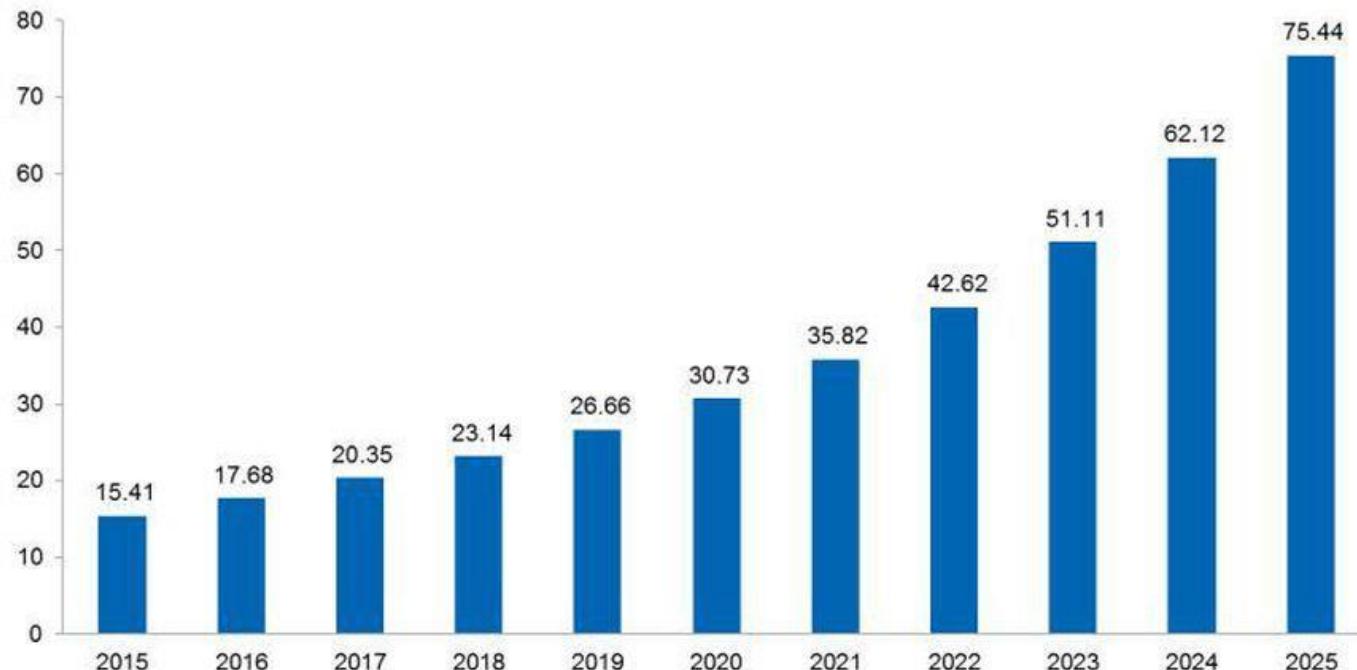
Source: IoT Analytics, Jan 2018

Source: The Top 10 IoT Segments in 2018 based on 1,600 real IoT projects, [IoT Analytics](#)

Rise of the IoT!

Figure 1. The IoT market will be massive

IoT installed base, global market, billions



Source: IHS

© 2016 IHS

<https://www.exuberantsolutions.com/iot-training.htm>

Companies in IoT (Only a few shown)

Professional Services	IT	SIEMENS	IoT	YAMAHA Rockwell NEC HITACHI
	IBM hp CSC accenture ARICENT GROUP Atos	Johnson Controls Honeywell Schneider Electric	Deloitte	YAMAHA xerox HITACHI
Datacenter Providers	IBM ORACLE	cisco	amazon	hp huawei lenovo
Communications	NTT at&t verizon	Communications Services Providers	vodafone	america m&v france telecom
	Telit HITACHI juniper	Infrastructure / Gateway	cisco	hp huawei fujitsu
Software	RTOS Microsoft MQX TinyOS	xively xBISQUARE GENIUS ThingWorx WIND RIVER	Software Stack	splunk osnapse sensinode ABO DATA ProSyst Jasper flexeye Megavendors
		EUROTECH Axeda zonoff	meshsystems blueforce arkessa	Microsoft ORACLE SAP IBM
Hardware	Healthcare	Transportation	Utilities	Design
	Medtronic johnson & johnson MCKESSON	Continental Michelin Michelin Ford	Siemens ABB	CONTINUUM frog design Quirby
	Smart Home	Consumer	Industrial	
	Whirlpool LG SAMSUNG MAYTAG CHAMBERLAIN	fitbit Google Wii JAWBONE	BOSCH PHILIPS cradlepoint Panasonic	
	Semiconductors	genuine	ODM / EMS	
	ST Qualcomm ARM Infineon intel RENESAS	InvenSense freescale Imagination	Kontrol FLEXTRONICS PEGATRON FOXCONN	

Everybody wants a pie of IoT!!!

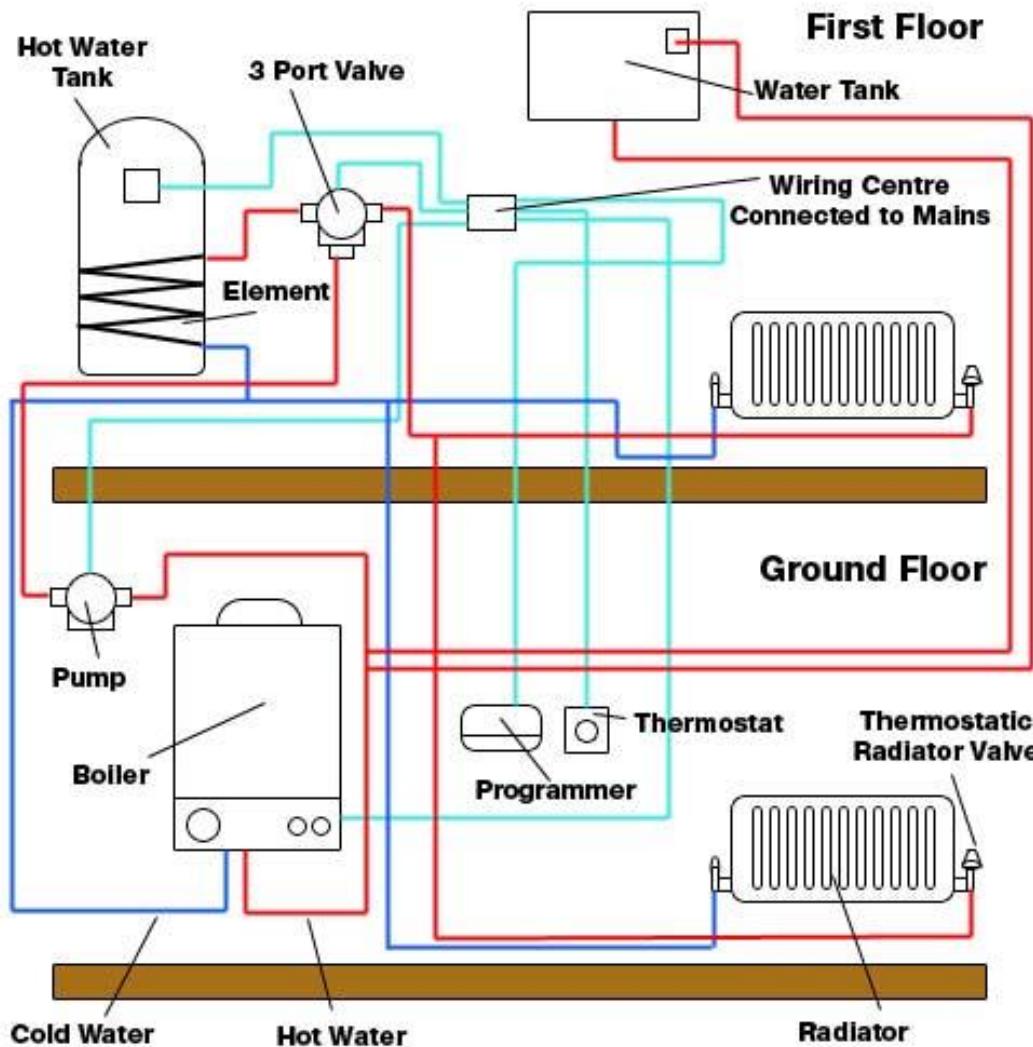
<https://www.quora.com/What-are-the-top-IoT-companies>

Use Case: Nest Learning Thermostat

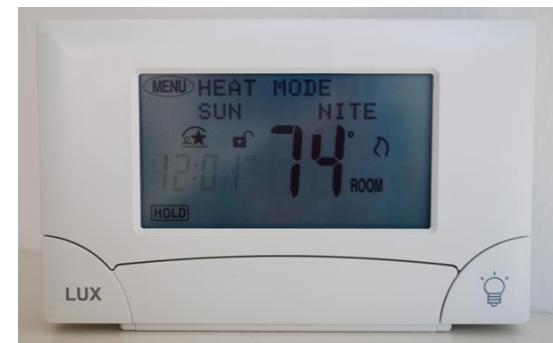


- Founded in 2010 by former Apple engineers
- Learns what temperature you like and builds a schedule around yours
- <https://www.youtube.com/watch?v=HhqD-ljcD6I>
- Google acquired Nest Labs for \$3.2 billion in cash in Jan. 2014

Thermostat?



Centralized Heating



Thermostat

Use Cases: Amazon Dash



https://en.wikipedia.org/wiki/Amazon_Dash

- Started in 2015
- Replenishment services
 - <https://www.youtube.com/watch?v=-OgPTC0EB48>
- Discontinued in 2019
 - Alexa
 - Subscriptions

Use Cases: Agriculture

- IoT based Soil-Crop-Atmosphere Screening
 - monitor pH level, temperature, airflow, water, manure, fertilizers, precipitation, nutrients and light
 - ML based algorithms to solve problems
- Objectives
 - Better crop selection and planning
 - Optimized inputs, Irrigation and Fertigation Schedule
 - Real time detection of diseases
 - Pest control and enhanced yield



Use Cases: Industrial IoT



<https://altizon.com/what-is-iiot-and-its-benefits/>

- IoT refers to consumer IoT while IIoT refers to Industrial IoT
 - IoT: simple and low-risk applications, low-cost sensors
 - IIoT: sophisticated high-risk applications, precision sensors
- Applications for industrial IoT
 - aerospace, defense, healthcare and energy
 - Improving productivity, safety, reliability

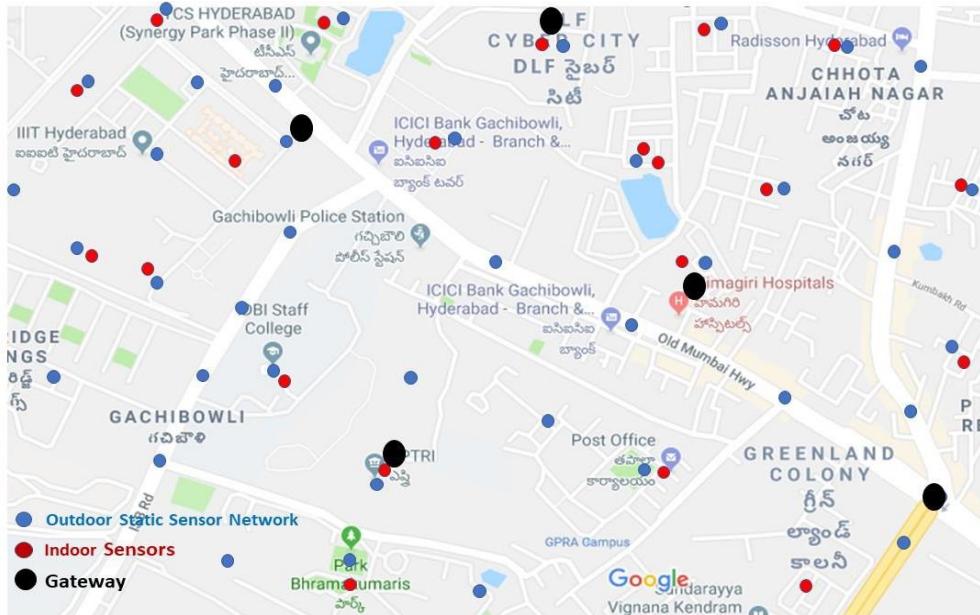
IoT: Necessity in Times of Covid

- Health
 - Remote monitoring
 - Telemedicine
 - Compliance
- Getting manufacturing back on track
 - Remote monitoring and control

What thing will you connect to the Internet?

Few of the IoT Activities at IIITH!!!

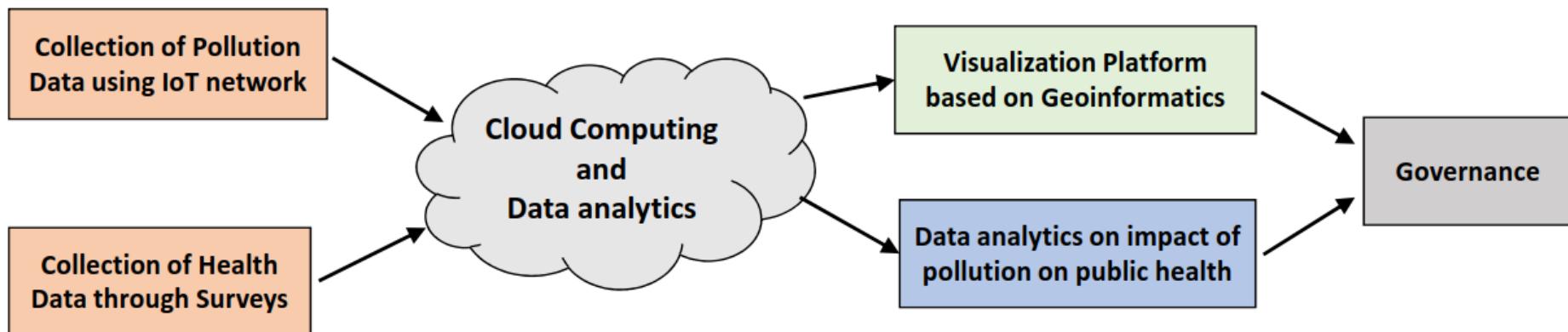
IoT Enabled Smart Cities: Pollution, Health and Governance



DST and PRIF-Funded project

Sachin Chaudhari (PI)
Aftab Hussain
Kavita Vemuri
K. Rajan
Dr. Shailaja Tetali

6 papers published and
one patent filed

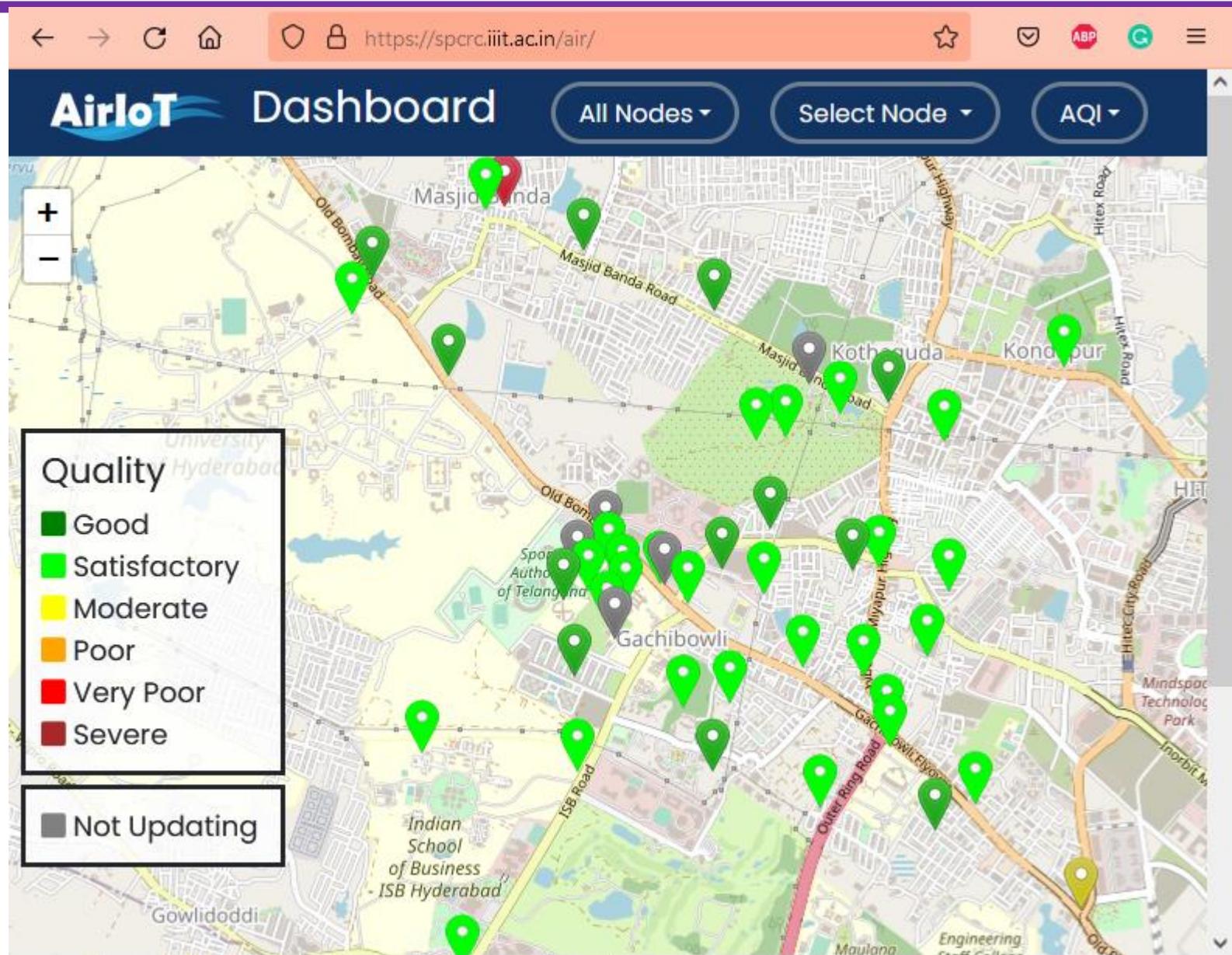


Development for APM

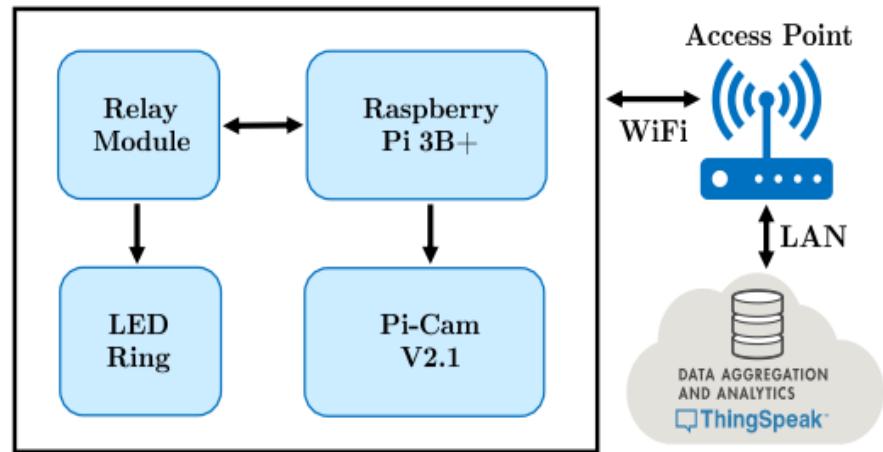
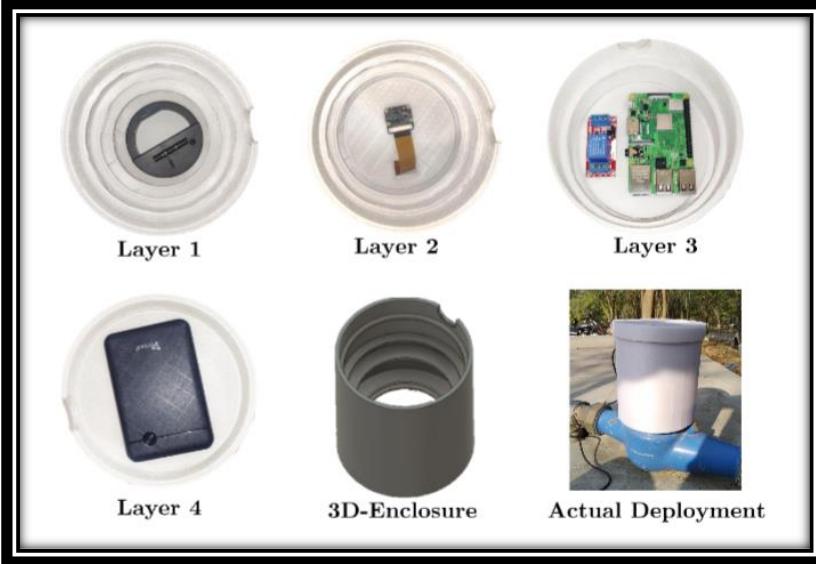


- Developed a product quality low-cost pollution node
- Patents filed
- Deployed 40 more nodes in Gachibowli region (extended IIITH region)
- Deployed 3 mobile nodes on GNITS buses
 - Plan to deploy 7 mobile nodes in collaboration with easyCommute and GNITS buses
- ML based calibration of PM sensors done
- Experiment with Gas sensors and their calibration going on

Web-based Dashboard



Making Analog Water Meters Smart!



- Developed an IoT and Learning (ML/DL)-based low-cost retrofit mechanism to digitize analog water meters to make them smart.
 - Low-Cost (we are working on making it optimized in terms of looks, cost, and robustness)
 - Retrofit model
 - ML based algorithm converts images to digits and send data to server
 - Patent Filed, Papers Published, Planning a start-up
- UG students enthusiastically worked and published an international conference and filed patent on this
 - Won the Water Challenge by the Telangana State Government

IoT Enabled E-bike chargers



- PI: Aftab Hussain
- Patent filed
- We have developed an E-bike charging system based on globally recognized OCPP and OneM2M standards
- It is compact, and low-cost specifically focused on 2-wheeler charging
- In the process of being commercialized through a partner company

Flexible pressure sensor array

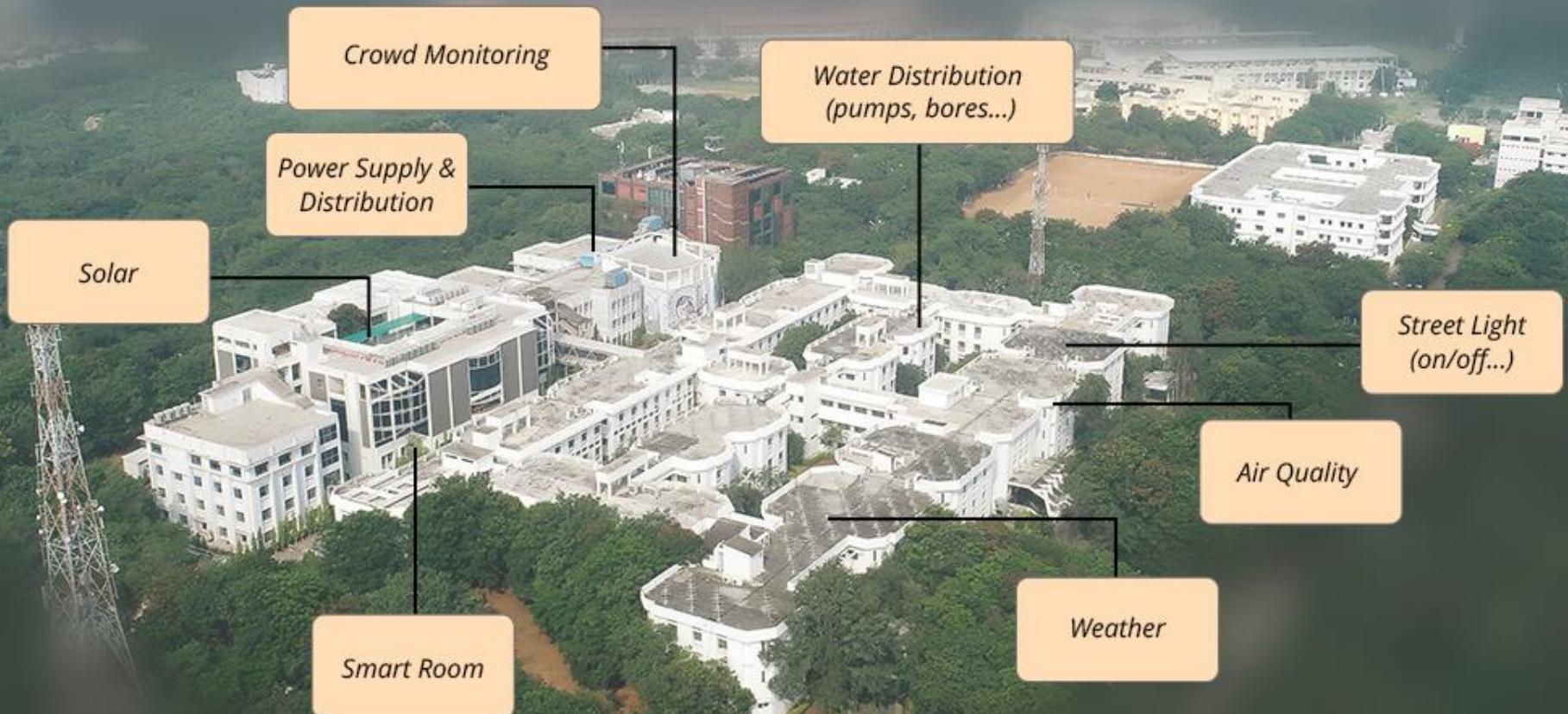


- PI: Aftab Hussain
- Patent filed
- Flexible pressure sensor: We have developed a flexible pressure sensor array that can provide real-time pressure distribution over a large area
- We have incorporated a startup for commercialization of the product

CoE on IoT for Smart Cities

- Started Jan. 2019
- Supported by India-EU collaboration on ICT Standardization, TSDSI and ETSI
- Faculty involved
 - S. Chaudhari (Coordinator), A. Hussain, R. Loganathan, V. Garg, D. Gangadharan, K. Vemuri, K. Rajan
- Activities Supported
 - **Knowledge initiatives:** Tutorials, Hackathons, Workshops
 - **Research collaboration** with LAAS-CNRS and INSA (Toulouse, France), Bordeaux metropole, NTNU (Norway), LTU (Sweden) and IITG
 - **Resulted in Living Lab project and Smart City Research Center**

India's First Living Lab for Smart Cities



Started April 2020

Living Lab: Team



Living Lab: Objectives



Research on IoT
for Smart Cities



Data for AI/ML



Test-bed for
smart city
deployments



Promote start-
ups

Living Lab: Themes



Water Monitoring and Distribution

Non-revenue water
Quality



Safety and Security

Crowd monitoring
Street Lights
Structural health of buildings



Health

Air Pollution + Weather
Social Distancing



Energy

Building Energy Efficiency
Solar energy
Smart rooms

Living Lab: Dashboard



Living Lab: Dashboard



IoT Courses at IIITH

IoT Workshop (MTech CASE)

Introduction to IoT (2nd Semester CSE)

Embedded System Workshop (3rd Semester CSE)

Communication and Controls in IoT (4th Semester ECE)

Talent Sprint: <https://iiit-h.talentsprint.com/iot/> (for professionals)

Importance of this subject

- Solve relevant problems in Indian Smart Cities
- Opportunity to work in Smart City Research Center (SCRC)
 - Sachin Chaudhari (SPCRC)
 - Aftab Hussain (CVEST)
 - Deepak Gangadharan (CSG)
 - Karthik Vaidyanathan (SERC)
- Startups!
 - CIE
 - Product lab
- Much more fun than theoretical subjects!

Course Details/ Logistics

Syllabus (Tentative)

- Introduction to IoT
- Sensing and Actuation
- Microcontroller based Embedded System Design
- Interfacing of Sensors and Actuators
- Basics of Networking
- Communication Protocols: WiFi/Bluetooth/Zigbee/LoRaWAN/NB-IoT,
- Data Protocols: MQTT/CoAP
- Sensor Networks, *Edge, Fog and Cloud Computing*,
- Interoperability in IoT
- Smart City Applications

Resources

BOOKS

- P. Lea, *Internet of Things for Architects*, Packt, 2018
- Raj Kamal, *Internet of Things*, McGraw Hill, 2018
- O. Hersistent, D. Boswarthick, O. Elloumi, *The Internet of Things*, Wiley, 2016
- D. Norris, *The Internet of Things*, McGraw Hill, 2015
- A. Bahga and V. Madisetti, *Internet of Things*, University Press, 2016

VIDEOS

- National Programme on Technology Enhanced Learning (NPTEL) and SWAYAM
 - Introduction to Internet of Things, Sudip Misra, IITK
 - https://swayam.gov.in/nd1_noc19_cs65/preview
- Research papers and online content

Course Portal

MOODLE: <https://courses.iiit.ac.in/>

Under Spring 2023

If you still need to get enroled, email us.

- News
- Discussion Forum
- Projects

Projects

- Themes:
 1. Mobile air pollution
 2. Smart agriculture
 3. 2-wheeler safety
 4. Water sensors
- Team of 4 students
 - TAs will circulate a form, please make your own teams
- Funded project
 - Teams will be reimbursed till maximum of Rs. 10 K per team for the components purchased
 - Check with lab if those components are already available
 - Please consult the faculty before purchasing the project specific sensors
 - Bills should be proper and in the name of IIIT Hyderabad.
 - Students will send soft-copy of bills to the faculty in advance.
 - Note that the reimbursement bills must be submitted by 15 Feb. 2023.

Teaching Assistants

- Rishabh Agarwal
- Vayur Shanbag
- Kirti Vignan Reddy
- Ivin Kuriakose

Exams and Evaluation

- Mark Distribution
 - First Quiz (20)
 - Final (50) : (MidSem as this a half course)
 - Project (30)
- Grading: Relative (TBD)

Questions?

- **That's all for today!**
- Next class on **Monday**!

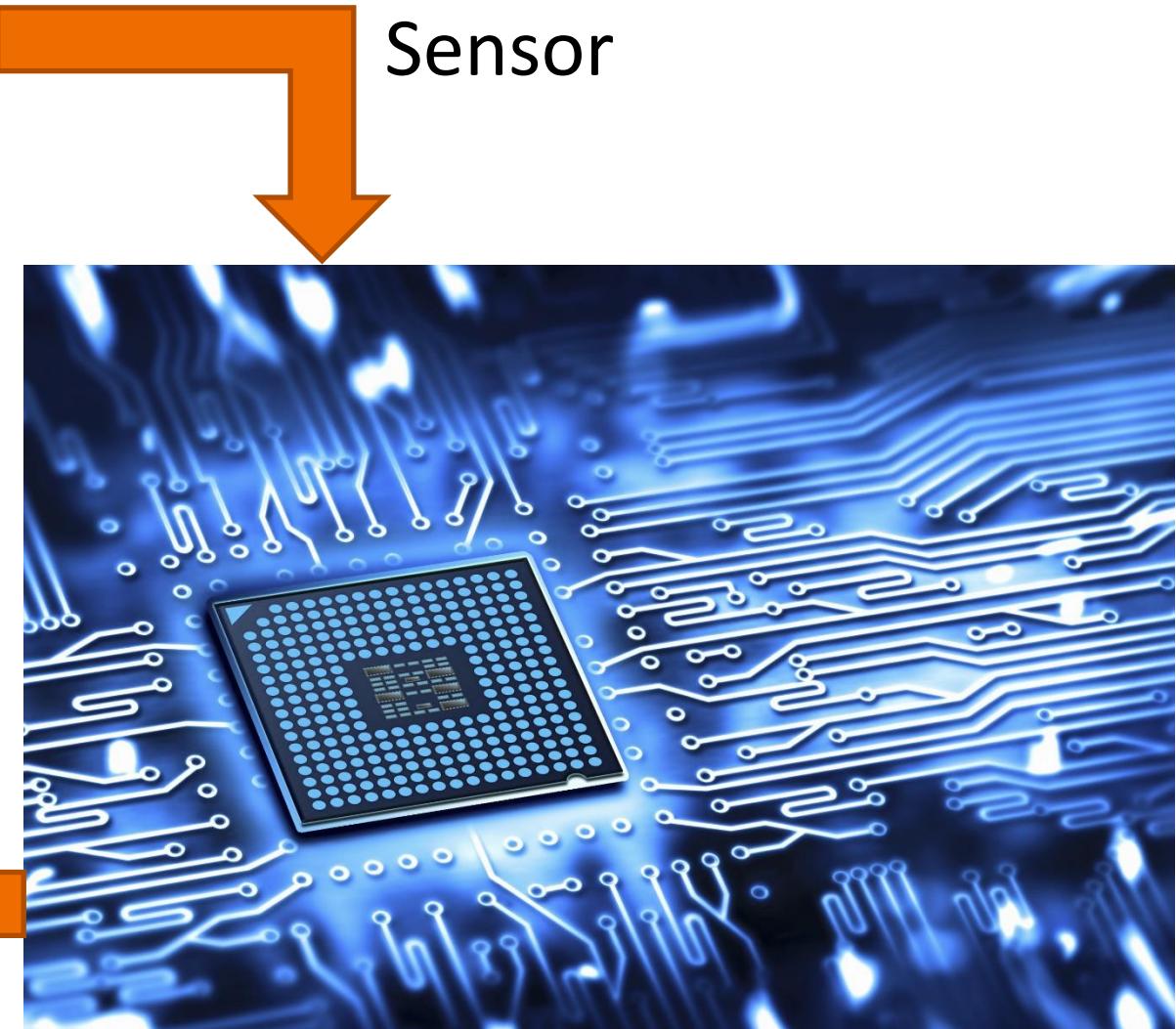
Sensors and Actuators

Dr. Aftab M. Hussain,
IIIT Hyderabad

Sensors and actuators



Actuator



Sensor



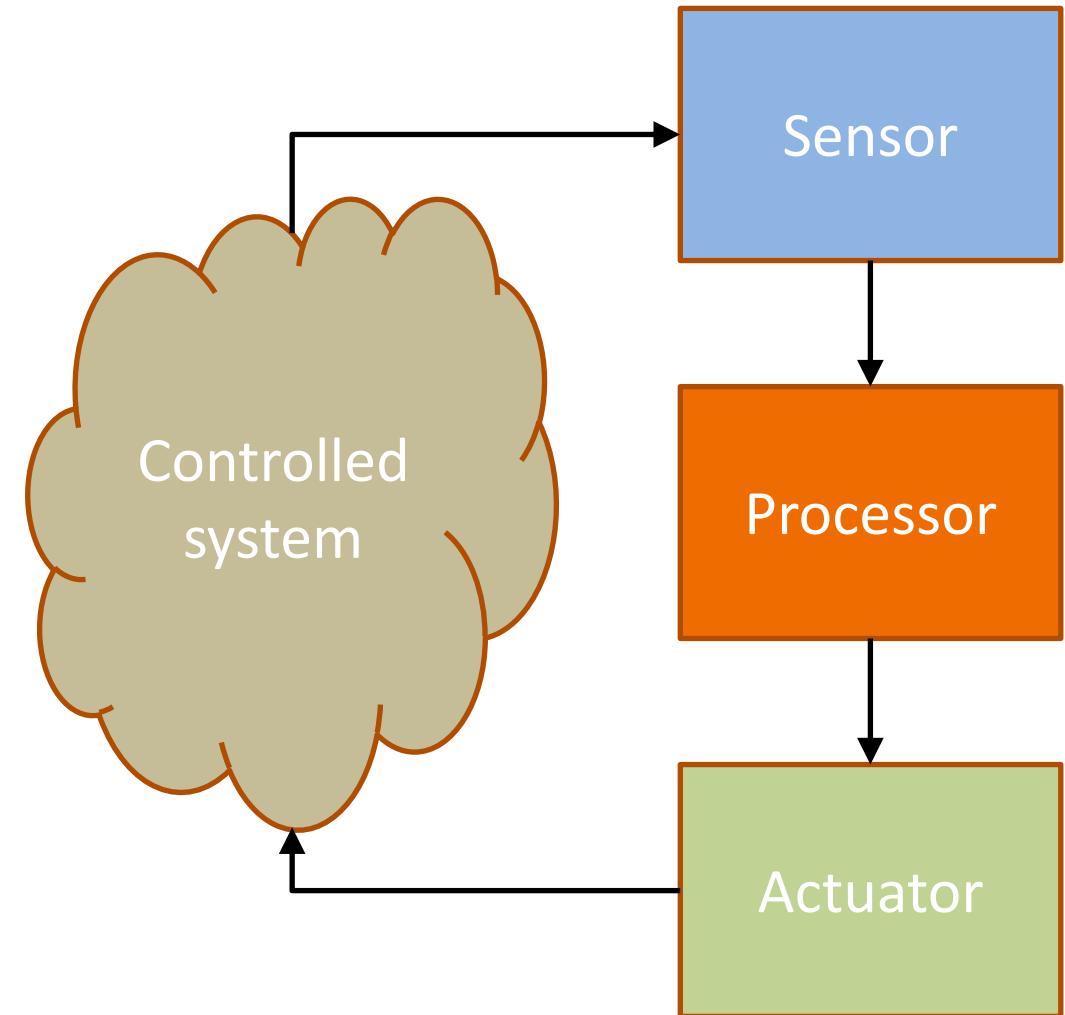
Transducers

- Transducers convert one form of energy into another
- Sensors and actuators are both transducers
 - Sensors convert a physical phenomenon into electrical signal
 - Actuators convert an electrical signal into a physical phenomenon
- Some things can do both
 - Thermocouple



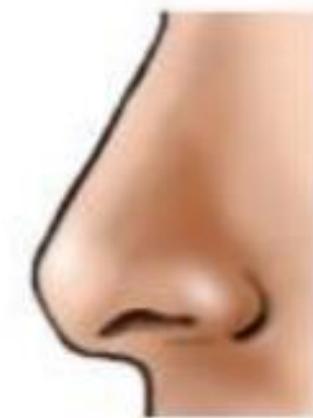
Transducers

- Physical signals need to be converted into proportional electrical signals (mostly analog voltage)
- Then we digitize the analog voltage using analog to digital converters (ADCs)
- The processor is able to process/store/transmit *only* digital signals (voltages)
- Actuators convert digital signals into physical signals (like light, sound, movement etc.)



Human sensors

- Human beings are equipped with 5 different types of sensors

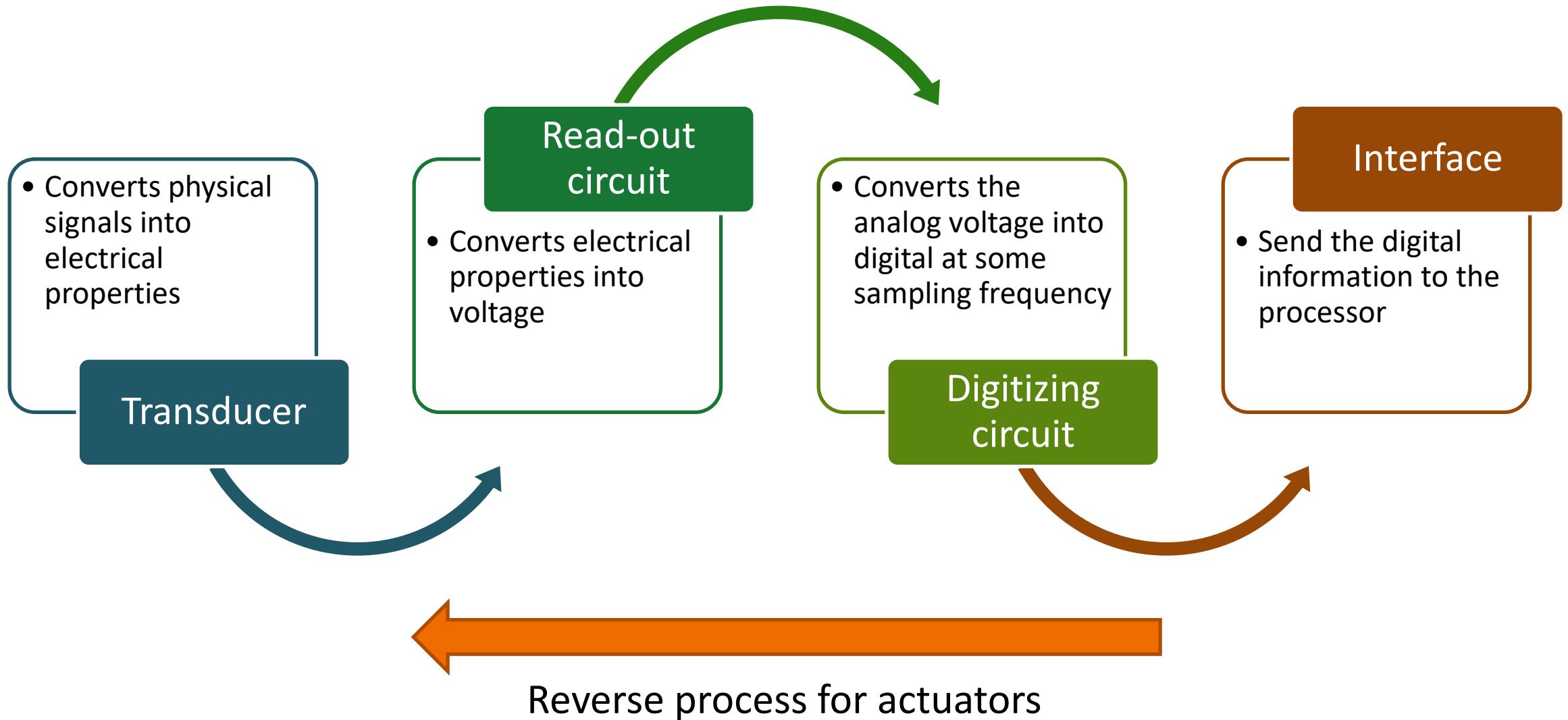


American scientists **David Julius and Ardem Patapoutian** have won the 2021 Nobel Prize for Physiology or Medicine for their discoveries of receptors for temperature and touch

So how to transduce to electronics?

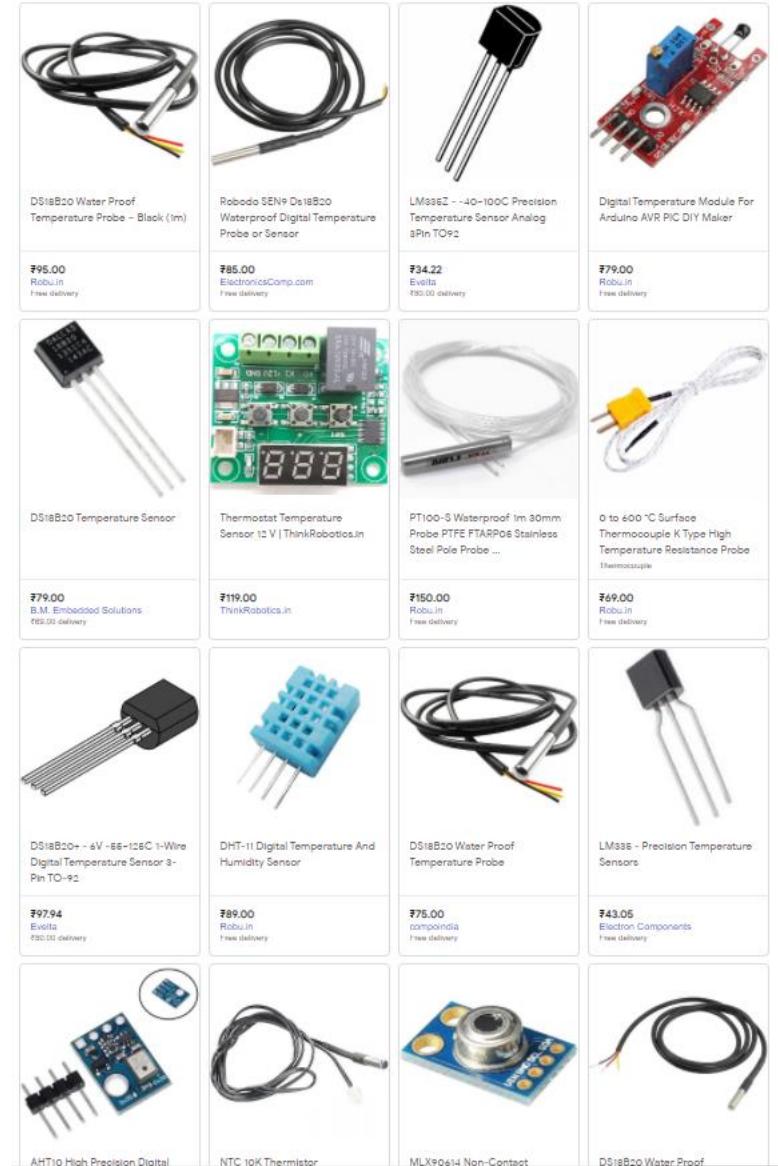
- We need to influence electronic properties using physical entities
- Electronic properties:
 - Resistance
 - Capacitance
 - Frequency
 - Charge density
- Sensors use the change in electronic properties because of physical phenomenon

Sensor systems



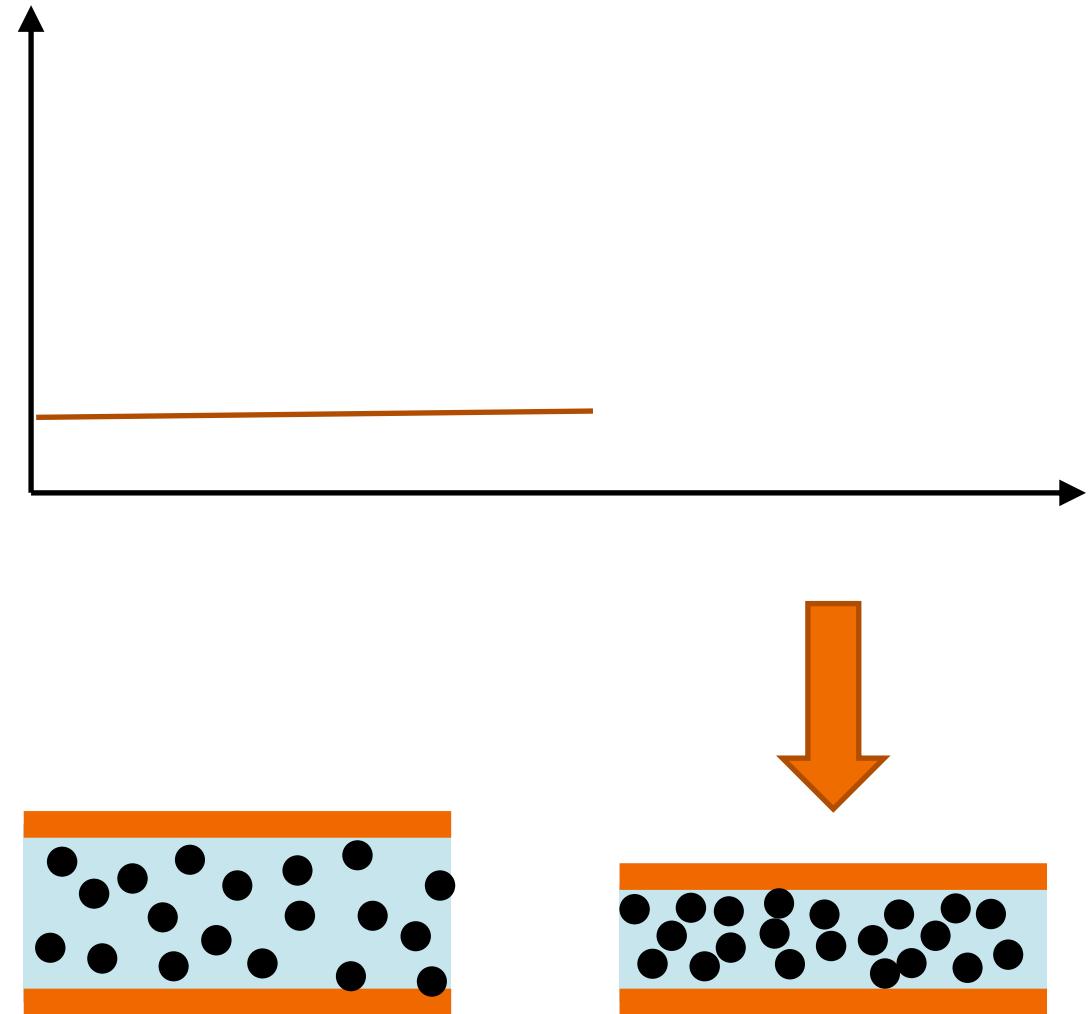
Sensor choice

- Lets say we want to measure the temperature of a room...
- We need a temperature sensor... but which one?
- To decide properly, we need to know:
 - Range
 - Sensitivity
 - Cost
 - Availability
 - Ease of installation etc
- Two main factors govern these parameters:
 - The physical principle involved in sensing (transducer)
 - The interface from transducer to processor (read-out circuit)



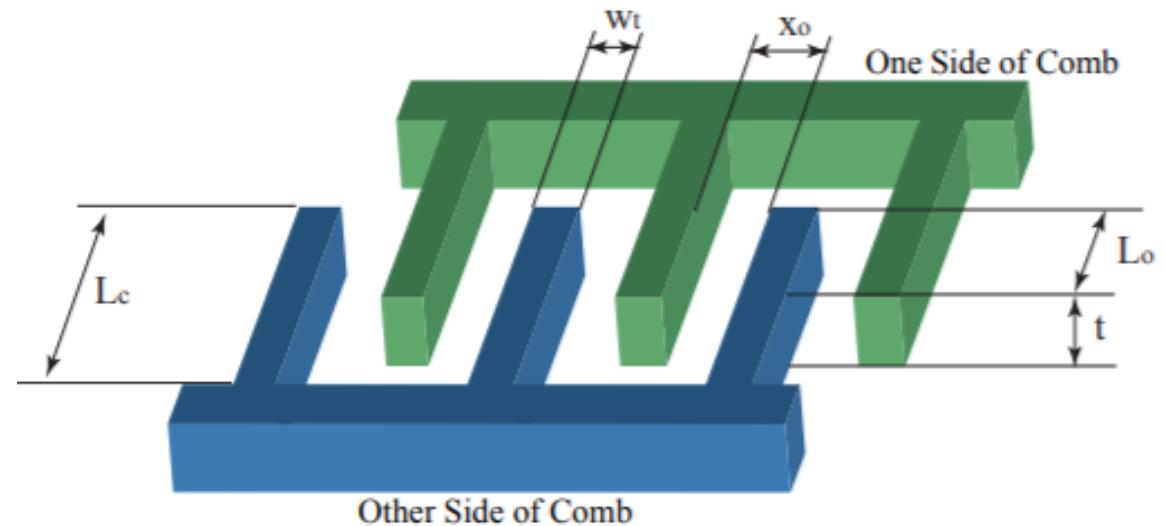
Flexible pressure sensor

- Force sensitive resistor (FSR) creates change in resistance with change in pressure
- One way of creating these is to produce a conductive polymer composite thin film using conductive particles
- When pressure is applied, the thin film compresses and causes the conductive particles to come together, reducing the resistance between the electrodes
- This can be passed through an RVC to make a force/pressure sensor



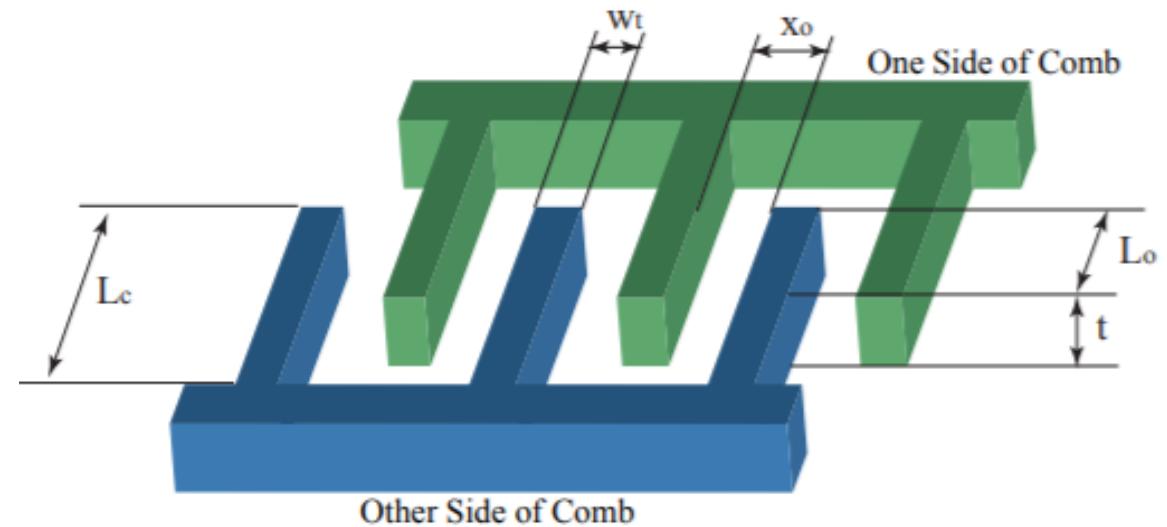
Accelerometers

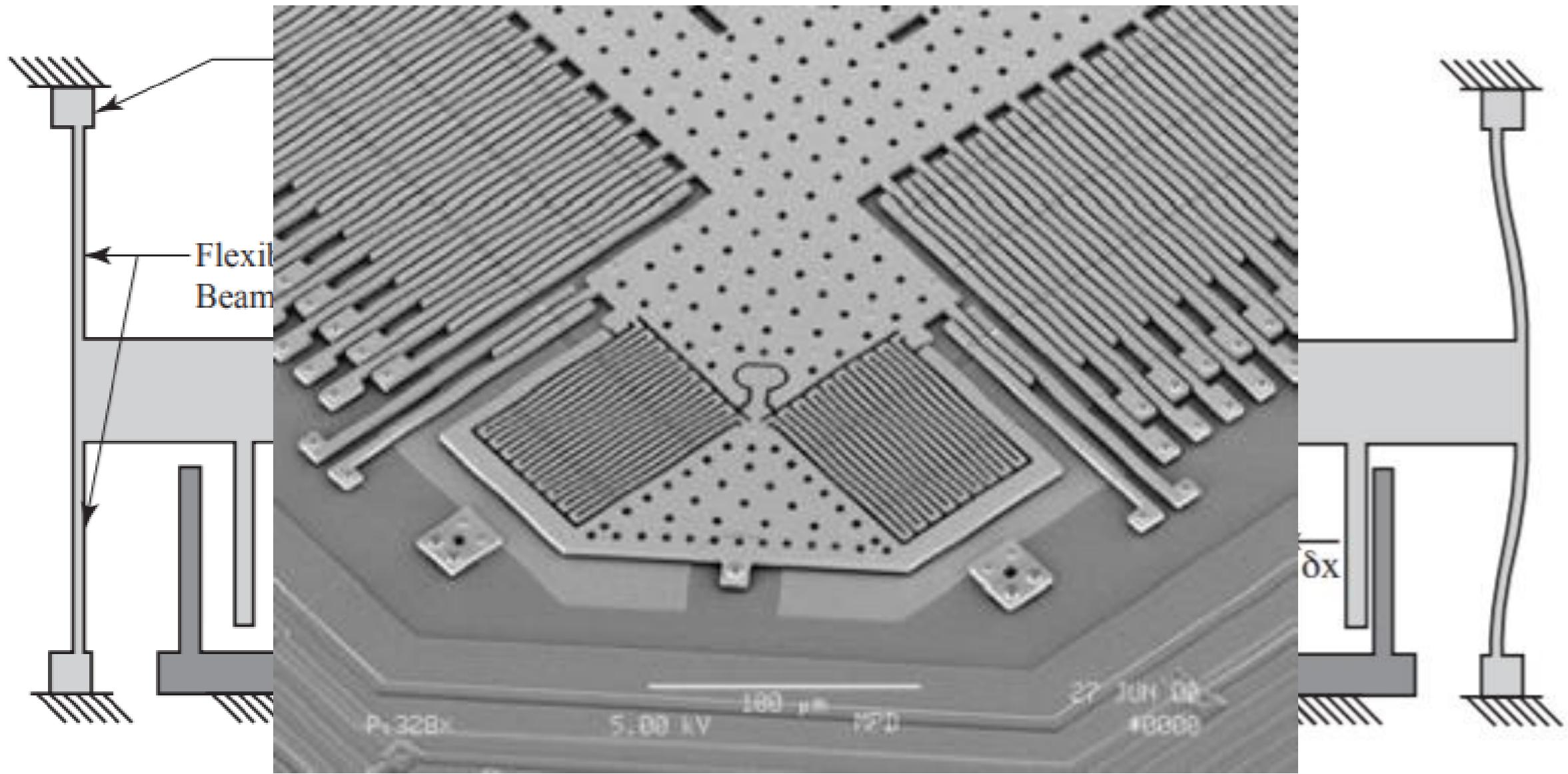
- If you have a linear displacement sensor, you can differentiate the output time series to get acceleration
- But some systems give output only for an accelerating system
- Classic example is the microcomb structure present in all of our smart phones!



Accelerometers

- Most commonly made using interdigitated fingers (or comb drive) with one side free to move and the other side static
- When this structure is accelerated, the free side tilts because of inertia, causing a change in capacitance of the structure
- For a given acceleration, the capacitance difference between the adjacent fingers changes
- With the stiffness of the structure known, the acceleration can be calculated





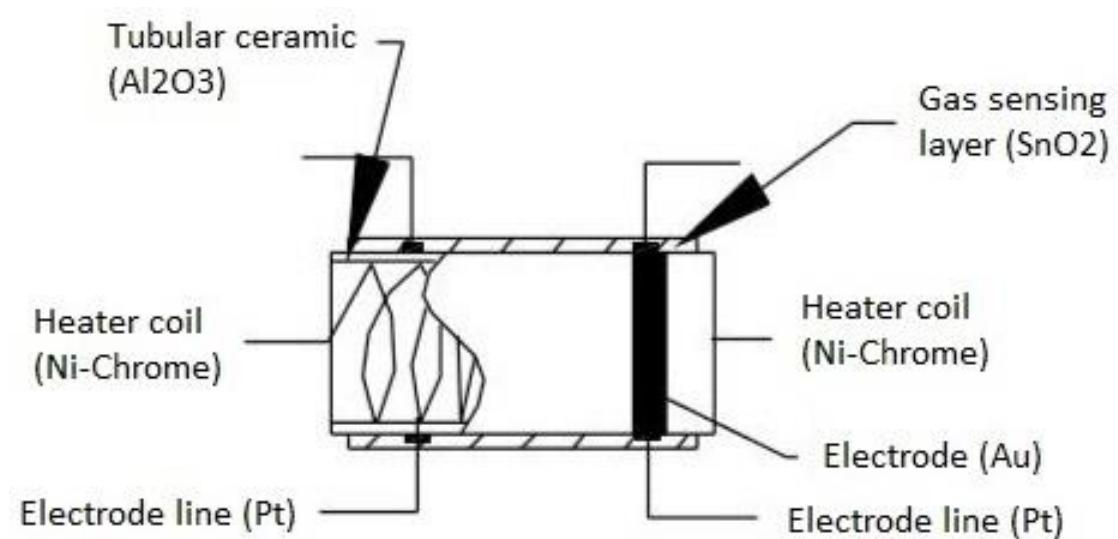
Humidity

- Humidity sensors rely on the fact that electrical properties of thin films changes because of adsorption of moisture on their surface
- This is used in capacitive humidity sensors to create changes in capacitance because of the change in dielectric constant of a metal oxide thin film upon exposure to moisture
- Humidity changes resistivity as well, but that is also very temperature sensitive



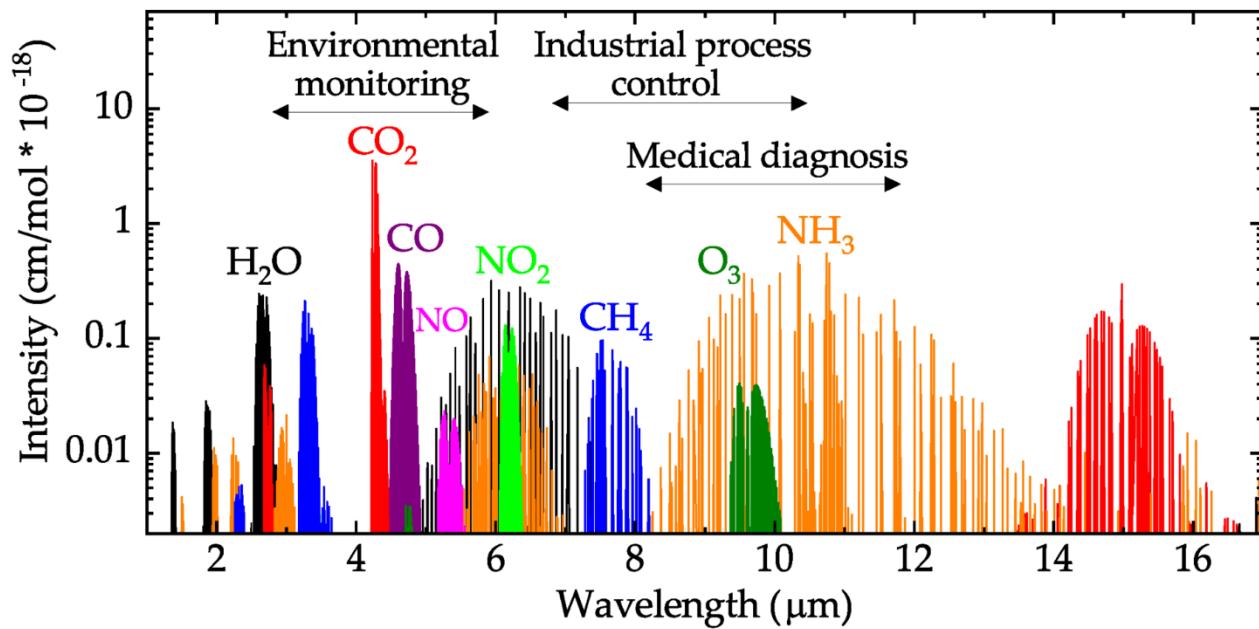
Gas sensors

- Most popular method is the metal oxide gas sensor with SnO_2 as the sensing layer
- Whenever certain gases are present near the thin film, they are adsorbed by it
- The resistance of the thin film changes and the current through it varies which represents the change in concentration of the gases
- The adsorption of different gases is a function of temperature – thus a heater is placed to determine selectivity



Gas sensors

- Non-disruptive infrared sensor (NDIR) is used for sensing gases based on their spectral response
- An IR LED with a very specific wavelength (depending on the gas to be sensed) is used as the source
- The light passes through the sample gas and a detector detects the incoming intensity
- Very selective to other gases
- Most commonly used for CO_2



Transduction

- Transducing into electrical domain

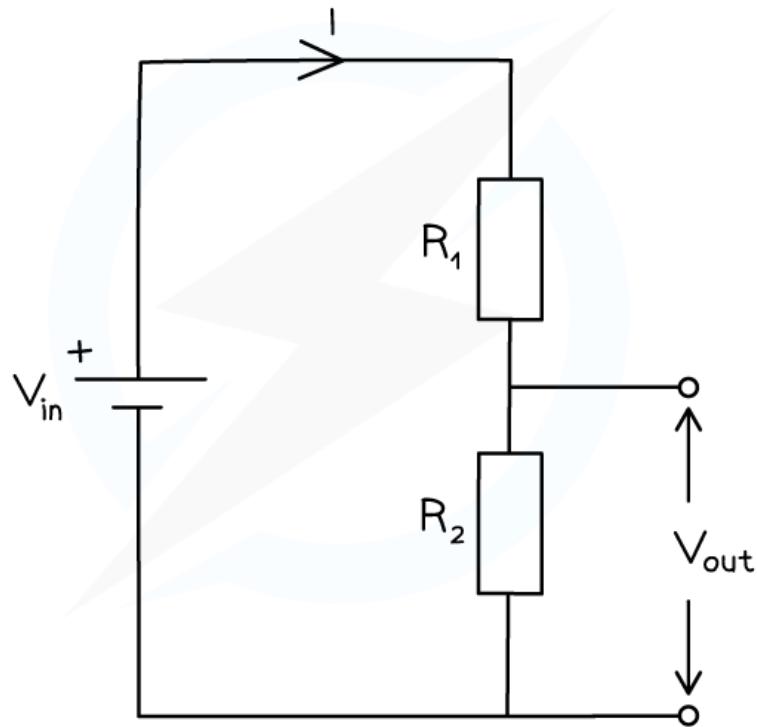
Sensed quantity	Electrical parameter
Strain, force, pressure	Resistance
Linear acceleration	Capacitance
Angular velocity	Capacitance
Rotation	Frequency
Fluid flow	Frequency (through rotation of turbine)
Temperature	Current, resistance, voltage
Humidity	Capacitance
Light	Current
Sound	Voltage
Magnetic field	Voltage
Gases	Resistance

From electrical parameters to processors

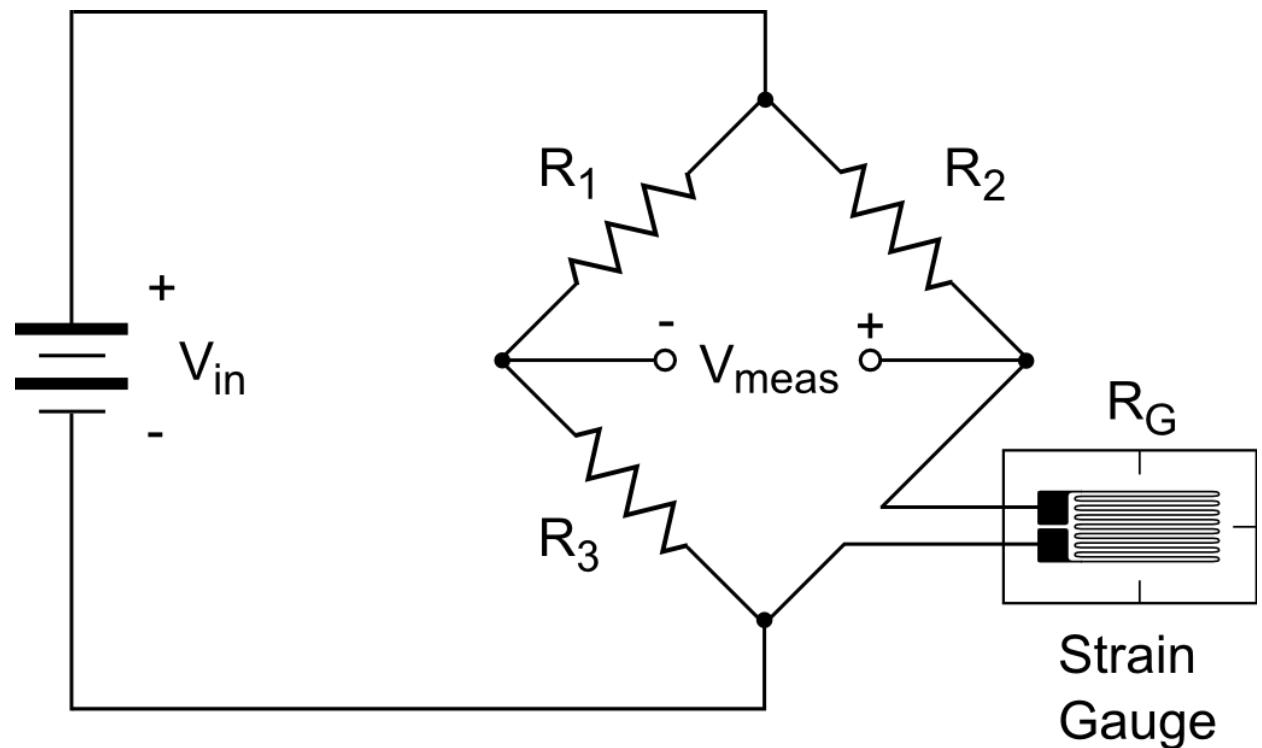
- The electrical parameter needs to be converted to voltage so that it can be read by the processor
- This requires:
 - Resistance to voltage convertor (RVC)
 - Capacitance to voltage convertor (CVC)
 - Frequency to voltage convertor (FVC)
- The voltage produced is then digitized for storage, processing and transmission

RVC example

POTENTIAL DIVIDER EQUATION: $V_{out} = \frac{R_2}{R_1 + R_2} V_{in}$



$$V_{meas} = V_{in} \frac{dR_G}{R}$$



Strain
Gauge

Sensor selection

- Range—Difference between the maximum and minimum value of the sensed parameter
- Resolution—The smallest change the sensor can differentiate
- Accuracy—Difference between the measured value and the true value
- Precision—Ability to reproduce repeatedly with a given accuracy
- Sensitivity—Ratio of change in output to a unit change of the input
- Zero offset—A nonzero value output for no input

Sensor selection

- Non Linearity—Percentage of deviation from the best-fit linear calibration curve
- Zero Drift—The departure of output from zero value over a period of time for no input
- Response time—The time lag between the input and output
- Operating temperature—The range in which the sensor performs as specified
- Deadband—The range of input for which there is no output
- Signal-to-noise ratio—Ratio between the magnitudes of the signal and the noise at the output

The calibration trap!

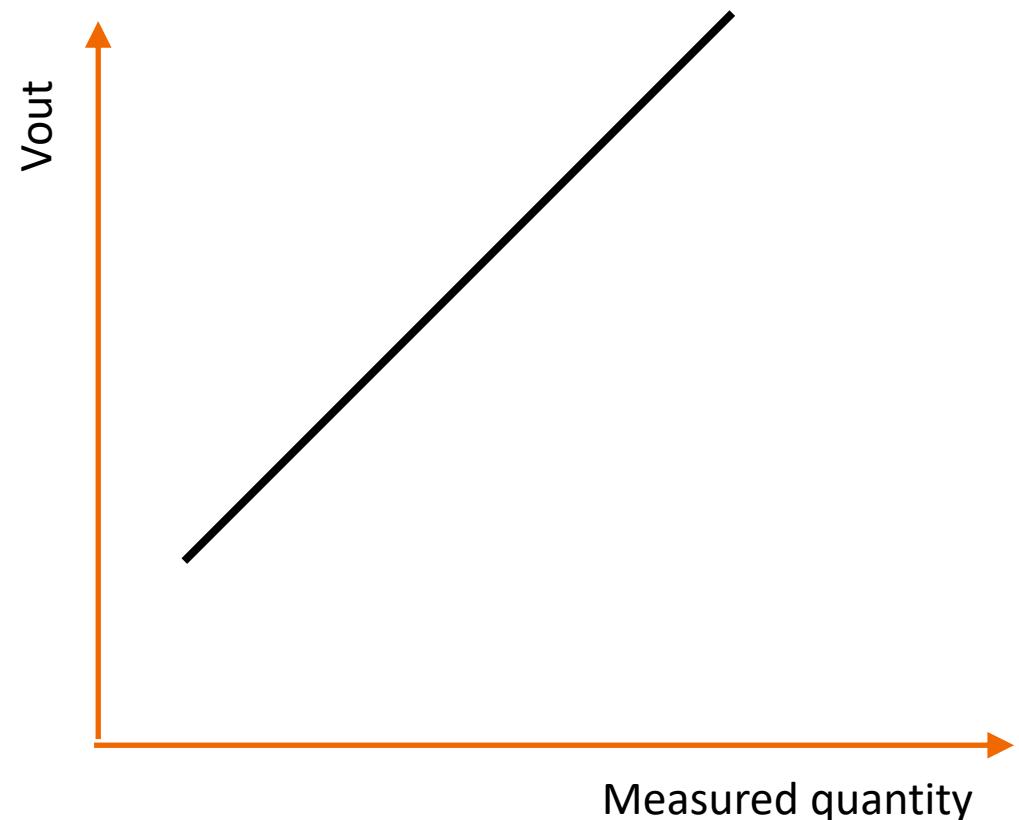
- All this is simple enough, however, the biggest hurdle in sensor deployment is calibration
- This ensures that the readings provided by the sensor are accurate and believable
- Particularly vital for medical applications
- Calibration is the comparison of measurement values delivered by a device under test (DUT) with those of a calibration standard



Sensor characteristics

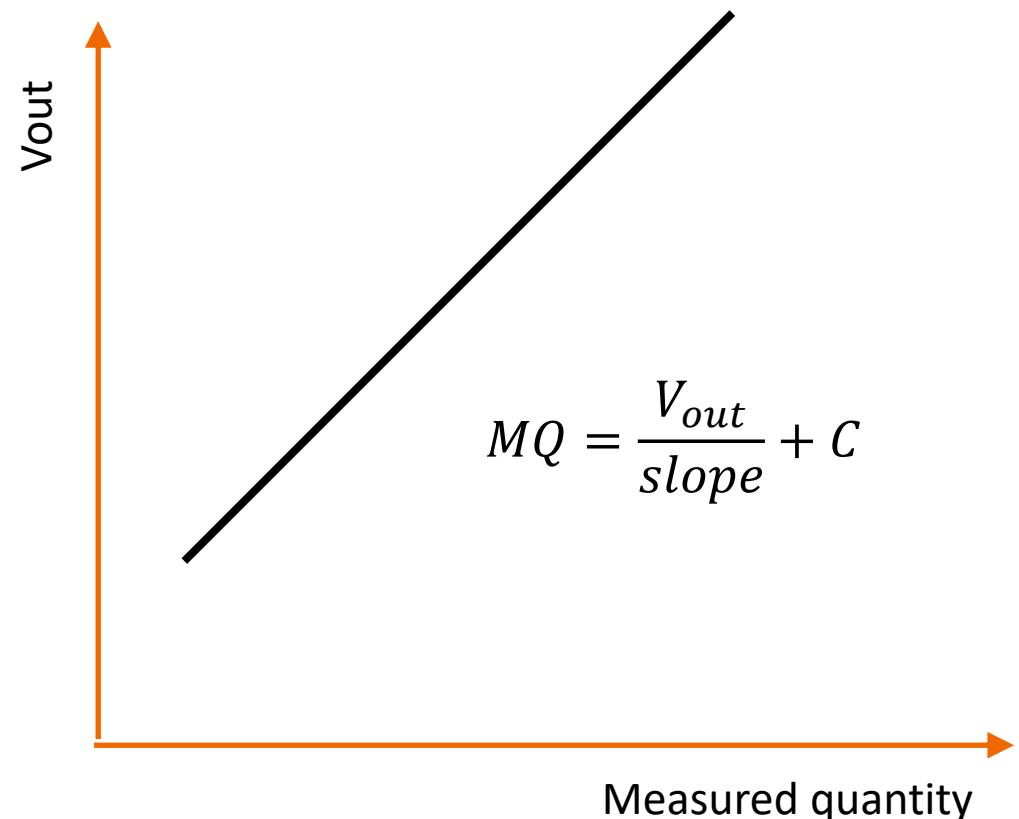
- Our eventual goal is to find out the value of the measured quantity based on the reading of the voltage
- For this, we need a function for MQ in the form of V_{out}
- This is typically the simplest for linear characteristics – that is why they are generally preferred

$$MQ = \frac{V_{out}}{\text{slope}} + C$$



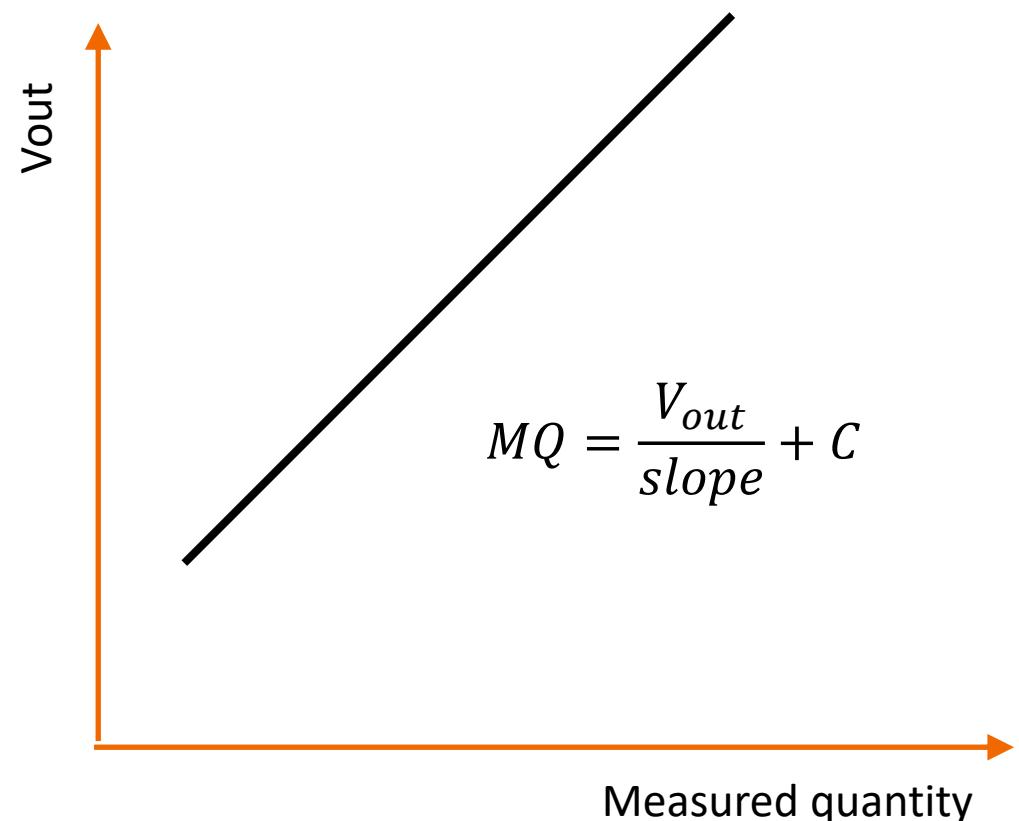
Sensor calibration

- We have two unknown quantities in the equation, once known, we can calculate the MQ for any V_{out}
- These are typically provided by the sensor manufacturer, however, the biggest problem in IoT design is shift in these values over time



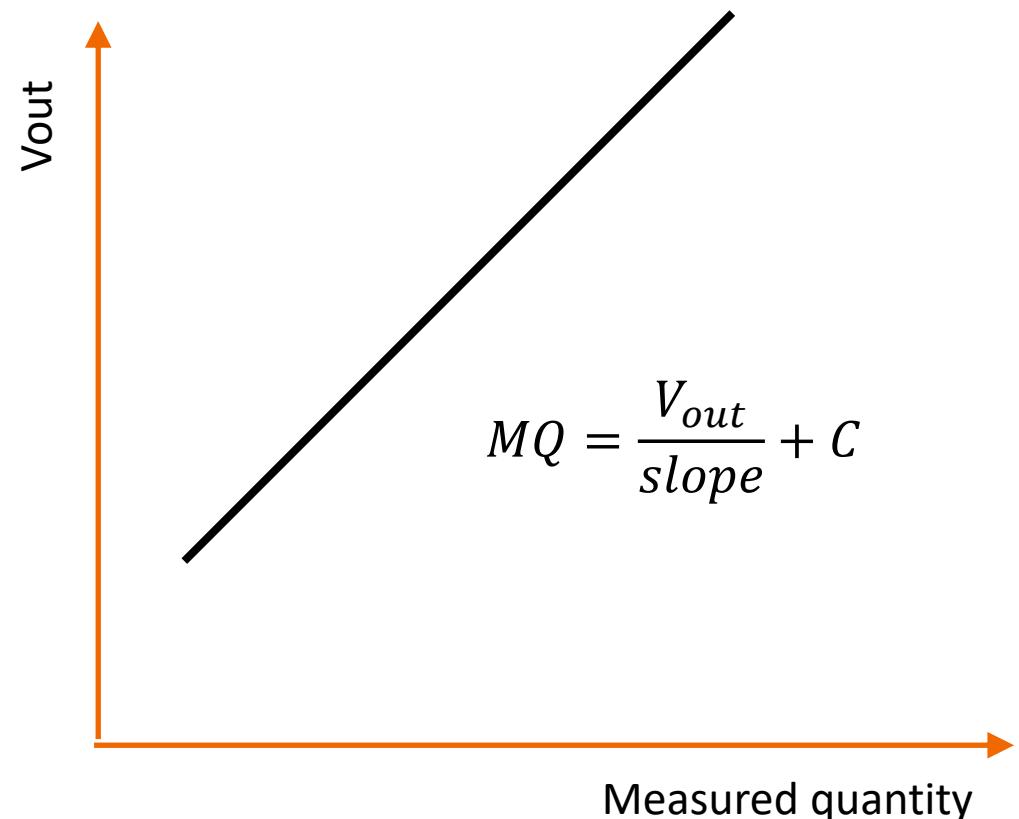
Sensor calibration

- Calibrating a sensor means determining these values, or if they are known, verifying them
- This can be done by subjecting the sensor to a known stimulant and measuring the readout voltage
- Two points are sufficient to determine the two unknown values
- However, the problem is to provide a *known stimulation* to the sensor



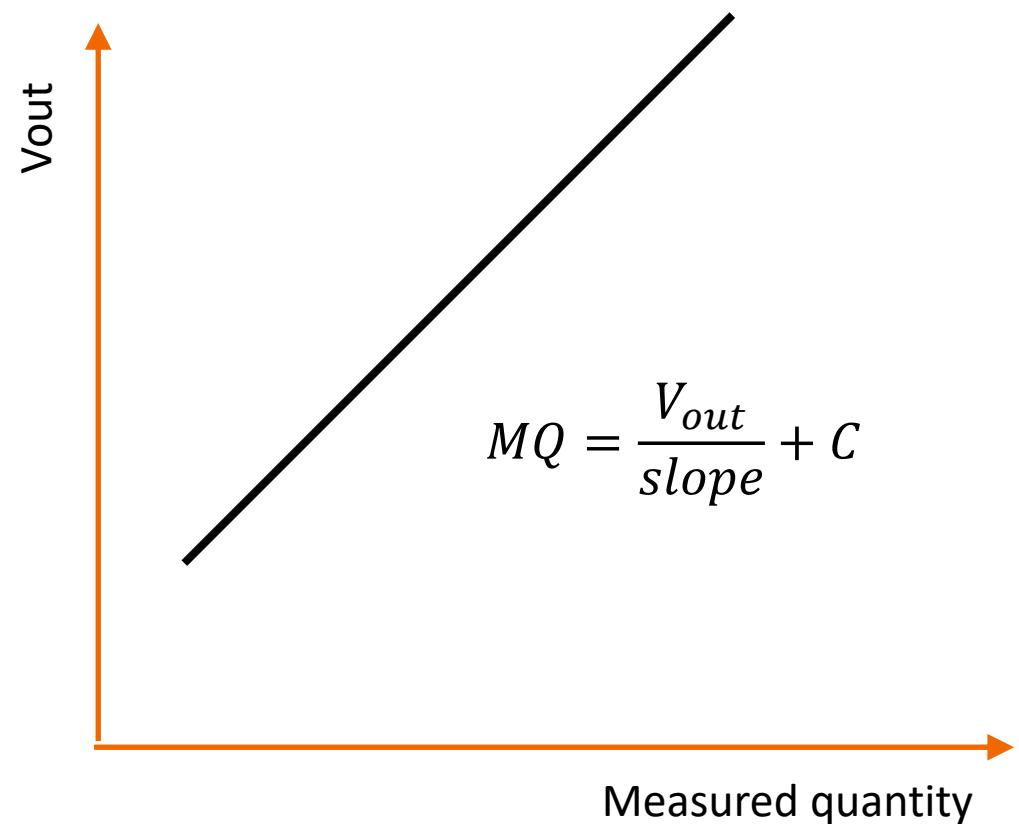
Sensor calibration

- One easy thing to do is to obtain the sensor output at zero applied input
- This is easily achieved for strain, force, light or acceleration sensors, but now so much for gas, temperature and pressure
- This zero-point provides the value of C, which can be itself be zero for no DC bias situation



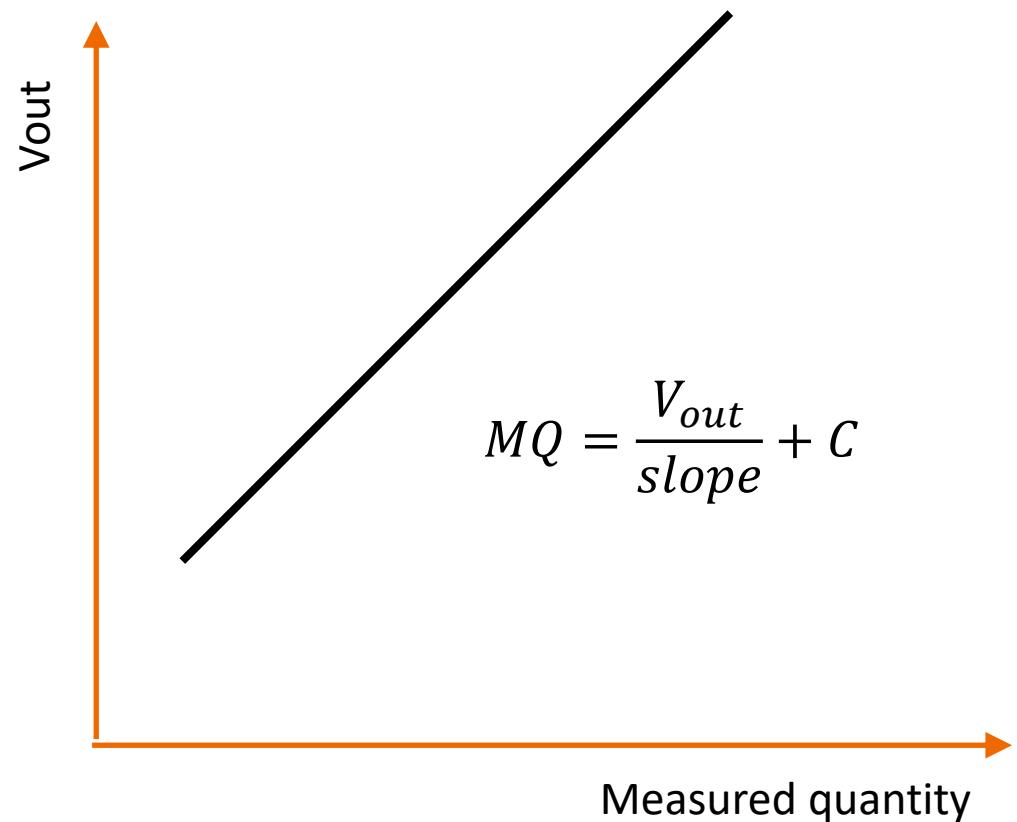
Sensor calibration

- For non-zero input, we require specific conditions that can be difficult to achieve for some sensors
- Sometimes, physics based calibration can be done – say phase change temperatures for temperature sensor
- In their absence, the best way forward is typically having a gold standard measurement system to determine the value of the quantity in the ambient
- Determining this gold standard is a major challenge



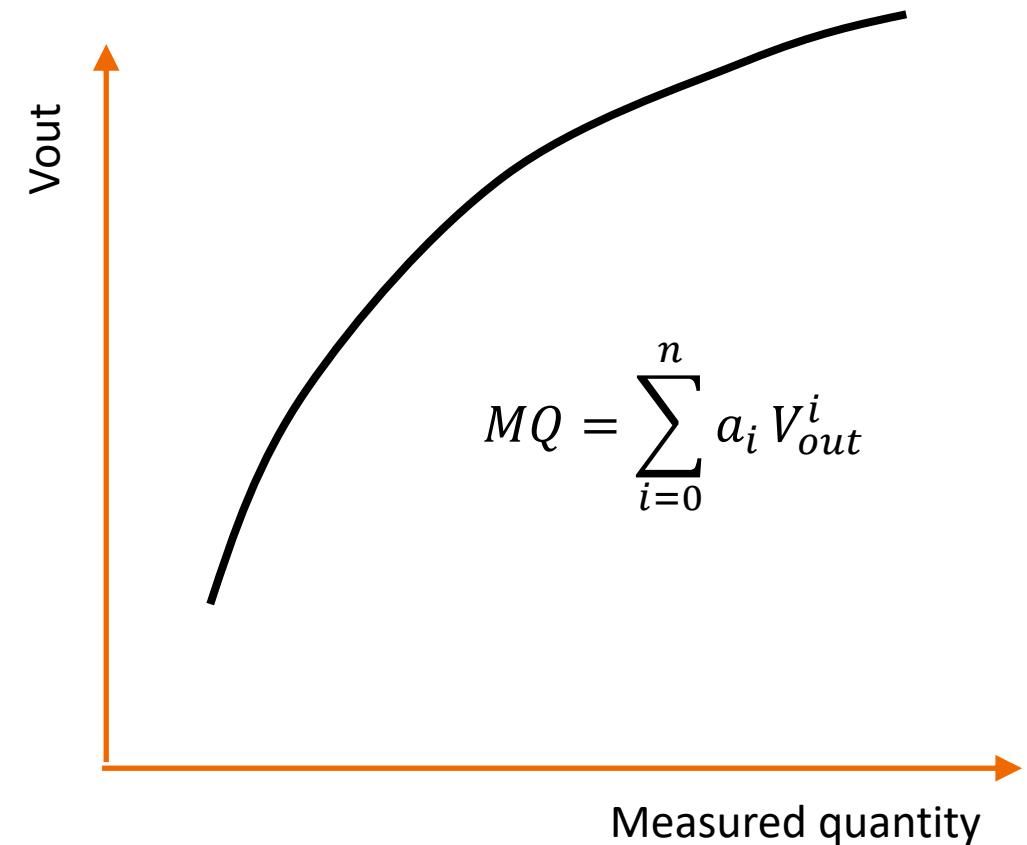
Sensor calibration

- Another major challenge in IoT deployments is that sensor calibration often drifts with time, i.e., the values of slope and C are functions of time
- There are several strategies:
 - Know the drift in advance and program it into the system logic
 - Recalibrate the sensor over-the-air based on a gold standard
 - Recalibrate by bring the sensor to a conditioned ambient



Non-linear systems

- No matter how much we try, there are always some systems with non-linear response
- We require multiple parameters to determine the calibration in this case
- New strategies:
 - Use of ML algorithms to determine sensed quantity



<https://ieeexplore.ieee.org/abstract/document/9335600>

Actuators

Dr. Aftab M. Hussain,
IIIT Hyderabad

Actuators

- Actuators are devices that take signal in electrical form and transform it into something that can influence the physical world
- We can almost say that this is the end goal of all IoT devices, i.e., influencing or altering the physical world in ways that enhance our safety/ease-of-living
- We can obtain actuation in many forms:
 - Movement
 - Temperature (heating/cooling)
 - Light
 - Sound
- These transducers are typically accompanied with their drive circuits (like RVC, CVC for sensors)

Actuators

Actuator	Physical principal
Motors	Electromagnetism
LED light	Electron-hole pair recombination
Incandescent light	Black body radiation
Electrical heaters	Joule heating
Speakers	Piezoelectricity/electromagnetism
Cooling	Peltier effect, adiabatic expansion

Actuator selection

- Continuous power output—The maximum force/torque attainable continuously
- Range—The range of linear/rotary motion/temperature/light intensity achievable
- Resolution—The minimum increment of output attainable
- Accuracy—Linearity of the relationship between the input and output
- Speed characteristics—output versus speed relationship
- No load operation—Typical operating speed/velocity with no external load
- Power requirement—Type of power (AC or DC), number of phases, voltage level, and current capacity

Actuator calibration

- Similar to sensor calibration
- Given a particular input, what output does the actuator produce?
- Is it on expected lines?
- Can be difficult to measure reliably depending on the type of actuator

Thank you

Dr. Aftab M. Hussain,
IIIT Hyderabad

Lecture 3 – Actuator drives and Sensor outputs

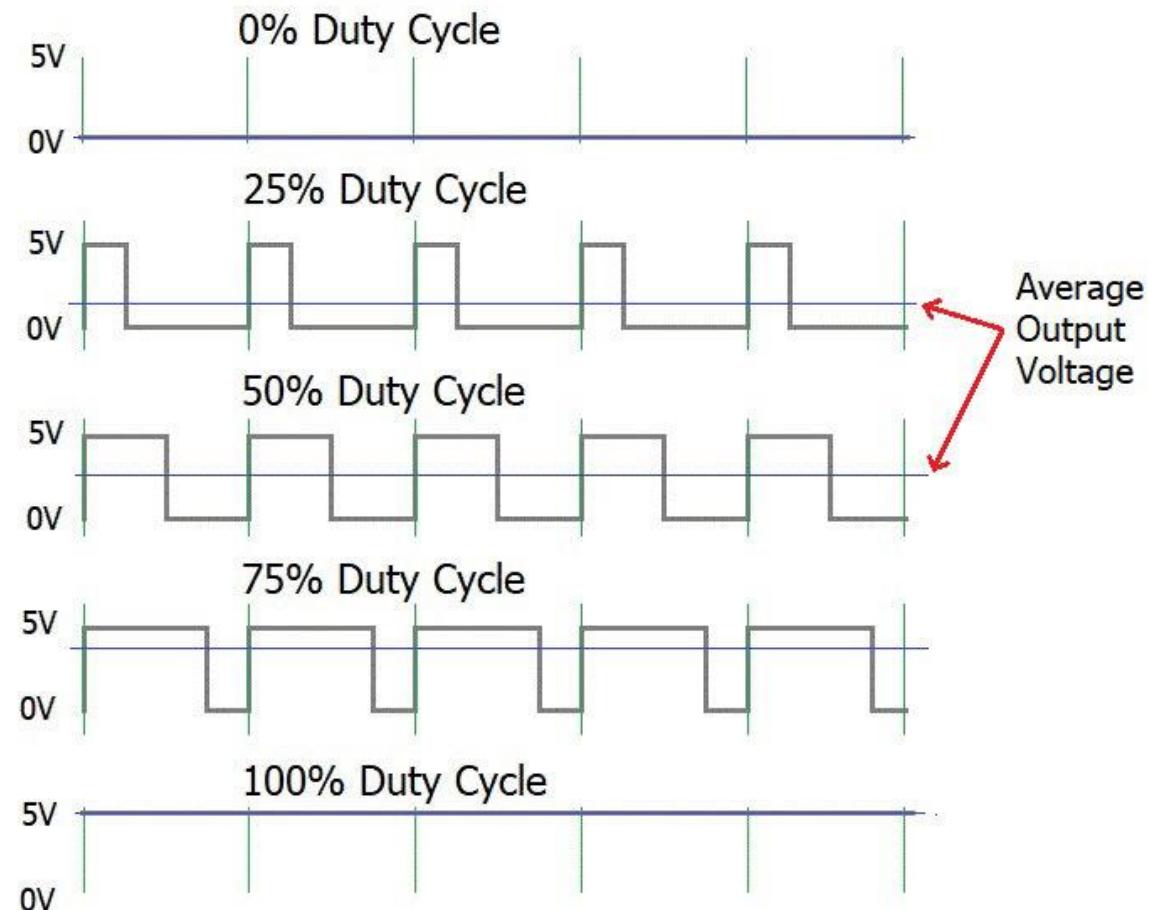
Dr. Aftab M. Hussain,
Assistant Professor, PATRIoT Lab, CVEST

Actuator drive

- There are many types of actuators:
 - Mechanical (electromagnetic drives)
 - Thermal (heating coil)
 - Optical (LED)
 - Acoustic (speaker)
- Most of these require an analog input voltage to control the intensity of actuation
- This analog signal needs to be obtained using an analog output pin (with a built-in DAC) or an external DAC

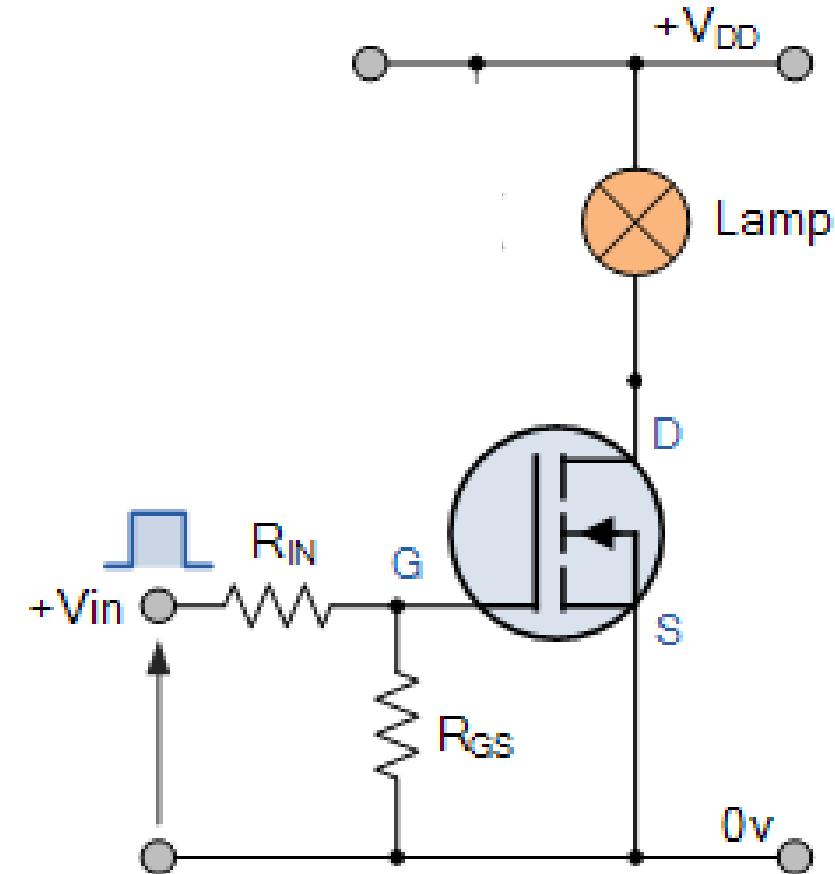
Actuator drive - PWM

- A very common way of skipping the DAC and getting “analog looking” voltage from digital output pins is the PWM method
- The output voltage is cycled from logic-0 to 1 at a very high rate with a particular duty cycle at a particular frequency
- This gives the illusion of the output being analog, say if this output is connected to an LED, the output intensity is perceived as the average of the PWM over time
- Arduino has no DAC, only PWM!



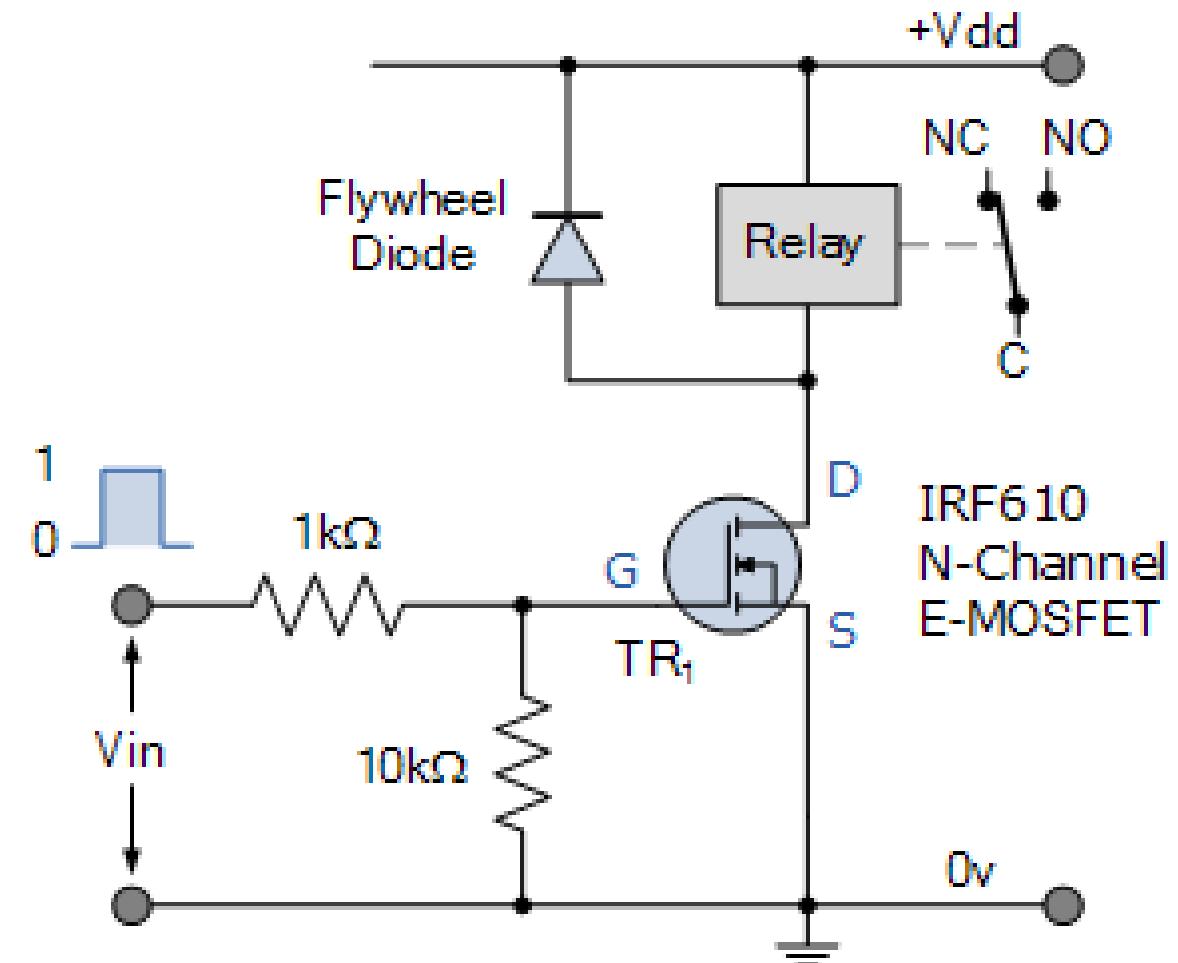
Actuator drive – loading effect

- A common problem with controlling certain peripherals with controllers is the loading effect
- If an actuator is to be driven with controller output, the current supplied by the controller output pin should be sufficient to drive it
- If it is not, the best way is to connect the actuator to a sufficient power source through a switch (transistor) and drive the using the controller output



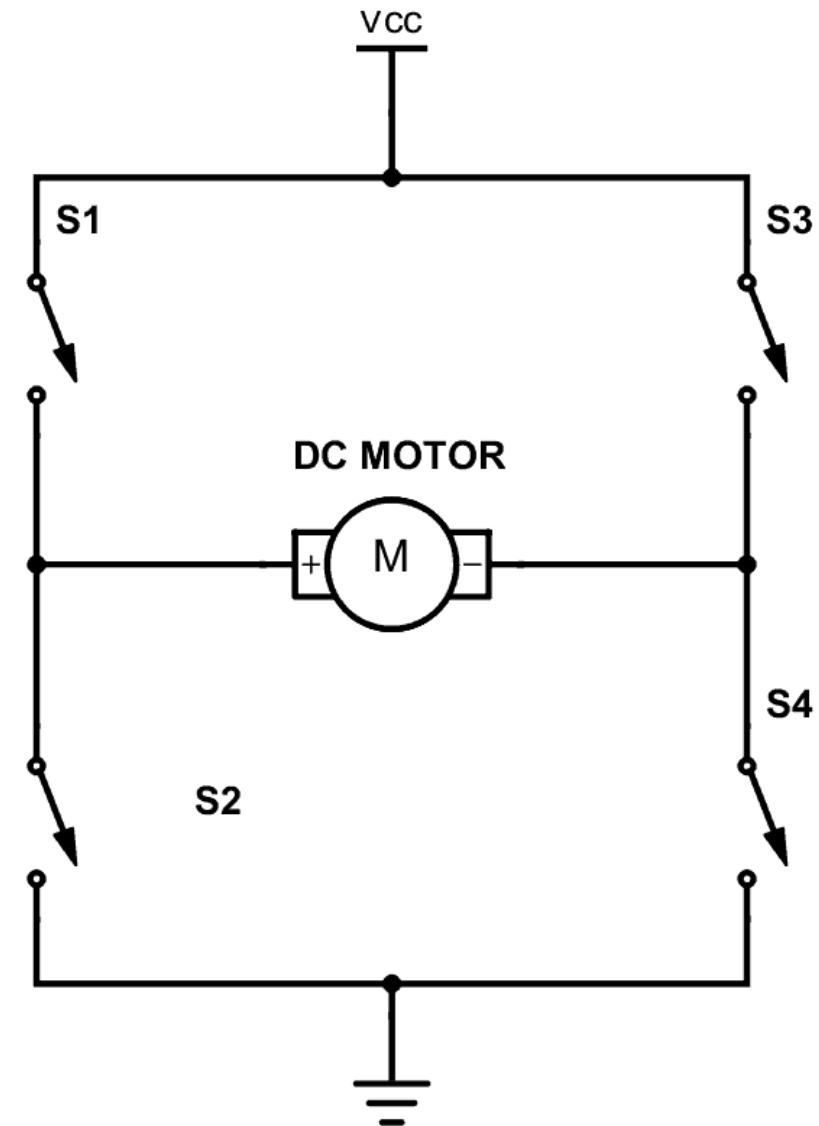
Actuator drive – loading loading effect

- In case of very large loads, a single stage of transistor switching may not be enough
- A common example is driving an AC load (say a light bulb) using a controller
- In this case, a relay is used as a switch to drive the power side using signal from the control side
- However, the power needed to switch the relay is often times more than the controller can provide, so we need a transistor circuit to power the relay



Actuator drive – H bridge

- A transistor switch is great to drive a motor forward, however, if the same supply is to be used to drive it in reverse, an H-bridge circuit is needed
- When S1 and S4 are closed and other two are open, the motor moves in forward direction
- When S2 and S3 are closed and other two are open, the motor moves in reverse
- The switches are generally realized using transistors
- When S1 and S3 or S2 and S4 are closed, the motor brakes (same voltage at the terminals)
- In other cases, the motor coasts

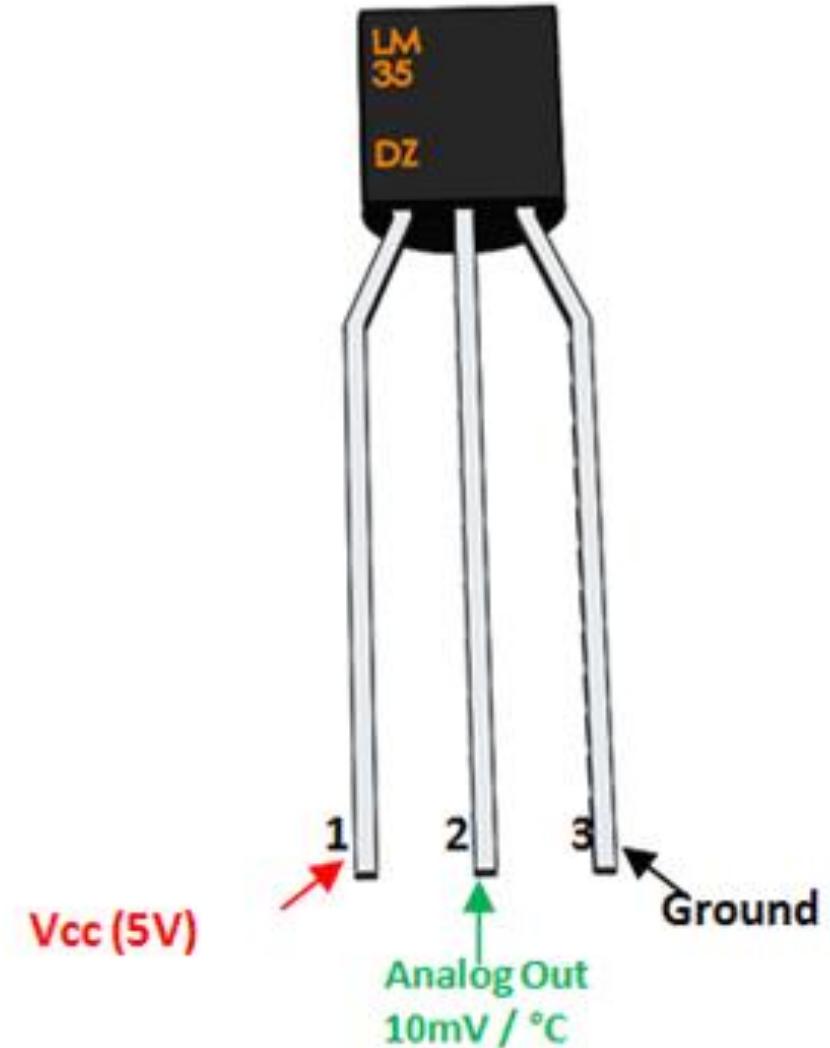


Sensors outputs

Dr. Aftab M. Hussain,
Assistant Professor, PATRIoT Lab, CVEST

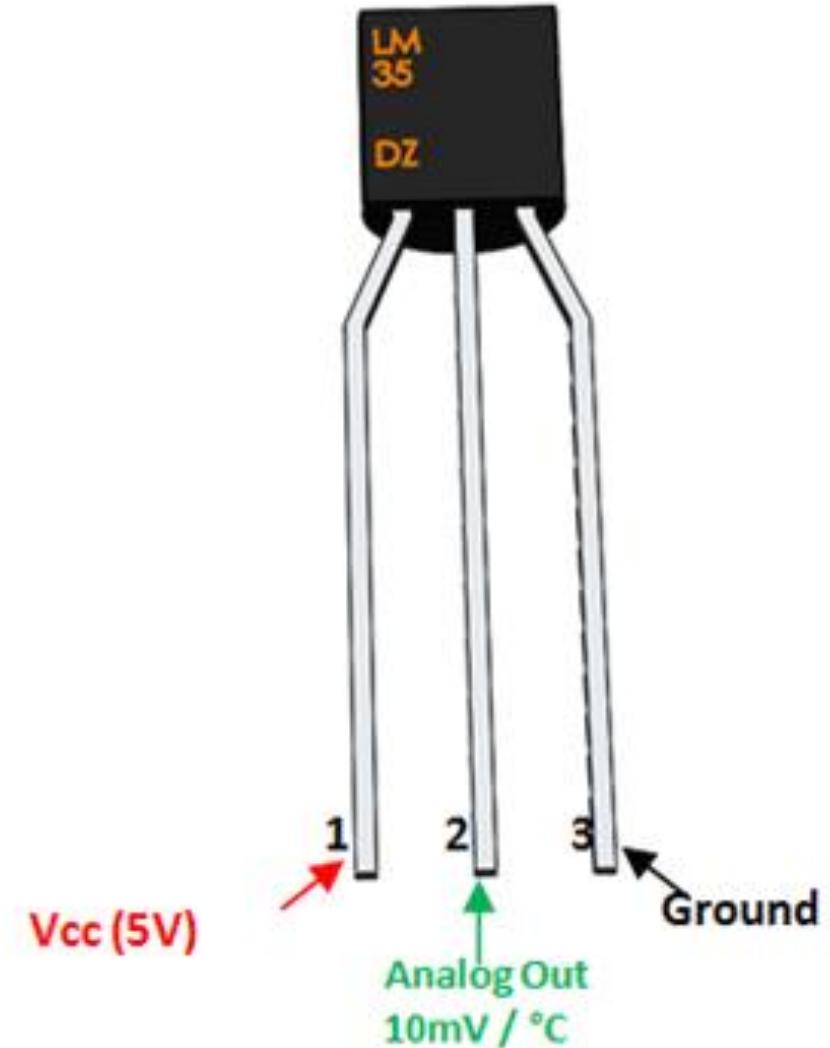
Sensor outputs – Analog

- Analog output is the simplest form of output for a sensor
- It has two pins – the analog output and the ground
- The analog output can be fed directly to some controllers that have a built-in analog to digital convertor (like Arduino), in other cases, an external ADC is required (like Raspberry pi)
- NodeMCU has one “analog read” pin



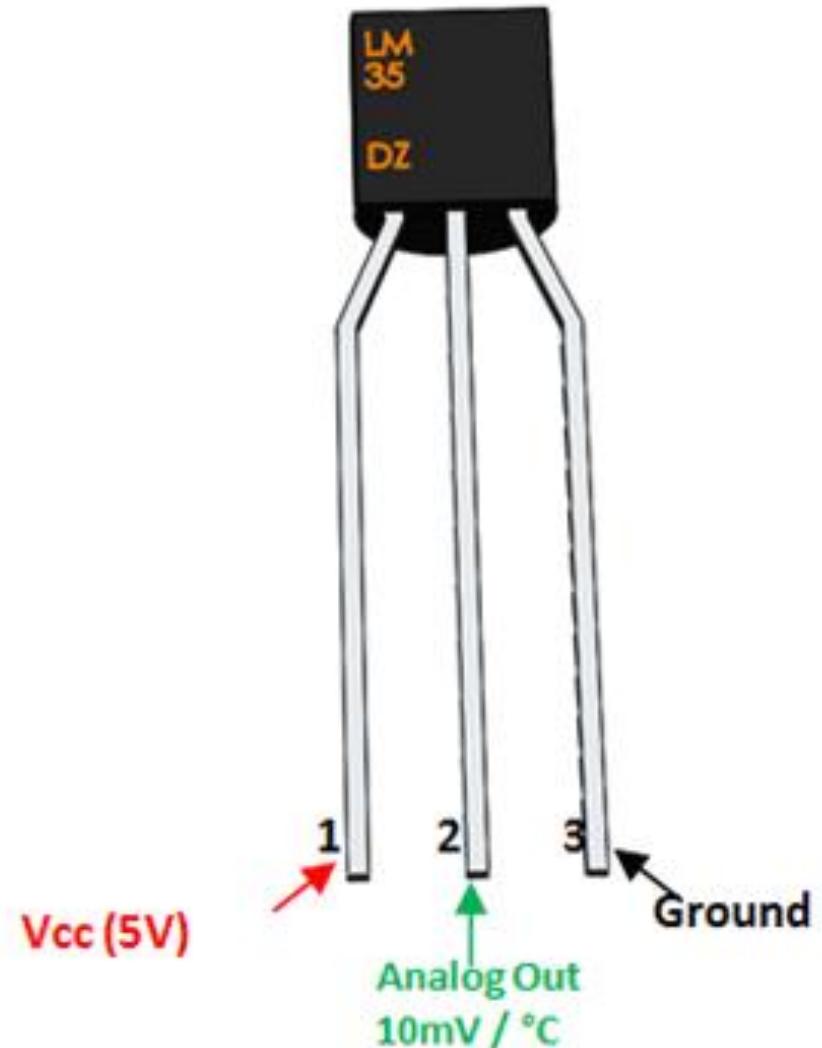
Sensor outputs – Analog

- Selecting an ADC can be tricky – because analog signals are continuous in time as well as amplitude, and their digital counterparts are discrete in both
- In time axis, the “discreteness” is measured by sampling rate (generally in ksps or Msps)
- In voltage axis, the “discreteness” is measured in the output bits



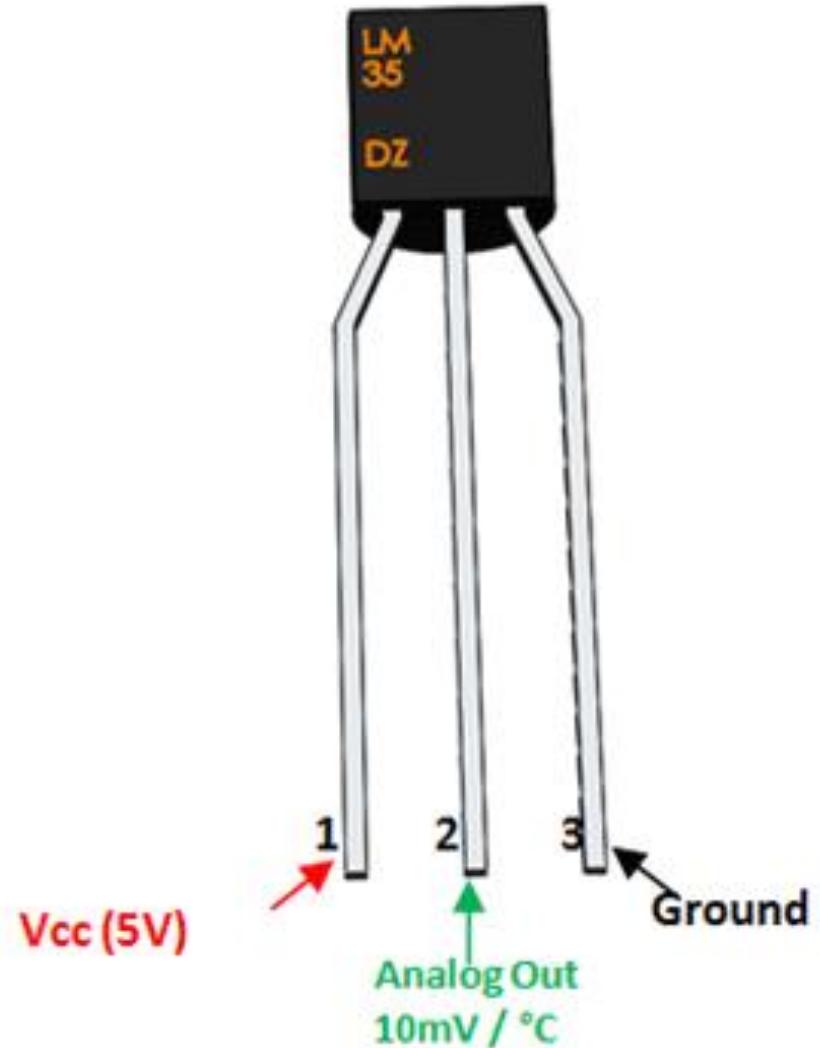
Sensor outputs – Analog

- Example: a 10-bit ADC can output a maximum of 1024 levels, so for a 5 V range, the difference in successive samples is $\sim 5 \text{ mV}$
- Clearly, more bits and higher sampling frequency is ideal
- However, these are competing goals because more bits take more time to process reducing the sampling rate



Sensor outputs – Analog

- Advantages:
 - Simple implementation with an onboard ADC (single wire)
 - Continuous output – so can be on demand
 - Infinite resolution
- Problems
 - Needs an ADC
 - Very susceptible to noise
 - The “loading” effect



Need of ADCs and DACs

- We know that sensor outputs (e.g., a voltage measured with a thermocouple or a speech signal recorded with a microphone) are analog quantities, varying continuously with time
- However most of the processing, storage and happens in digital format
- An ADC (Analog-to-Digital Converter) is used to convert an analog signal to the digital format
- The reverse conversion (from digital to analog) is also required (mostly for actuator operation)
- For example, music stored in a DVD in digital format must be converted to an analog voltage for playing out on a speaker
- A DAC (Digital-to-Analog Converter) is used to convert a digital signal to the analog format

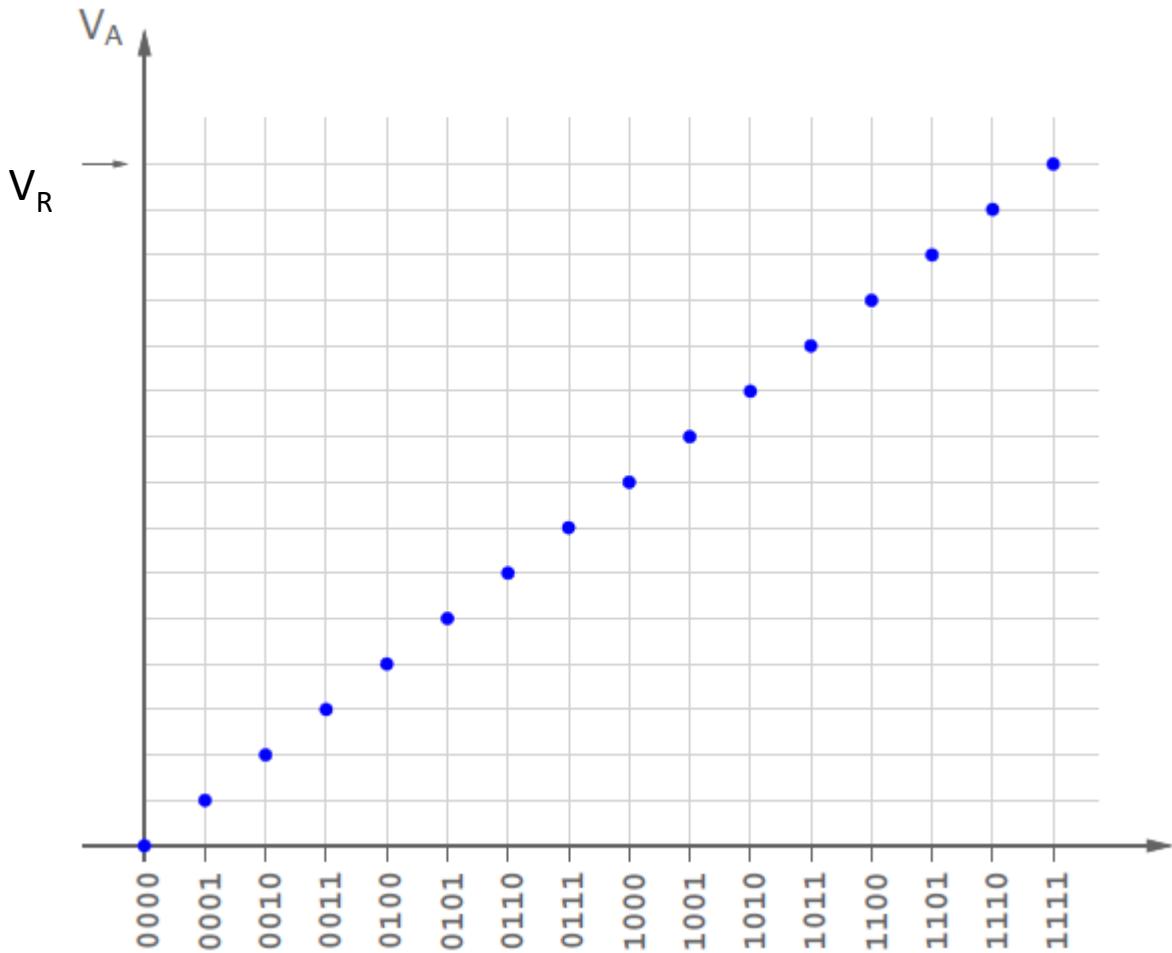
DAC

- A DAC is a circuit that takes digital signal and provides a corresponding analog output
- For a four bit DAC with input $A_3A_2A_1A_0$, the output voltage is:

$$V_A = \frac{V_R}{15} [(8A_3 + 4A_2 + 2A_1 + A_0)]$$

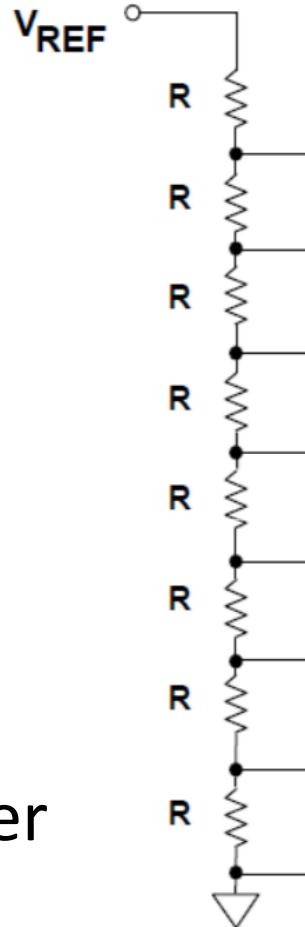
- In general, the output is:

$$V_A = \frac{V_R}{2^n - 1} \sum_{k=0}^{n-1} A_k 2^k$$



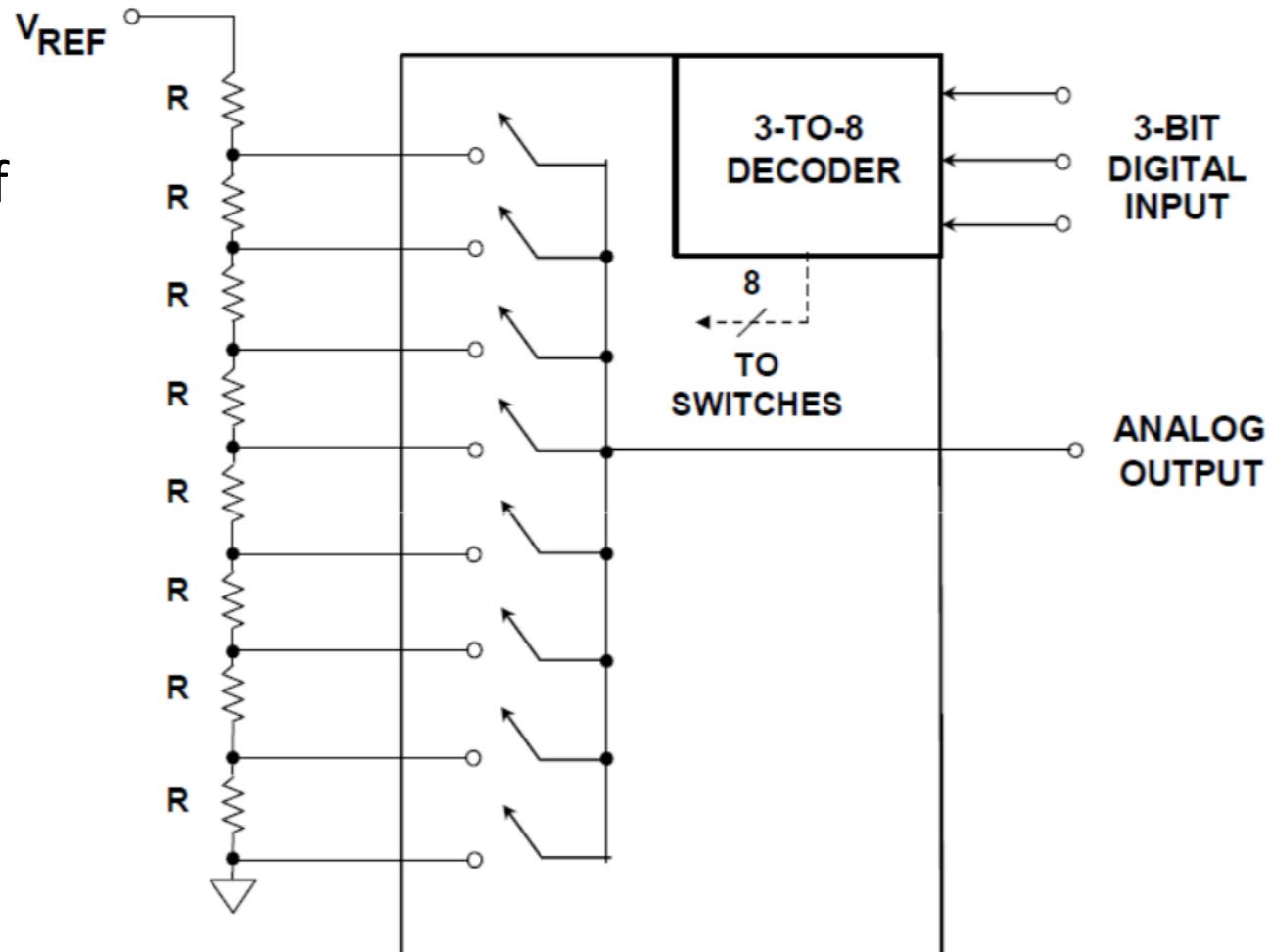
DAC – Kelvin divider

- The most intuitive way of implementing a DAC is using the Kelvin divider
- In this case, we use 2^n resistors of equal values in series with the reference voltage
- The voltage is thus divided into 2^n equal intervals
- The output is then tapped from an appropriate position using a decoder circuit in conjunction with 2^n switches



DAC – Kelvin- divider

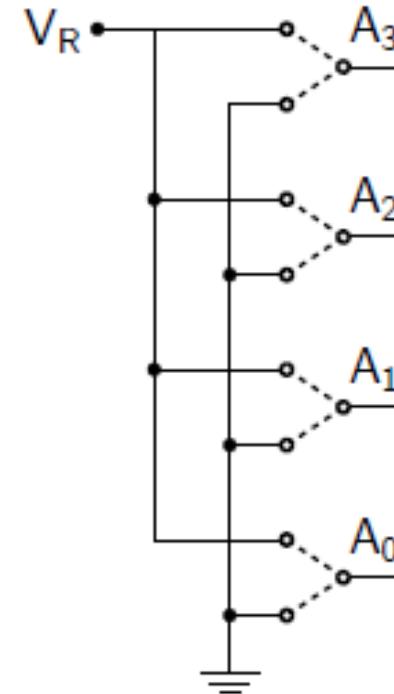
- Advantages:
 - Simple design
 - It is configurable to non-linear DACs if specific step size is required for particular applications
- Disadvantages
 - 2^n switches and resistors can explode exponentially
 - Large power consumption
 - Decoding circuit can become complicated for large values of n



DAC – Weighted resistors

- We can connect the input signals to switches such that the V_R appears at a terminal if the input is high
- Such a connection can be used in conjunction with resistors of specific value to obtain the analog voltage
- If the input bit A_k is 1, the terminal gets connected to V_R ; else, it gets connected to ground
- Thus,

$$I_k = \frac{A_k V_R - 0}{R_k} = \frac{A_k V_R}{R_k}$$



DAC – Weighted resistors

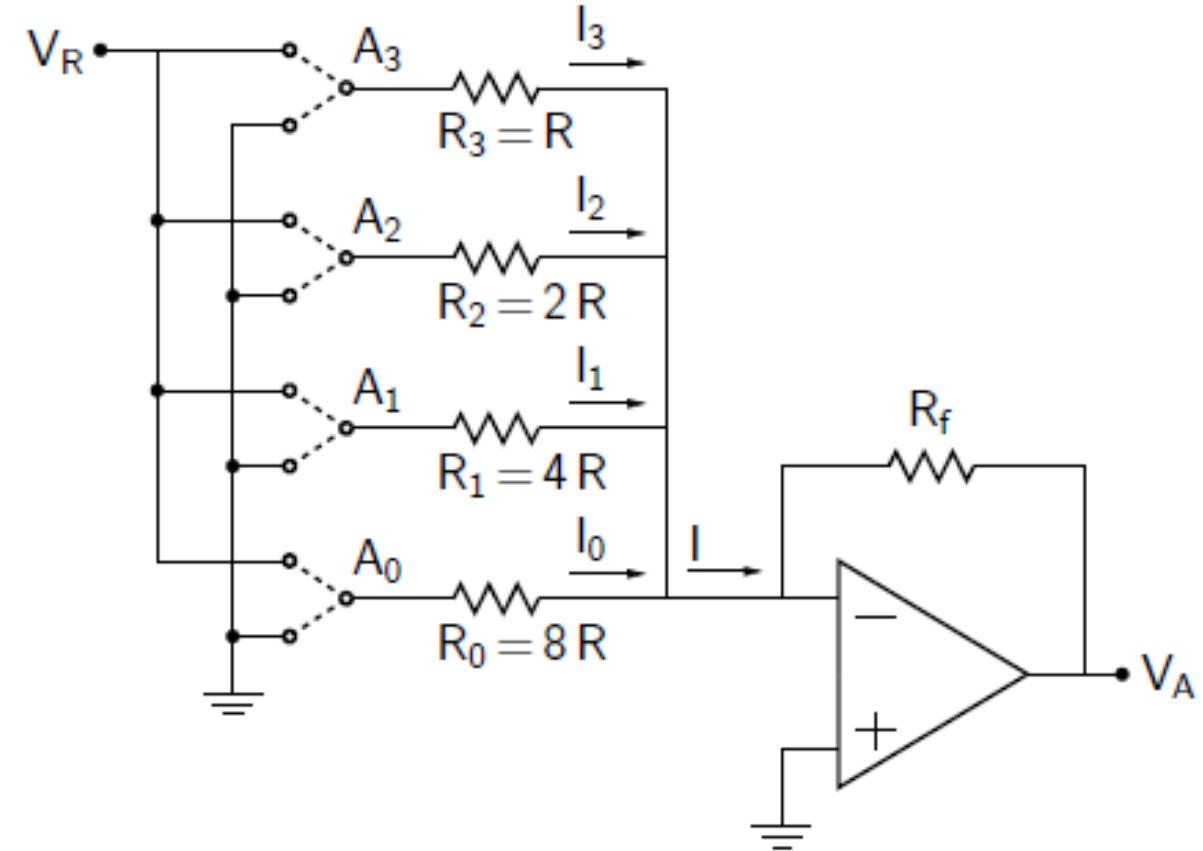
- From the non-inverting opamp, we have:

$$V_A = -R_f I = -R_f \sum_{k=0}^{n-1} \frac{A_k V_R}{R_k}$$

- If we use $R_k = \frac{2^{n-1} R}{2^k}$:

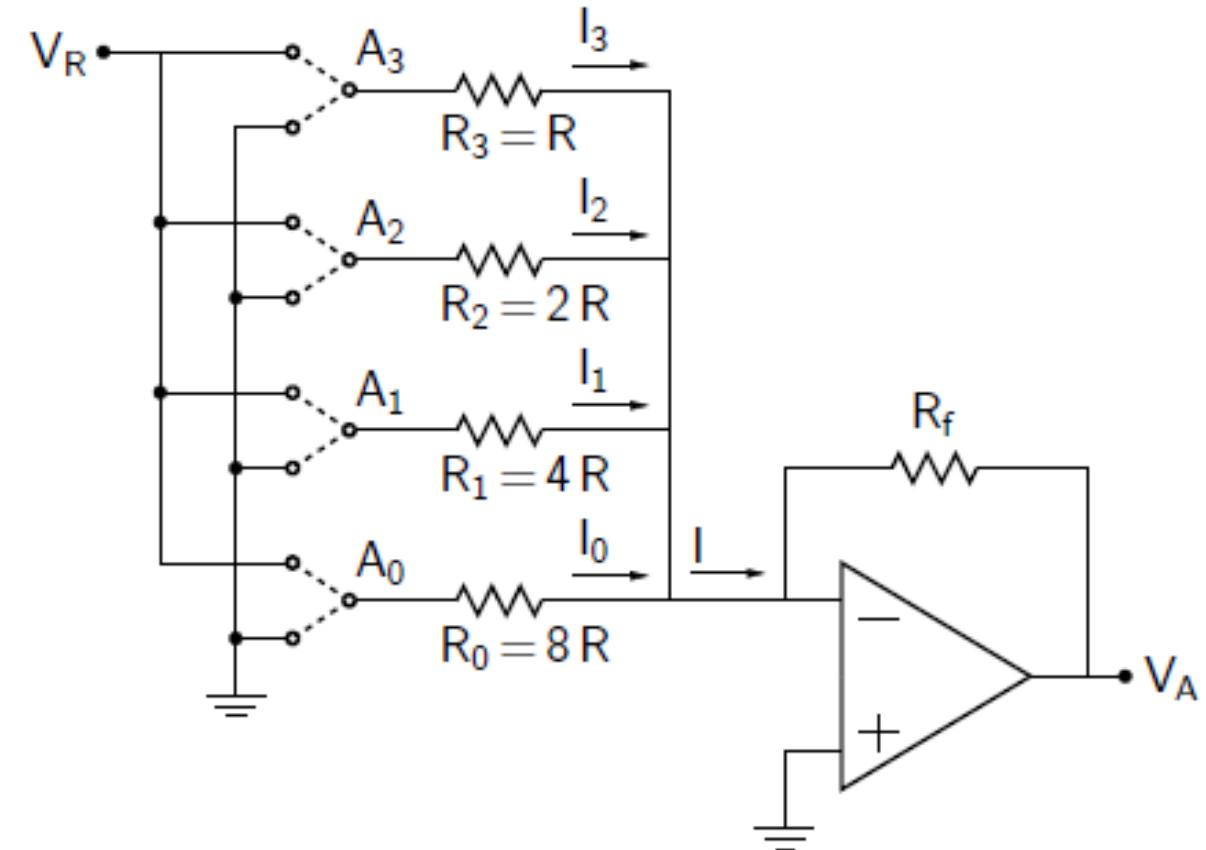
$$V_A = -R_f I = -\left(\frac{R_f}{R}\right) \frac{V_R}{2^{n-1}} \sum_{k=0}^{n-1} A_k 2^k$$

- Thus, with only n resistors and n switches, the circuit can be made
- R_f can be used to introduce additional gain



DAC – Weighted resistors

- Advantages:
 - n resistors and $2n$ switches
 - Current is only drawn when input is applied, no static power consumption
 - The R_f value can be varied to obtain a gain along with D to A conversion
- Problems:
 - Hard to find resistors of exact values such as $R, 2R \dots 16R, 32R$ etc.



DAC – R-2R ladder

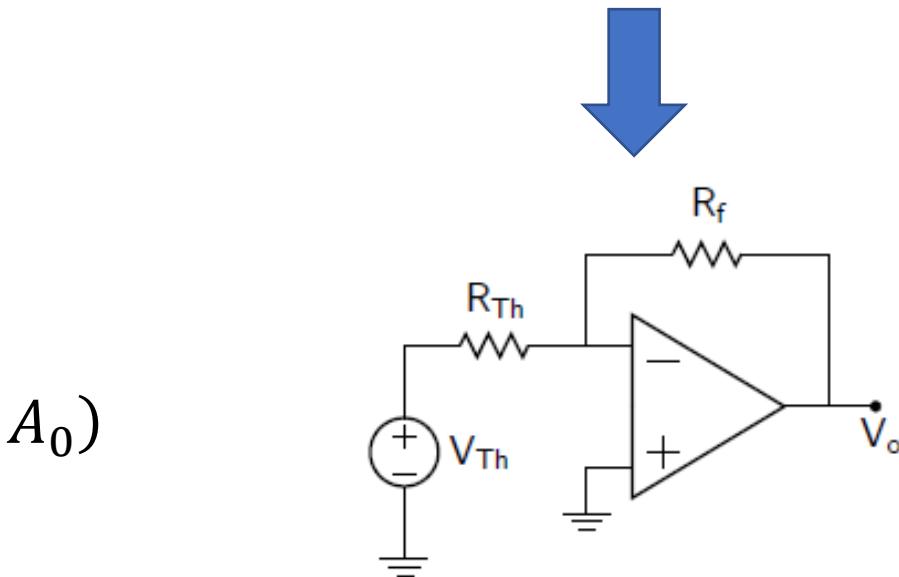
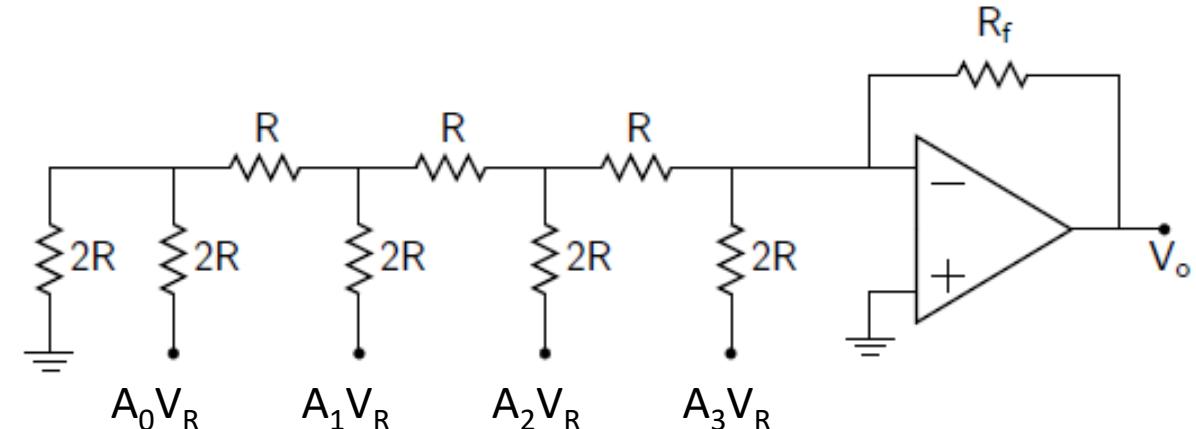
- The R-2R ladder circuit consists of the input voltage applied to a resistance network made of resistors of values R and $2R$ as shown
- The output can be thought of as an Thevenin equivalent circuit with,

$$R_{th} = R$$

$$V_{th} = \frac{V_R}{16} (A_3 2^3 + A_2 2^2 + A_1 2^1 + A_0)$$

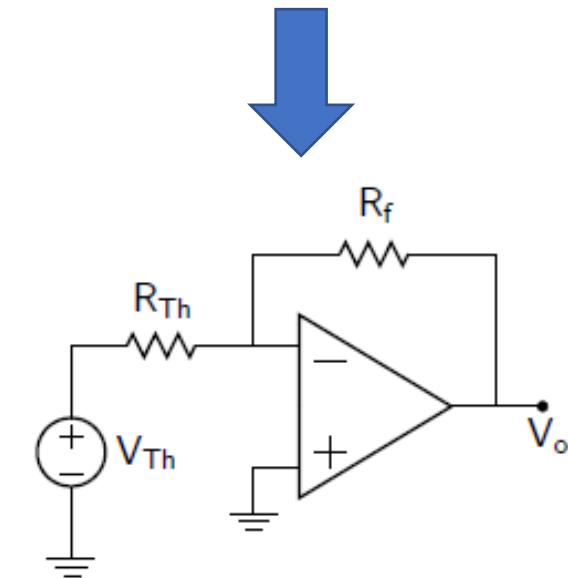
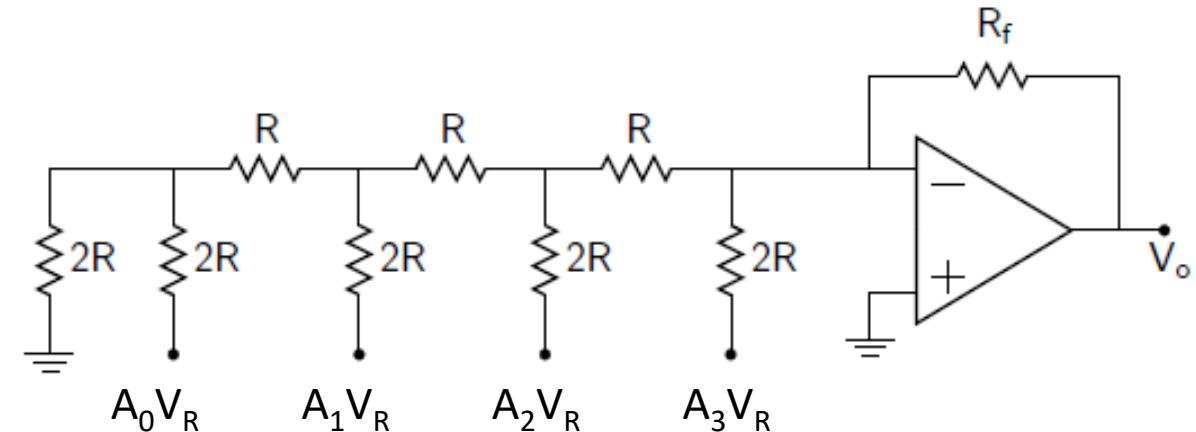
- Thus,

$$V_o = - \left(\frac{R_f}{R} \right) \frac{V_R}{16} (A_3 2^3 + A_2 2^2 + A_1 2^1 + A_0)$$



DAC – R-2R ladder

- Advantages:
 - Does not require resistors of specific values
 - Can be made using only $2n$ resistors and $2n$ switches
 - No static power consumption
- Due to its many advantages, R-2R ladder is often used in fabricating commercial DACs

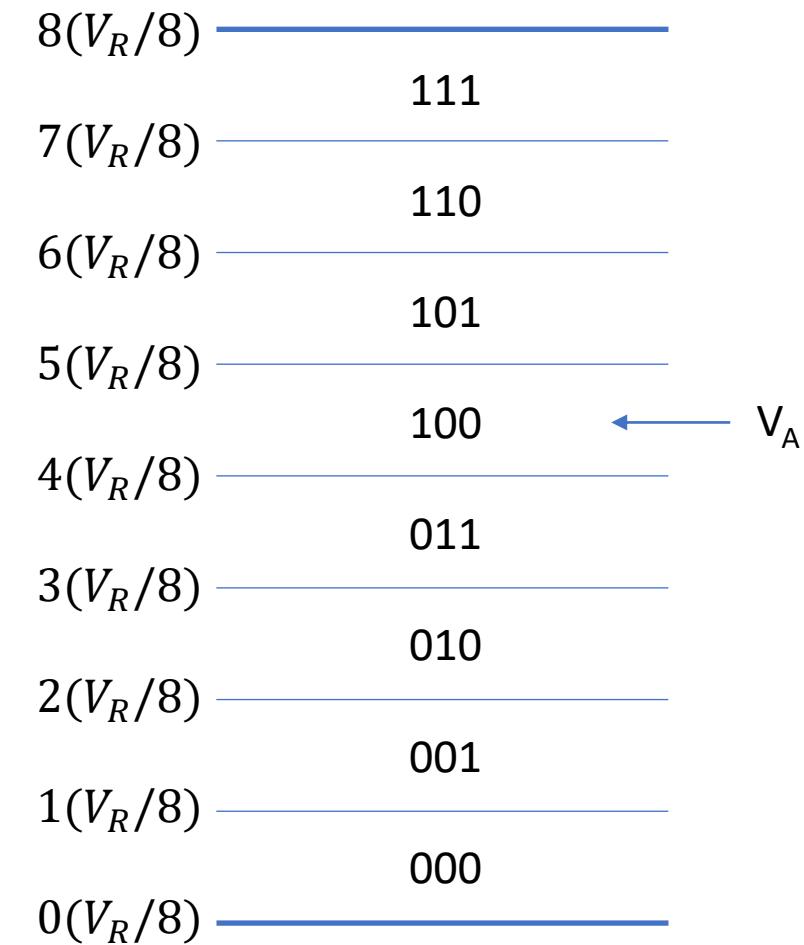


Lecture 4 – ADC and Digital signals

Dr. Aftab M. Hussain,
Assistant Professor, PATRIoT Lab, CVEST

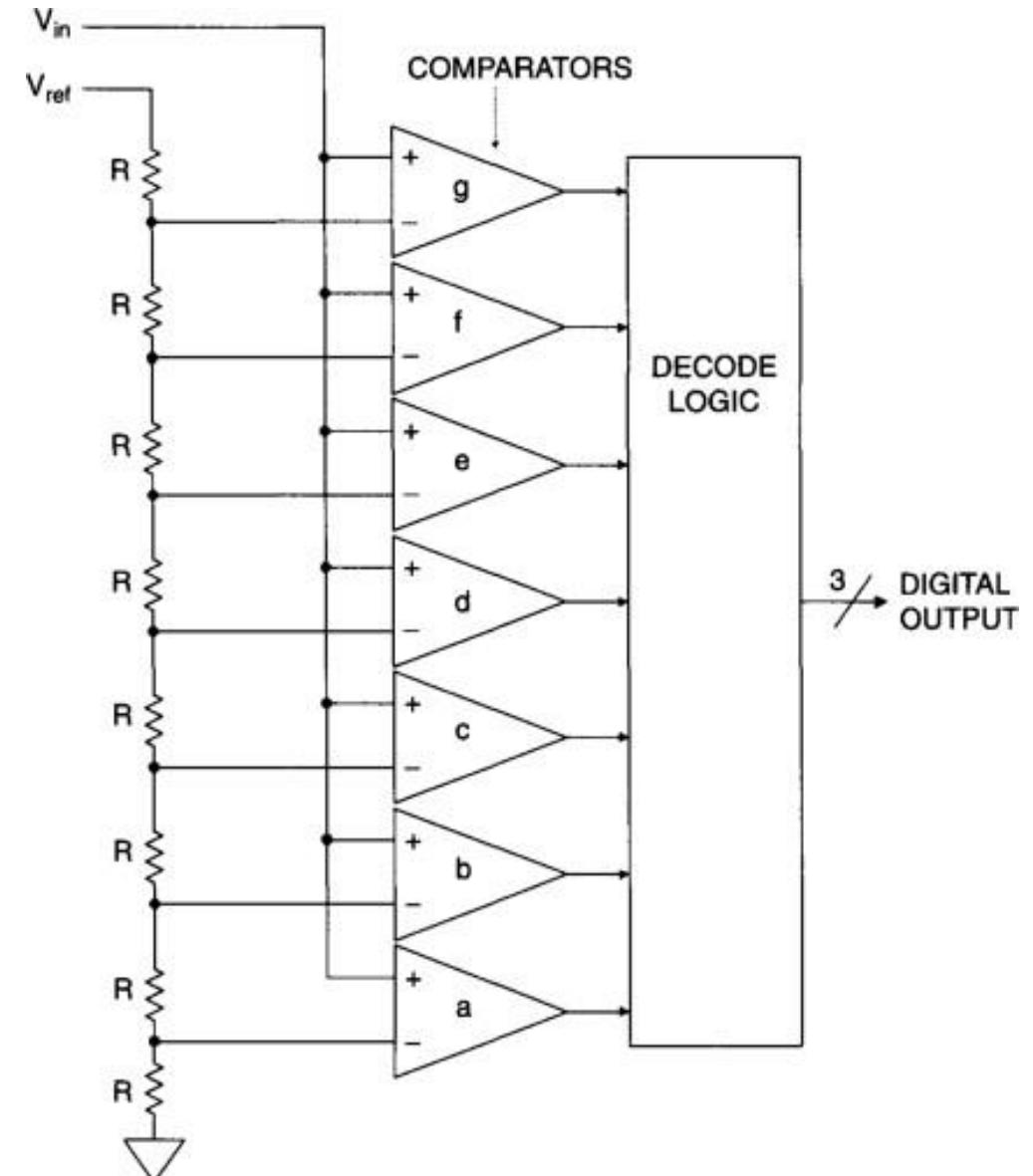
ADCs

- To convert from analog to digital, we may think of dividing the reference voltage by 2^n and consider each voltage interval (corresponding to 000, 001, etc.) as a bin
- If the input voltage V_A falls in the 100 bin; therefore, the output of the ADC would be 100
- Thus, the basic idea behind an ADC is simple:
 - Generate reference voltages V_1, V_2, \dots , etc.
 - Compare the input V_A with each of V_i to figure out which bin it belongs to
 - If V_A belongs to bin k , convert k to the binary format



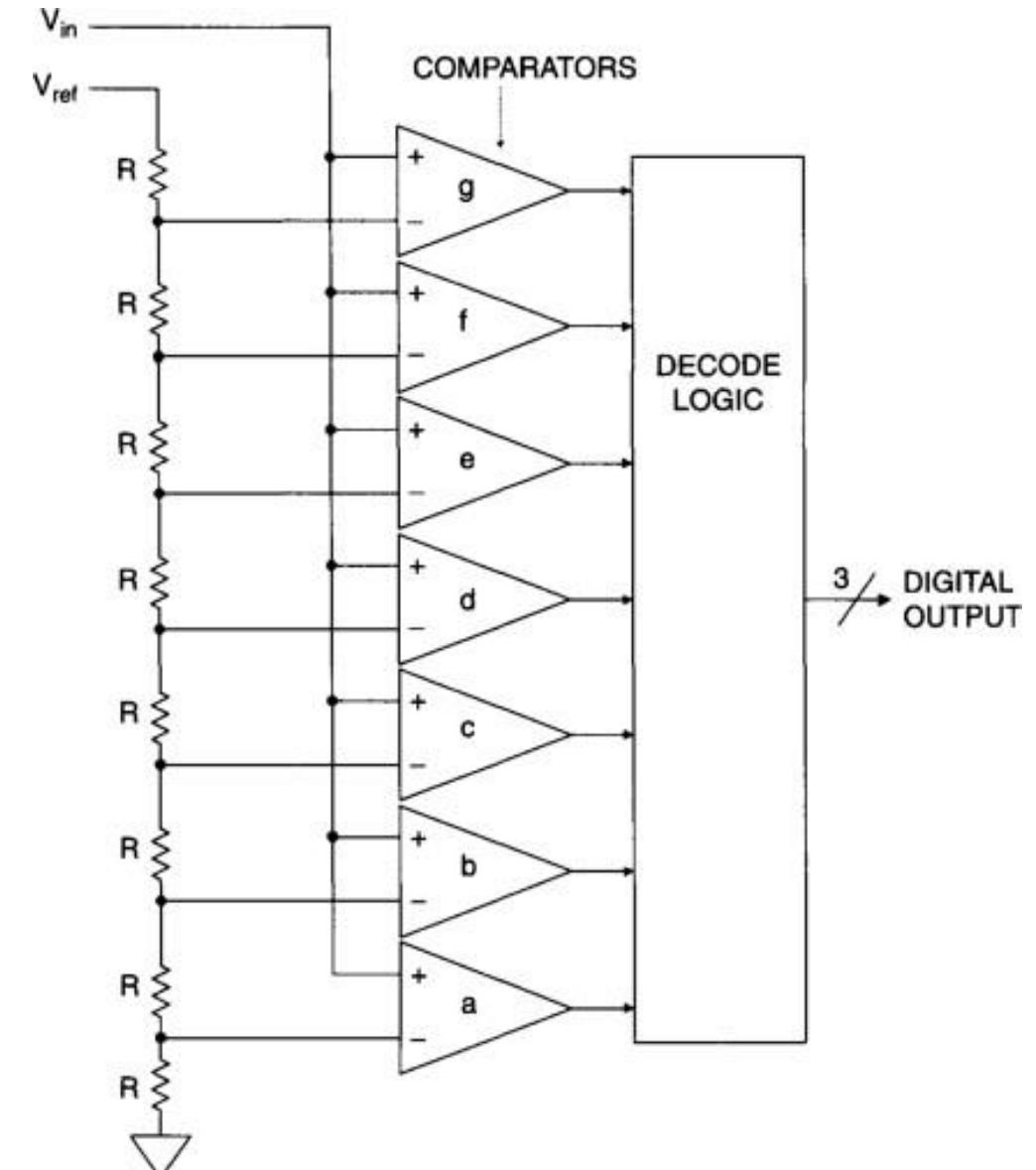
ADC – parallel/flash

- In case of the parallel ADC, the input voltage is compared with the V_{ref} divided into bins using a voltage divider
- The output of the comparators depends on the level of the input voltage with respect to these bins
- This output is decoded into the digital output



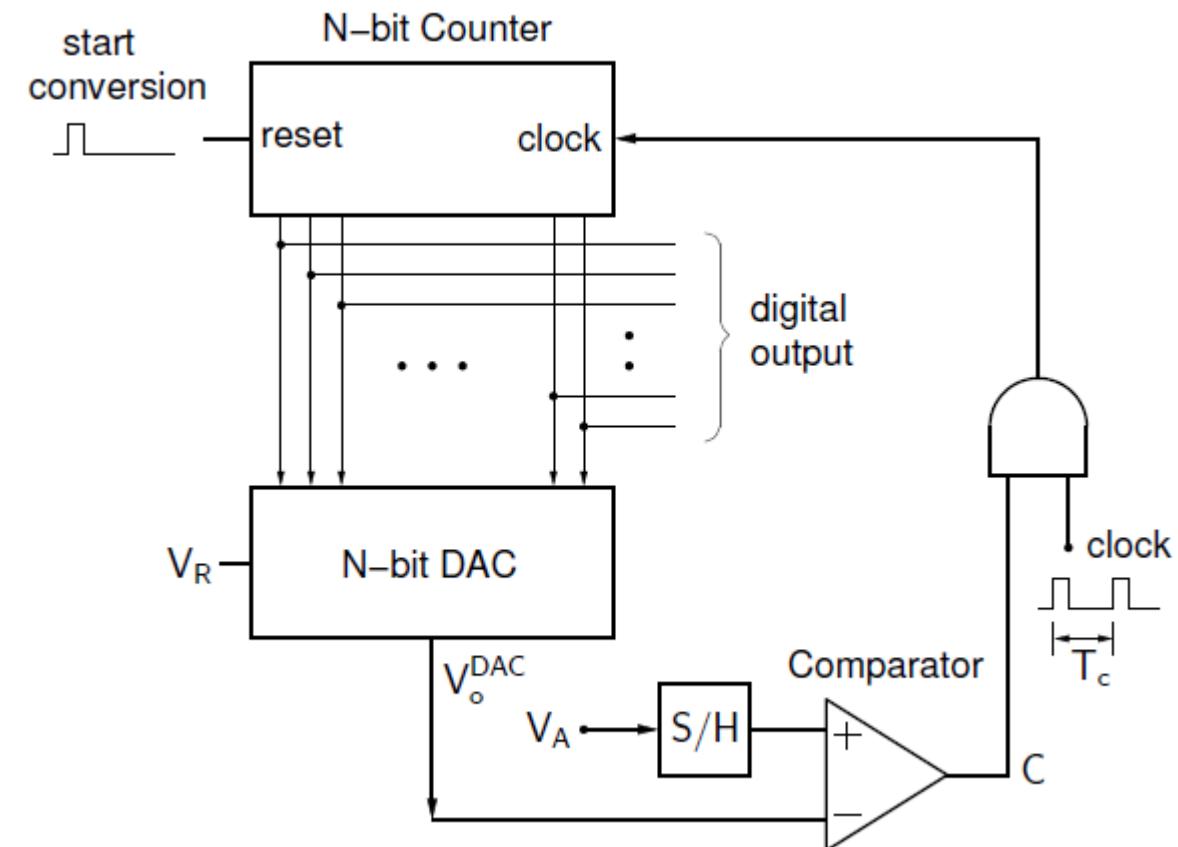
ADC – parallel/flash

- Advantages:
 - Speed – the ADC is not called flash for nothing! Flash ADCs handling 10+ Gsps are commercially available
- Disadvantages:
 - Number of comparators and resistors is 2^n
 - Static power is consumed because V_{ref} is continuously subjected to voltage divider
 - The comparators may not settle to the correct output value together for a changing input – leading to error in output



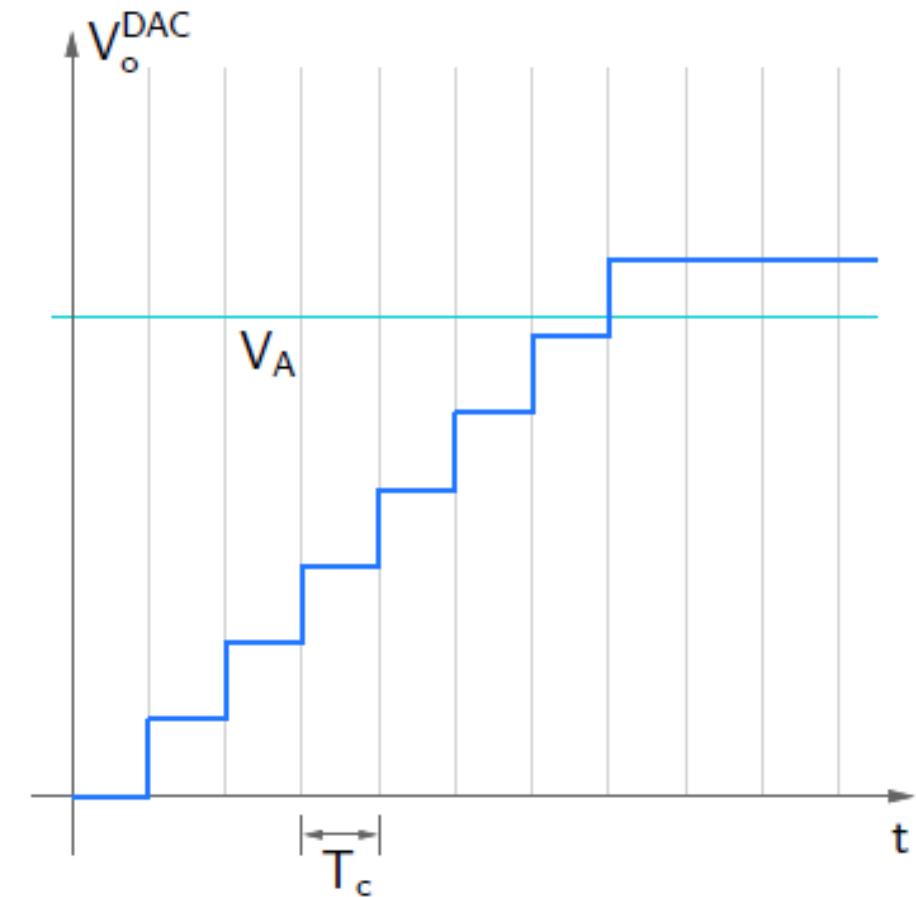
ADC – Ramp type (or counting)

- An interesting way of making an ADC is using an internal DAC to compare the output with the input signal
- We start a digital counter at the start of every conversion
- The digital counter output is converted into analog and compared with the input signal
- When the comparison just becomes high, the counter is stopped
- The output of the counter is the digital equivalent of the analog input



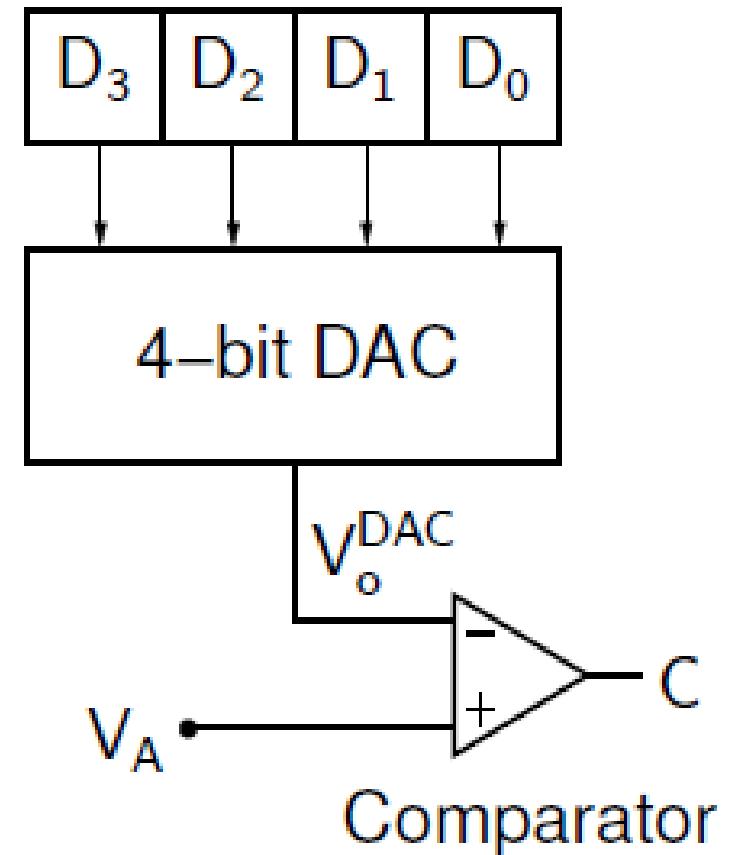
ADC – Ramp type (or counting)

- Advantages:
 - Simpler circuit compared to the flash ADC for large value of n
- Disadvantages:
 - Requires a DAC
 - Is very slow – in worst case, requires 2^n clock cycles to complete. On average 2^{n-1} . This reduces the sampling frequency

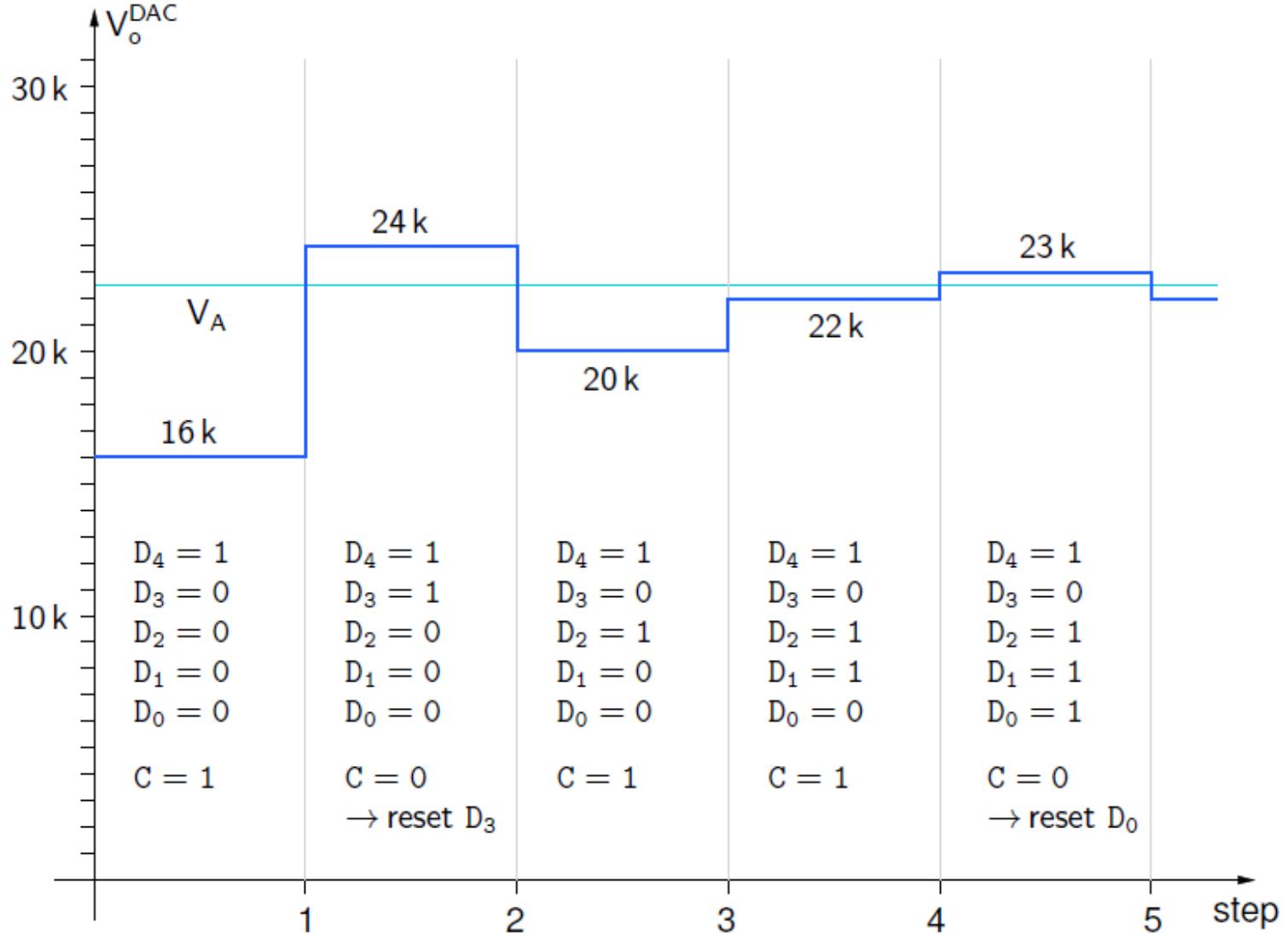
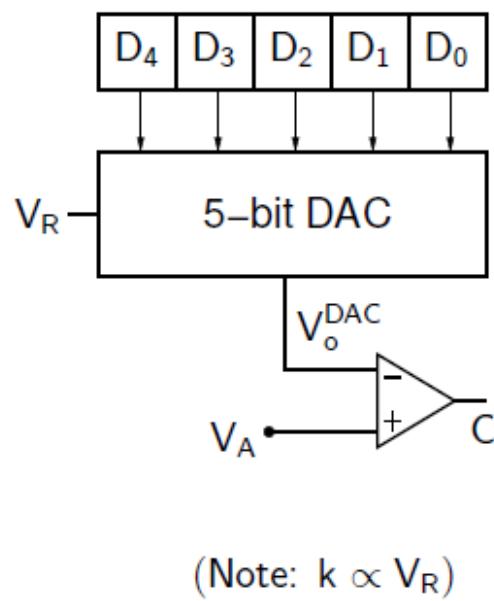


ADC – Successive approximation

- We can use a DAC and adjust one bit at a time to obtain the correct digital output
- Suppose we have a 4-bit DAC
 - Start with $D_3 D_2 D_1 D_0 = 0000$
 - Set MSB to 1 ($D_3 = 1$) keeping other bits unchanged
 - If $V_{DAC} > V_A$, set D_3 back to 0, else keep D_3 at 1
 - Repeat these steps for successively lower bits
- At the end of four steps, the digital output is given by $D_3 D_2 D_1 D_0$



ADC – Successive approximation



Sensors outputs - digital

Dr. Aftab M. Hussain,

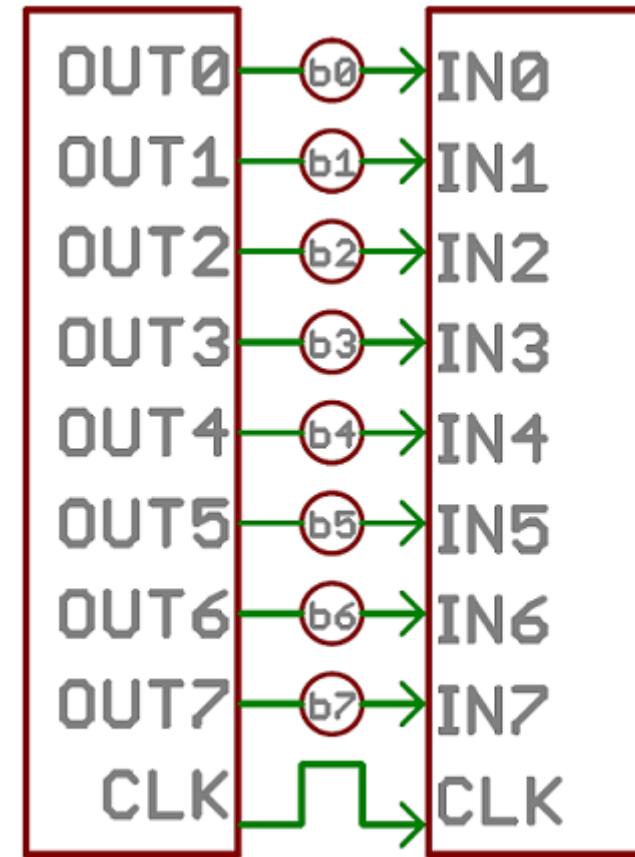
Assistant Professor, PATRIoT Lab, CVEST

Sensor outputs – Digital

- Digital communication is preferred over analog because it is less susceptible to noise
- There are multiple ways in which you can obtain digital
 - Parallel – with each bit on a separate wire
 - Serial – with bit transmitted one after the other
- In serial communication we can different protocols:
 - UART (asynchronous)
 - SPI (synchronous)
 - I2C (synchronous)

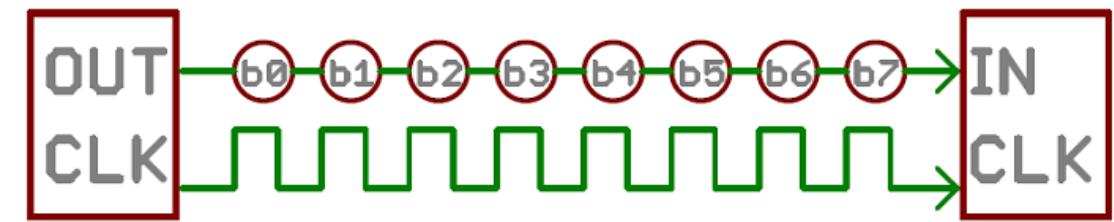
Sensor outputs – Digital – Parallel

- Parallel interfaces transfer multiple bits at the same time
- They usually require **buses** of data - transmitting across eight, sixteen, or more wires
- Advantages:
 - Very high data rates (single clock transfer)
 - Easy to implement
- Disadvantages:
 - Large number of data lines required, specially if number of peripherals are large



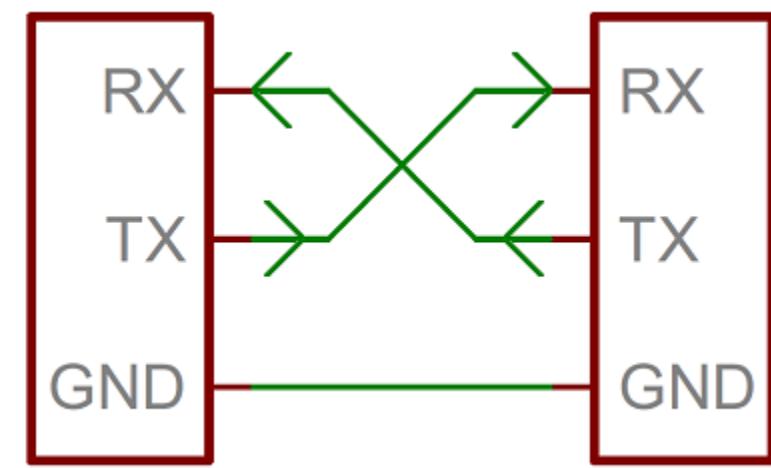
Sensor outputs – Digital – Serial

- Serial interfaces stream their data, one single bit at a time
- These interfaces can operate on as little as one wire
- Serial interfaces can be synchronous and asynchronous
- A synchronous serial interface always pairs its data line(s) with a clock signal, so all devices on a synchronous serial bus share a common clock
- Asynchronous means that data is transferred without support from an external clock signal



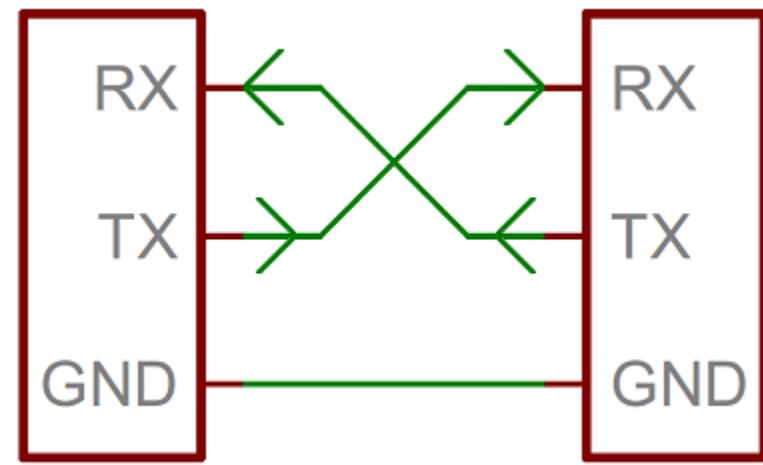
Sensor outputs – Digital – UART

- A universal asynchronous receiver/transmitter (UART) is a serial communication protocol that employs two lines Tx and Rx for communication
- UART support is commonly found inside microcontrollers
- For example, the Arduino Uno - based on the "old faithful" ATmega328 - has just a single UART, while the Arduino Mega - built on an ATmega2560 - has a whopping four UARTs
- NodeMCU has two UARTs



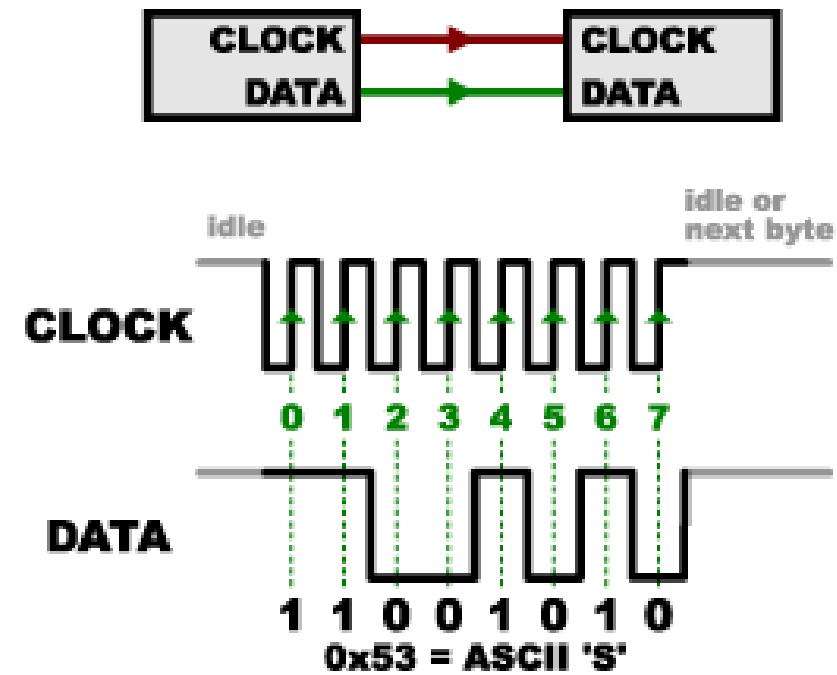
Sensor outputs – Digital – UART

- Advantages:
 - Two line communication
 - Simple to implement in software
 - Legacy protocol
- Disadvantages:
 - No synchronization means we have to make “baud rates” equal manually before communication
 - Low data rate – general baud rate is 9600 bits per second
 - Hardware implementation is complex
 - Needs start and stop bits to sync – which can be wasteful
 - Rx and Tx pins can be very confusing!



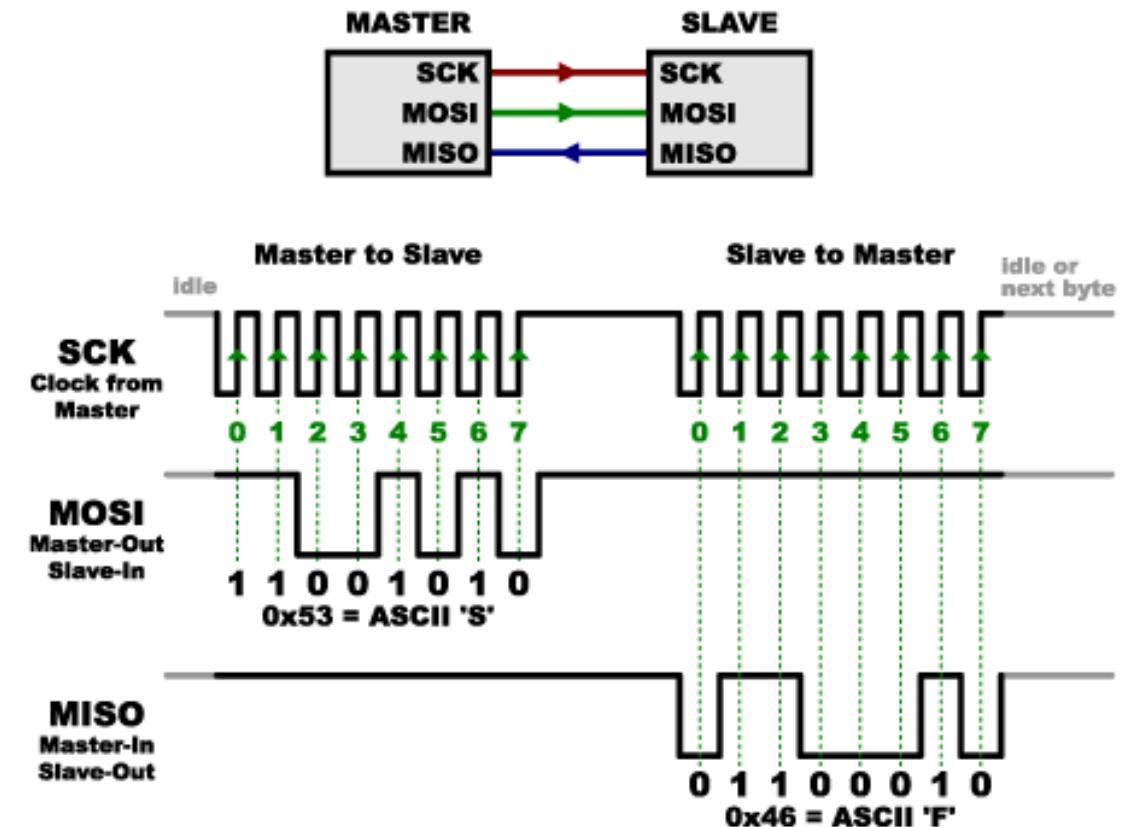
Sensor outputs – Digital – SPI

- SPI is serial peripheral interface
- It's a "synchronous" data bus, which means that it uses separate lines for data and a "clock" that keeps both sides in perfect sync
- The clock is an oscillating signal that tells the receiver exactly when to sample the bits on the data line
- When the receiver detects that edge, it will immediately look at the data line to read the next bit
- Because the clock is sent along with the data, specifying the speed isn't important, although devices will have a top speed at which they can operate



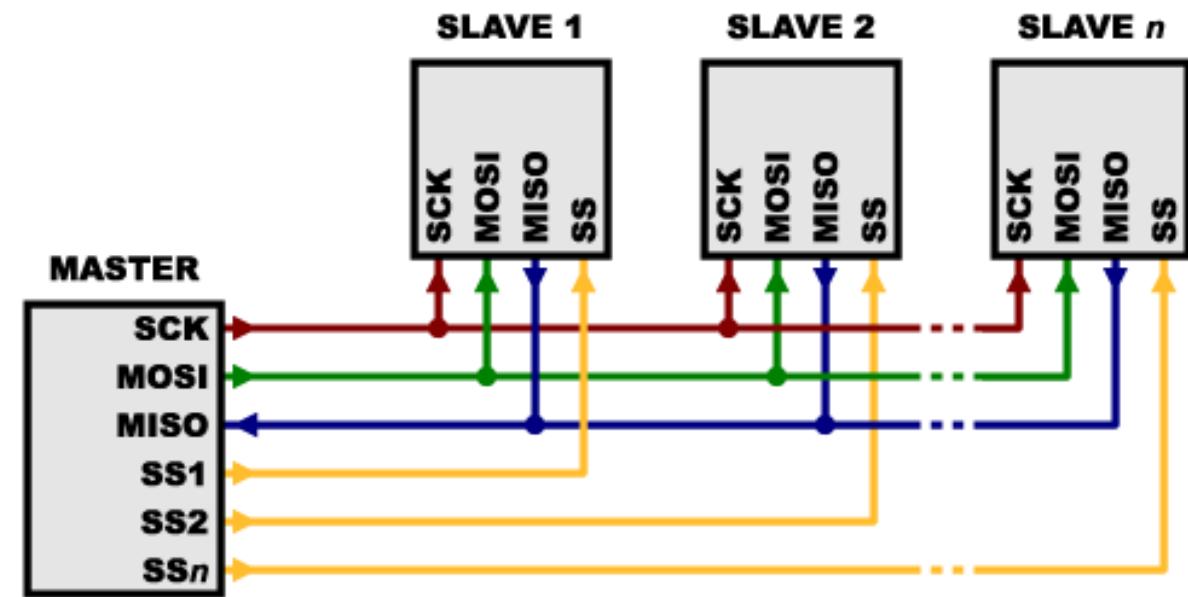
Sensor outputs – Digital – SPI

- We can also configure SPI for duplex communication
- In SPI, only one side generates the clock signal (usually called CLK or SCK for Serial Clock)
- The side that generates the clock is called the "master", and the other side is called the "slave"
- When data is sent from the master to a slave, it's sent on a data line called MOSI, for "Master Out / Slave In"
- If the slave needs to send a response back to the master, the master will continue to generate a prearranged number of clock cycles, and the slave will put the data onto a third data line called MISO, for "Master In / Slave Out"



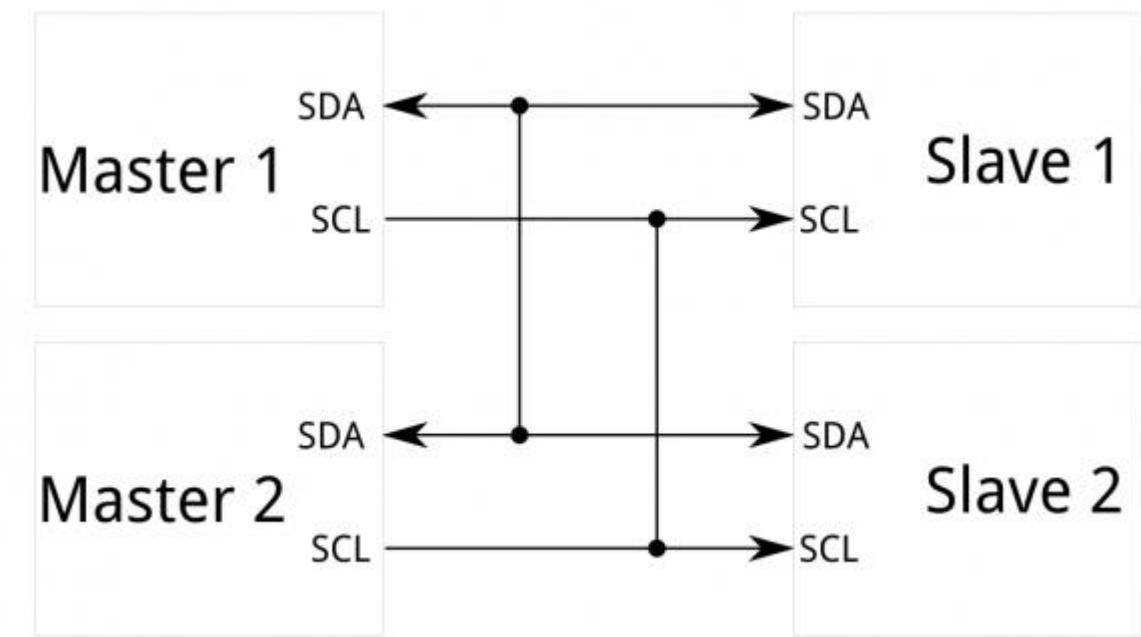
Sensor outputs – Digital – SPI

- Lastly, we can configure SPI for multiple slaves using the same lines for Sclk, MOSI and MISO, but different “slave select” lines
- With this, the slave with its slave select that is enabled will communicate with the master on the same bus, while the others await their turn
- SPI has lots of advantages:
 - Its synchronous so no prearranged baud rates and no start/stop bits
 - Multiple devices on a single bus
- Disadvantages:
 - Too many lines in case of many slaves
 - Only one master per bus



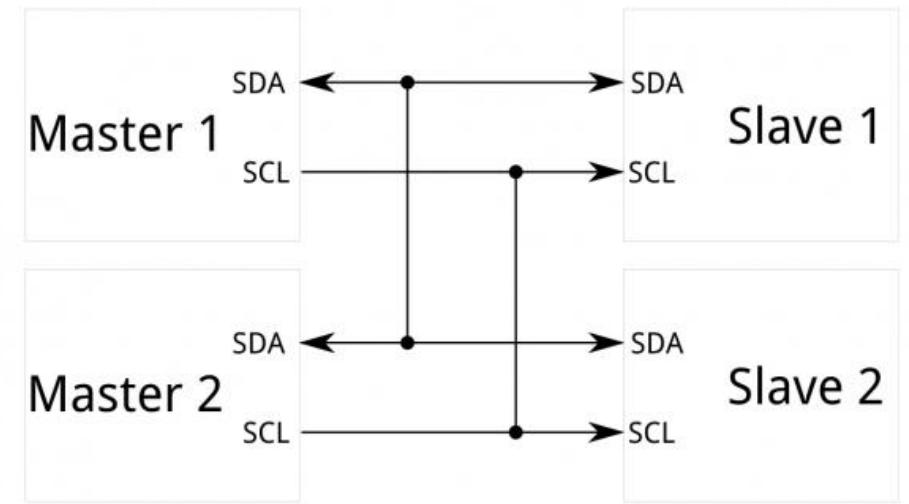
Sensor outputs – Digital – I²C

- The Inter-integrated Circuit (I²C or I2C) Protocol is a protocol intended to allow multiple slaves to communicate with one or more "master" chips
- I²C requires a mere two wires, like asynchronous serial, but those two wires can support up to 1008 slave devices
- Also, unlike SPI, I²C can support a multi-master system, allowing more than one master to communicate with all devices on the bus



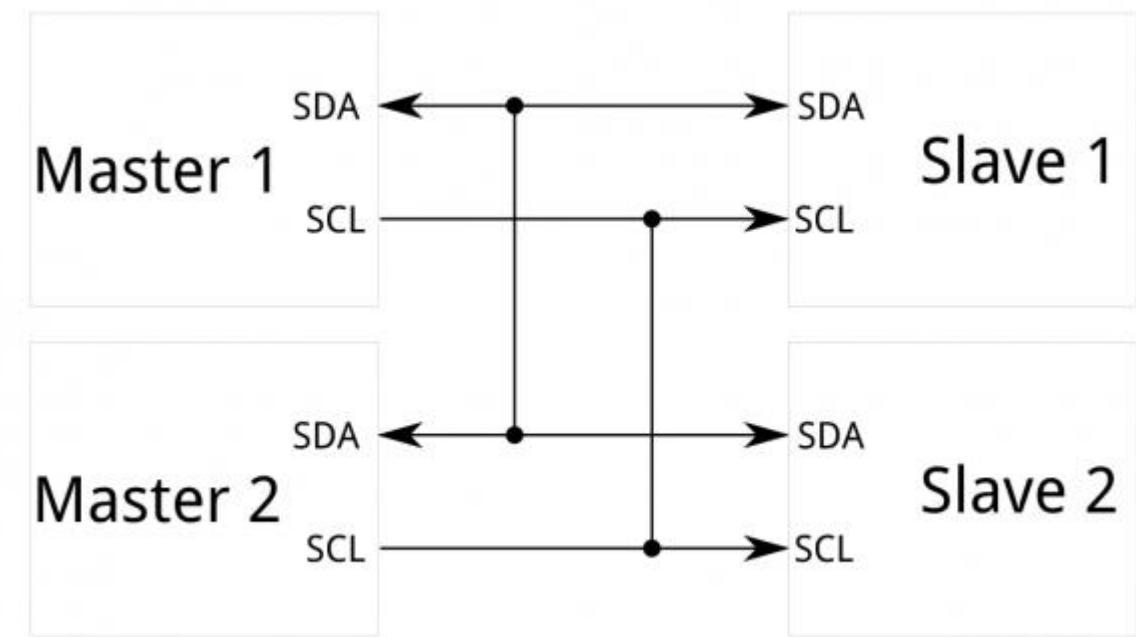
Sensor outputs – Digital – I2C

- Each I²C bus consists of two signals: SCL and SDA. SCL is the clock signal, and SDA is the data signal
- The clock signal is always generated by the current bus master
- Unlike UART or SPI connections, the I2C bus drivers are "open drain", meaning that they can pull the corresponding signal line low, but cannot drive it high
- Thus, there can be no bus contention where one device is trying to drive the line high while another tries to pull it low
- Each signal line has a pull-up resistor on it, to restore the signal to high when no device is asserting it low



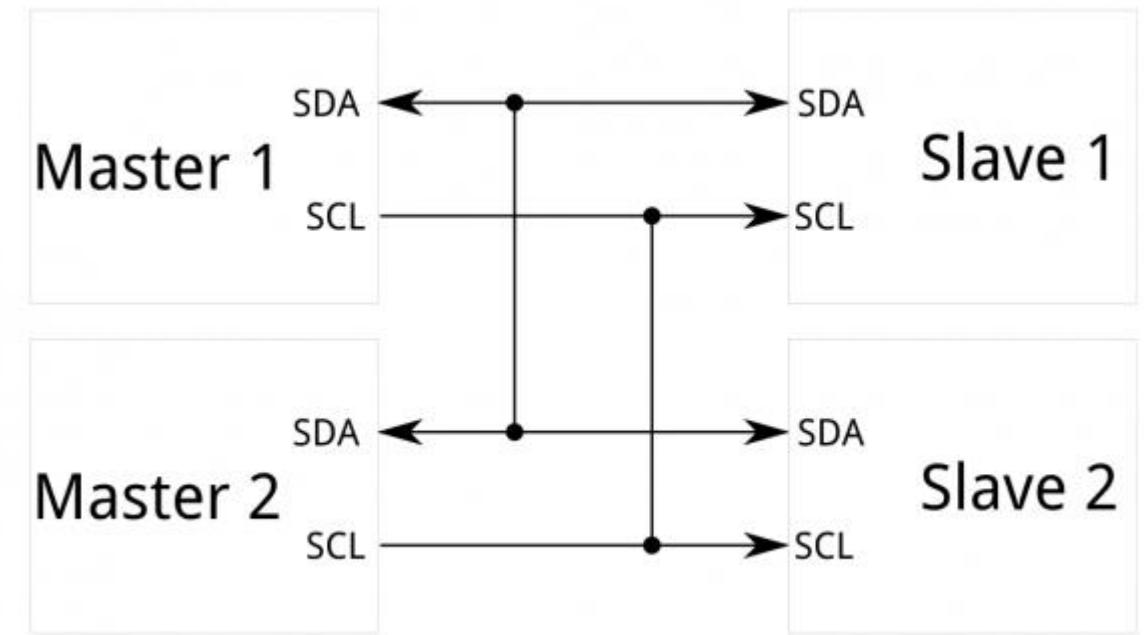
Sensor outputs – Digital – I²C

- Because the devices on the bus don't actually drive the signals high, I²C allows for some flexibility in connecting devices with different I/O voltages
- In general, in a system where one device is at a higher voltage than another, it may be possible to connect the two devices via I²C without any level shifting circuitry in between them
- The trick is to connect the pull-up resistors to the lower of the two voltages
- Although this only works in cases where the lower of the two system voltages exceeds the high-level input voltage of the the higher voltage system - for example, a 5V Arduino and a 3.3V peripheral



Sensor outputs – Digital – I2C

- In practice, most I2C peripherals have a defined address – or changeable address based on some external hardware pins
- The device address is first put on the SDA after the SCL is activated so that the correct slave can listen and respond
- Devices are addressed using a 10-bit address with a total of 1008 addresses possible
- In practice, if more than one I2C peripheral is to be connected, make sure there is only one pull up resistance for the complete bus



Lecture 5 – Interrupts and Timers

Dr. Aftab M. Hussain,
Assistant Professor, PATRIoT Lab, CVEST

Need for interrupts

- Interrupts are exactly what the term means
- This is a very natural way to respond to unexpected events or random events
- We routinely respond to interruptions in our daily lives and then resume the original activity

Need for interrupts

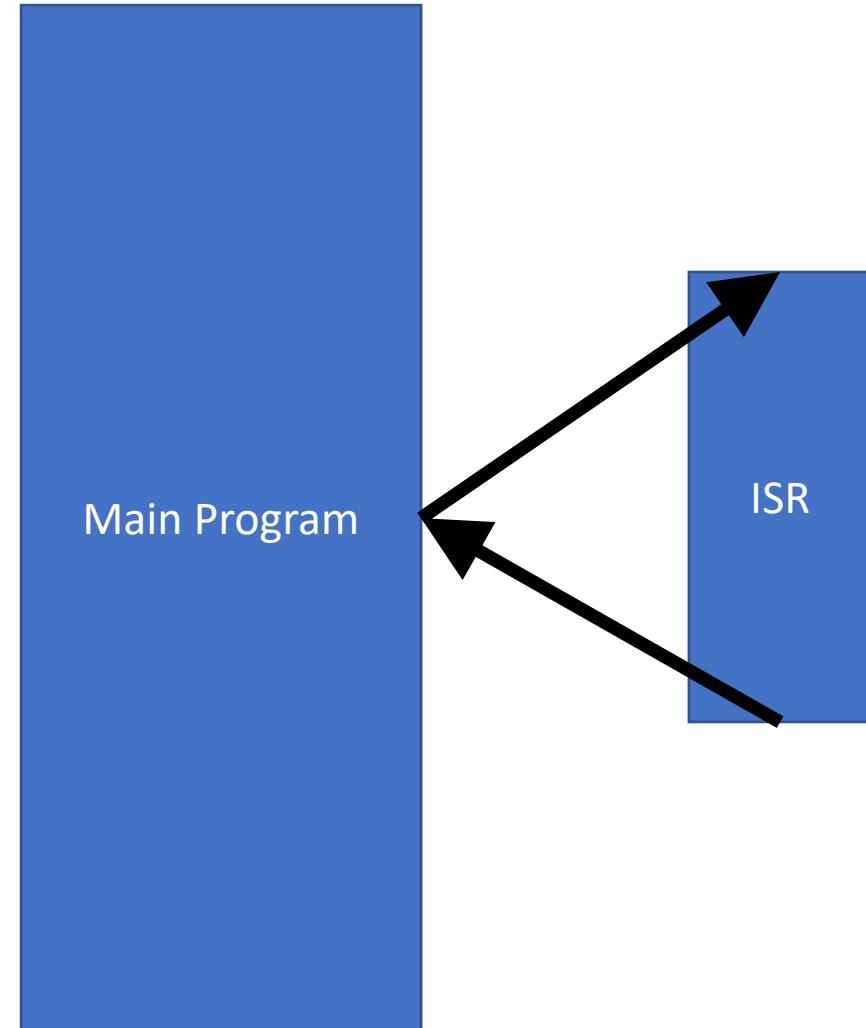
- Interrupts are often used to capture events that can occur at *random* in the external world
- The most common example is of a button being pressed
- The software for what happens when the button is pressed is known, but when to execute it is unknown – the time of the press
- Interrupts can be used to separate non-time-critical functions (within the main loop) from the time-critical functions that are executed on demand in response to interrupts

Introduction

- An interrupt is the automatic transfer of software execution in response to an event that is asynchronous with the current software execution
- This event is called a trigger
- The event can either be a busy to ready transition in an external I/O device (like the UART input/output) or an internal event (like bus fault, memory fault, or a periodic timer)
- This should interrupt the flow of the embedded system program to follow another set of instructions
- These instructions are called the interrupt service routine (ISR)
- Once the ISR is completed, the code is expected to execute at the point of break

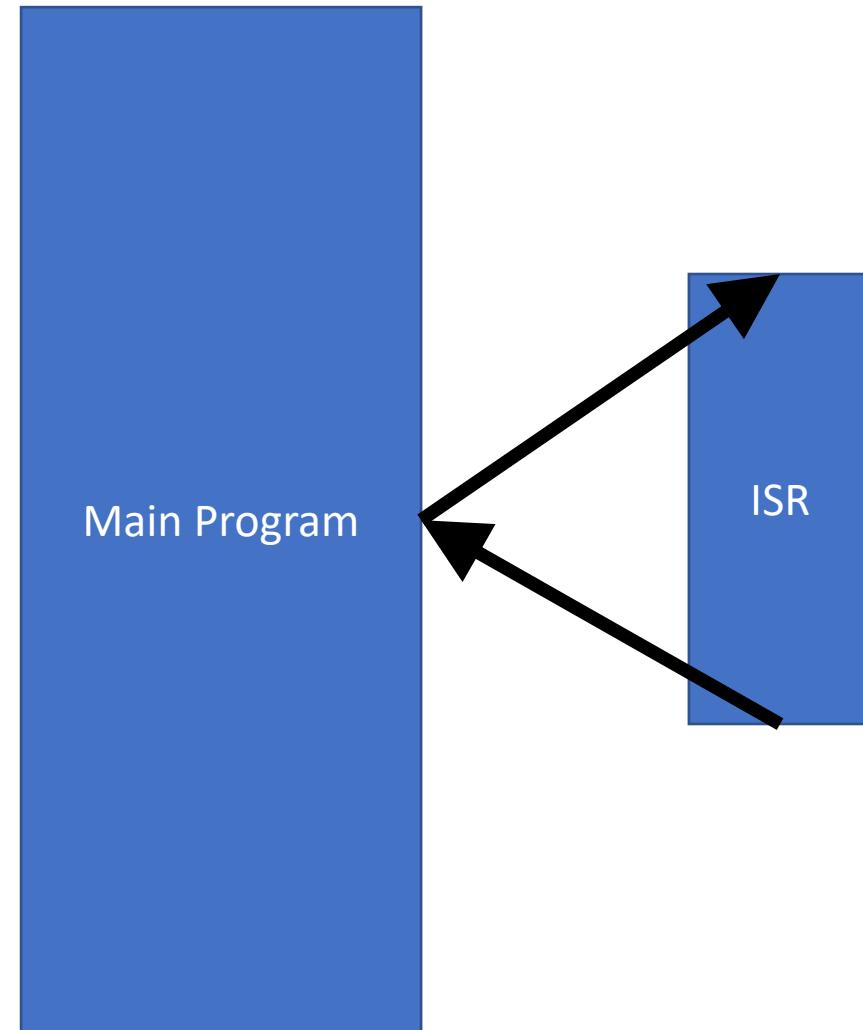
How they work

- The ISR is stored in a specific part of the memory with the address of the ISR stored in dedicated registers
- When the interrupt is triggered, the current state of the program has to be stored temporarily and the program execution needs to move to the ISR address
- Following the completion of the ISR, the program execution returns to the original position



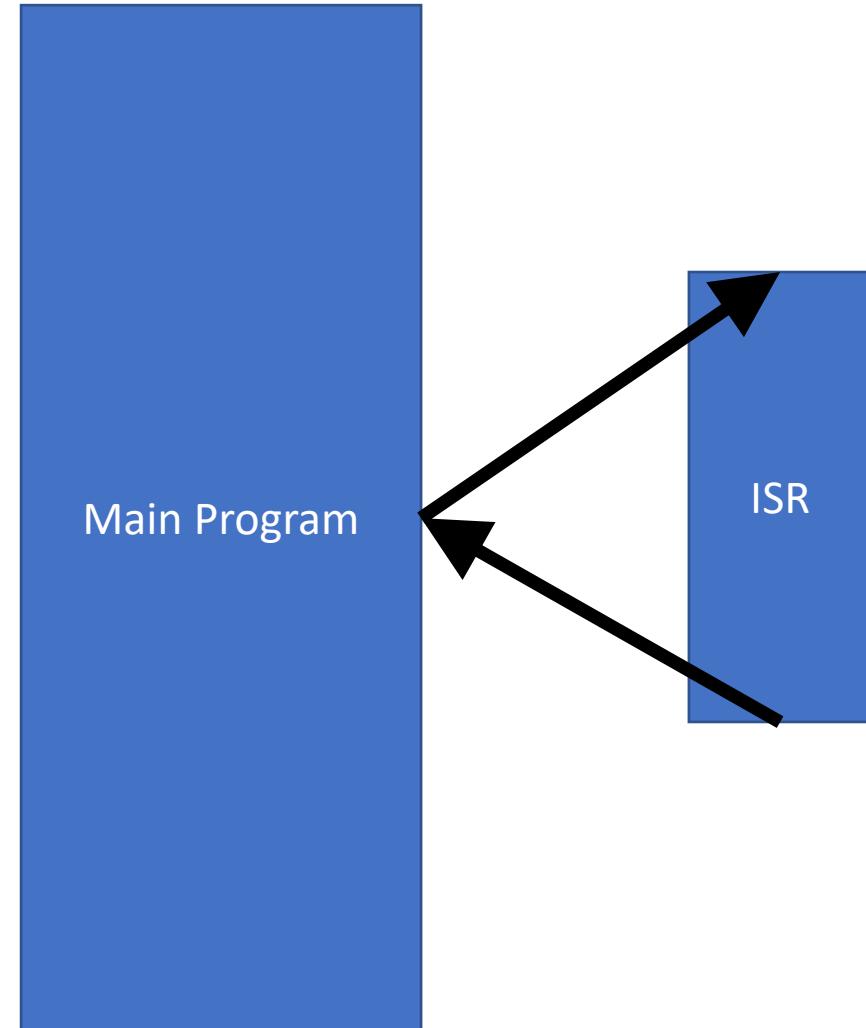
How they work

- The controller needs to recognize that the code involves possibilities of an interrupt
- This is done by using an enabling interrupts function generally provided as standard for any controller supporting interrupts
- This makes the controller check for the trigger flag for the interrupts that are enabled and for the type of interrupts in play
- Once an interrupt is being processed, other interrupts are automatically disabled (except in special cases)

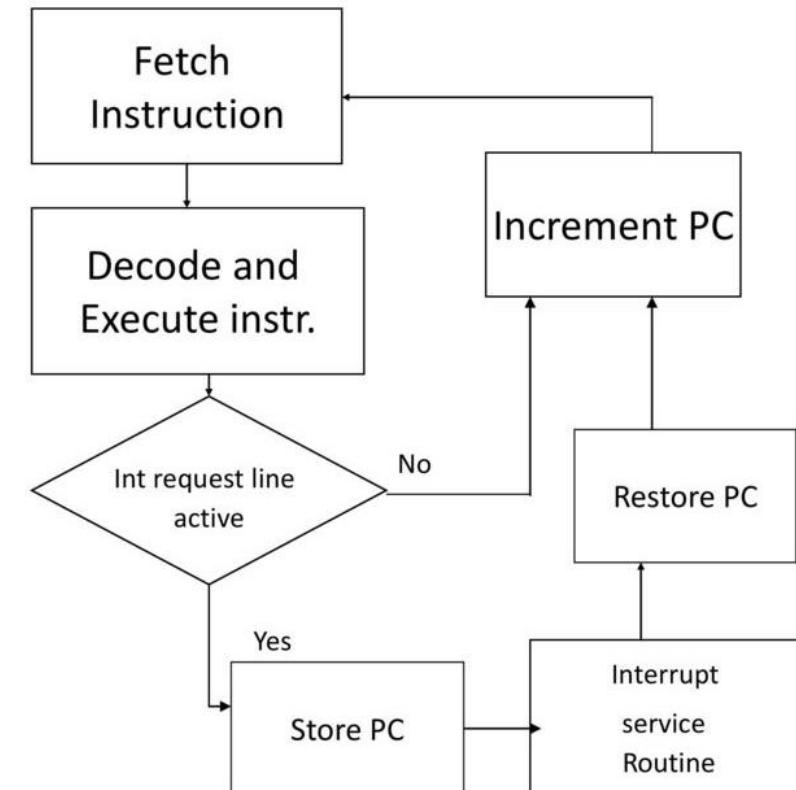
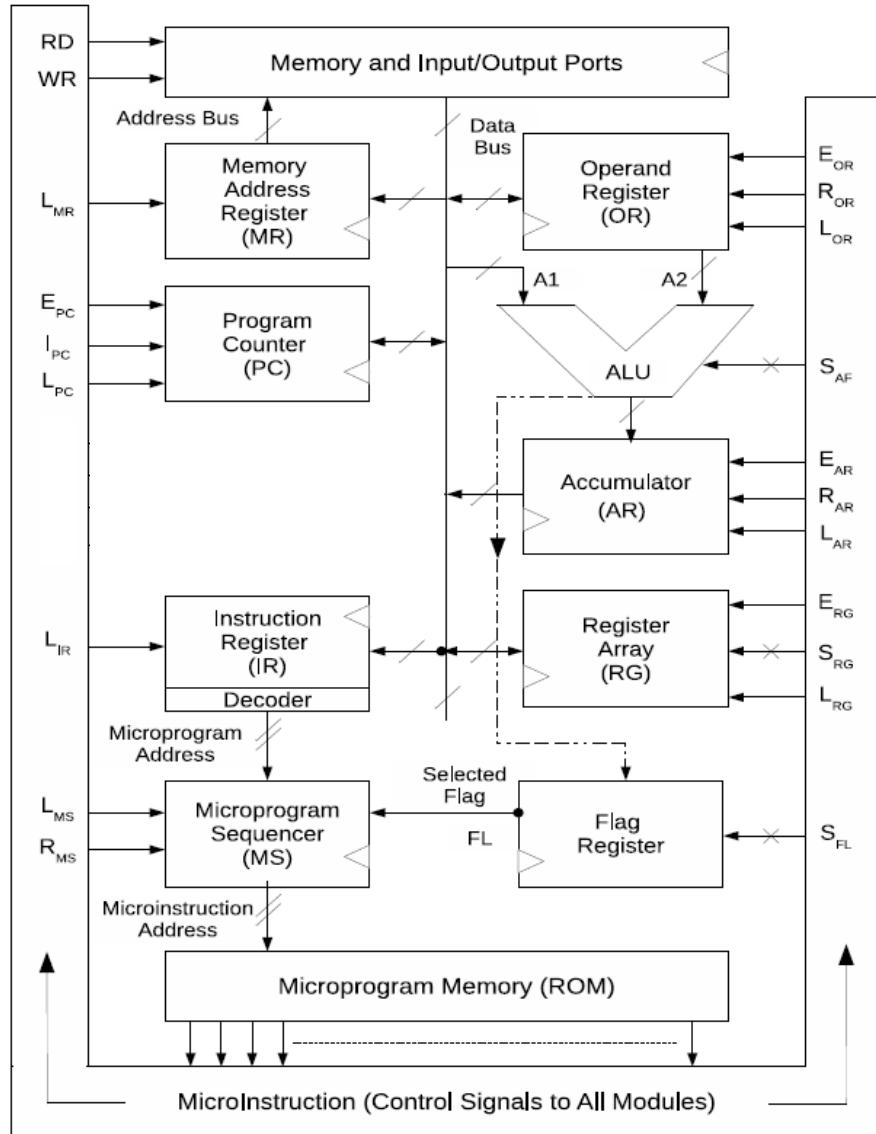


Interrupt setup

- Apart from the global interrupt enable, many other registers have to be set
- Interrupt control register – used for configuring the interrupt pin, type of interrupt, level, edge triggered etc.
- Interrupt priority – used to determine which interrupt are serviced if more than one occur simultaneously



Interrupt setup



Interrupts in Arduino

- External interrupts
 - Used to monitor changes in I/O pins
 - The triggering event can be configured depending on the application using the control register
- On Arduino uno, there are two external interrupts INT0 and INT1 at pin 2 and 3 that can be used for multiple event types
- In atmega328, all of the IO pins can be used as pin change interrupts (PCINT) to detect toggle in signal level

ATMega328P and Arduino Uno Pin Mapping

Arduino function		Arduino function
reset	(PCINT14/RESET) PC6	1 28 PC5 (ADC5/SCL/PCINT13)
digital pin 0 (RX)	(PCINT16/RXD) PD0	2 27 PC4 (ADC4/SDA/PCINT12)
digital pin 1 (TX)	(PCINT17/TXD) PD1	3 26 PC3 (ADC3/PCINT11)
digital pin 2	(PCINT18/INT0) PD2	4 25 PC2 (ADC2/PCINT10)
digital pin 3 (PWM)	(PCINT19/OC2B/INT1) PD3	5 24 PC1 (ADC1/PCINT9)
digital pin 4	(PCINT20/XCK/T0) PD4	6 23 PC0 (ADC0/PCINT8)
VCC	VCC	7 22 GND
GND	GND	8 21 AREF
crystal	(PCINT6/XTAL1/TOSC1) PB6	9 20 AVCC
crystal	(PCINT7/XTAL2/TOSC2) PB7	10 19 PB5 (SCK/PCINT5)
digital pin 5 (PWM)	(PCINT21/OC0B/T1) PD5	11 18 PB4 (MISO/PCINT4)
digital pin 6 (PWM)	(PCINT22/OC0A/AIN0) PD6	12 17 PB3 (MOSI/OC2A/PCINT3)
digital pin 7	(PCINT23/AIN1) PD7	13 16 PB2 (SS/OC1B/PCINT2)
digital pin 8	(PCINT0/CLKO/ICP1) PB0	14 15 PB1 (OC1A/PCINT1)

Digital Pins 11,12 & 13 are used by the ICSP header for MOSI, MISO, SCK connections (Atmega168 pins 17,18 & 19). Avoid low-impedance loads on these pins when using the ICSP header.

Interrupts in Arduino

- To use INT0 and INT1, the corresponding bits need to be set in the GICR
- The register is set when the controller encounters the instruction determining the use of one of the external interrupt pins
- If this bit is set along with the global interrupt enable, the processor checks for interrupts (interrupt flag) after every instruction

General Interrupt Control Register - GICR									
Bit	7	6	5	4	3	2	1	0	
Bit Name	INT1	INT0	-	-	-	-	IVSEL	IVCE	
Read/Write	RW	RW	RW	RW	RW	RW	RW	RW	RW
Initial value	0	0	0	0	0	0	0	0	0

Interrupts in Arduino

- Then we need to set the external interrupt control register (EICR), sometimes also referred to as MCUCR
- This determines the trigger event for the interrupts
- The interrupt is triggered only if:
 - The global interrupt bit is set
 - The corresponding GICR bit is set
 - The MCUCR is set
 - The correct event occurs at the external pin

Bit	7	6	5	4	3	2	1	0
Bit Name	SE	SM2	SM1	SM0	ISC11	ISC10	ISC01	ISC00
Read/Write	RW	RW	RW	RW	RW	RW	RW	RW
Initial value	0	0	0	0	0	0	0	0

Interrupts in Arduino

Interrupt Sense Control		
ISCx1	ISCx0	Interrupt Generated Upon
0	0	The low Level of INTx pin
0	1	Any logical change in INTx pin
1	0	Falling edge of INTx
1	1	Rising edge of INTx

x- Interrupt number, either 0 or 1

Bit	7	6	5	4	3	2	1	0
Bit Name	SE	SM2	SM1	SM0	ISC11	ISC10	ISC01	ISC00
Read/Write	RW	RW	RW	RW	RW	RW	RW	RW
Initial value	0	0	0	0	0	0	0	0

Interrupts in Arduino IDE

- `attachInterrupt(digitalPinToInterrupt(pin), ISR, mode)`
- This single line of code in Arduino IDE does all of the required bit setting in the controller
- The parameters are:
 - pin: the Arduino pin number (2, 3)
 - ISR: the ISR to call when the interrupt occurs
 - mode: defines when the interrupt should be triggered. Four constants are predefined as valid values:
 - LOW to trigger the interrupt whenever the pin is low
 - CHANGE to trigger the interrupt whenever the pin changes value
 - RISING to trigger when the pin goes from low to high
 - FALLING for when the pin goes from high to low

ATMega328P and Arduino Uno Pin Mapping

Arduino function		Arduino function
reset	(PCINT14/RESET) PC6	1 28 PC5 (ADC5/SCL/PCINT13)
digital pin 0 (RX)	(PCINT16/RXD) PD0	2 27 PC4 (ADC4/SDA/PCINT12)
digital pin 1 (TX)	(PCINT17/TXD) PD1	3 26 PC3 (ADC3/PCINT11)
digital pin 2	(PCINT18/INT0) PD2	4 25 PC2 (ADC2/PCINT10)
digital pin 3 (PWM)	(PCINT19/OC2B/INT1) PD3	5 24 PC1 (ADC1/PCINT9)
digital pin 4	(PCINT20/XCK/T0) PD4	6 23 PC0 (ADC0/PCINT8)
VCC	VCC	7 22 GND
GND	GND	8 21 AREF
crystal	(PCINT6/XTAL1/TOSC1) PB6	9 20 AVCC
crystal	(PCINT7/XTAL2/TOSC2) PB7	10 19 PB5 (SCK/PCINT5)
digital pin 5 (PWM)	(PCINT21/OC0B/T1) PD5	11 18 PB4 (MISO/PCINT4)
digital pin 6 (PWM)	(PCINT22/OC0A/AIN0) PD6	12 17 PB3 (MOSI/OC2A/PCINT3)
digital pin 7	(PCINT23/AIN1) PD7	13 16 PB2 (SS/OC1B/PCINT2)
digital pin 8	(PCINT0/CLKO/ICP1) PB0	14 15 PB1 (OC1A/PCINT1)

Digital Pins 11,12 & 13 are used by the ICSP header for MOSI, MISO, SCK connections (Atmega168 pins 17,18 & 19). Avoid low-impedance loads on these pins when using the ICSP header.

Example button press code

- Easy to implement a simple button press routine
- Connect the push button to pin 2 and ground
- The global interrupt , GICR and MCUCR are all set using the attachInterrupt function
- With this code, the LED will turn on when the button is pressed and off when released
- Challenge: Can we think of a way to create a code for a button that performs different tasks if short-pressed or long-pressed?

```
const byte ledPin = 13;
const byte interruptPin = 2;
volatile byte state = LOW;

void setup() {
    pinMode(ledPin, OUTPUT);
    pinMode(interruptPin, INPUT_PULLUP);
    attachInterrupt(digitalPinToInterrupt(interruptPin), blink, CHANGE);
}

void loop() {
    digitalWrite(ledPin, state);
}

void blink() {
    state = !state;
}
```

Internal interrupts

- Apart from the external pin based interrupts, we have many internal interrupts in most controllers
- In Arduino, the total number of internal interrupts is around 20 (including timer based interrupts)
- Some of the most used are (in order of decreasing priority):
 - Reset – external interrupt with no RELI()
 - INT0,1 – external pins
 - PCINT – external pins
 - Watchdog timer-out
 - ADC conversion complete
 - Analog comparator

ATMega328P and Arduino Uno Pin Mapping

Arduino function		Arduino function
reset	(PCINT14/RESET) PC6	1 28 PC5 (ADC5/SCL/PCINT13) 27 PC4 (ADC4/SDA/PCINT12)
digital pin 0 (RX)	(PCINT16/RXD) PD0	3 26 PC3 (ADC3/PCINT11) 4 25 PC2 (ADC2/PCINT10)
digital pin 1 (TX)	(PCINT17/TXD) PD1	5 24 PC1 (ADC1/PCINT9) 6 23 PC0 (ADC0/PCINT8)
digital pin 2	(PCINT18/INT0) PD2	7 22 GND 8 21 AREF
digital pin 3 (PWM)	(PCINT19/OC2B/INT1) PD3	9 20 AVCC 10 19 PB5 (SCK/PCINT5) 11 18 PB4 (MISO/PCINT4)
digital pin 4	(PCINT20/XCK/T0) PD4	12 17 PB3 (MOSI/OC2A/PCINT3) 13 16 PB2 (SS/OC1B/PCINT2) 14 15 PB1 (OC1A/PCINT1)
VCC	VCC	digital pin 13 digital pin 12 digital pin 11(PWM) digital pin 10 (PWM) digital pin 9 (PWM)
GND	GND	VCC
crystal	(PCINT6/XTAL1/TOSC1) PB6	
crystal	(PCINT7/XTAL2/TOSC2) PB7	
digital pin 5 (PWM)	(PCINT21/OC0B/T1) PD5	
digital pin 6 (PWM)	(PCINT22/OC0A/AIN0) PD6	
digital pin 7	(PCINT23/AIN1) PD7	
digital pin 8	(PCINT0/CLKO/ICP1) PB0	

Digital Pins 11,12 & 13 are used by the ICSP header for MOSI, MISO, SCK connections (Atmega168 pins 17,18 & 19). Avoid low-impedance loads on these pins when using the ICSP header.

Internal interrupts

- ADC conversion complete
 - This interrupt signals that conversion from analog to digital is complete and the result is available as `analogRead`
 - Useful for sampling fast changing analog signals at the fastest rate possible
 - To enable this, we need to set the ADCSR register with bits like
 - `ADEN` – enable ADC
 - `ADIE` – enable ADC interrupt
 - The `ADIF` (ADC interrupt flag) is enabled when the conversion is over and the data register is updated – this is only done when `ADIE` and global interrupt is set
- Once done, the ISR can be programmed as:

```
ISR (ADC_vect)
{
    //code
}
```

Internal interrupts

- Analog comparator
 - This interrupt is triggered when the comparison output between pin AIN1 and AIN0 toggles in a particular direction
 - The incoming voltage is on the AIN0 (positive) pin which is pin 12 on the actual chip, and D6 on the Arduino board. The reference voltage (negative) pin is pin 13 on the chip, and D7 on the Arduino
 - To enable this, we set the ACSR:
 - ACIE – Analog Comparator Interrupt Enable
 - ACIS1 – Analog Comparator Interrupt Mode Select
- ACI - Analog Comparator Interrupt Flag is set when triggered

```
ISR (ANALOG_COMP_vect)
{
    //code
}
```

ATMega328P and Arduino Uno Pin Mapping

Arduino function		Arduino function
reset	(PCINT14/RESET) PC6	1 28 PC5 (ADC5/SCL/PCINT13) analog input 5
digital pin 0 (RX)	(PCINT16/RXD) PD0	2 27 PC4 (ADC4/SDA/PCINT12) analog input 4
digital pin 1 (TX)	(PCINT17/TXD) PD1	3 26 PC3 (ADC3/PCINT11) analog input 3
digital pin 2	(PCINT18/INT0) PD2	4 25 PC2 (ADC2/PCINT10) analog input 2
digital pin 3 (PWM)	(PCINT19/OC2B/INT1) PD3	5 24 PC1 (ADC1/PCINT9) analog input 1
digital pin 4	(PCINT20/XCK/T0) PD4	6 23 PC0 (ADC0/PCINT8) analog input 0
VCC	VCC	7 22 GND GND
GND	GND	8 21 AREF analog reference
crystal	(PCINT6/XTAL1/TOSC1) PB6	9 20 AVCC VCC
crystal	(PCINT7/XTAL2/TOSC2) PB7	10 19 PB5 (SCK/PCINT5) digital pin 13
digital pin 5 (PWM)	(PCINT21/OC0B/T1) PD5	11 18 PB4 (MISO/PCINT4) digital pin 12
digital pin 6 (PWM)	(PCINT22/OC0A/AIN0) PD6	12 17 PB3 (MOSI/OC2A/PCINT3) digital pin 11(PWM)
digital pin 7	(PCINT23/AIN1) PD7	13 16 PB2 (SS/OC1B/PCINT2) digital pin 10 (PWM)
digital pin 8	(PCINT0/CLKO/ICP1) PB0	14 15 PB1 (OC1A/PCINT1) digital pin 9 (PWM)

Digital Pins 11,12 & 13 are used by the ICSP header for MOSI, MISO, SCK connections (Atmega168 pins 17,18 & 19). Avoid low-impedance loads on these pins when using the ICSP header.

Multiple interrupts

- There are cases when multiple interrupt flags will be enabled simultaneously
- In this case, the interrupt with the highest priority gets serviced
- The priority is determined beforehand in the boot-loader or can be software set in more advanced processors
- By default, interrupts are disabled during an ISR, however, there are controllers that “remember” the interrupts that got triggered during an ISR and service them once the ISR is done

VectorNo.	Program Address ⁽²⁾	Source	Interrupt Definition
1	0x0000 ⁽¹⁾	RESET	External Pin, Power-on Reset, Brown-out Reset and Watchdog System Reset
2	0x0002	INT0	External Interrupt Request 0
3	0x0004	INT1	External Interrupt Request 1
4	0x0006	PCINT0	Pin Change Interrupt Request 0
5	0x0008	PCINT1	Pin Change Interrupt Request 1
6	0x000A	PCINT2	Pin Change Interrupt Request 2
7	0x000C	WDT	Watchdog Time-out Interrupt
8	0x000E	TIMER2 COMPA	Timer/Counter2 Compare Match A
9	0x0010	TIMER2 COMPB	Timer/Counter2 Compare Match B
10	0x0012	TIMER2 OVF	Timer/Counter2 Overflow
11	0x0014	TIMER1 CAPT	Timer/Counter1 Capture Event
12	0x0016	TIMER1 COMPA	Timer/Counter1 Compare Match A
13	0x0018	TIMER1 COMPB	Timer/Counter1 Compare Match B
14	0x001A	TIMER1 OVF	Timer/Counter1 Overflow
15	0x001C	TIMER0 COMPA	Timer/Counter0 Compare Match A
16	0x001E	TIMER0 COMPB	Timer/Counter0 Compare Match B
17	0x0020	TIMER0 OVF	Timer/Counter0 Overflow
18	0x0022	SPI, STC	SPI Serial Transfer Complete
19	0x0024	USART, RX	USART Rx Complete
20	0x0026	USART, UDRE	USART, Data Register Empty
21	0x0028	USART, TX	USART, Tx Complete
22	0x002A	ADC	ADC Conversion Complete
23	0x002C	EE READY	EEPROM Ready
24	0x002E	ANALOG COMP	Analog Comparator
25	0x0030	TWI	2-wire Serial Interface
26	0x0032	SPM READY	Store Program Memory Ready

Nested interrupts

- In certain cases, interrupts can be enabled during the servicing of an ISR
- Thus, when a particular ISR is being processed, an interrupt can trigger and cause the program execution to go to that ISR
- This may also lead to recursive interrupts if permitted by the processor architecture
- This method is generally not recommended and should be avoided unless absolutely necessary
- Simple solutions like have multiple short ISRs is recommended

Timers

Dr. Aftab M. Hussain,
Assistant Professor, PATRIoT Lab, CVEST

Timers

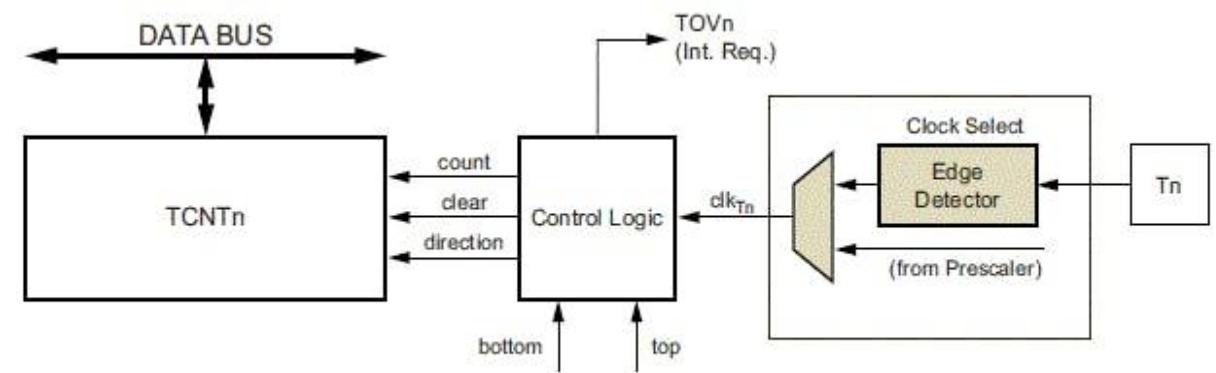
- Internal timers are used to calculate time using the known clock frequency obtained from the PLL output
- In essence, timers are just counters being provided with a particular clock frequency
- Once a particular count is over, a timer generally creates an interrupt that can trigger a particular application or even other timers to register their ticks
- Many Arduino functions uses timers, for example the time functions:
 - `delay()`
 - `millis()`
 - `micros()`
 - `delayMicroseconds()`

Timers

- A `delay()` function is most commonly used to stall the program for a fixed amount of time
 - `delay(x)` stalls the program for x millisecond
- However, during this time, the program is not able to perform any other task
- Say a blinking LED is required while checking for an analog input
- If this is done using a `delay()` function, the delay function can inhibit the `analogRead` function

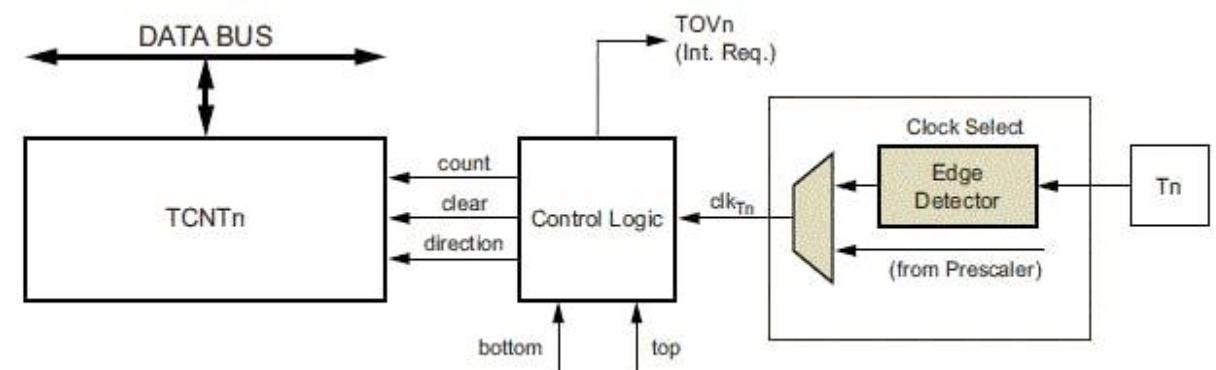
Timers

- Another way of making the same code is using internal timers
- In Arduino, we have 3 timers, called timer0, timer1 and timer2
- Timer0 and timer2 are 8bit timers, where timer1 is a 16bit timer
- Thus, maximum count for timer0 and timer2 is 256 while that for timer1 is 65536
- These timers are implemented using TCNT_n counter connected to the databus with a prescaled clock applied to each



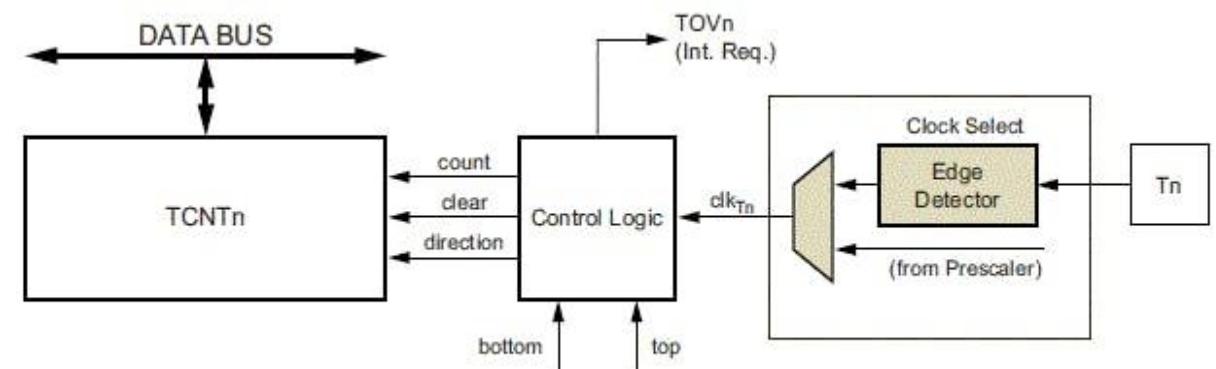
Timers

- All the timer values can be compared with a given value to generate an interrupt at a predefined time
- The registers involved in this are:
 - TCNT_n - Timer/Counter Register. The actual timer value is stored here
 - TCCR_n – Timer/Counter Control Register. Determines the settings for the timer
 - OCR_n - Output Compare Register
 - TIMSK_n - Timer/Counter Interrupt Mask Register. To enable/disable timer interrupts
 - TIFR_n - Timer/Counter Interrupt Flag Register. Indicates a pending timer interrupt



Timers

- Apart from this, the prescaler needs to be set to scale the internal clock of the controller as needed
- This increases the range of the timer, however, decreases the precision with which the timing is measured
- For instance, if the clock of the controller is 16 MHz and the prescalar is 8, the actual clock applied to the timer is $16/8 = 2$ MHz



Timers

- Let us take the same example of an LED blinking at 0.5Hz
- To calculate the timer frequency you use the CPU frequency as 16Mhz for Arduino
- If we use timer0, the maximum timer counter is 256
- We set the TCCR0B to a particular prescalar using CS00, CS01 and CS02, say 64
- The clock frequency for timer is: $16\text{ M} / 64 = 250000\text{ Hz}$
- We cannot get the 1 sec clock with this, but we can get 1 ms clock
- If we need 1000Hz operation, we divide result through the desired frequency: $250000 / 1000\text{Hz} = 250$
- Thus, the OCROA value should be 249 (result – 1)

	7 bit	6 bit	5 bit	4 bit	3 bit	2 bit	1 bit	0 bit
TCCR0B	FOC0A	FOC0B	-	-	WGM02	CS02	CS01	CS00

Timer/Counter Control Register 0 B

CS02	CS01	CS00	DESCRIPTION
0	0	0	Timer/Counter0 Disabled
0	0	1	No Prescaling
0	1	0	Clock / 8
0	1	1	Clock / 64
1	0	0	Clock / 256
1	0	1	Clock / 1024

Timers

- An important function of all the three timers is the Clear Timer on Compare match
- When the timer counter (TCNT_n) reaches the compare match register (OCR_n), the timer will be cleared to zero
- This helps the timer frequency stay at the desired value
- This mode is enabled by setting the WGM02 bit in the TCCRnB register

	7 bit	6 bit	5 bit	4 bit	3 bit	2 bit	1 bit	0 bit
TCCR0B	FOC0A	FOC0B	-	-	WGM02	CS02	CS01	CS00

Timer/Counter Control Register 0 B

CS02	CS01	CS00	DESCRIPTION
0	0	0	Timer/Counter0 Disabled
0	0	1	No Prescaling
0	1	0	Clock / 8
0	1	1	Clock / 64
1	0	0	Clock / 256
1	0	1	Clock / 1024

Timers

```
/*
This program turns on and off a LED on pin 13 each 1 second
*/

int timer=0;
bool state=0;
void setup() {
  pinMode(13,OUTPUT);

TCCR0A=(1<<WGM01);      //Set the CTC mode
OCR0A=0xF9; //Value for OCR0A for 1ms

TIMSK0|=(1<<OCIE0A);    //Set the interrupt request
sei(); //Enable interrupt

TCCR0B|=(1<<CS01);      //Set the prescale 1/64 clock
TCCR0B|=(1<<CS00);

void loop() {
  //in this way you can count 1 second because the interrupt
  if(timer>=1000){
    state=!state;
    timer=0;
  }
  digitalWrite(13,state);
}

ISR(TIMER0_COMPA_vect){ //This is the interrupt request
  timer++;
}
```

Internal timer implementations – millis()

- Measuring a time period using millis, is simply a matter of comparing current time to the time value stored in a variable. As you go round a loop you continuously perform a simple bit of maths: millis() - stored_time
- This gives you the elapsed time in milliseconds
- The millis() function is driven by a millisecond timer interrupt that increments an unsigned long every time it activates and just returns the value of that variable
- This is implemented internally using timer0 (64 bit prescaler)

```
LOOP
...
// An event happens
if (event==true) stored_time = millis();
...
elapsed_time = millis() - stored_time;
...
END_LOOP
```

Internal timer implementations – millis()

- The millis value is stored in an unsigned int of 32 bit
- Thus, it can store a maximum value of 4,294,967,296
- This translates to roughly 49.71 days before it overflows back to 0
- It is always prudent to define the variable that is supposed to contain millis values as unit32_t
- millis can also be used to create non-blocking delays

```
void setup (void) {  
}  
  
#define LED 13  
  
void loop(void){  
    static uint8_t tog=0;  
    static uint32_t oldtime=millis();  
  
    if ( (millis()-oldtime) > 500) {  
        oldtime = millis();  
  
        tog = ~tog; // Invert  
        if (tog) digitalWrite(LED,HIGH); else digitalWrite(LED,LOW);  
    }  
}
```

Watchdog timers

- The ATmega328P has a Watchdog Timer which is a useful feature to help the system recover from scenarios where the system hangs or freezes due to errors in the code written or due to conditions that may arise due to hardware issues
- The watchdog timer uses an internal 128kHz clock source
- When enabled, it starts counting from 0 to a value selected by the user
- If the watchdog timer is not reset by the time it reaches the user selected value, the watchdog resets the microcontroller
- This hard reset is used to make sure the controller does not get stuck in infinite loops
 - Particularly helpful in IoT applications where wifi and lorawan connectivity is needed

Watchdog timers

- Configure the watchdog timer through one of the registers known as WDTCSR
 - WDIE - Enables Interrupts. This will give you the chance to include one last dying wish before the board is reset. This is a great way of performing interrupts on a regular interval should the watchdog be configured to not reset on time-out
 - WDE - Enables system reset on time-out. Whenever the Watchdog timer times out the microcontroller will be reset. This is probably what you were all looking for. Set this to '1' to activate
 - WDPO/WDP1/WDP2/WDP3 - These four bits determine how long the timer will count for before resetting

Bit	Name
7	WDIF
6	WDIE
5	WDP3
4	WDCE
3	WDE
2	WDP2
1	WDP1
0	WDPO

WDP 3	WDP 2	WDP 1	WDP 0	Time-out (ms)
0	0	0	0	16
0	0	0	1	32
0	0	1	0	64
0	0	1	1	125
0	1	0	0	250
0	1	0	1	500
0	1	1	0	1000
0	1	1	1	2000
1	0	0	0	4000
1	0	0	1	8000

Watchdog timers

- To make life easy, Arduino implementations have some functions to configure the WDTCSR

- We have:

`wdt_enable(WDTO_8S)`

options: `WDTO_1S`,

`WDTO_2S`, `WDTO_4S`,

`WDTO_8S`

`wdt_disable()`

`wdt_reset()`

```
#include<avr/wdt.h> /* Header for watchdog timers in AVR */

void setup() {
    Serial.begin(9600); /* Define baud rate for serial communication */
    Serial.println("Watchdog Demo Starting");
    pinMode(13, OUTPUT);
    wdt_enable(WDTO_2S); /* Enable the watchdog with a timeout of 2 seconds */
}

void loop() {
    for(int i = 0; i<20; i++) /* Blink LED for some time */
    {
        digitalWrite(13, HIGH);
        delay(100);
        digitalWrite(13, LOW);
        delay(100);
        wdt_reset(); /* Reset the watchdog */
    }
    while(1); /* Infinite loop. Will cause watchdog timeout and system reset. */
}
```

Watchdog timers

- To let the watchdog timer know that everything is running ok and that it needn't panic or take any action you are going to have to keep resetting the timer within your main loop
- This is done by periodically entering in:

 wdt_reset();

```
#include<avr/wdt.h> /* Header for watchdog timers in AVR */

void setup() {
    Serial.begin(9600); /* Define baud rate for serial communication */
    Serial.println("Watchdog Demo Starting");
    pinMode(13, OUTPUT);
    wdt_enable(WDTO_2S); /* Enable the watchdog with a timeout of 2 seconds */
}

void loop() {
    for(int i = 0; i<20; i++) /* Blink LED for some time */
    {
        digitalWrite(13, HIGH);
        delay(100);
        digitalWrite(13, LOW);
        delay(100);
        wdt_reset(); /* Reset the watchdog */
    }
    while(1); /* Infinite loop. Will cause watchdog timeout and system reset. */
}
```

Summary

- We discussed many important features of hardware design using microcontrollers
- The specific register names and bit positions will be different for different controllers. Timer sizes and clock frequencies will be different
- However, the basic concepts of interrupts, timers, watchdogs are generally the same
- These are very helpful features in advanced IoT applications

EC5.204 Communications & Controls in IoT

Networking Basics

Instructors: Sachin Chaudhari

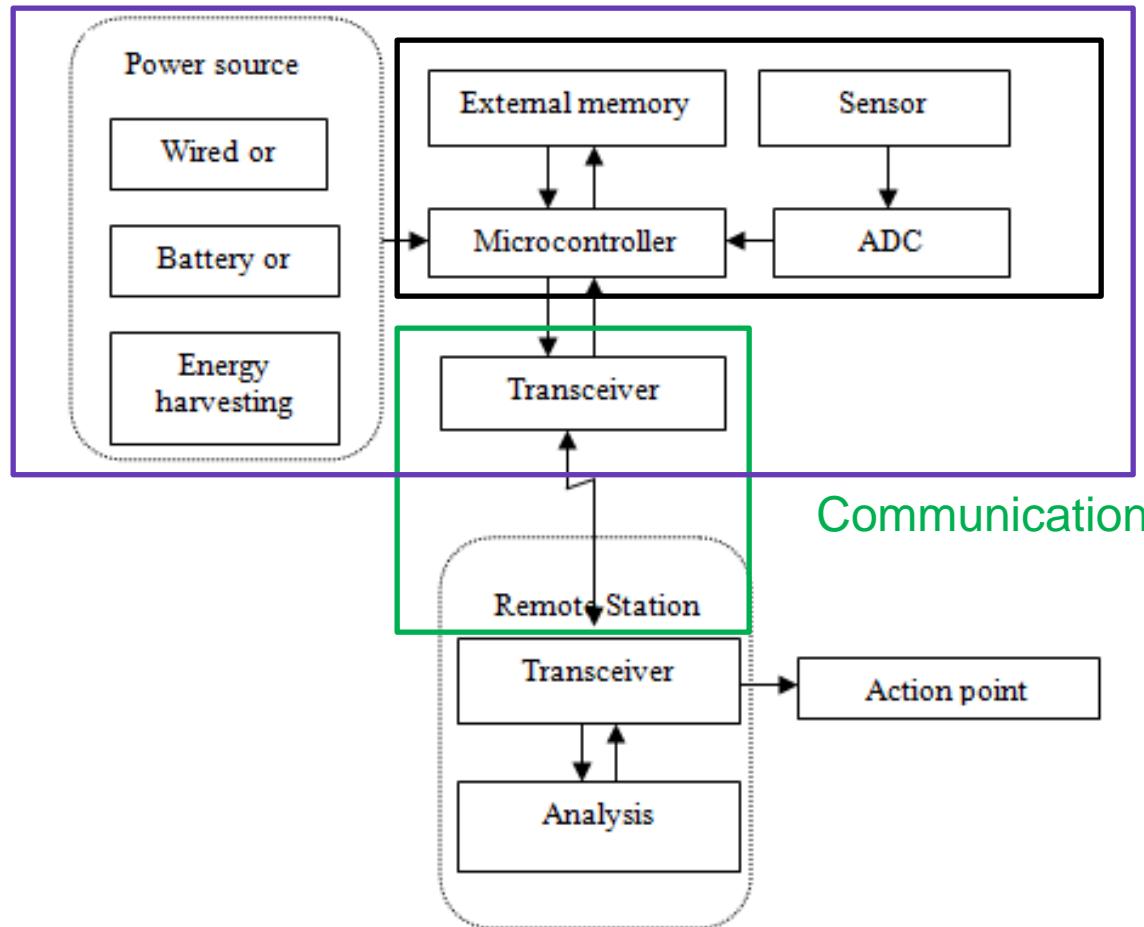
Jan. 23, 2023



**INTERNATIONAL INSTITUTE OF
INFORMATION TECHNOLOGY**

HYDERABAD

Block Diagram of Sensor Node

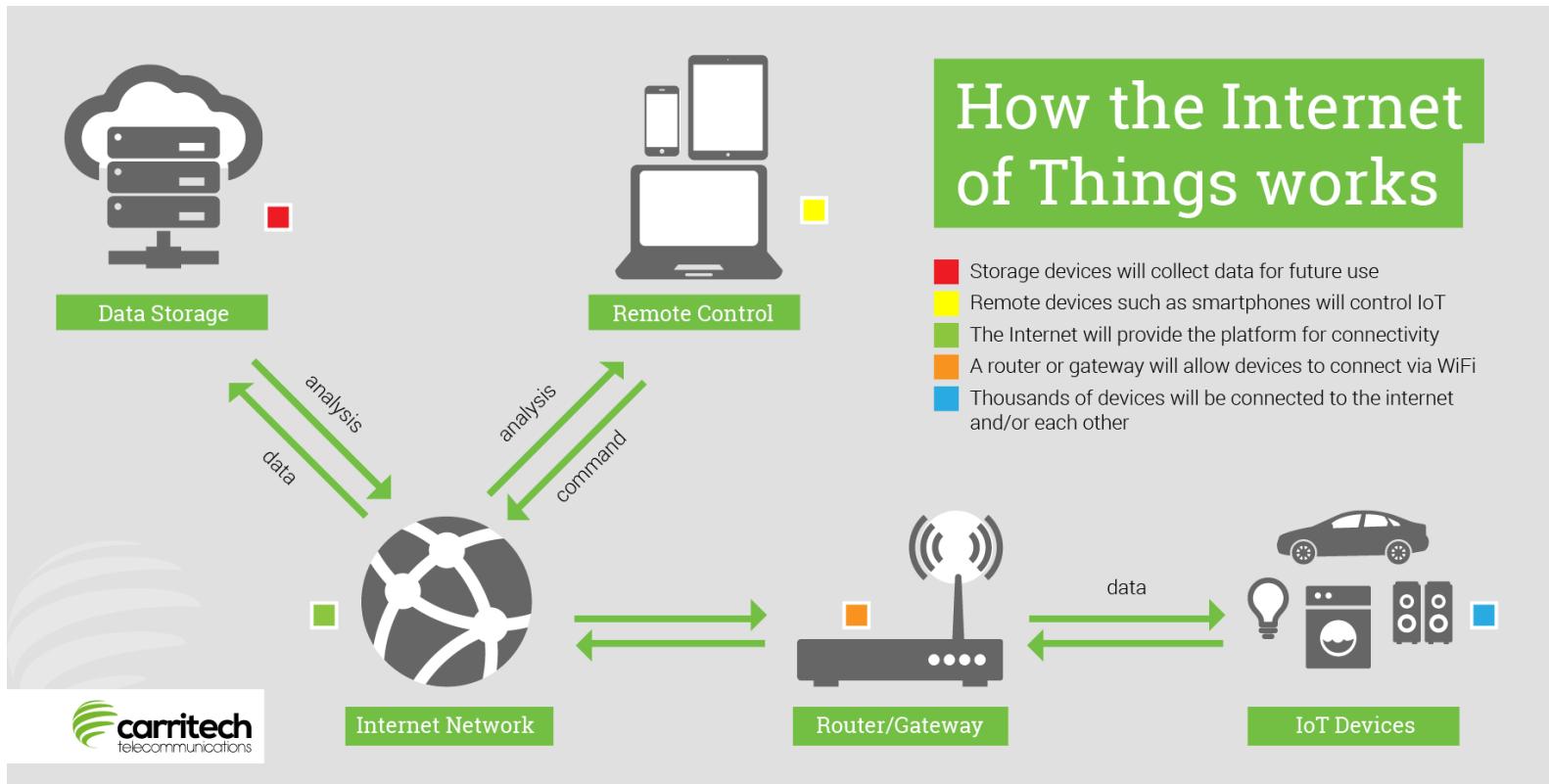


Till Now in Course

https://www.researchgate.net/publication/269310409_A_review_of_sensor_networks_Technologies_and_applications/figures?lo=1&utm_source=google&utm_medium=organic

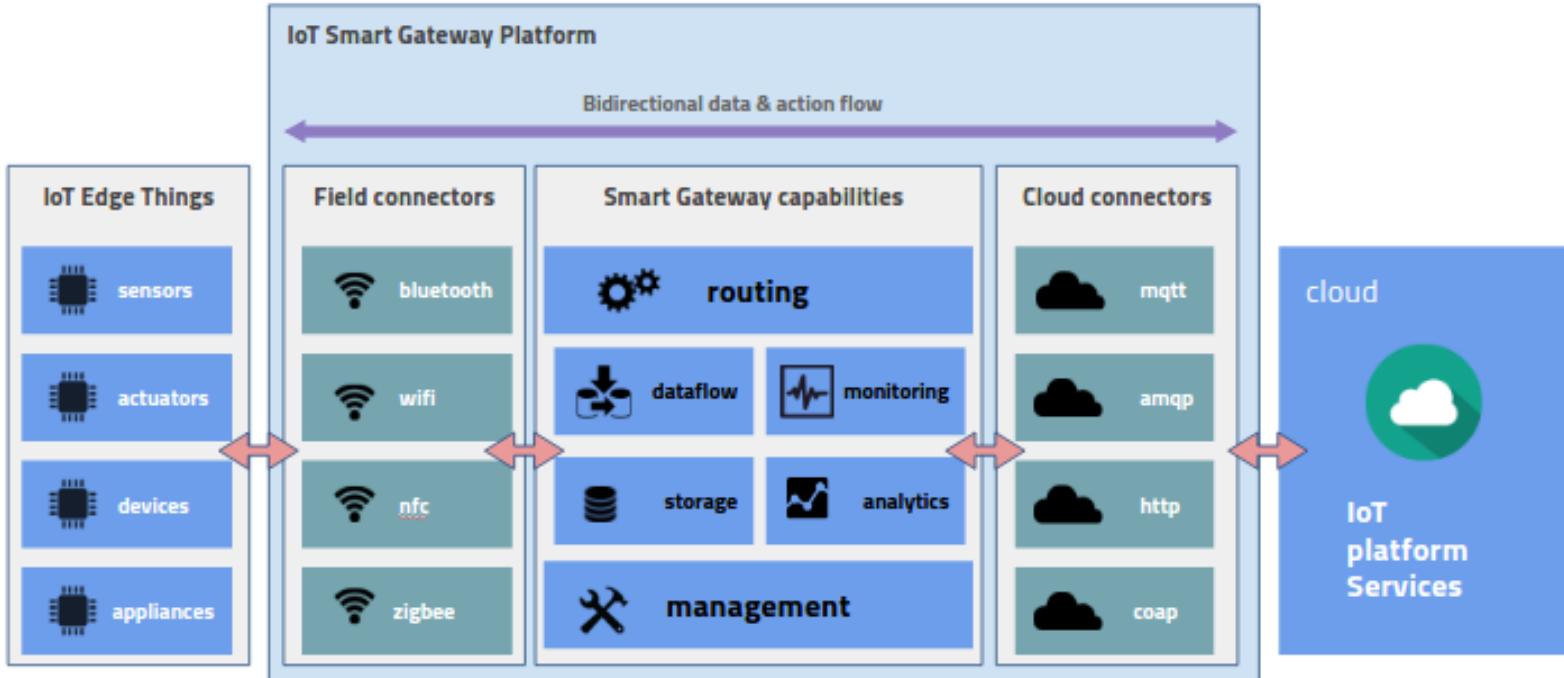
[Leverage: Intro to IoT] To be smart, a thing doesn't need to have super storage or a supercomputer inside of it . All a thing has to do is connect to super storage or to a super computer.

How does IoT work?



Picture Credit: <http://www.carritech.com/news/internet-of-things/>

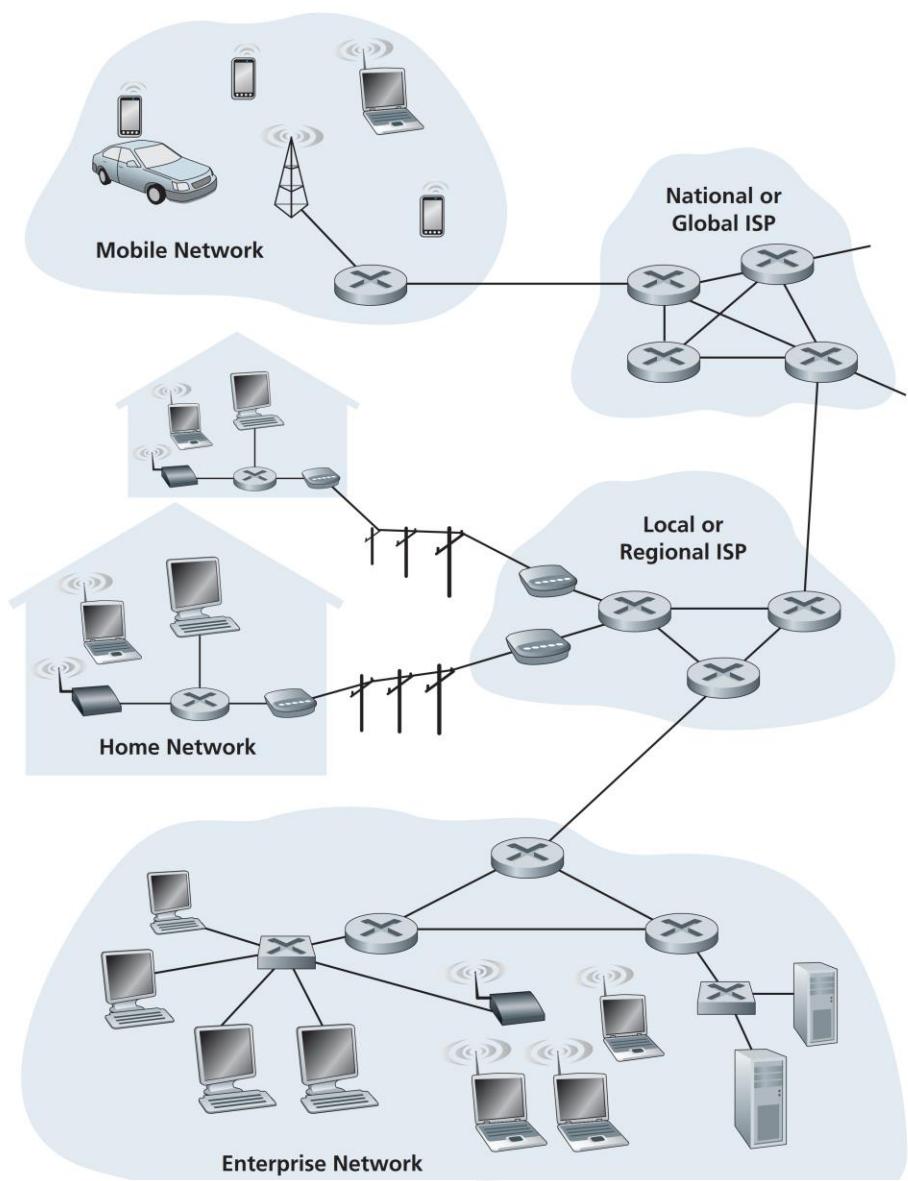
IoT Network Setup



Picture Credit: <https://www.iotcentral.io/blog/the-iot-architecture-at-the-edge>

Main Reference

- [Kurose2012] J. Kurose and K. Ross, *Computer Networking*, Pearson, 2012.



Some Pieces of Internet

The Internet is a computer network that interconnects hundreds of millions of computing devices throughout the world

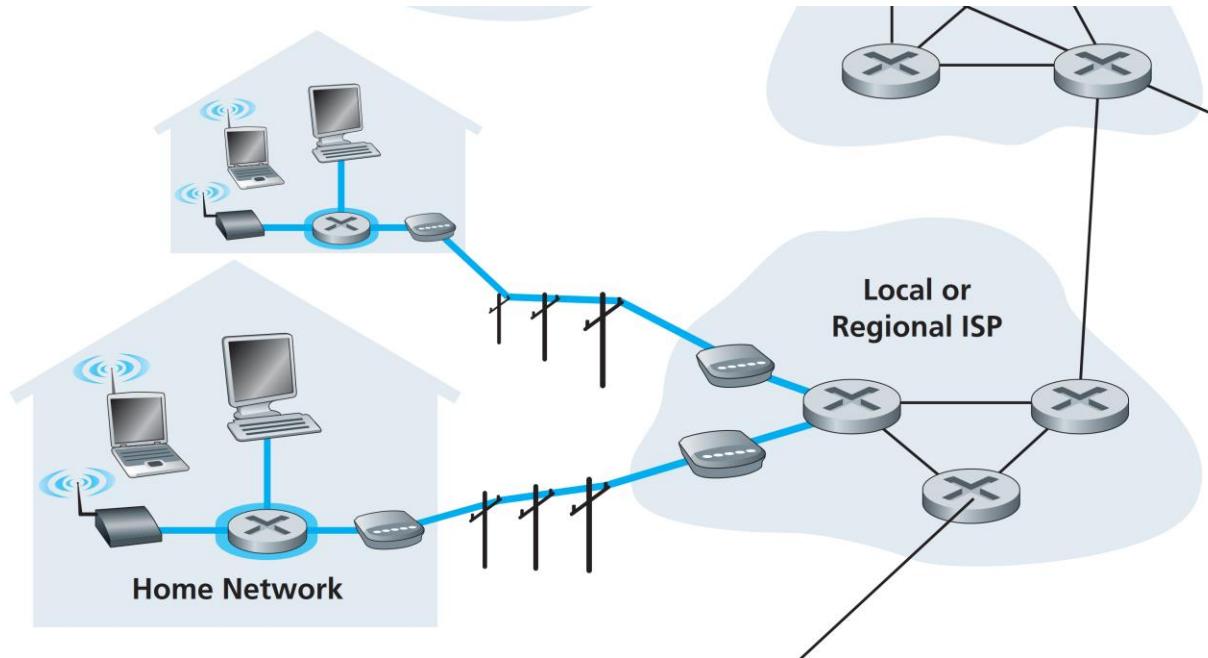
Key:



Few Internet Terminologies

- Host or end-devices
 - computing devices connected to the internet
- Communication links
 - Connect the different elements in the network
- Packet switches
 - takes a packet arriving on one of its incoming communication links and forwards that packet on one of its outgoing communication links
 - Most prominent
 - Routers: Network core
 - Link-layer switches: access network
- Route or Path
 - The sequence of communication links and packet switches traversed by a packet from the sending end system to the receiving end system
- Internet service providers (ISPs)

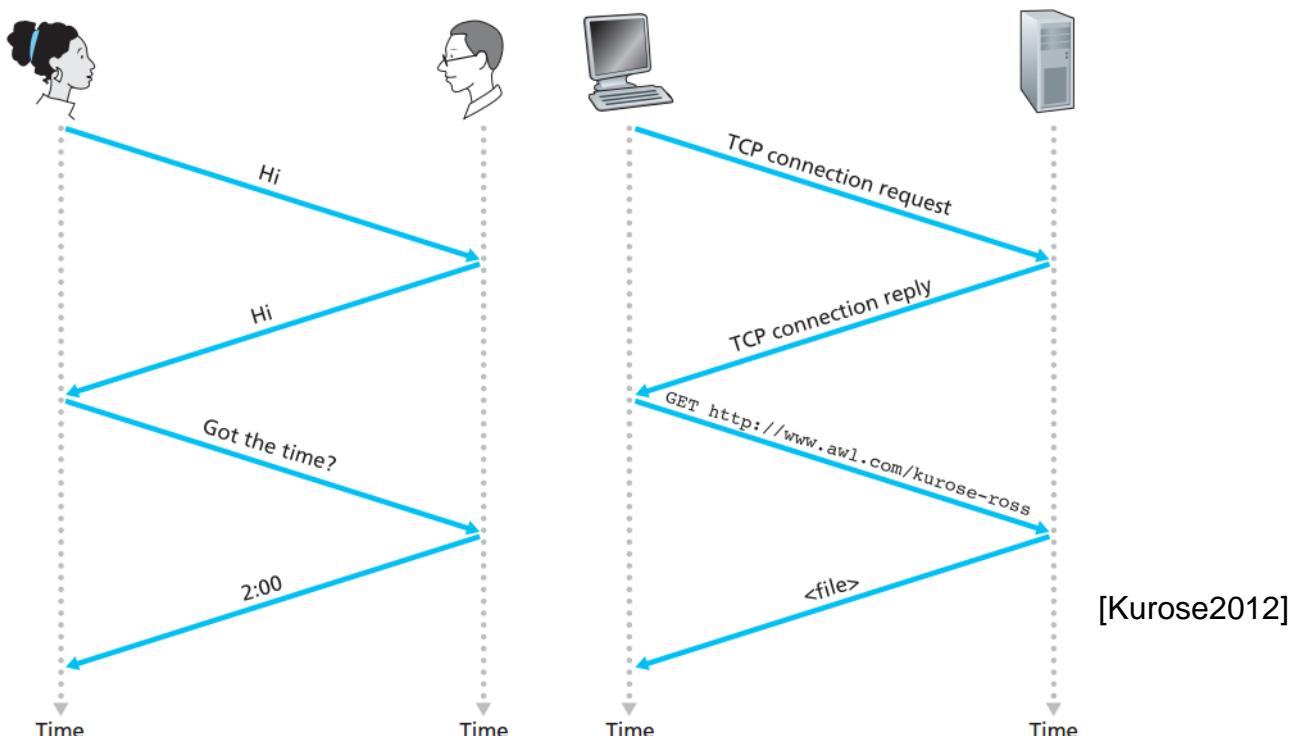
Example of route



Communication Protocol Basics

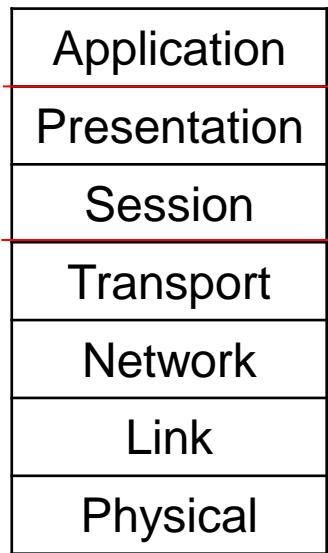
Protocol

- A protocol defines the **format** and the **order** of messages exchanged between two or more communicating entities as well as the actions taken on the transmission and/or receipt of a message or other event. [Kurose2012]

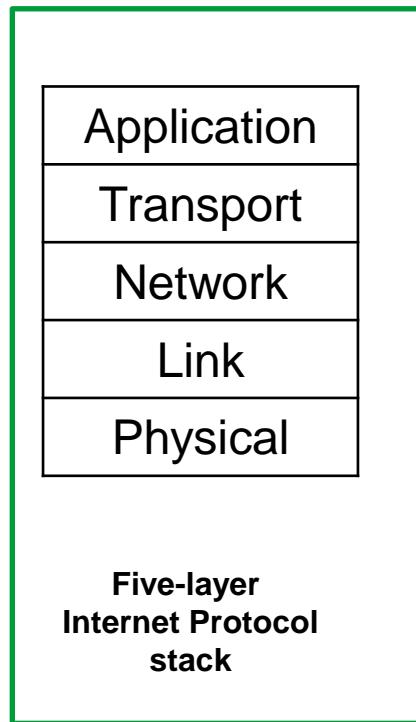


Analogy of human protocol and a computer network protocol

Internet protocol stack and OSI model



Seven-layer
Open Systems Interconnection
(OSI) model



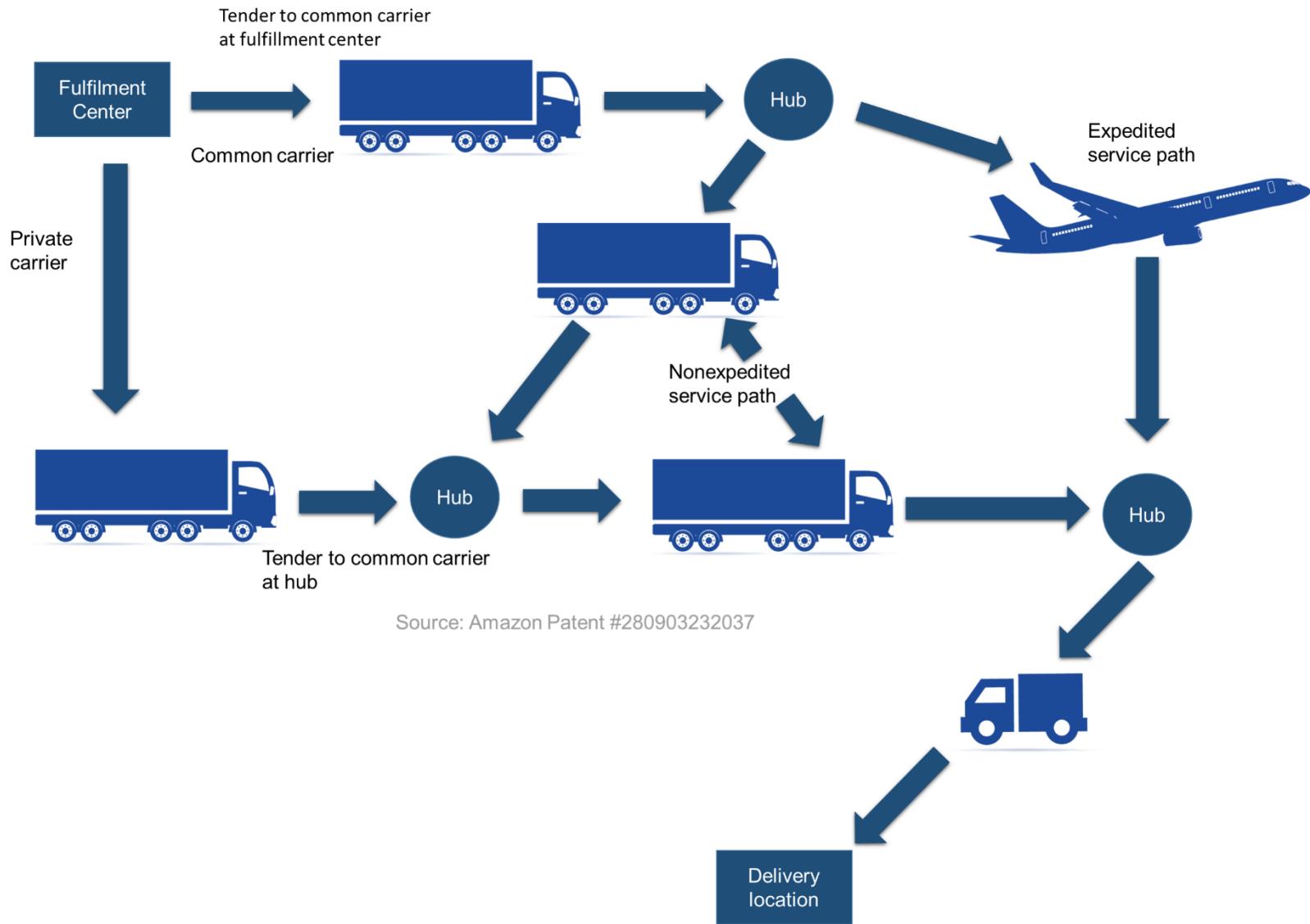
Protocols Layers and Their Service Models

- A layered architecture allows us to discuss a well-defined, specific part of a large and complex system.
- Provides modularity, making it much easier to change the implementation of the service provided by the layer.
- As long as the layer provides the same service to the layer above it and uses the same services from the layer below it, the remainder of the system remains unchanged when a layer's implementation is changed.

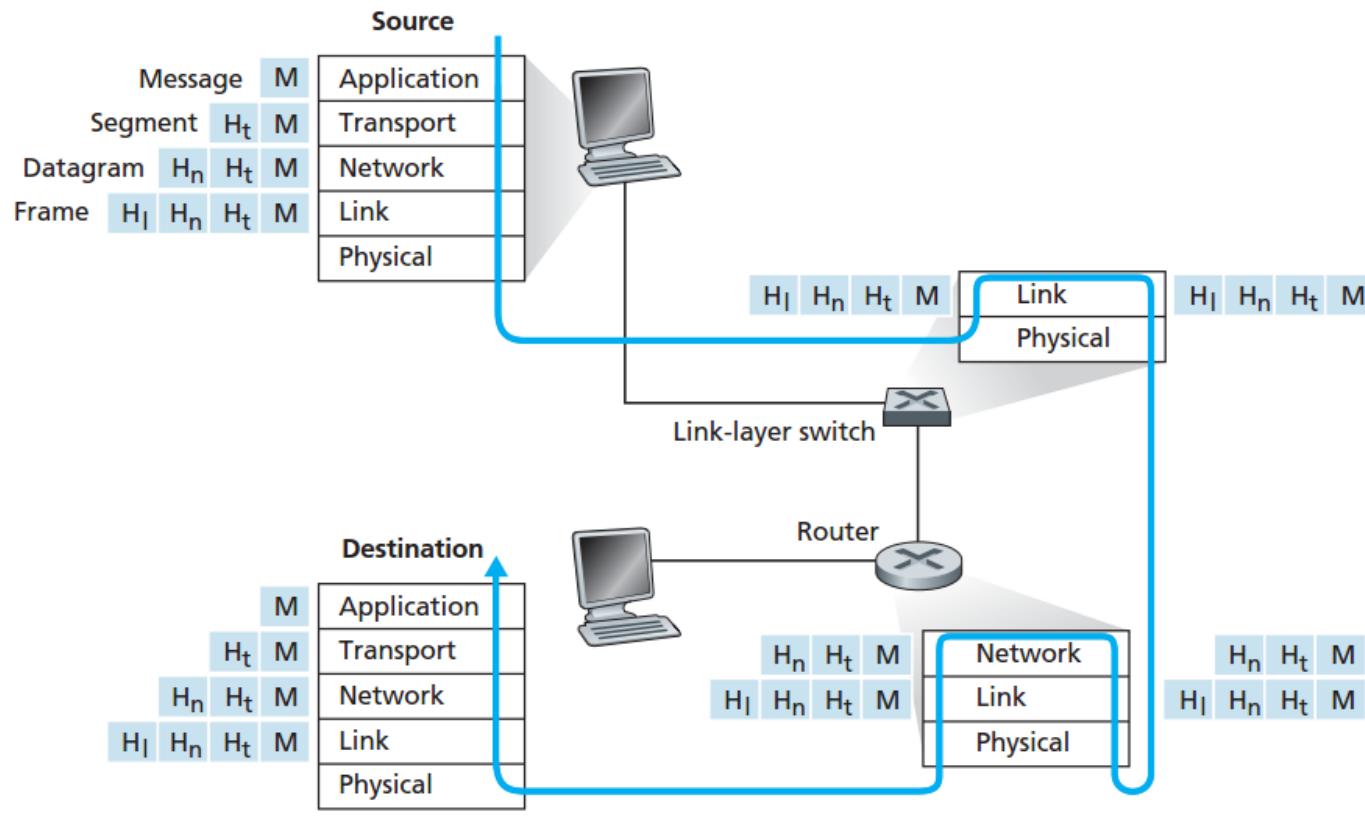
Internet protocol stack: Toy Example

- Sending a courier from company branch in Hyderabad to company branch in New York
 - Application Layer: Individuals giving parcels
 - Transport Layer: office boy or admin assistant
 - Network Layer: Speed post/ Blue Dart (representative)
 - Link Layer: Different drivers (and vehicles)
 - Physical Layer: Road/Air/Water

Analogy: e-Commerce supply chain



Encapsulation of data across layers



[Kurose2012]

Questions?

Internet Protocol Stack: Application Layer

- The application layer is where **network applications** and their application-layer protocols reside
- Example of network applications: www, file sharing, text chat, electronic commerce, instant messaging, video chat
- The *Application layer* provides applications the ability to access the services of the other layers and defines the protocols that applications use to exchange data
- Many services: file transfer, web surfing, web chat, email clients, virtual terminals, various file and data operations
- Protocol Examples: HTTP, SMTP, FTP, DNS, MQTT, TelNet

Communication between Applications

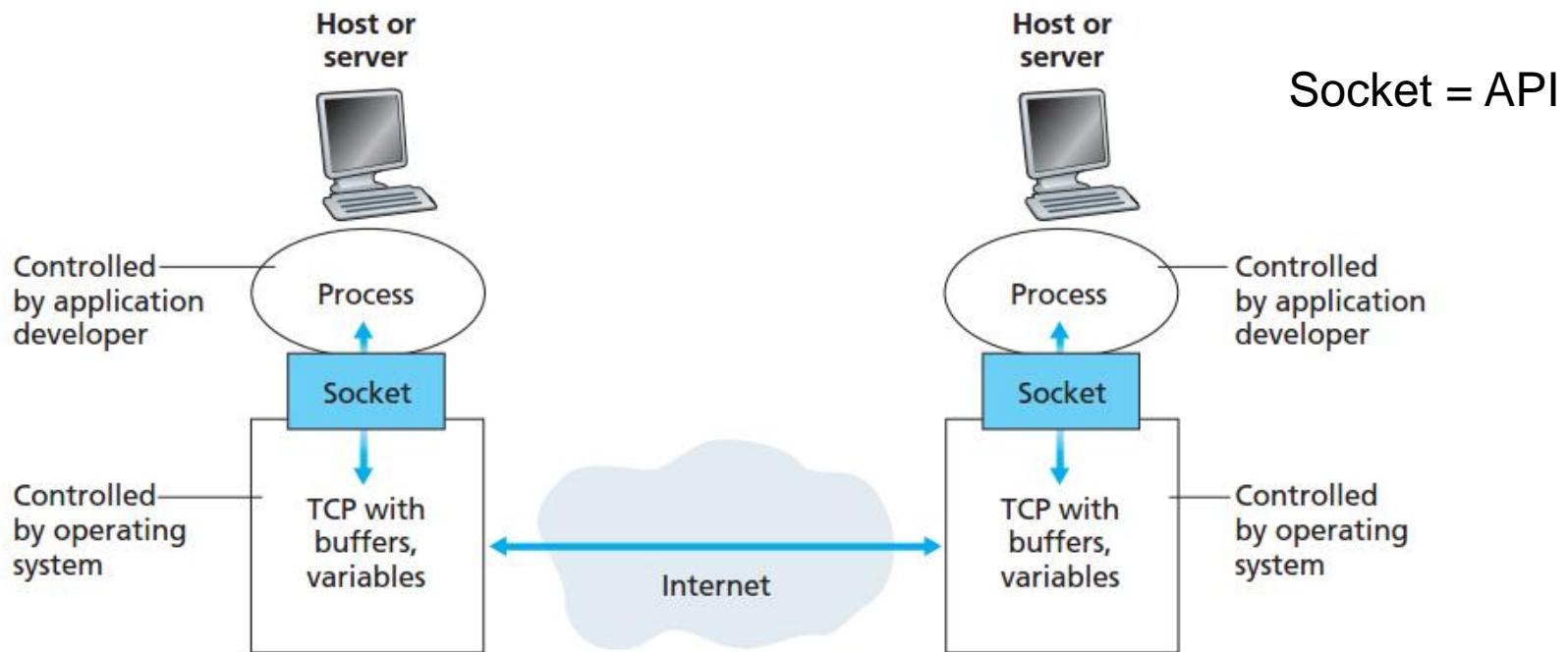


Figure 2.3 ♦ Application processes, sockets, and underlying transport protocol

- In network applications, programs/processes are running on different end-systems or hosts and communicating over host
- A process can be thought of as a program running within an end system

Application Layer: Examples

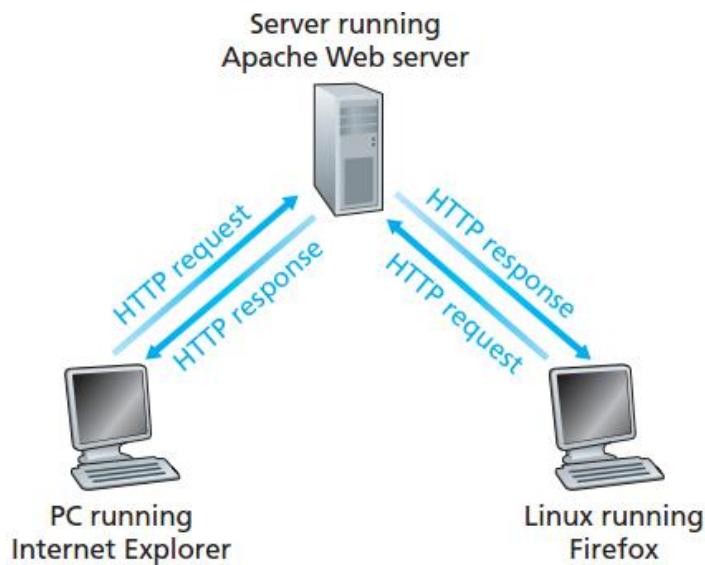


Figure 2.6 ♦ HTTP request-response behavior

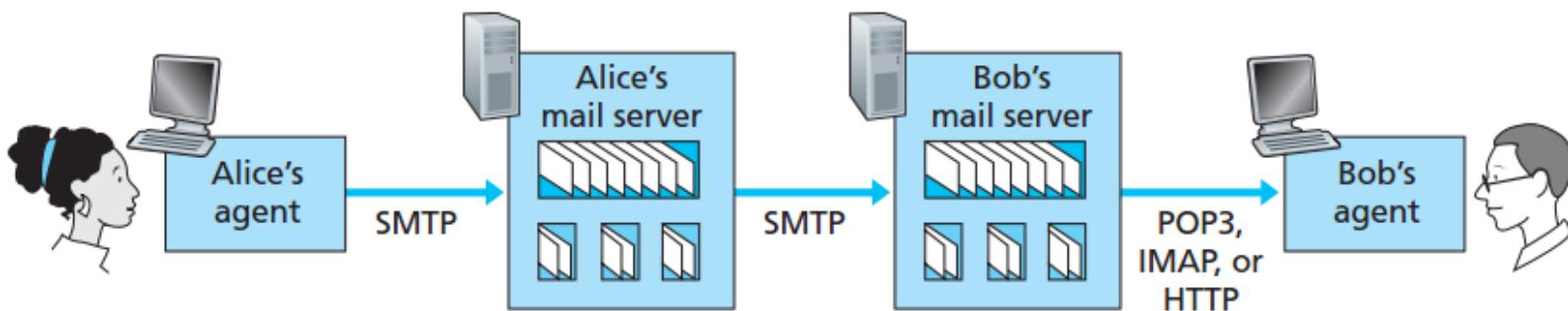


Figure 2.18 ♦ E-mail protocols and their communicating entities

Application Layer

Application layer protocols defines:

- Types of message exchanged, for example, request and response messages
- Syntax of the various message types, such as the fields in the message and how the fields are delineated
- The semantics of the field, that is, the meaning of the information
- Rules for determining when and how a process sends messages and responds to the messages

Example: HTTP request message

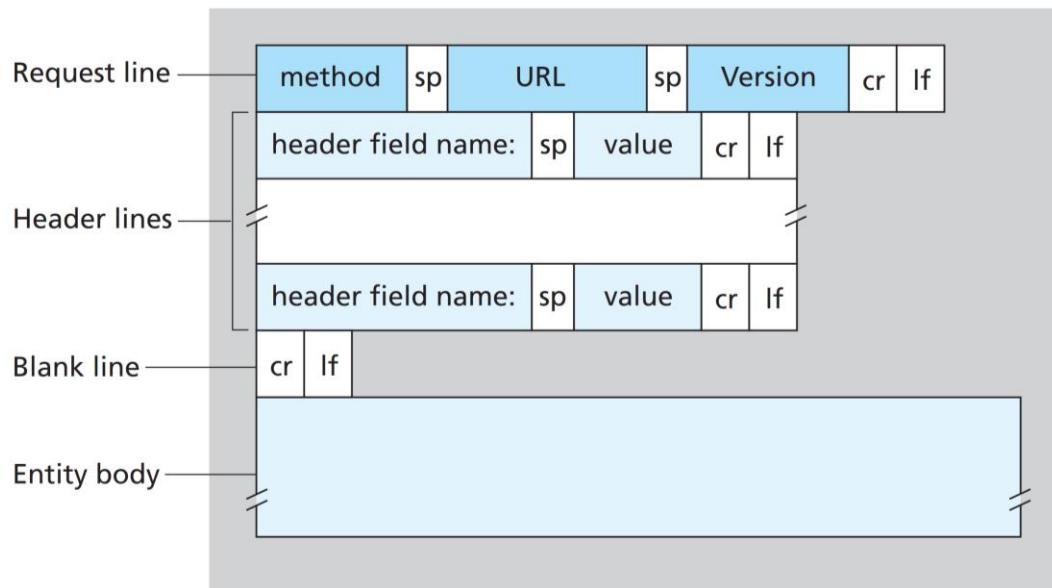


Figure 2.8 ♦ General format of an HTTP request message

www.somechool.edu/page.html

request line
(GET, POST,
HEAD commands)

header lines

Carriage return,
line feed
indicates end
of message

extra carriage return, line feed

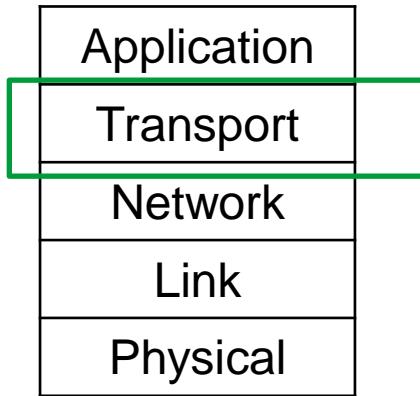
GET /somedir/page.html HTTP/1.1
Host: www.someschool.edu
User-agent: Mozilla/4.0
Connection: close
Accept-language:fr

Addressing Mechanism

- A host has one IP address
- How does the sending host identifies the receiving process running in the receiving host?
- Port Number:
 - Web server : port 80
 - SMTP : port 25

Questions?

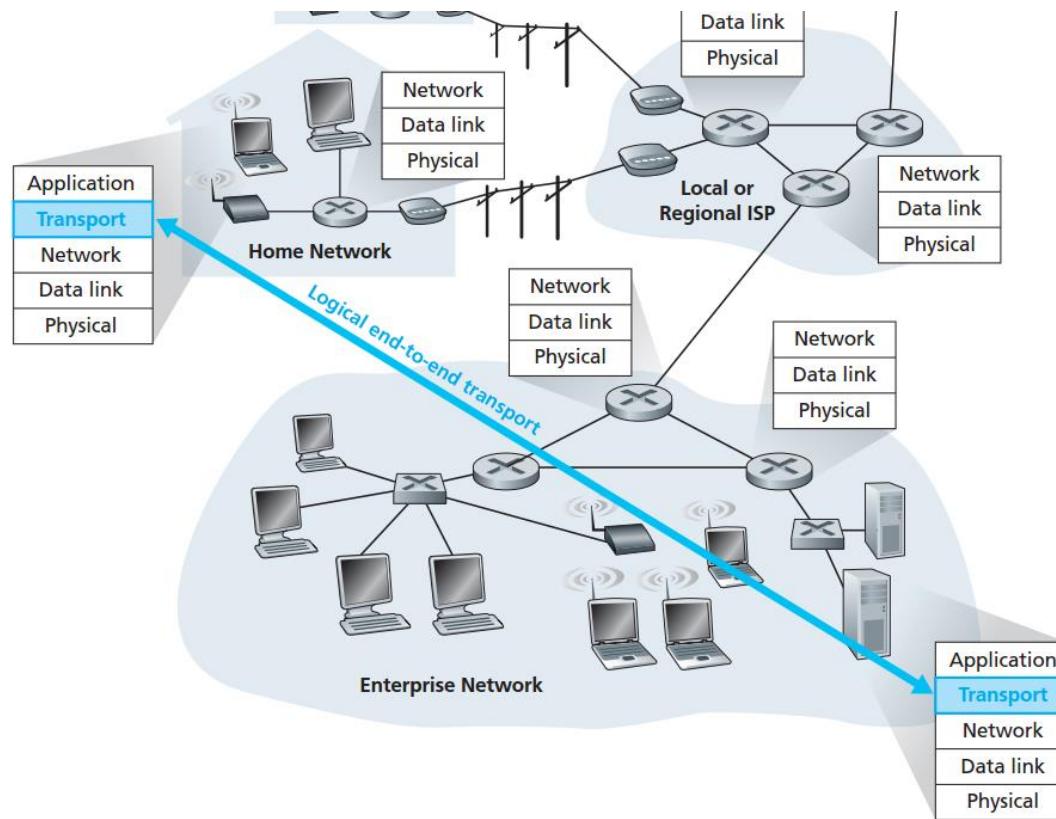
Internet protocol stack: *Transport Layer*



Five-layer
Internet Protocol
stack

Internet Protocol Stack: Transport Layer

- It provides logical communication between application processes running on different hosts



Internet Protocol Stack: *Transport Layer*

- It provides logical communication between application processes running on different hosts
- Provides two transport protocols
 - TCP (connection-oriented)
 - provides reliability
 - flow control
 - congestion control
 - Breaks long messages
 - Application layer protocols using TCP: SMTP, HTTP, FTP
 - UDP (connectionless)
 - No frills service
 - No reliability, no flow control, no congestion control
 - Application layer protocols: IP telephony or video (Youtube, Skype)
- Multiplexing and demultiplexing of data

Multiplexing and Demultiplexing

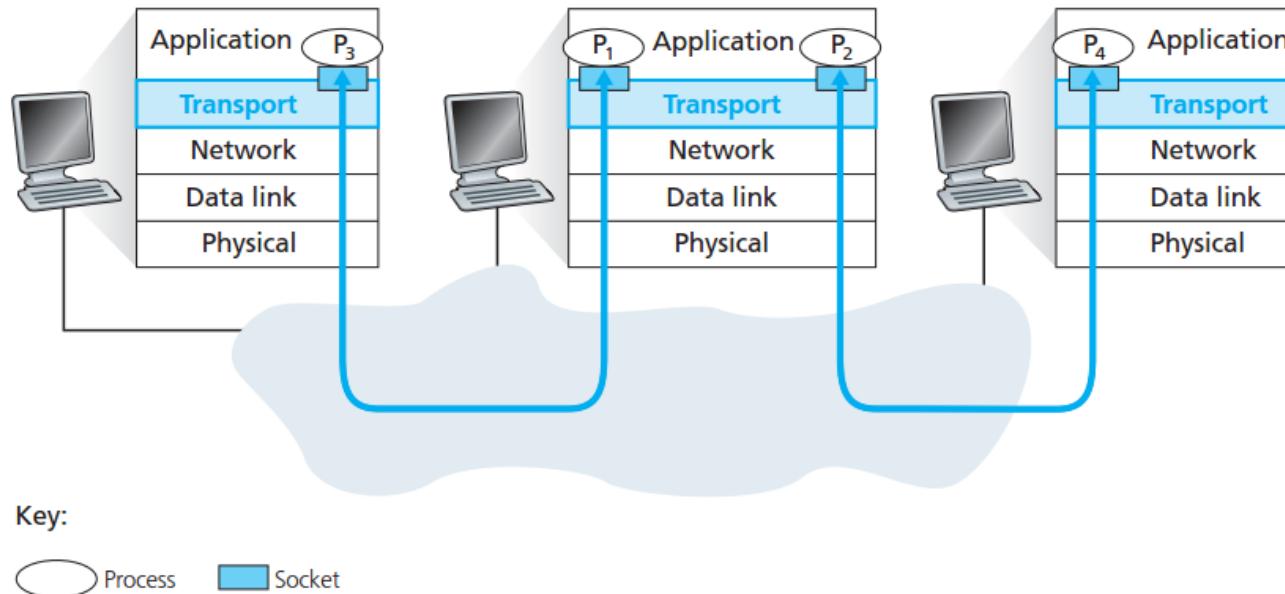


Figure 3.2 ♦ Transport-layer multiplexing and demultiplexing

TCP: Multiplexing and Demultiplexing

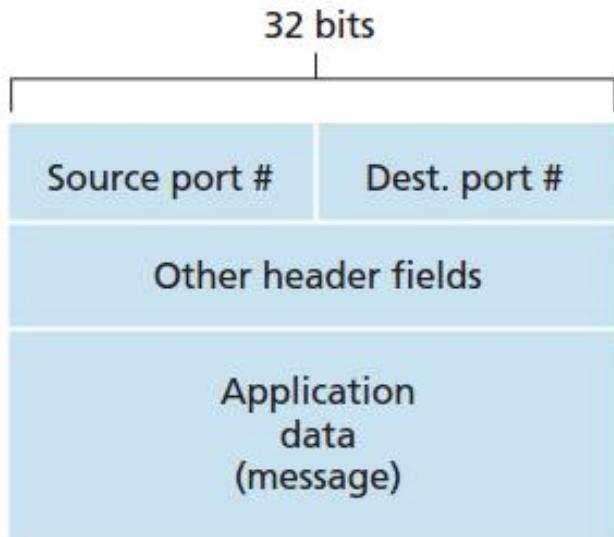


Figure 3.3 ♦ Source and destination port-number fields in a transport-layer segment

TCP: Use of IP and Port Addresses

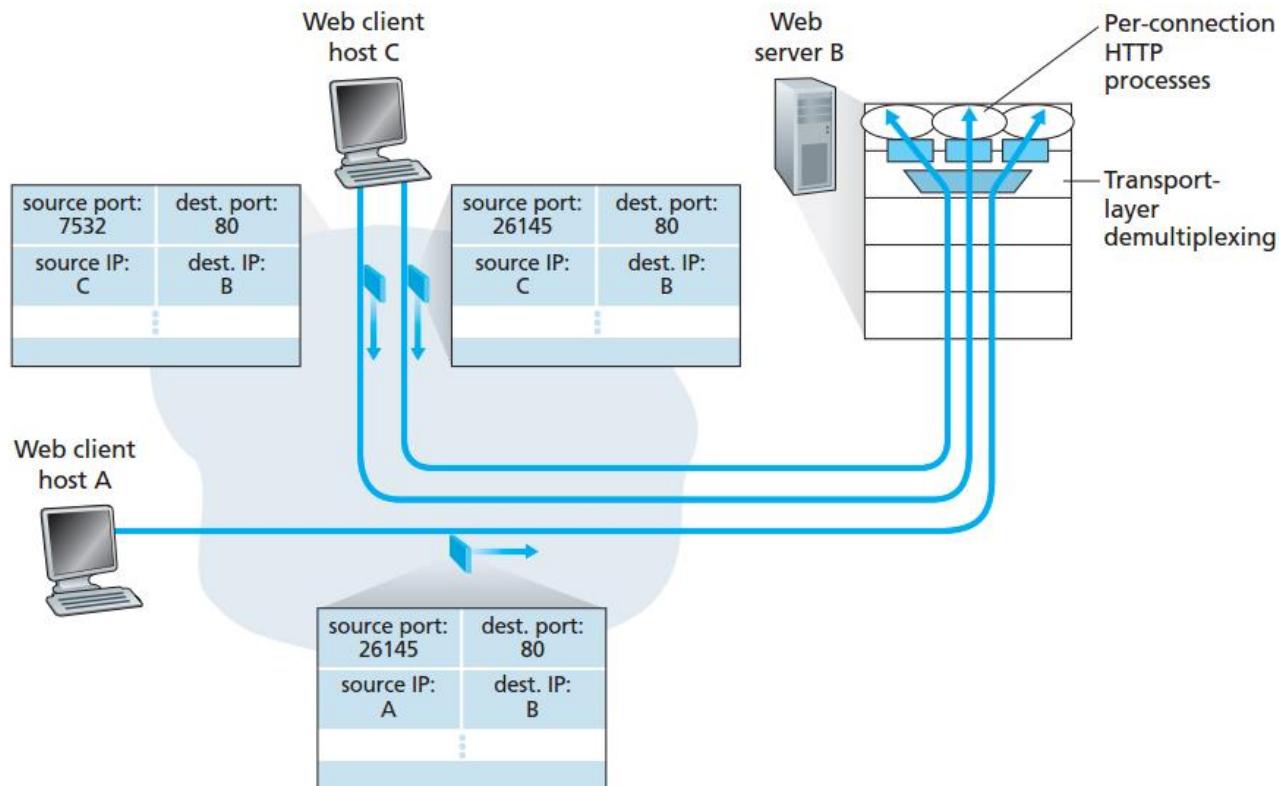


Figure 3.5 ♦ Two clients, using the same destination port number (80) to communicate with the same Web server application

TCP: Connection Oriented

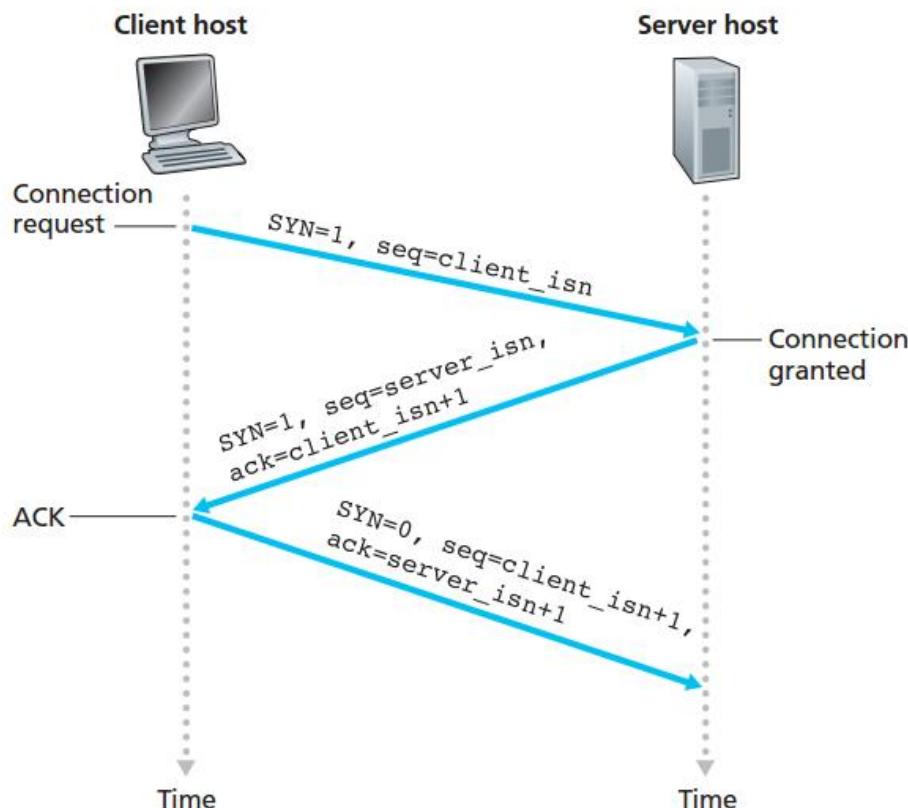


Figure 3.39 ♦ TCP three-way handshake: segment exchange

TCP: Reliable data transfer

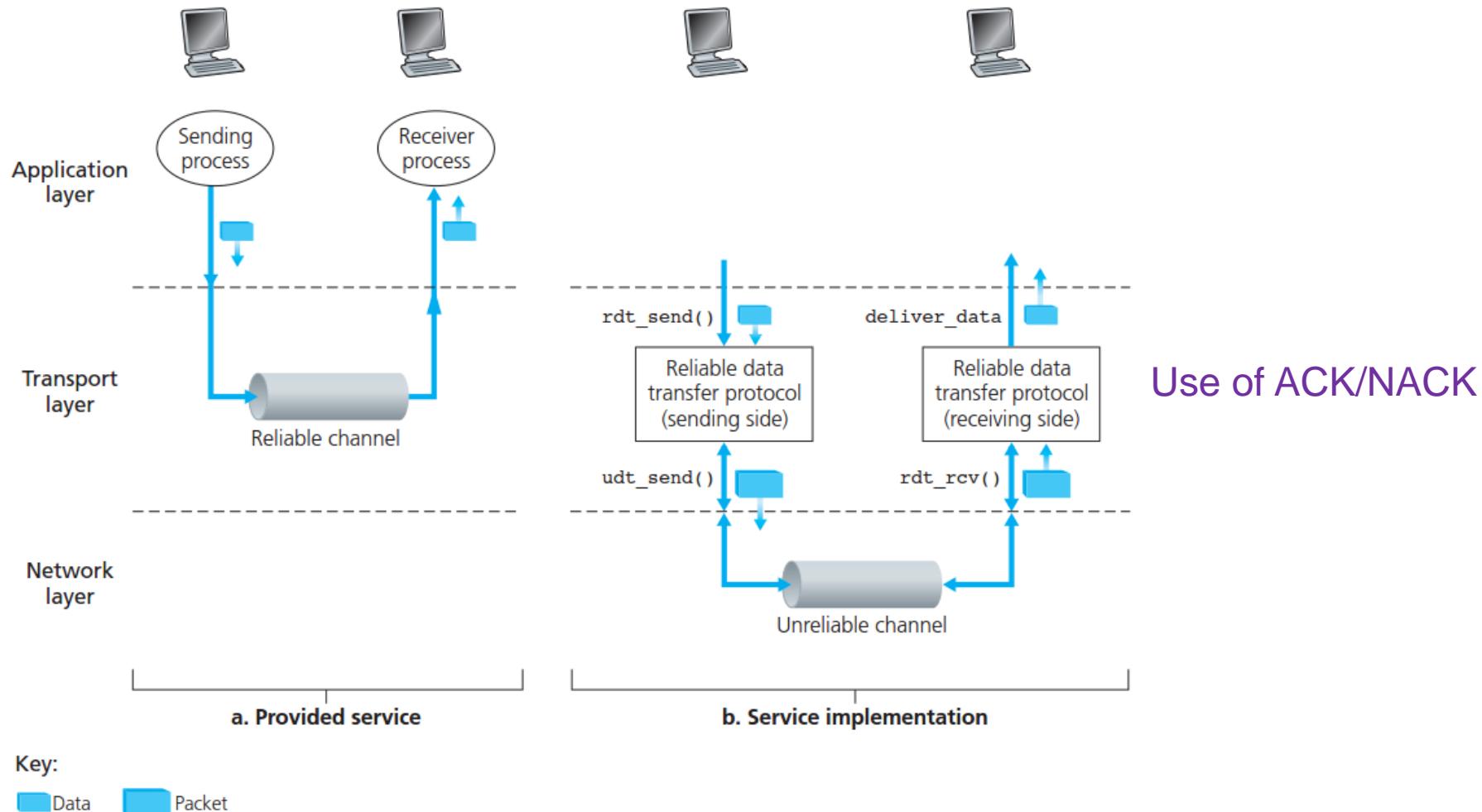


Figure 3.8 ♦ Reliable data transfer: Service model and service implementation

TCP: Use of ACK

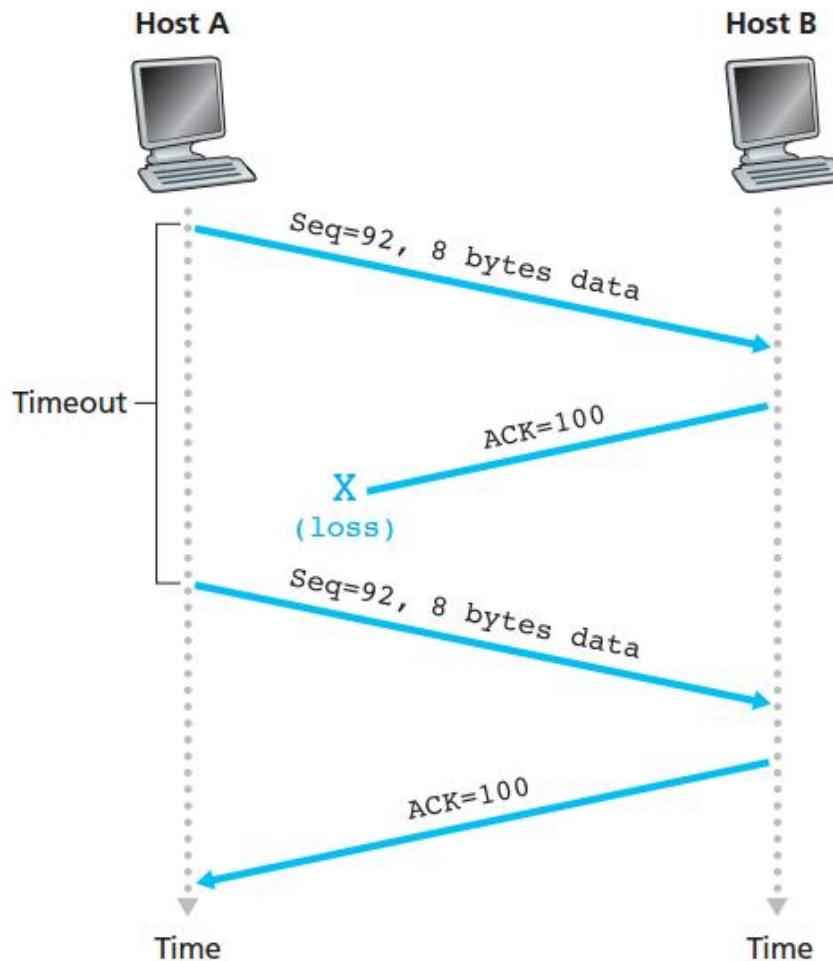


Figure 3.34 ♦ Retransmission due to a lost acknowledgment

TCP: Congestion Control

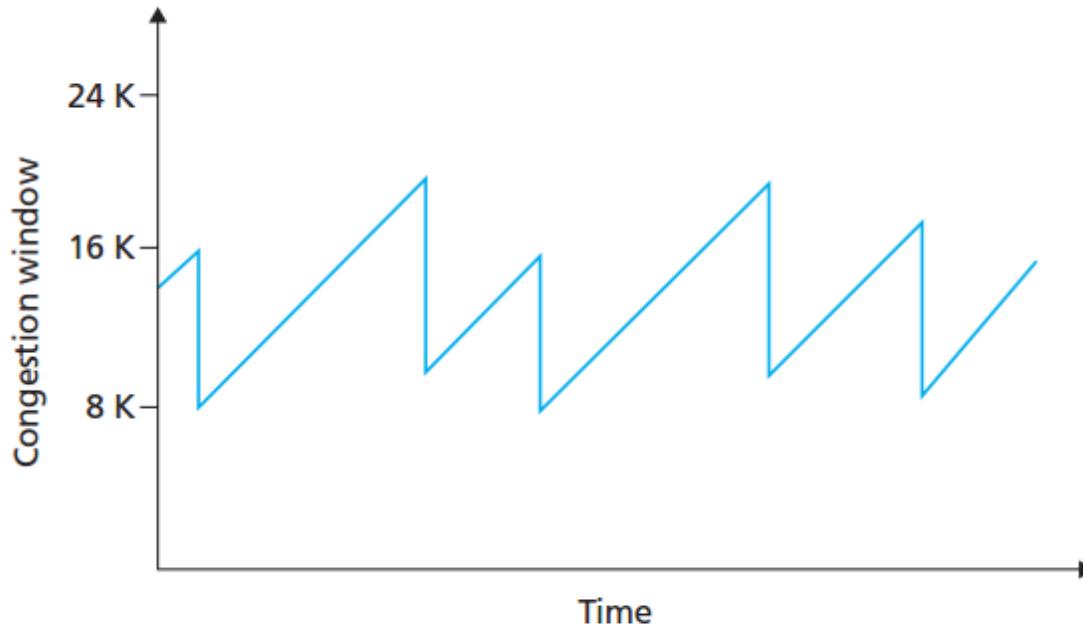


Figure 3.54 ♦ Additive-increase, multiplicative-decrease congestion control

TCP: Header

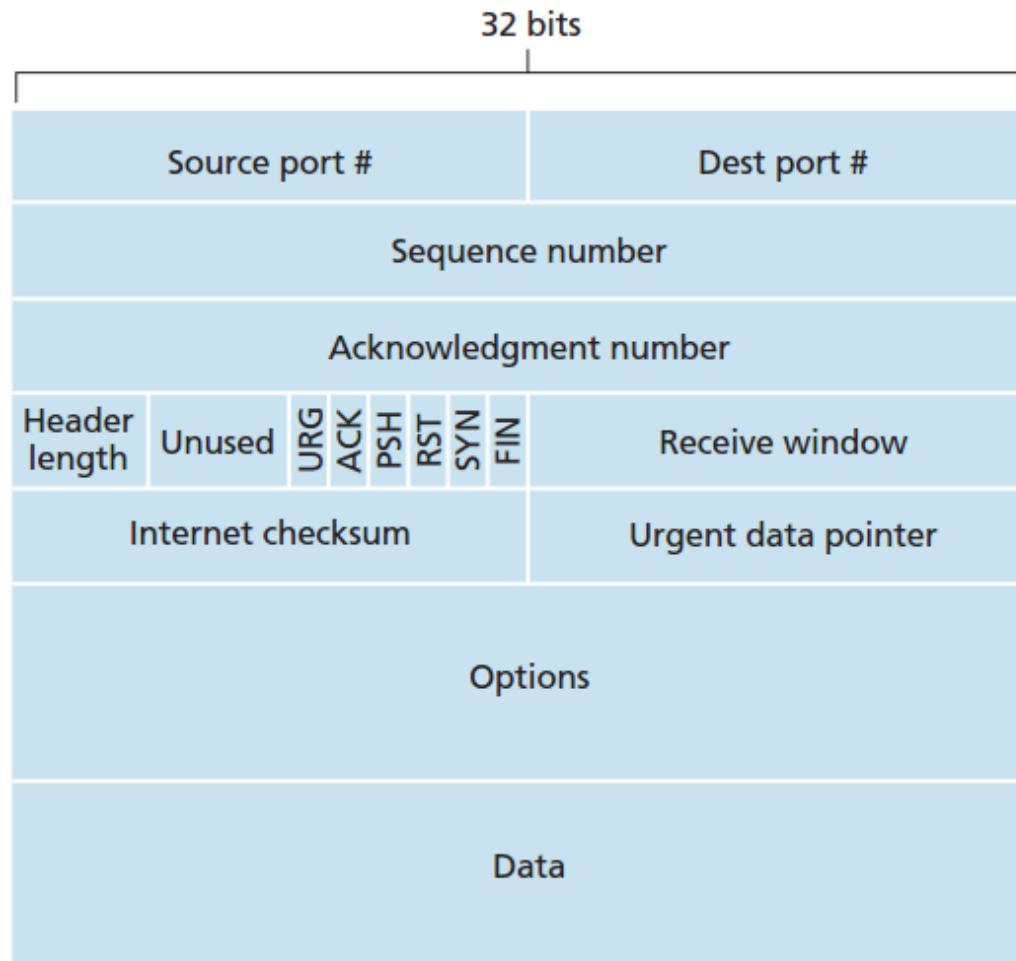


Figure 3.29 ♦ TCP segment structure

TCP: *Dividing the packets*

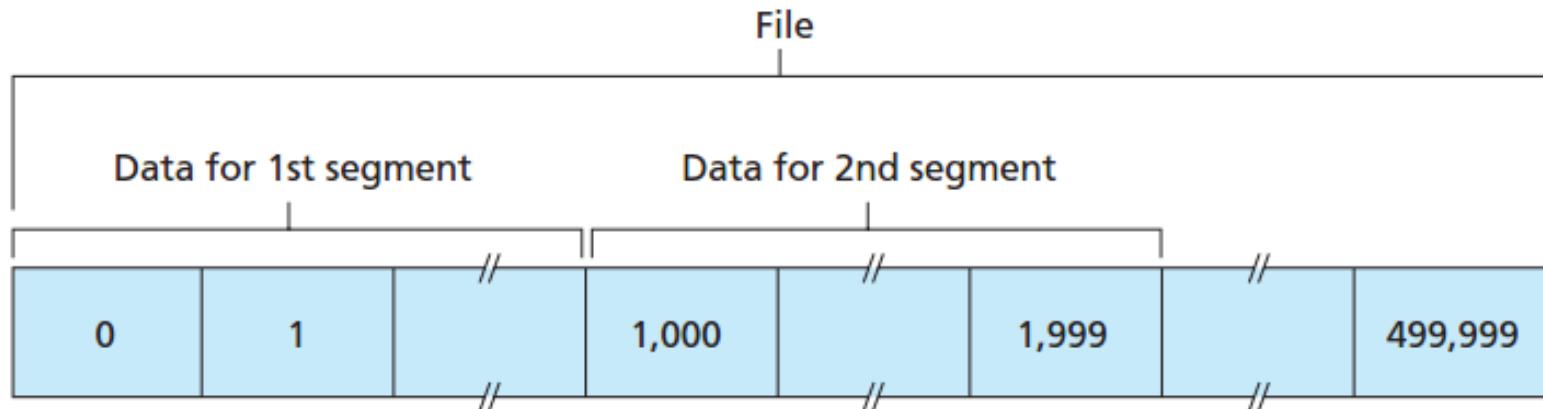
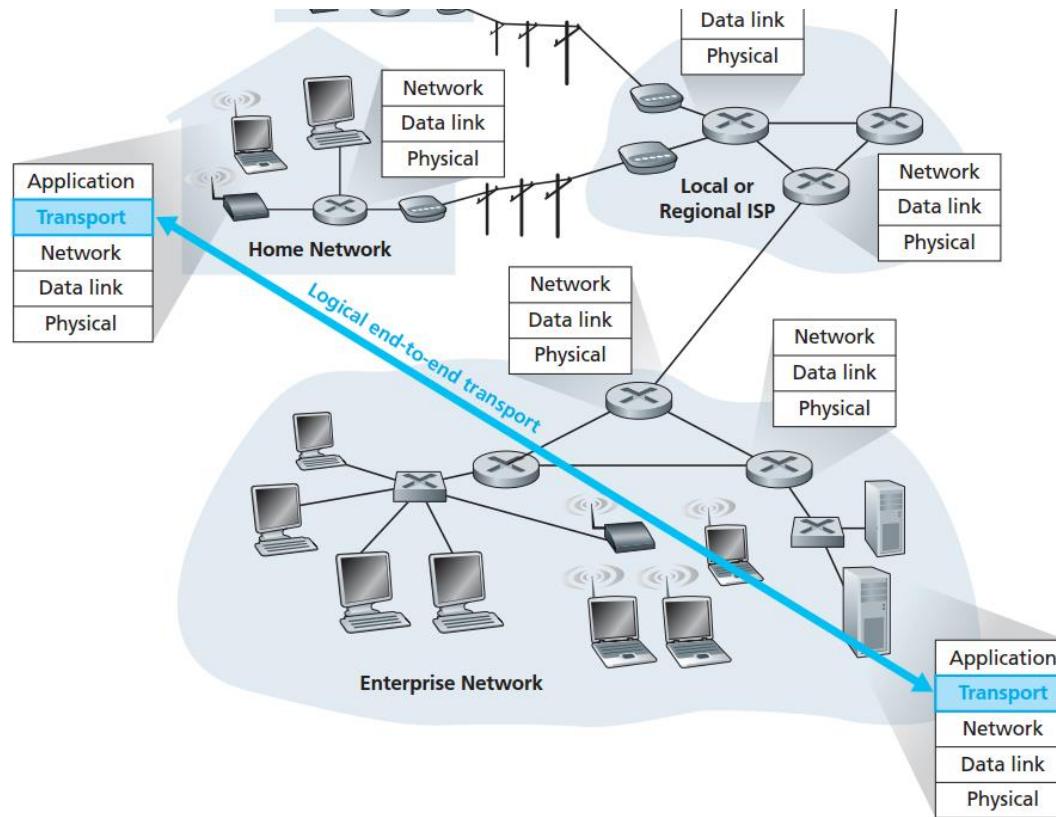


Figure 3.30 ♦ Dividing file data into TCP segments

Internet Protocol Stack: Transport Layer

- Resides at the end-host!

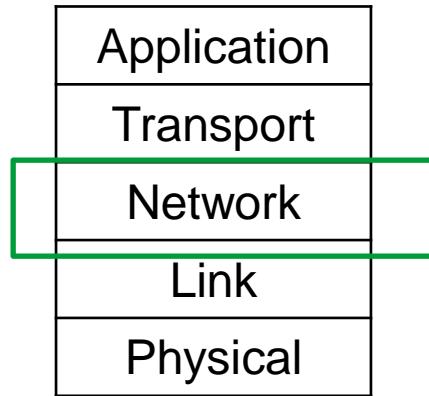


Questions?

Quick Quiz

- <https://forms.office.com/r/mLfgsuz7yY>
- True or False:
 - Route decider
- MCQ:
 - Network Application

Internet protocol stack: *Network Layer*



Five-layer
Internet Protocol
stack

Internet Protocol Stack: *Network Layer*

- Network Layer
 - Each host and router has network and below layers
 - Responsible for addressing, packaging, and routing functions
 - Includes Internet protocol (IP): defines the fields in the datagram as well as how the end systems and routers act on these fields
 - Includes routing protocols such as IGMP, OSPF, BGP
 - Includes other supplementary protocols such as internet message control protocol (ICMP) and address resolution protocol (ARP)
 - Uses IPv4 and IPv6 addresses

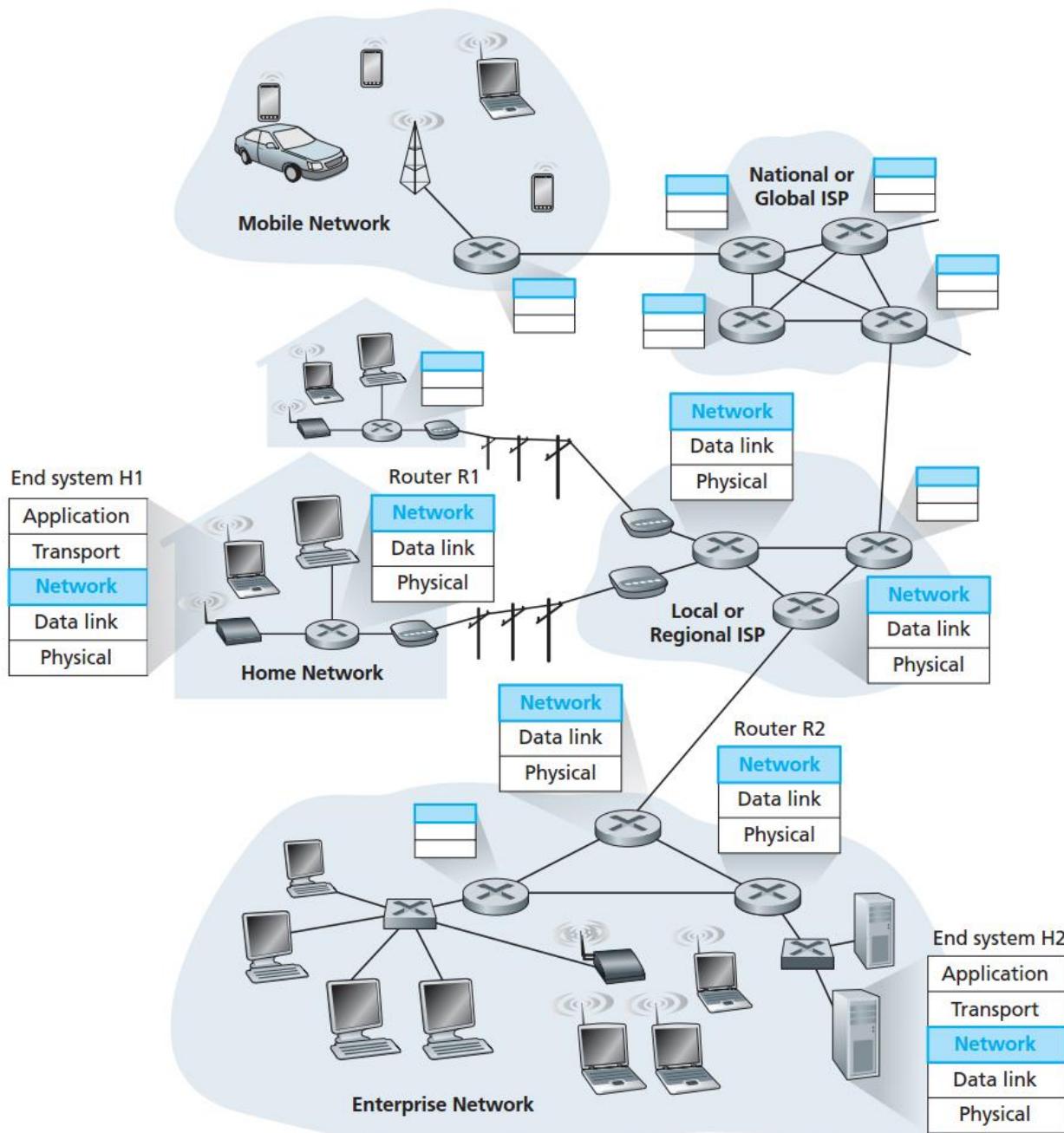


Figure 4.1 ♦ The network layer

Network Layer: Forwarding and Routing

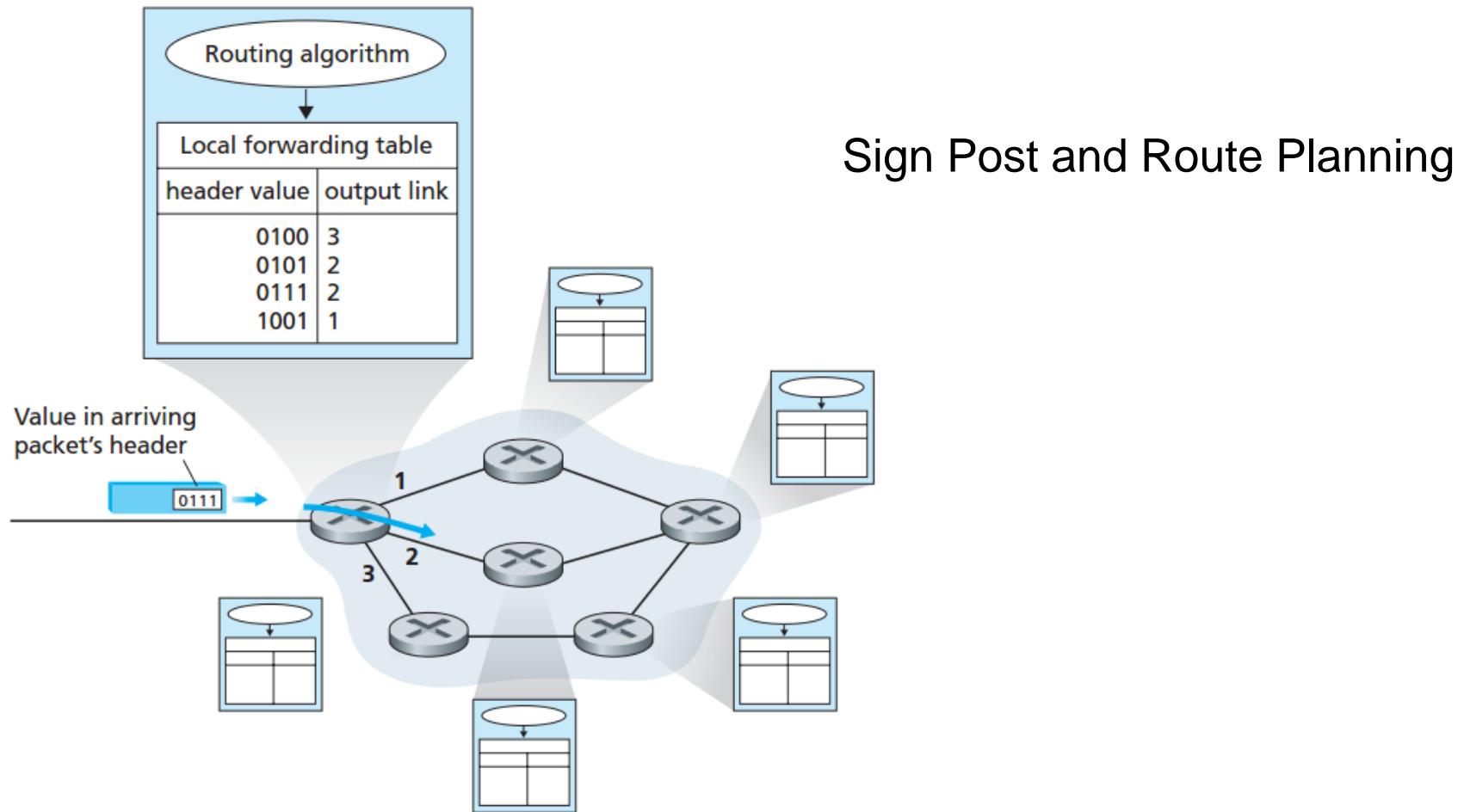


Figure 4.2 ♦ Routing algorithms determine values in forwarding tables

Network Layer: IPv4 datagram

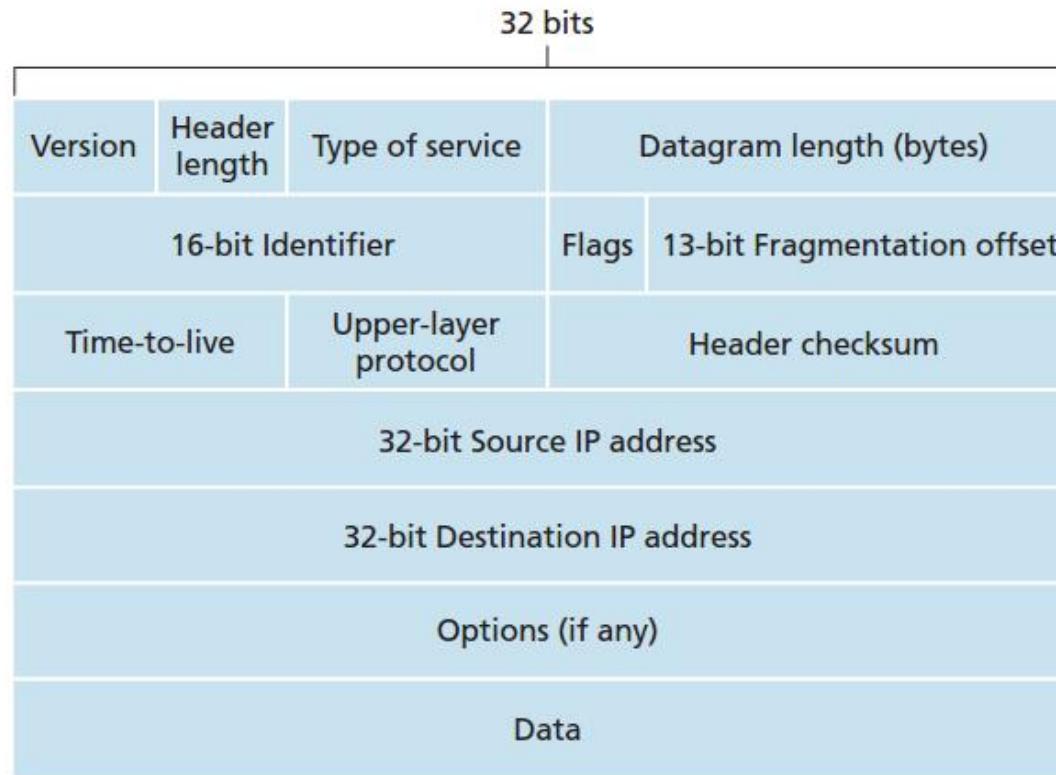
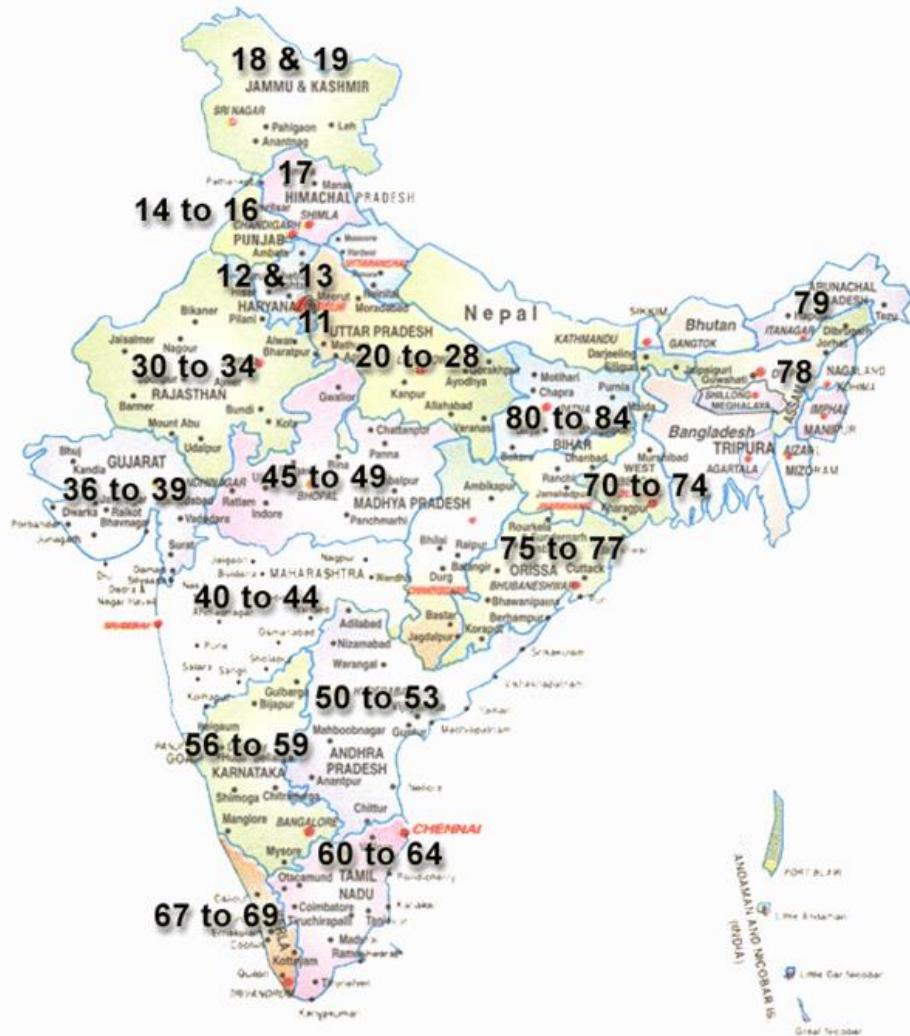
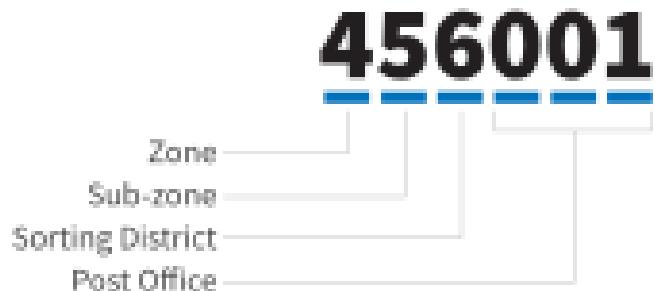


Figure 4.13 ♦ IPv4 datagram format

Example: Indian Postal Codes



Network Layer: Services offered

- Logical transport
- Guaranteed delivery
- Guaranteed delivery with bounded delay
- In-order packet delivery
- Guaranteed minimum bandwidth
- Security services

Network Layer: Routing Architecture

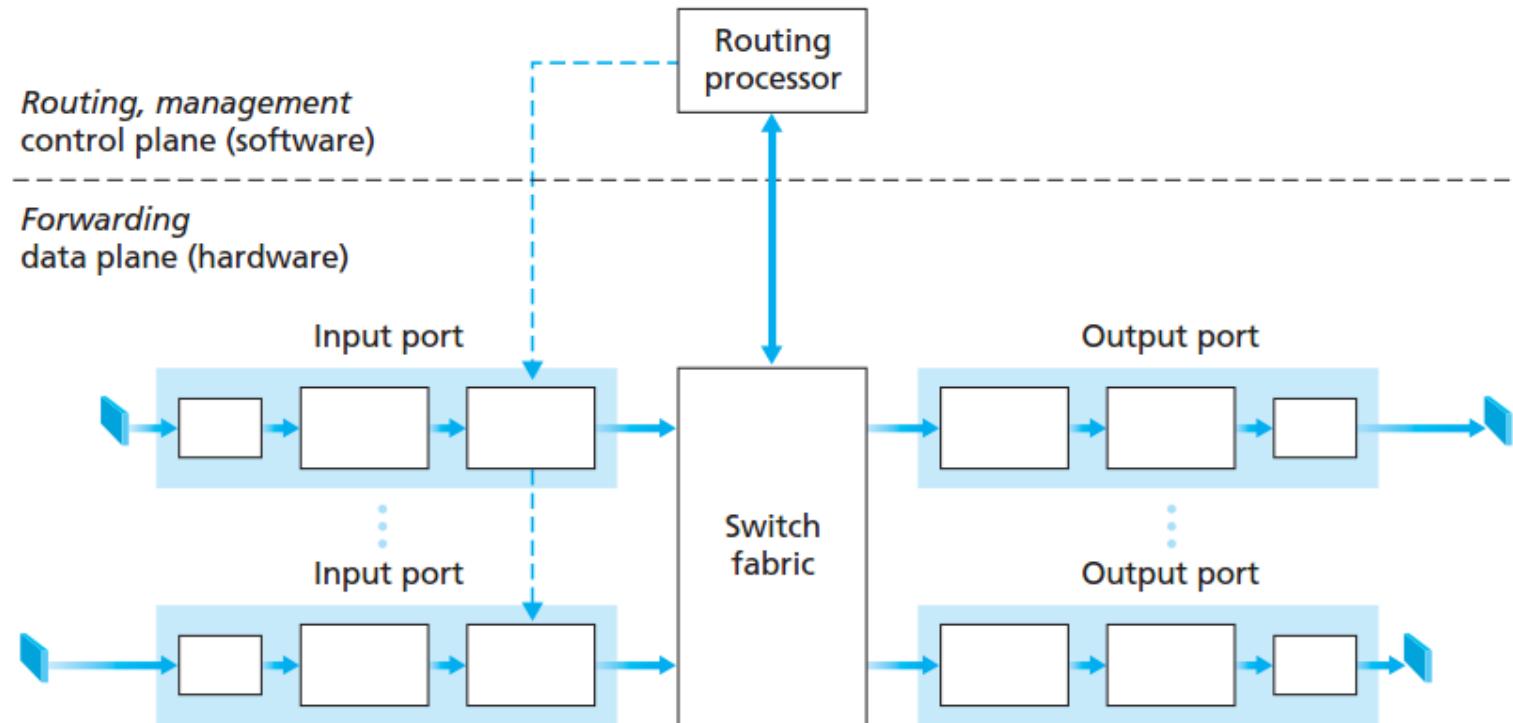


Figure 4.6 ♦ Router architecture

Questions?

Communications & Controls in IoT

Networking Basics-2

Instructors: Sachin Chaudhari

Jan. 30, 2023



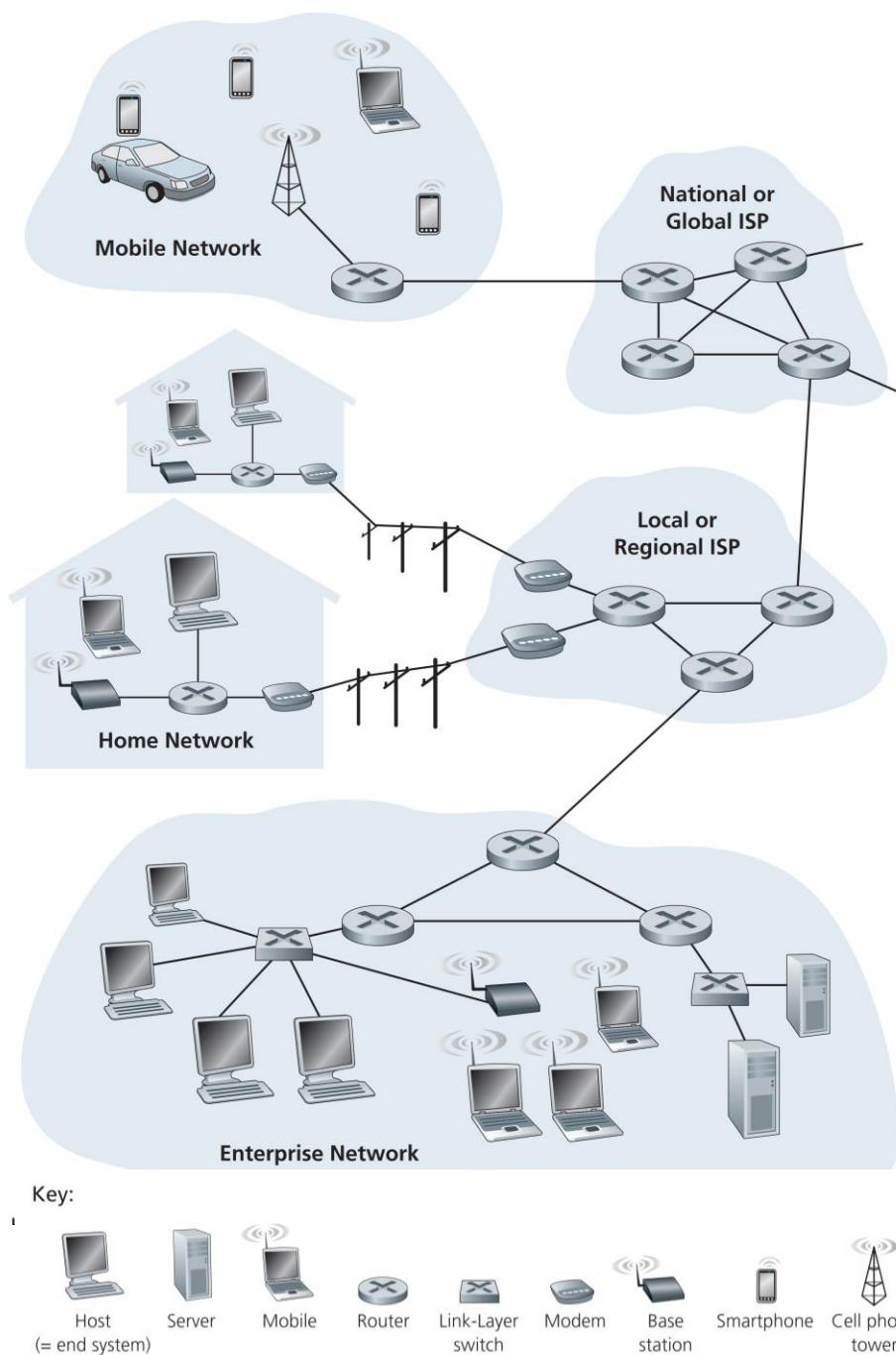
**INTERNATIONAL INSTITUTE OF
INFORMATION TECHNOLOGY**

HYDERABAD

Main Reference

- [Kurose2012] J. Kurose and K. Ross, *Computer Networking*, Pearson, 2012.

Recap



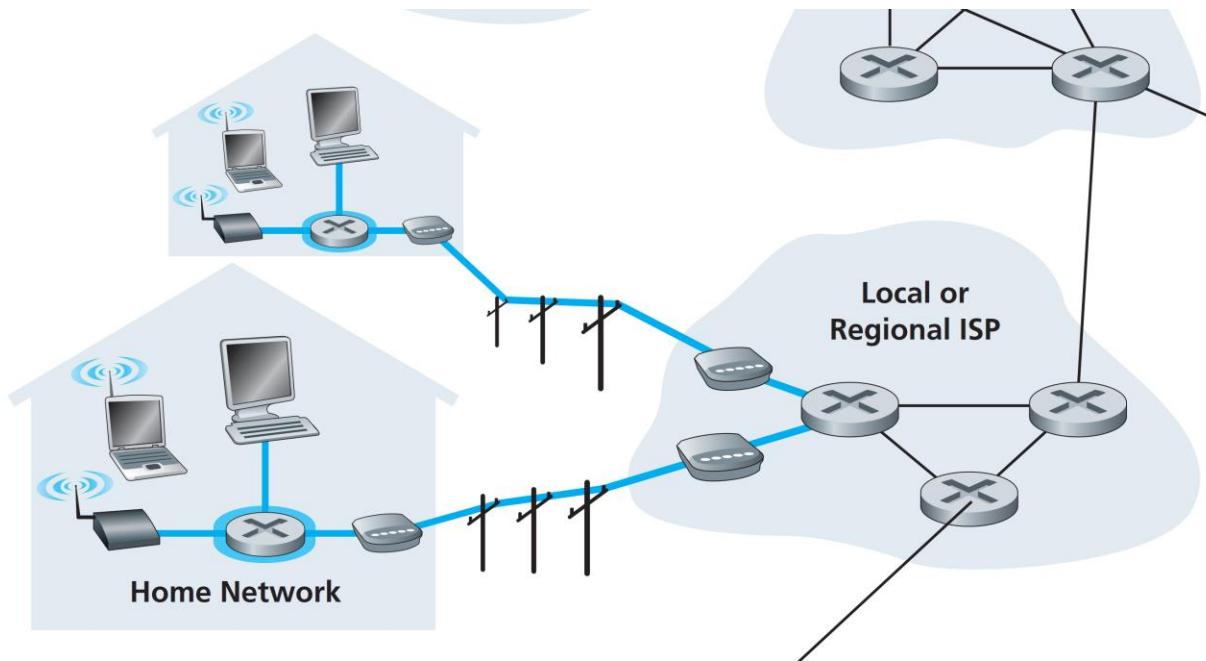
Some Pieces of Internet

The Internet is a computer network that interconnects hundreds of millions of computing devices throughout the world

Few Internet Terminologies

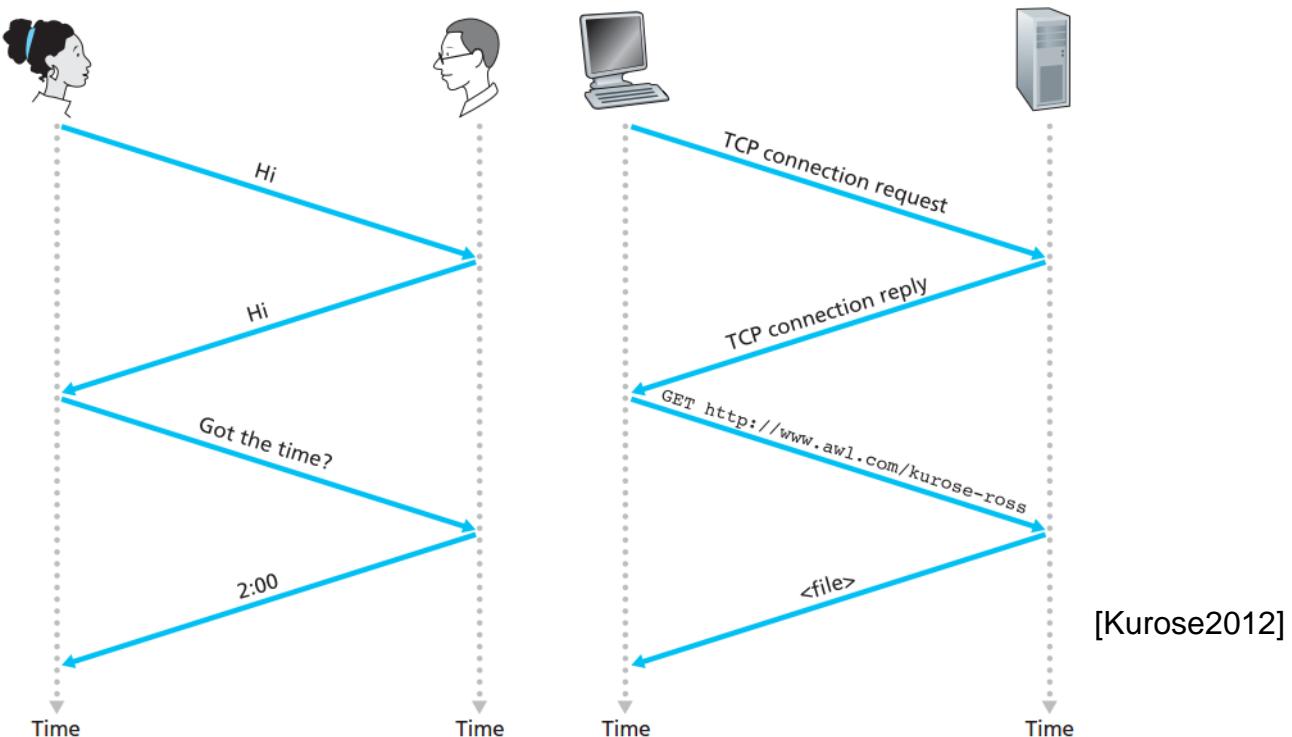
- Host or end-devices
 - computing devices connected to the internet
- Communication links
 - Connect the host in the network
- Packet switches
 - takes a packet arriving on one of its incoming communication links and forwards that packet on one of its outgoing communication links
 - Most prominent
 - Routers: Network core
 - Link-layer switches: access network
- Route or Path
 - The sequence of communication links and packet switches traversed by a packet from the sending end system to the receiving end system
- Internet service providers (ISPs)

Example of route



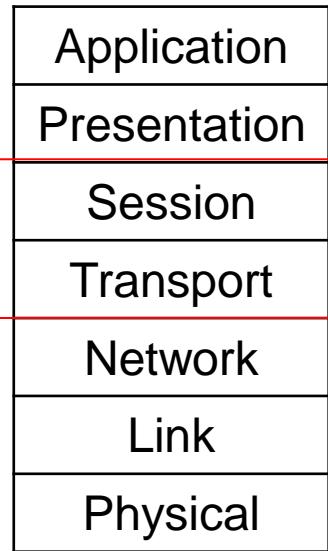
Protocol

- A protocol defines the format and the order of messages exchanged between two or more communicating entities as well as the actions taken on the transmission and/or receipt of a message or other event. [Kurose2012]

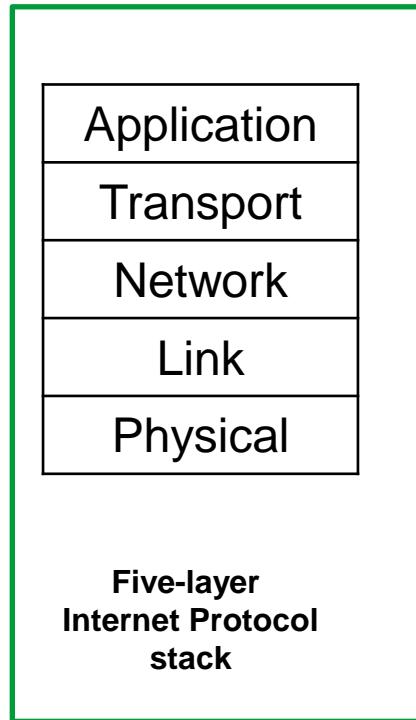


Analogy of human protocol and a computer network protocol

Internet protocol stack and OSI model



Seven-layer
Open Systems Interconnection
(OSI) model



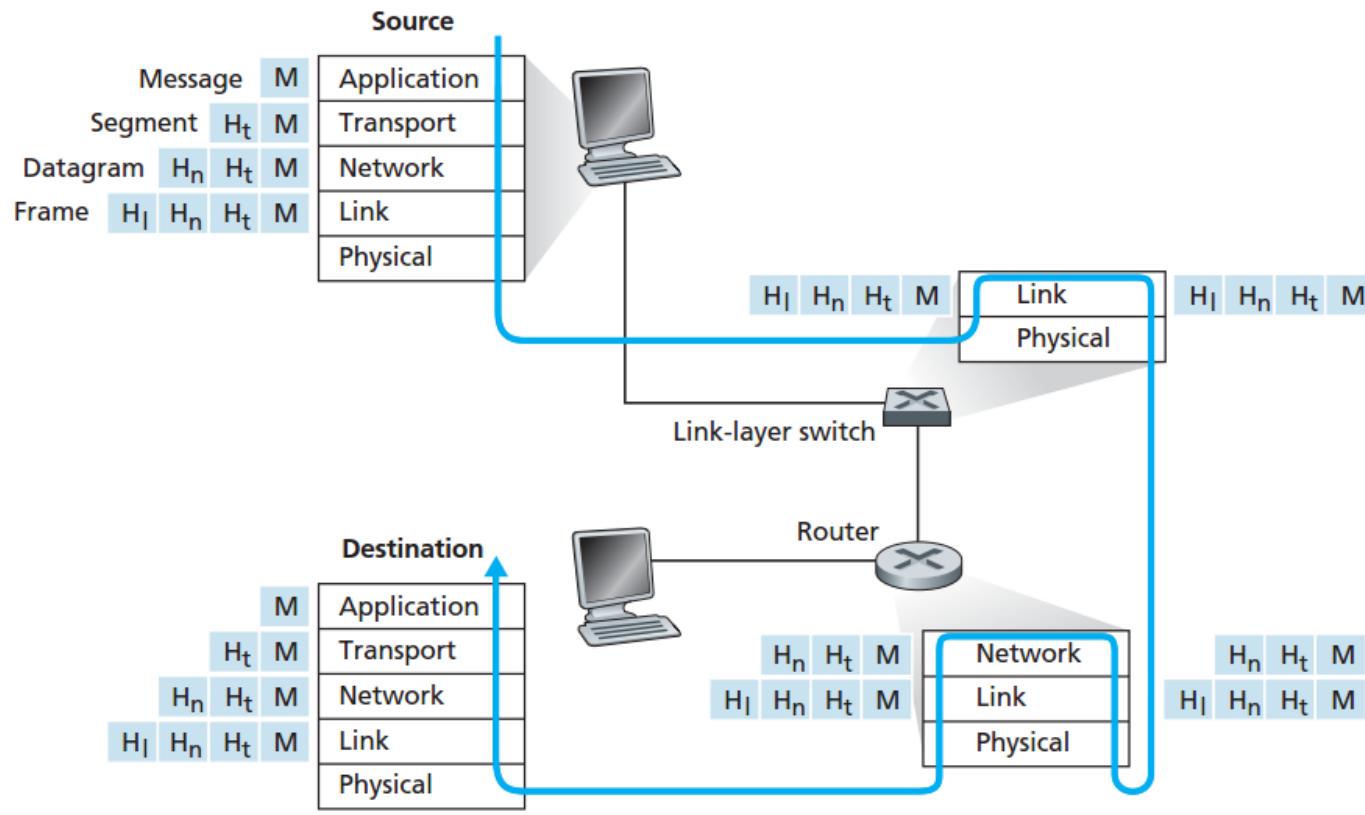
Protocols Layers and Their Service Models

- A layered architecture allows us to discuss a well-defined, specific part of a large and complex system.
- Provides modularity, making it much easier to change the implementation of the service provided by the layer.
- As long as the layer provides the same service to the layer above it and uses the same services from the layer below it, the remainder of the system remains unchanged when a layer's implementation is changed.

Internet protocol stack: Toy Example

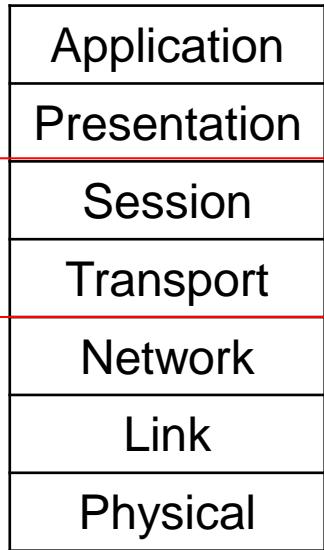
- Sending a courier from company branch in Hyderabad to company branch in New York
 - Application Layer: Individuals giving parcels
 - Transport Layer: office boy or admin assistant
 - Network Layer: Speed post/ Blue Dart (representative)
 - Link Layer: Different drivers (and vehicles)
 - Physical Layer: Road/Air/Water

Encapsulation of data across layers

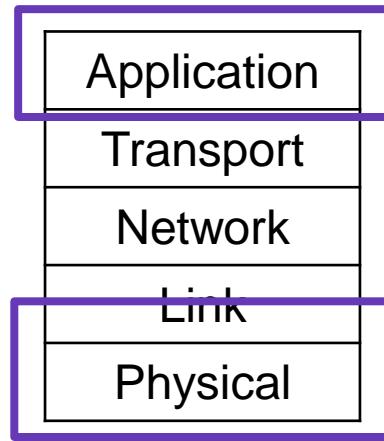


[Kurose2012]

Focus in this course



Seven-layer
Open Systems Interconnection
(OSI) model

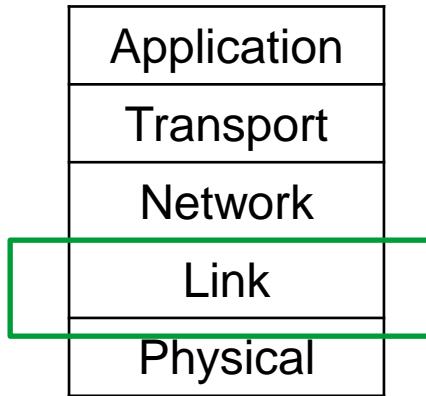


Five-layer
Internet Protocol
stack

Application
MAC and PHY

Today's Class

Internet protocol stack: *Link Layer*



Five-layer
Internet Protocol
stack

Internet Protocol Stack: *Link Layer*

- Also called Data Link Layer
- Responsible for moving data packets from one node to another across an individual link
- Implemented in network adapter or network interface card (NIC)
- Provides medium access control (MAC) for multiple nodes sharing a common medium
- Offer services like framing, reliable delivery, flow control, error detection and correction,
- Each NIC has 48 bit unique LAN/MAC/physical address
- Examples of link-layer protocols include Ethernet, WiFi, PPP

Link Layer: Various multiple access channels

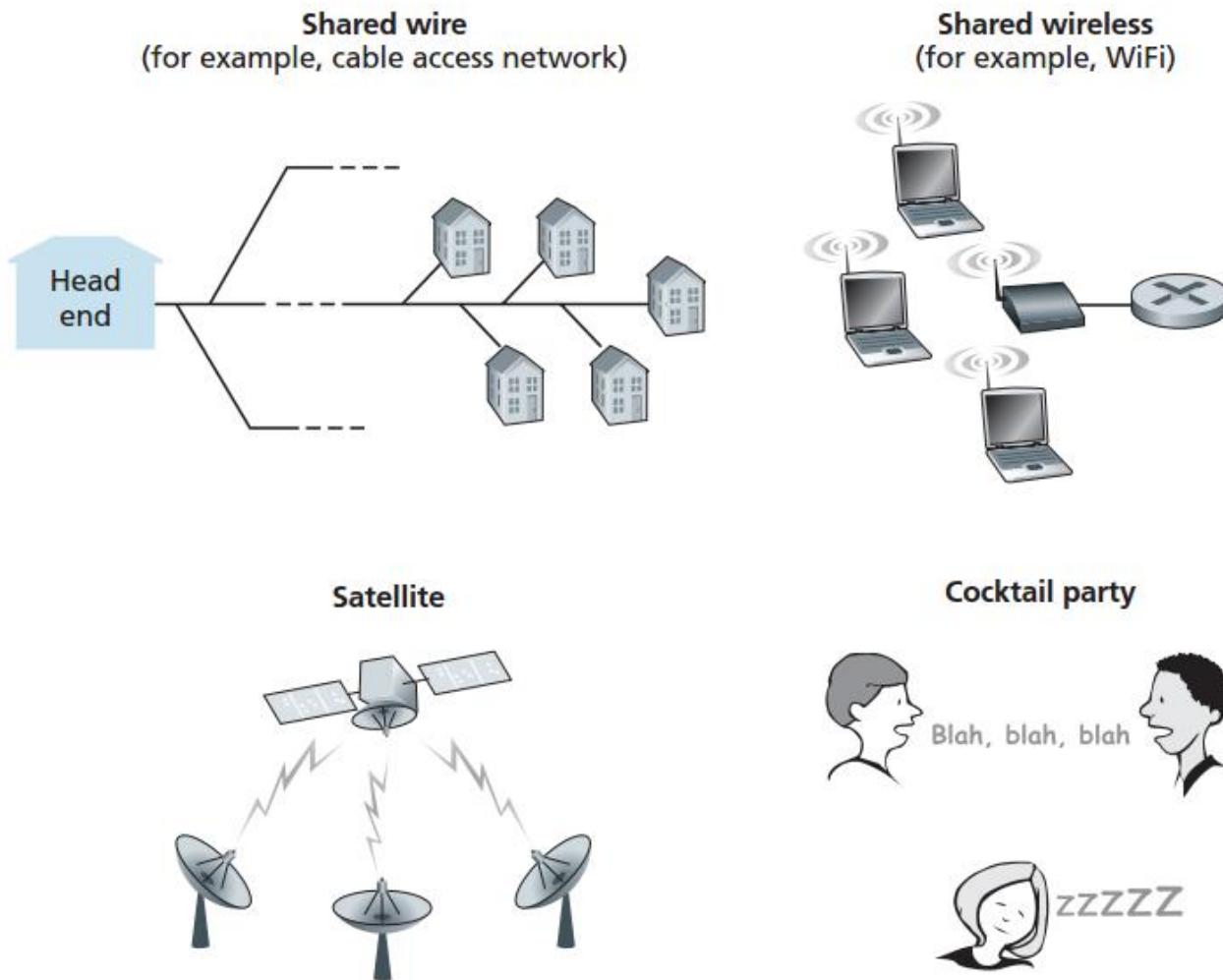


Figure 5.8 ♦ Various multiple access channels

Link Layer

- Different link layer protocols can be used over different links along a route
 - Trip from Hyderabad office to New York office

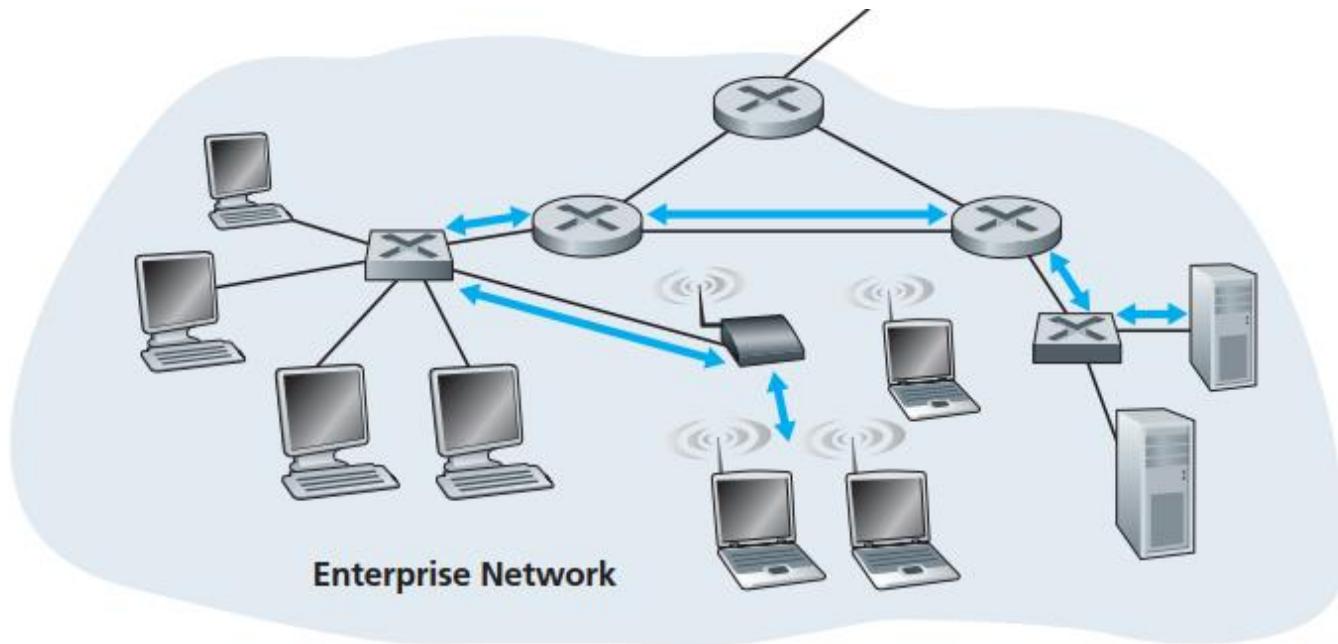
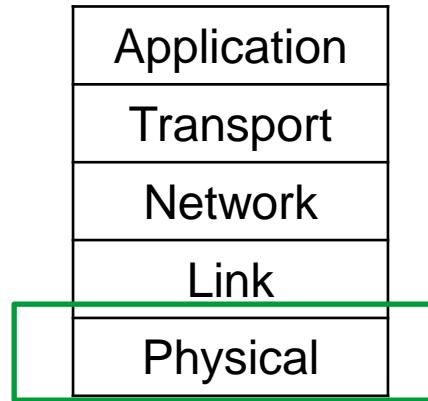


Figure 5.1 ♦ Six link-layer hops between wireless host and server

Link Layer: MAC and IP addresses

- MAC address is like social security number
- IP address is like house address
- Address resolution protocol to translate IP to MAC address and vice-versa

Internet protocol stack: *Physical Layer*

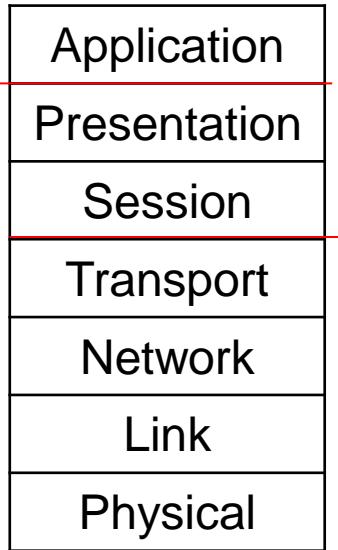


Five-layer
Internet Protocol
stack

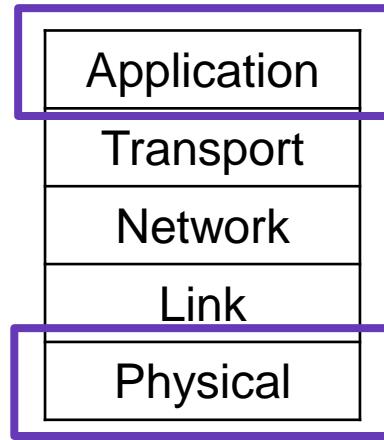
Internet Protocol Stack: *Physical Layer*

- Defines the means of transmitting raw bits rather than logical data packets over a physical link/medium connecting two nodes on the same network
- Provides interface (such as electrical, optical, and electromagnetic) to the transmission medium (such as twisted-pair copper wire, optical, air)
- Signal processing of bits and physical signals: Modulation, error correction and detection (Channel Coding), Bit Interleaving, Synchronization, Carrier sensing and collision detection, etc.
- Example: WLAN 802.11, LR-WPANs 802.15.4, Ethernet 802.3, Bluetooth 802.15.1

Focus in this course



Seven-layer
Open Systems Interconnection
(OSI) model

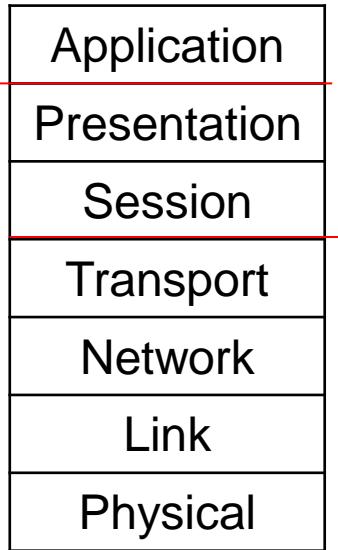


Five-layer
Internet Protocol
stack

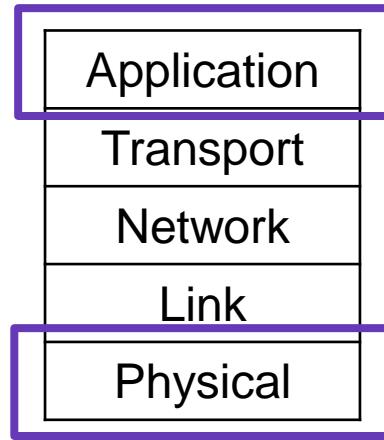
Application
MAC and PHY

Questions?

Focus in this course



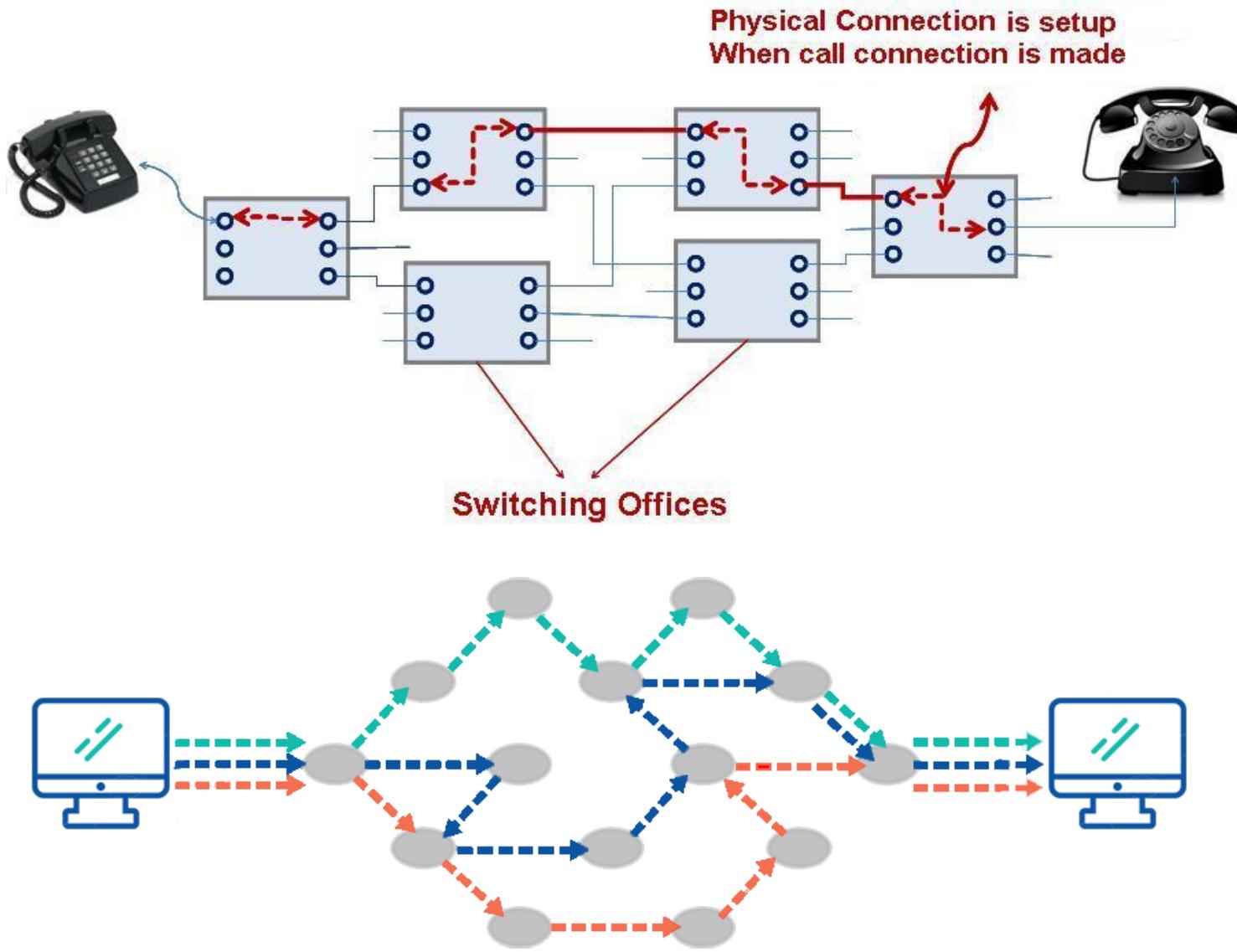
Seven-layer
Open Systems Interconnection
(OSI) model



Five-layer
Internet Protocol
stack

Application
MAC and PHY

Circuit Switched vs Packet Switched



Statistical Multiplexing

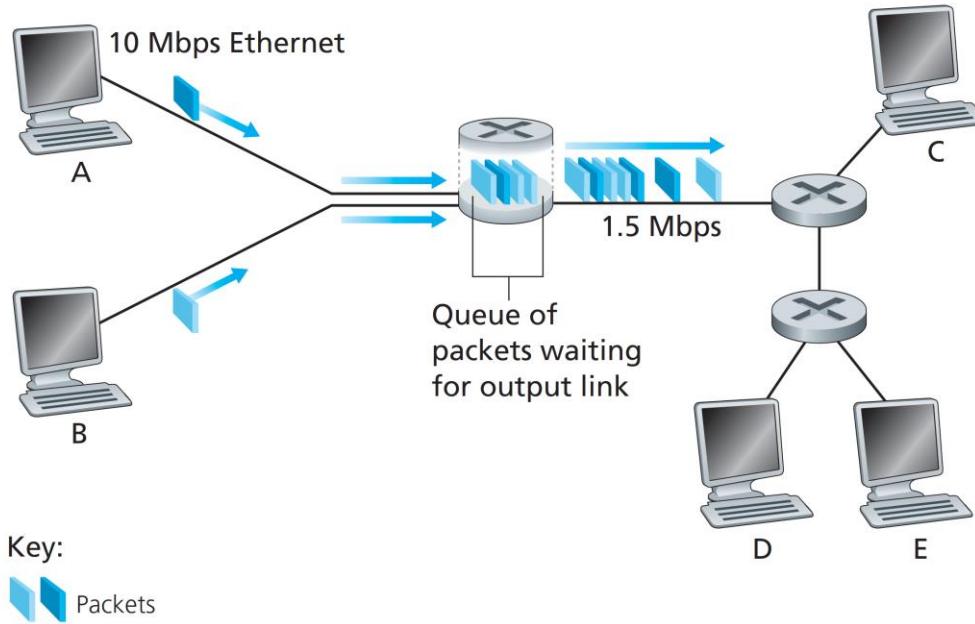
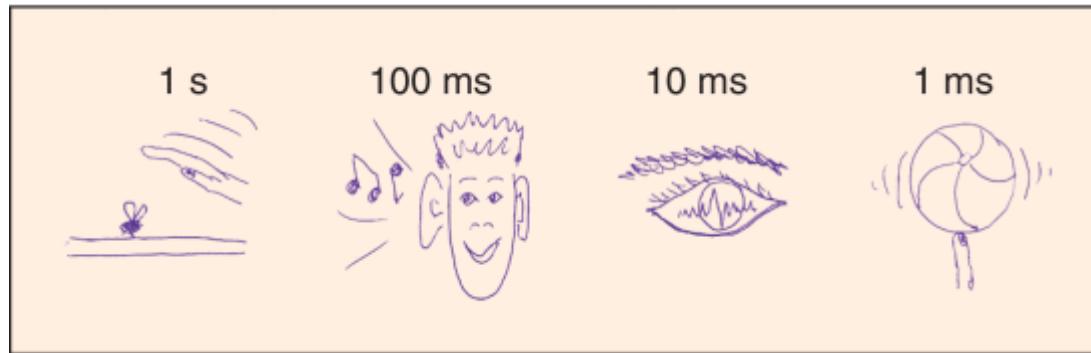


Figure 1.12 ♦ Packet switching

- Number of users = 35
- Ten slots in TDM
- Probability of user being active = $p = 0.1$
- For this case, probability of outage is 0.0004

Statistical Multiplexing: An example

- Frame duration generally ranges from 1ms to 10ms



Human reaction and interaction times (coarse)

Image Credit: G. P. Fettweis, "A 5G wireless communications vision", *Microwave J.*, pp. 24-36, Dec. 2012..

Circuit Switched vs Packet Switched

Circuit Switching	Packet Switching
Physical path between source and destination	No physical path
All packets use same path	Packets travel independently
Reserve the entire bandwidth in advance	Does not reserve
Bandwidth Wastage	No Bandwidth wastage
No store and forward transmission	Supports store and forward transmission

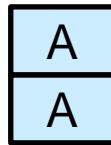
Medium Access Control (MAC) protocols

Medium Access Control (MAC)

- One of the two sublayers of data link layer
- Acts as an interface between the logical link control (LLC) and the network's physical layer
- Provides channel access control mechanisms across a shared physical medium



- Provides addressing mechanisms



A



B

Link Layer: Various multiple access channels

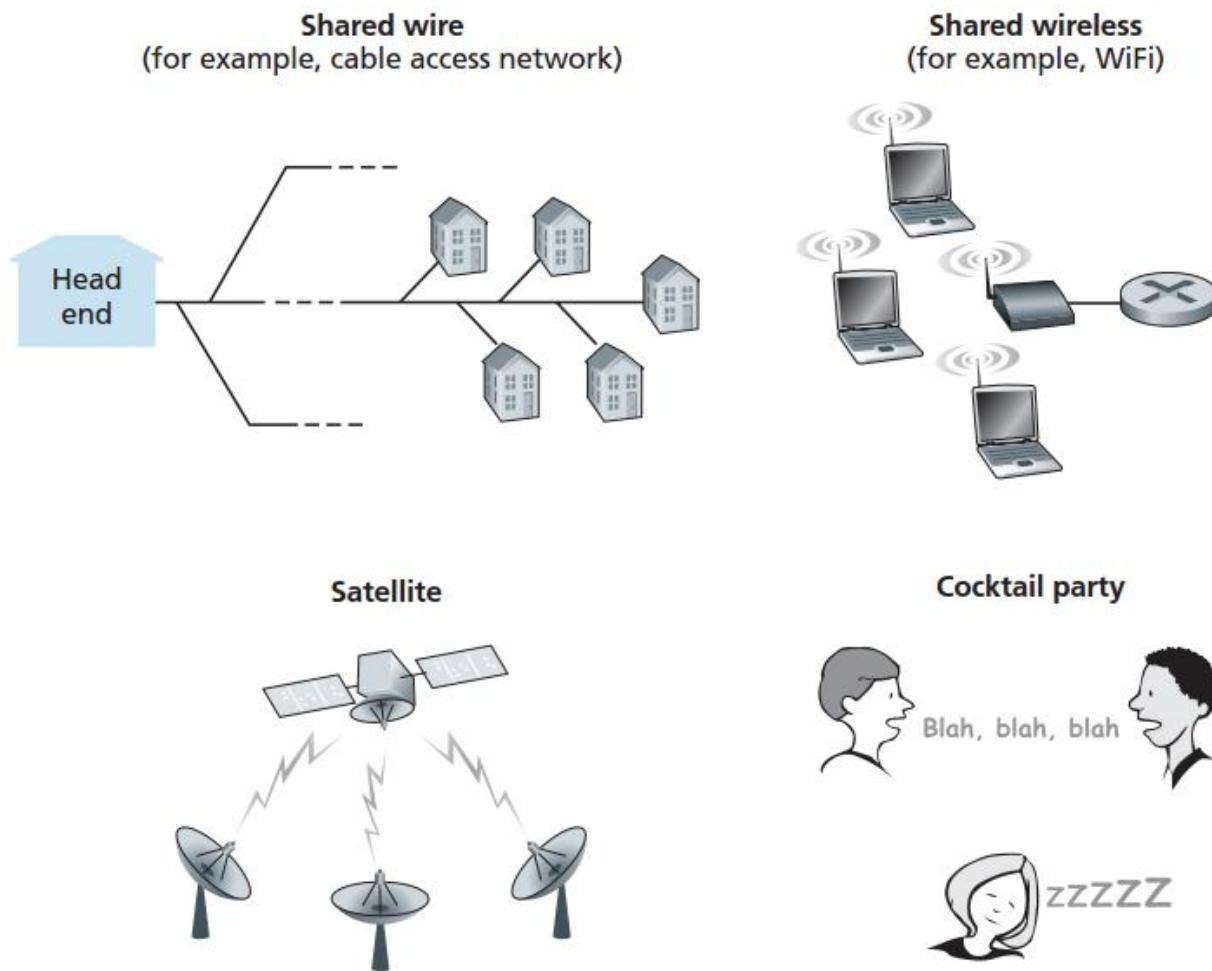


Figure 5.8 ♦ Various multiple access channels

Another Analogy: Roads



<https://www.freepik.com/vectors/travel>

Conversation Etiquettes

- Meeting
 - Give everyone a chance to speak
 - Don't monopolize the conversation
- Class
 - Don't speak until you are spoken to
- General
 - Don't interrupt when someone is speaking
 - Don't fall asleep when someone is speaking
 - What happens if somebody sleeps?

Desirable Properties of MAC protocols

A MAC protocol for a broadcast channel of rate R bps should have following desirable properties

- When only one node has to send data, that node has throughput of R bps
- When M nodes have to send data, each of the nodes should have average rate of R/M bps
- The protocol is decentralized so that there is no master node with single point of failure
- The protocol is simple so that it is inexpensive to implement

Types of MAC protocols

- Channel Partitioning Protocols (or Fixed Assignment Protocols)
 - TDMA, FDMA, CDMA, SDMA
- Random Access Protocols
 - Aloha, Slotted Aloha, CSMA/CA
- Taking Turn Protocols (or Demand Assignment Protocols)
 - Token Ring
 - Polling

Channel Partitioning Protocols

Time Division Multiple Access (TDMA)

- Time Division Multiple Access
 - TDMA is a digital technique that divides a single channel or band into time slots
 - Examples: T1 carrier systems (digital transmission of multiplexed telephone calls), 2G cellular system GSM

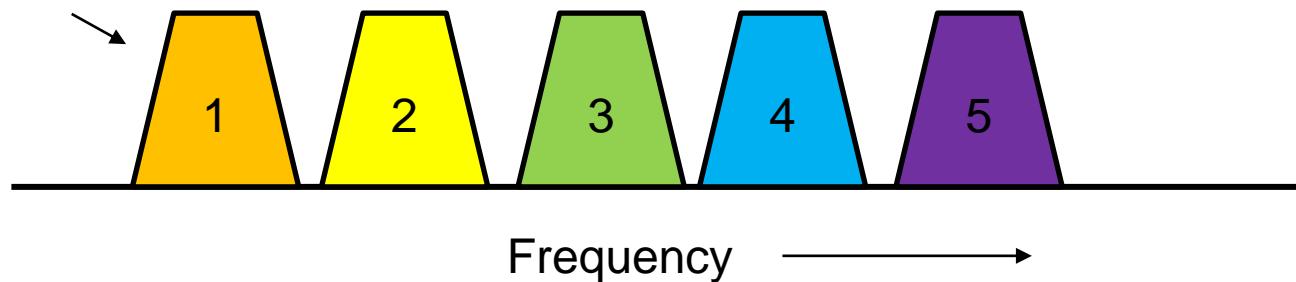


- Advantages
 - No Collision
 - Fair usage
- Disadvantages
 - Wastage of resources
 - Delay
 - Need Synchronization

Frequency Division Multiple Access (FDMA)

- FDMA divides the shared medium bandwidth into individual channels
- Examples: Cable television system, FM stations

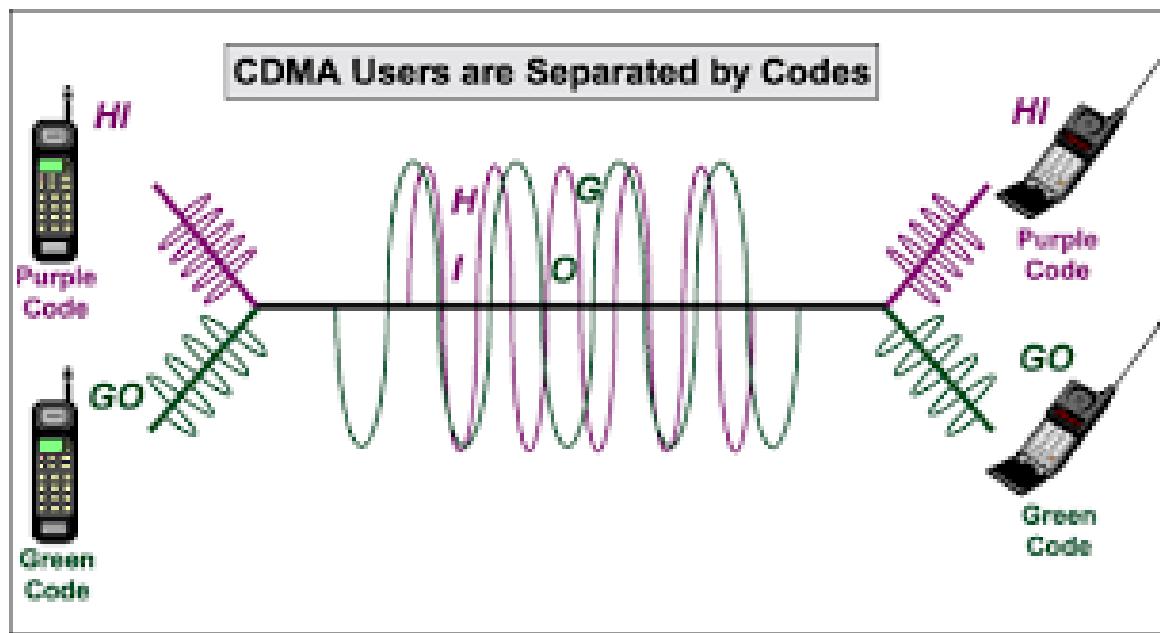
One band per user



- Advantages
 - No Collision
 - Fair usage
- Disadvantages
 - Wastage of resources
 - Need synchronization

Code Division Multiple Access (CDMA)

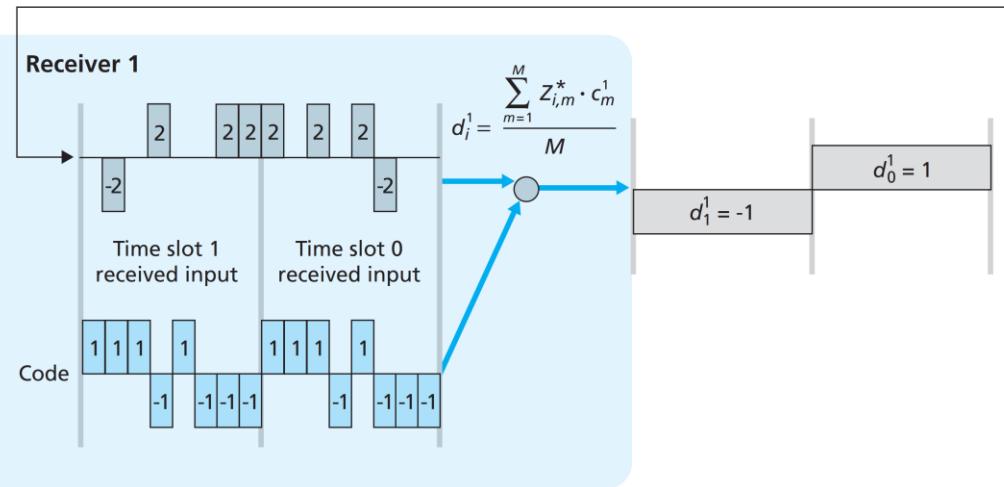
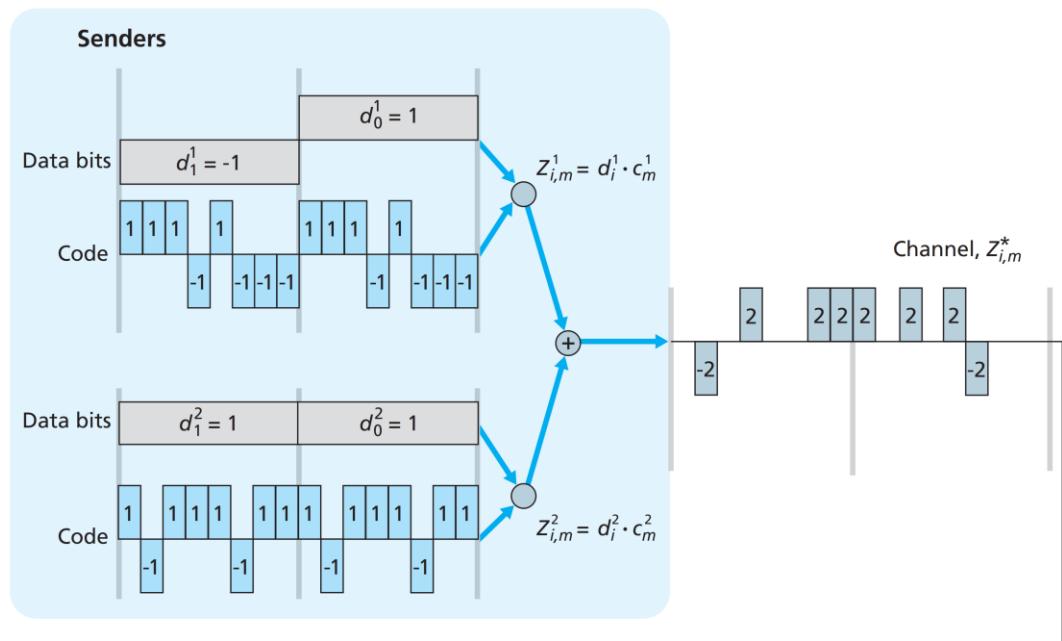
- It is also known as spread spectrum because it takes the digitized version of an analog signal and spreads it out over a wider bandwidth at a lower power level.
- Example: 2G IS-95, 3G (WCDMA)



Source: <http://www.electronicdesign.com/communications/fundamentals-communications-access-technologies-fdma-tdma-cdma-ofdma-and-sdma>

CDMA

- Use of higher rate PN sequences
- Low auto and cross correlation

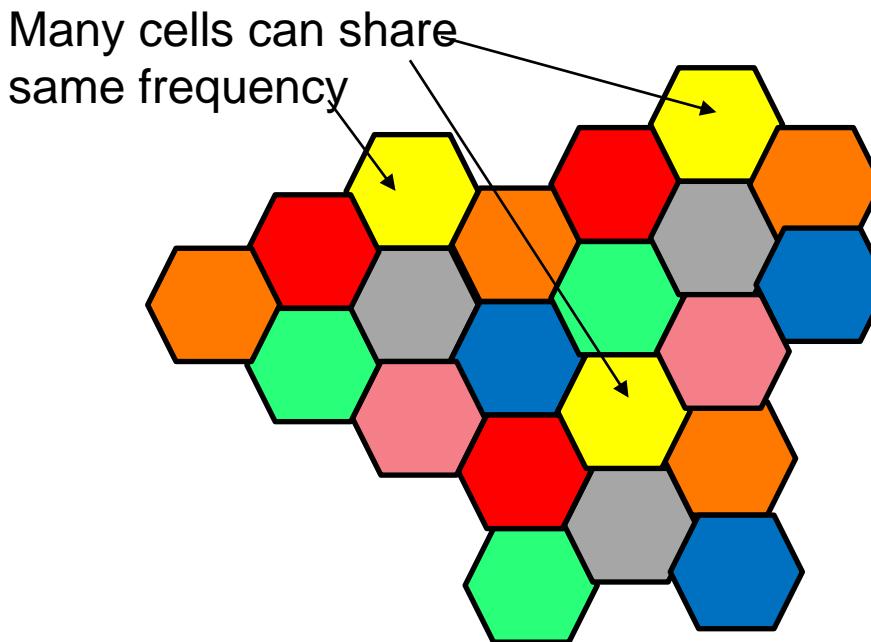


CDMA

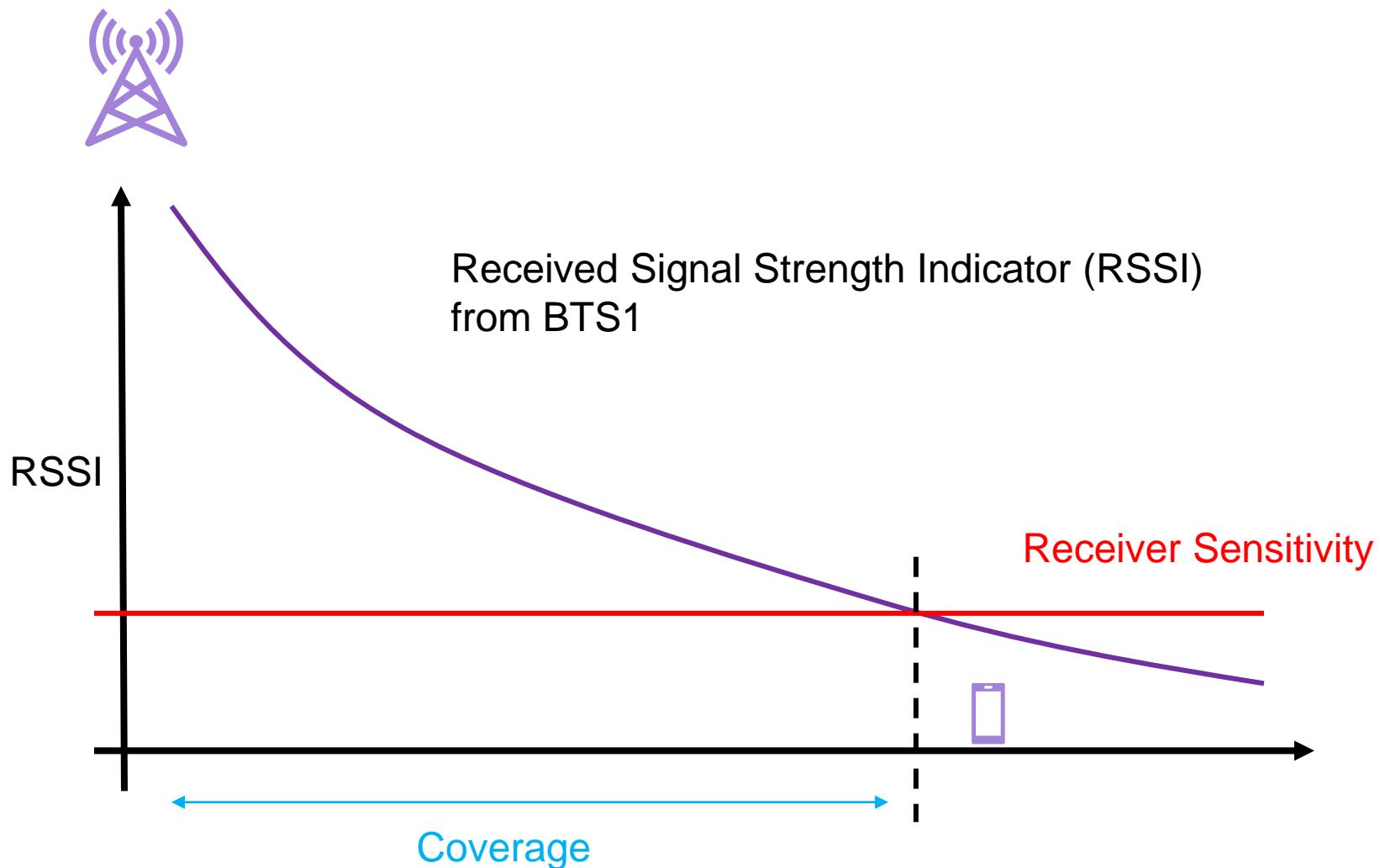
- Advantages
 - No collisions
 - Asynchronous CDMA possible
 - Better efficiency than TDMA and FDMA
- Disadvantages
 - Need extra processing
 - Power control is needed

Space Division Multiple Access (SDMA)

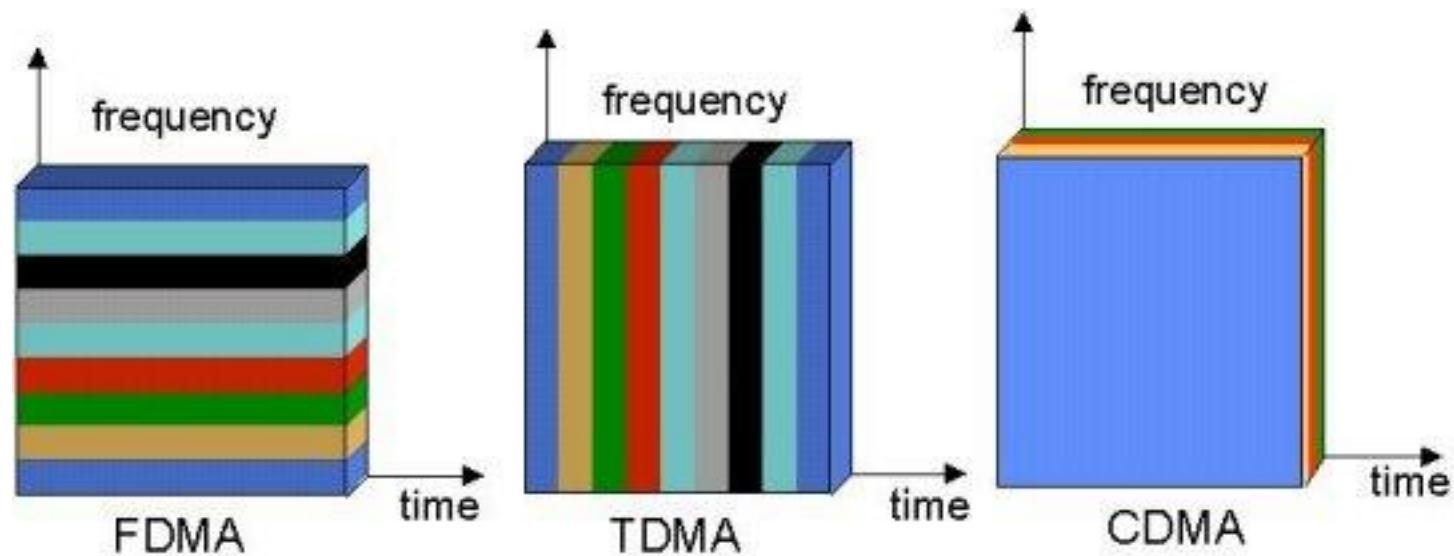
- Space Division Multiple Access
 - SDMA uses physical separation methods that permit the sharing of wireless channels. For instance, a single channel may be used simultaneously if the users are spaced far enough from one another to avoid interference. Known as frequency reuse, the method is widely used in cellular radio systems. Cell sites are spaced from one another to minimize interference.



Coverage in Free-Space



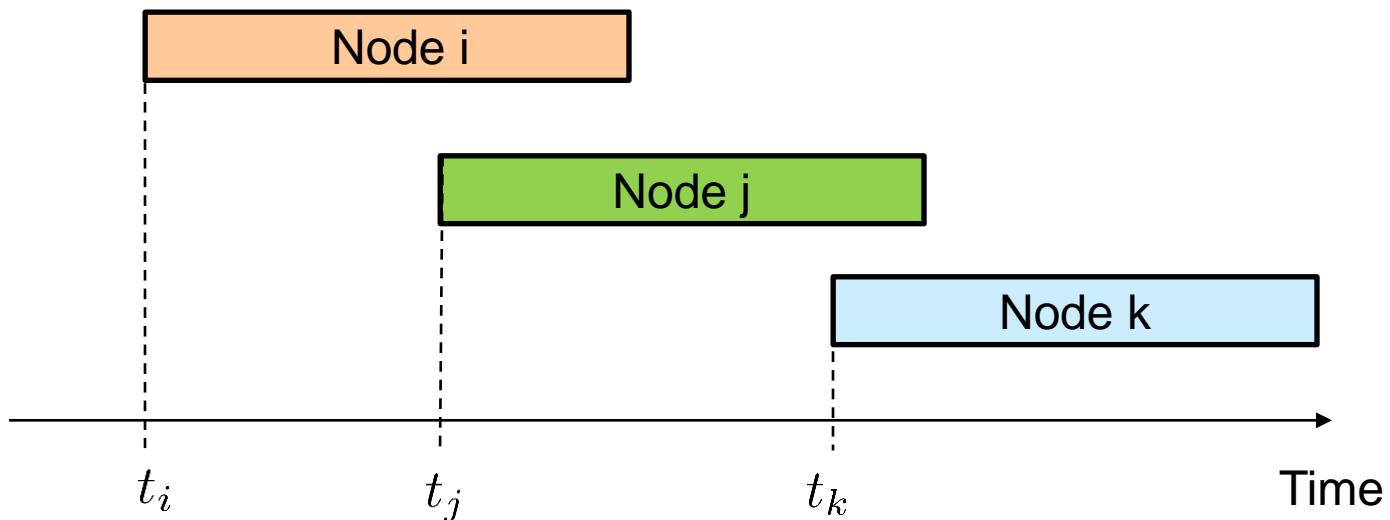
Difference between different TDMA/FDMA/CDMA



Random Access Protocols

Aloha

- When you have data, send it
- If data doesn't go through, resend it after random delay
 - Send with probability p or wait for one transmission frame with probability $1-p$

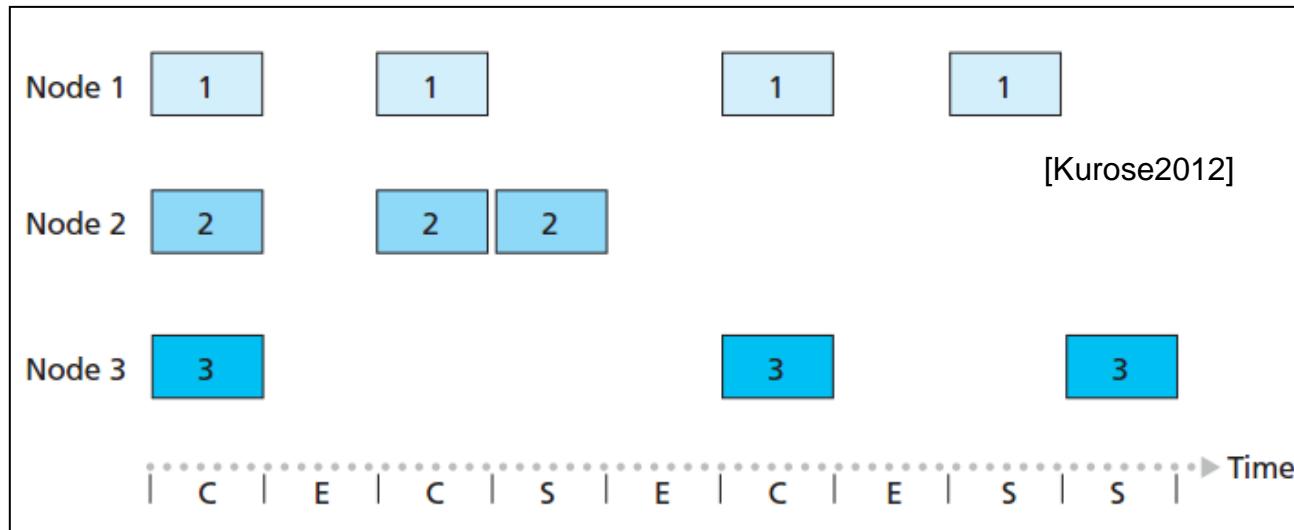


Aloha

- Advantages
 - Full instantaneous rate
 - Fully decentralized
- Disadvantages
 - Low efficiency:
 - Probability of success is $p(1 - p)^{2(N-1)}$
 - 18.5% for large N
 - Suitable only for light loaded network
 - Unstable for certain channel conditions
 - Avalanche of retransmission attempts
 - Nice animation
 - <http://www.wirelesscommunication.nl/reference/chaptr06/aloha/alohplay.htm>

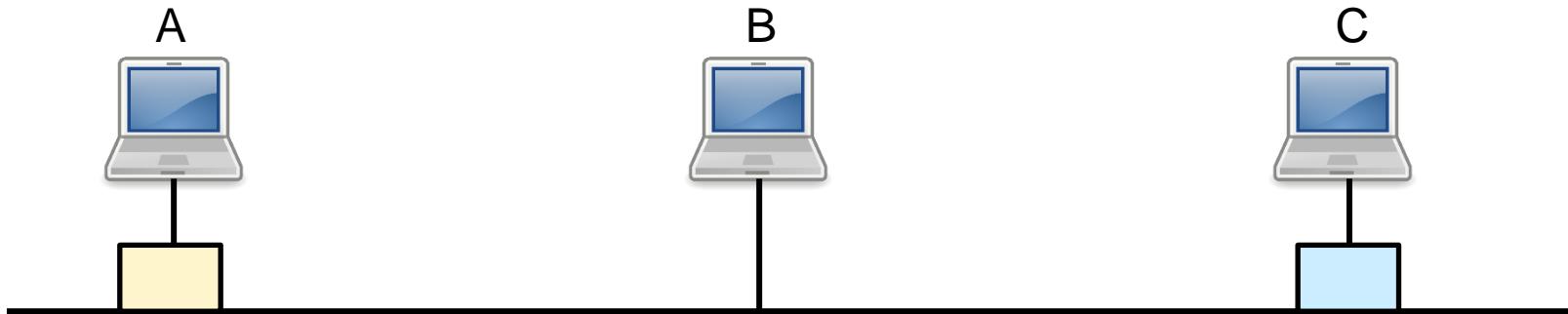
Slotted Aloha

- Time is divided into equal time slots
- Sensor node can send data only at the beginning of a slot
- If have data to send, send at the start of slot. If collision, send in the next slot with probability p and do not transmit with probability $1-p$
- Requires time synchronization between nodes
- Better than Aloha but still low
 - Probability of success is $Np(1 - p)^{N-1}$
 - Asymptotic numbers: Efficiency: 37%; Wastage: 37%; Collisions: 26%

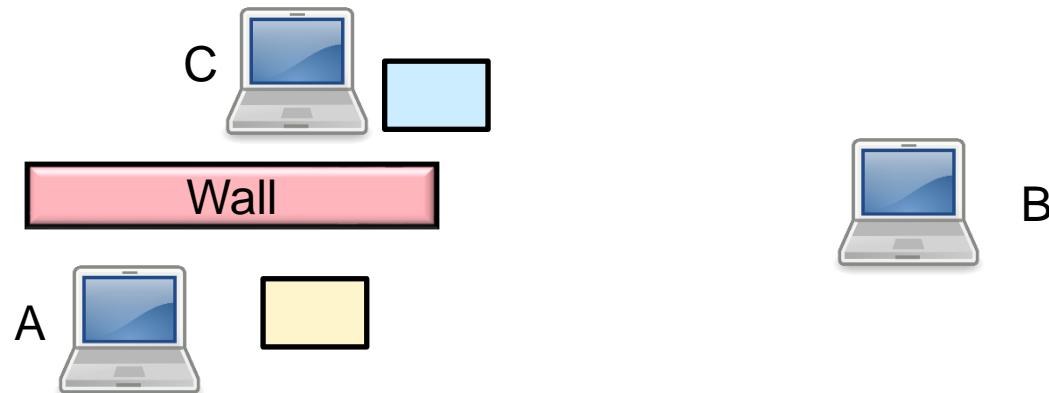


Carrier Sense Multiple Access (CSMA)

- Listen before sending
- Send only if channel is idle
- Collisions can still happen (Hidden Node Problem)
- If collision, back-off for random delay and transmit again

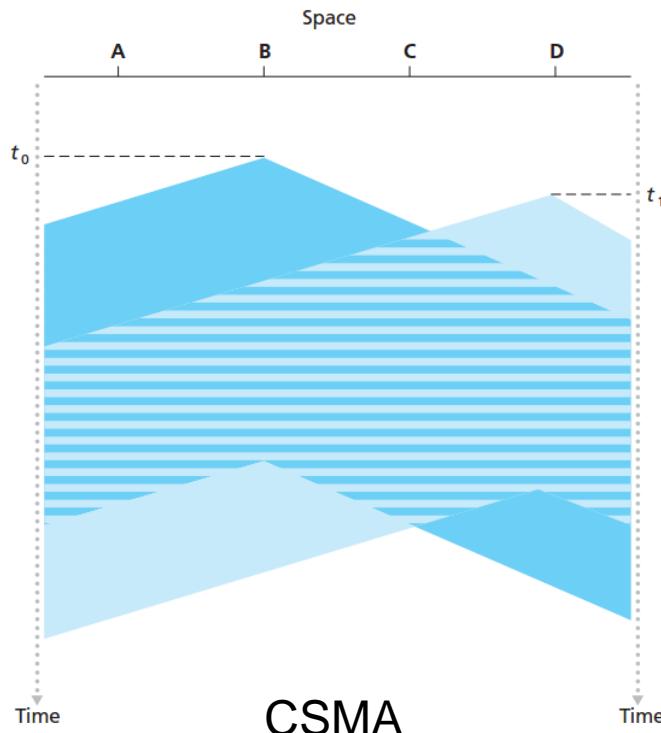


- Hidden Node Problem in Wireless Networks



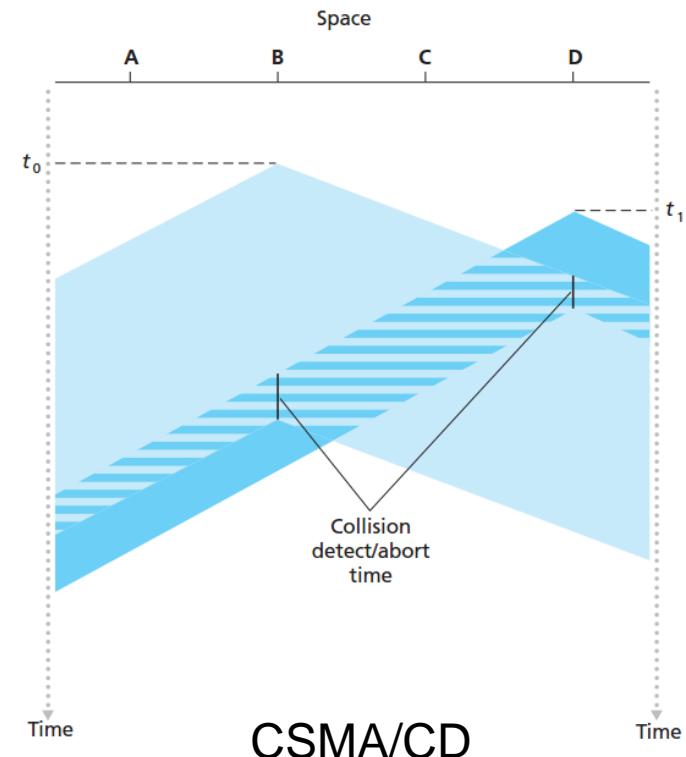
CSMA with collision detection (CD)

- Listen while transmitting!
- Stop transmitting as soon as collision is detected
- Wait for random duration before retry (binary exponential backoff)
- Improves CSMA performance at the cost of complexity
- Used in original Ethernet (wired LAN technology IEEE 802.3)



[Kurose2012]

CSMA



CSMA/CD

Binary Exponential Backoff

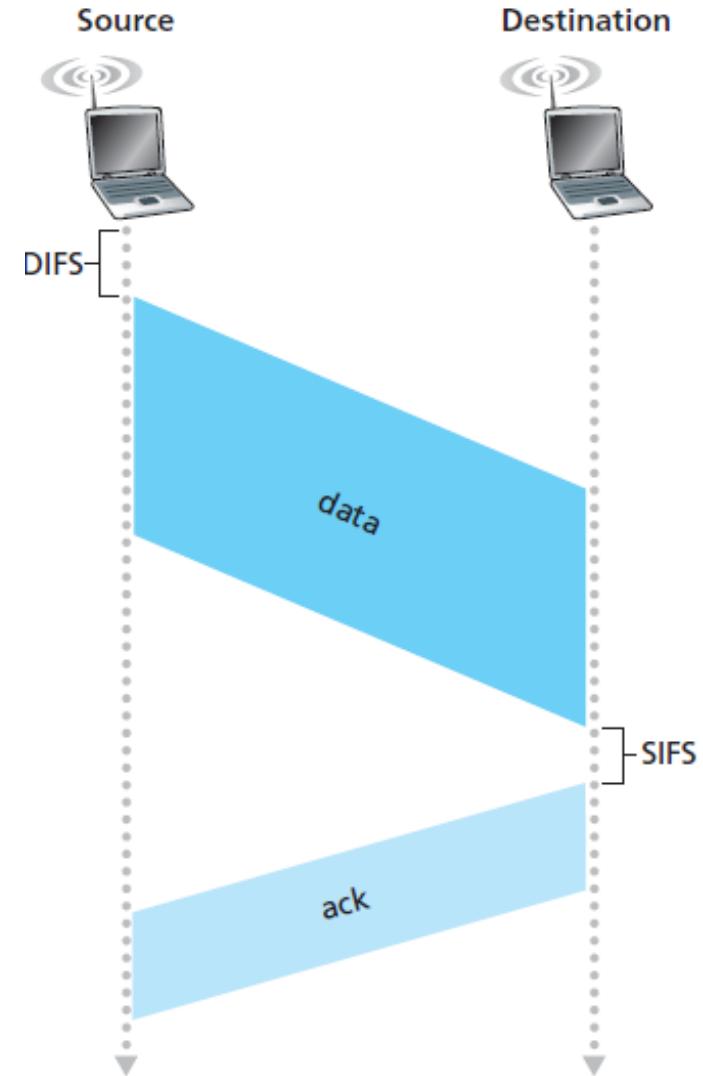
- A node, which has already observed n collisions for a packet, chooses value K at random from $\{0, 1, \dots, 2^n - 1\}$
- For ethernet, node $K * 512$ bit times
- Why exponential backoff?
 - Large backoff time and small number of nodes
 - long wait times and channel idle
 - Small backoff time and large number of nodes
 - large number of collisions
- What happens if a new user wants to transmit when there are several users already in contention?

CSMC/CD

- Advantages
 - Saves resources
- Disadvantages
 - Needs extra hardware and processing
 - Possible only in wired

CSMA/CA (Collision Avoidance)

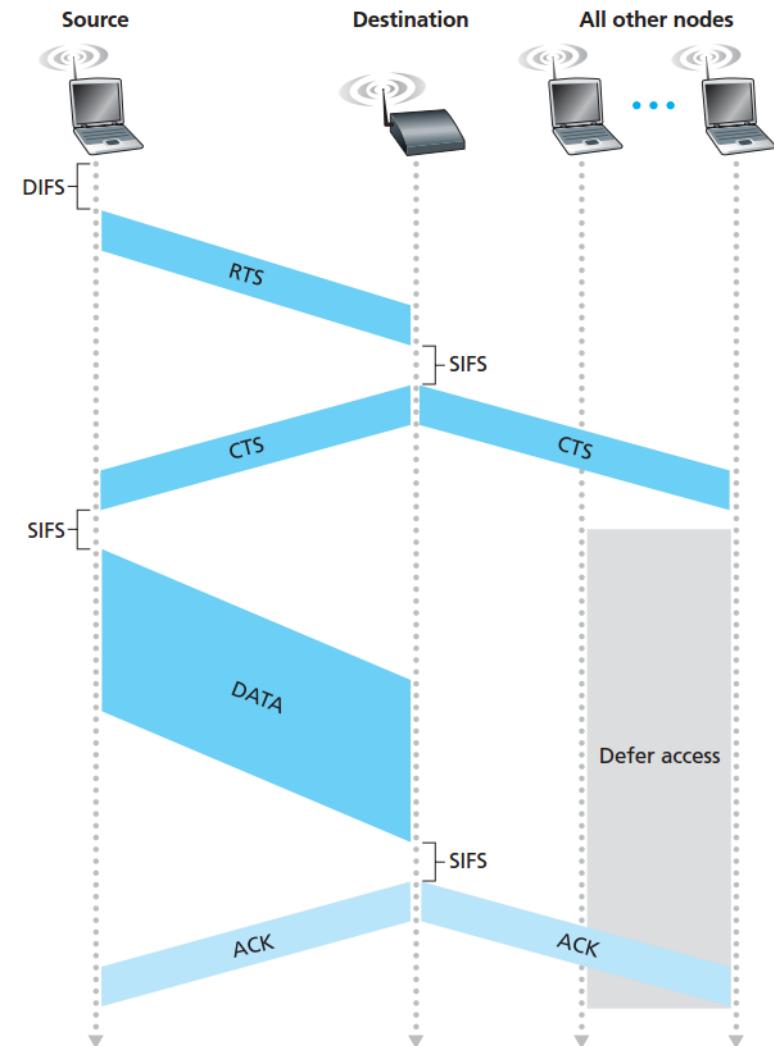
- Listens for an idle channel
- Once channel is detected
 - idle waits for Distributed Inter Frame Spacing (DIFS)
 - Performs exponential random backoff
 - Counter is frozen if the channel is busy
- Once the counter hits zero, sends the complete frame and waits for an ack
 - If ack, repeat the procedure for next packet
 - If no ack, repeat with a longer window
- Suitable for wireless networks
- Used in most of the 802.11 (WLAN) technologies



[Kurose2012]

CSMA/CA with RTS and CTS

- Three-way handshake :
 - RTS-CTS-Data
- CSMA and use of ready to send (RTS) and clear to send (CTS)
 - In RTS/CTS access mode, prior to the data transmission the sending node will send a RTS packet to announce the upcoming transmission
 - When the destination node receives the RTS it will send a CTS packet after a short inter-frame space (SIFS) interval
 - Both the RTS and CTS packets are short control packets
- Removes hidden node issues



Communications & Controls in IoT

Communication Techniques

Instructor: Sachin Chaudhari

Feb. 06, 2023



**INTERNATIONAL INSTITUTE OF
INFORMATION TECHNOLOGY**

HYDERABAD

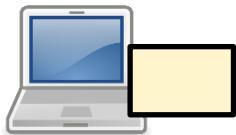
Main Reference

- [Kurose2012] J. Kurose and K. Ross, *Computer Networking*, Pearson, 2012.

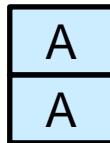
Recap: *MAC protocols*

Medium Access Control (MAC)

- One of the two sublayers of data link layer
- Acts as an interface between the logical link control (LLC) and the network's physical layer
- Provides channel access control mechanisms across a shared physical medium



- Provides addressing mechanisms



A



B

Types of MAC protocols

- Channel Partitioning Protocols (or Fixed Assignment Protocols)
 - TDMA, FDMA, CDMA, SDMA
- Random Access Protocols
 - Aloha, Slotted Aloha, CSMA/CA
- Taking Turn Protocols (or Demand Assignment Protocols)
 - Token Ring
 - Polling

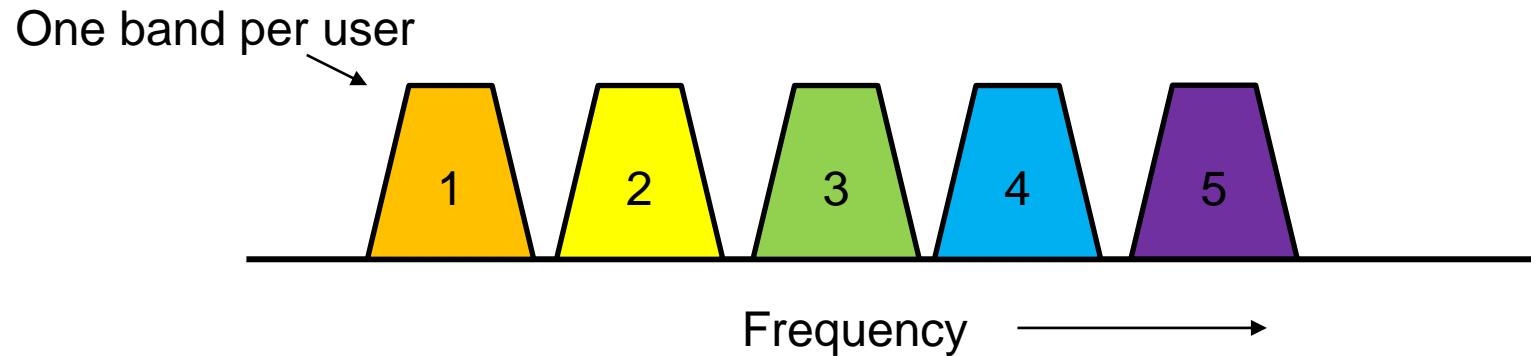
Time Division Multiple Access

- Time Division Multiple Access
 - TDMA is a digital technique that divides a single channel or band into time slots
 - Examples: T1 carrier systems (digital transmission of multiplexed telephone calls), 2G cellular system GSM



Frequency Division Multiple Access

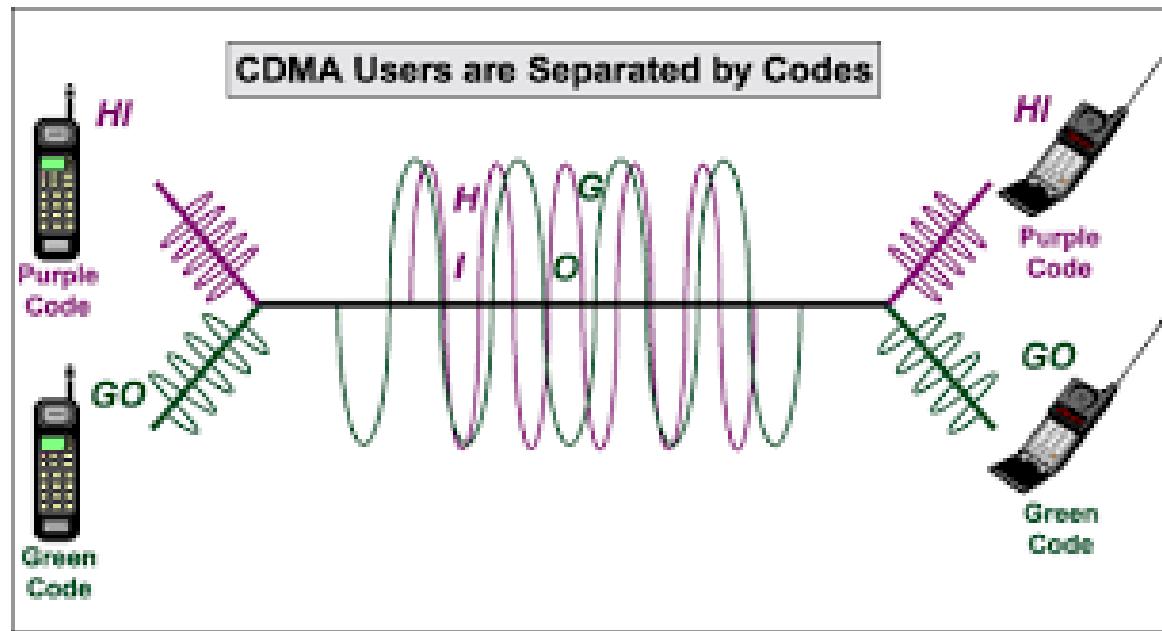
- Frequency Division Multiple Access
 - FDMA divides the shared medium bandwidth into individual channels
 - Examples: Cable television system, FM stations



Code Division Multiple Access

- Code Division Multiple Access

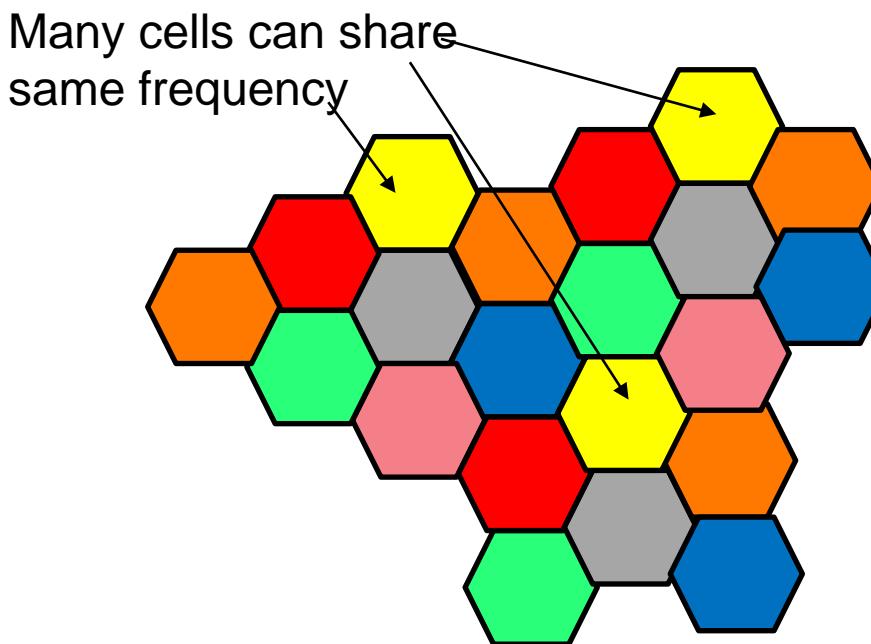
- It is also known as spread spectrum because it takes the digitized version of an analog signal and spreads it out over a wider bandwidth at a lower power level.
- Example: 2G IS-95, 3G (WCDMA)



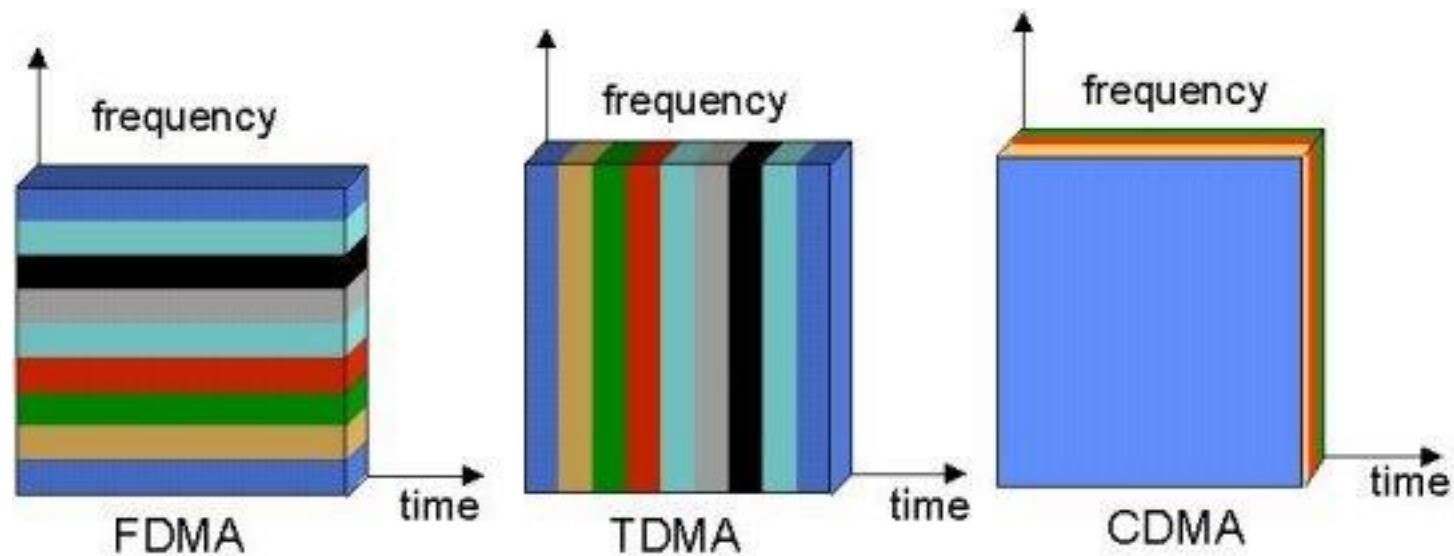
Source: <http://www.electronicdesign.com/communications/fundamentals-communications-access-technologies-fdma-tdma-cdma-ofdma-and-sdma>

Space Division Multiple Access

- Space Division Multiple Access
 - SDMA uses physical separation methods that permit the sharing of wireless channels. For instance, a single channel may be used simultaneously if the users are spaced far enough from one another to avoid interference. Known as frequency reuse, the method is widely used in cellular radio systems. Cell sites are spaced from one another to minimize interference.

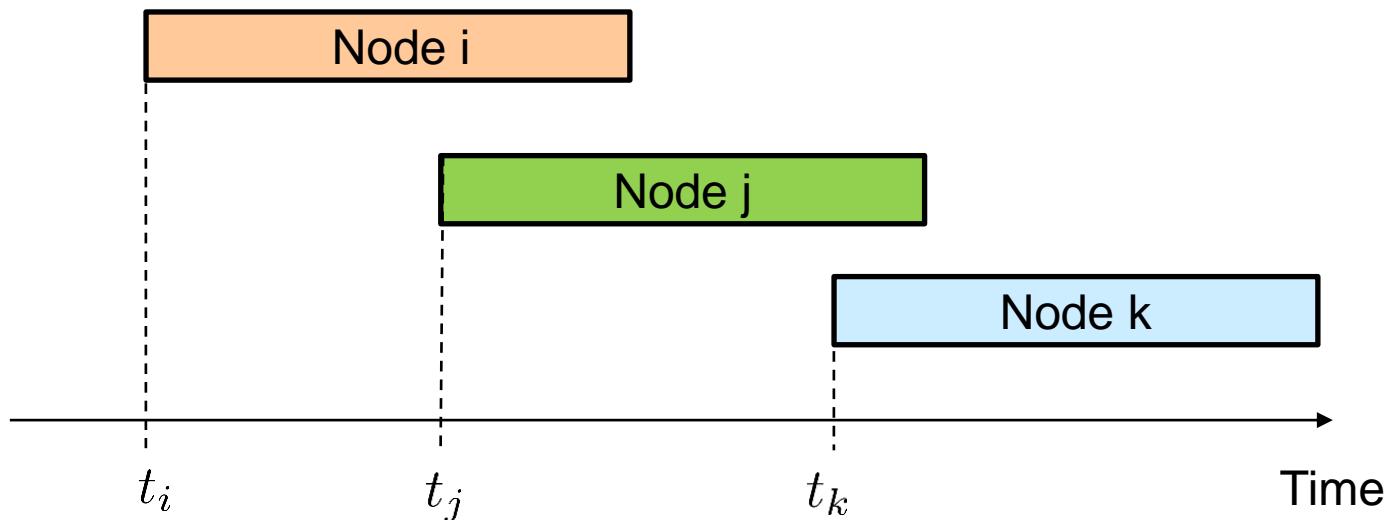


Difference between different TDMA/FDMA/CDMA



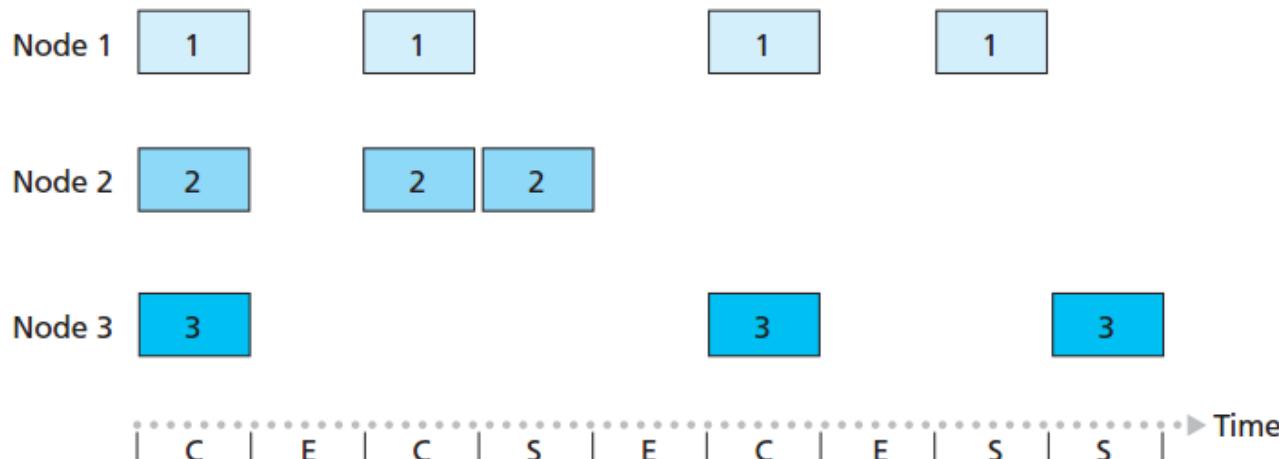
Aloha

- When you have data, send it
- If data doesn't go through, resend it after random delay
- Low efficiency: 18.5% for large N
- Suitable only for light loaded network



Slotted Aloha

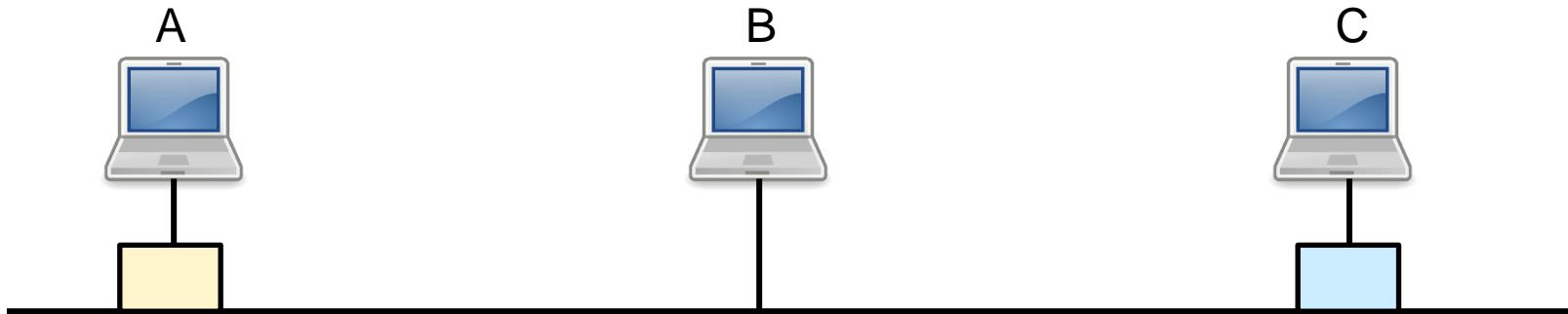
- Time is divided into equal time slots
- Sensor node can send data only at the beginning of a slot
- If have data to send, send at the start of slot. If collision, send in the next slot with probability p and do not transmit with probability $1-p$
- Requires time synchronization between nodes
- Efficiency: 37%; Better than Aloha but still low



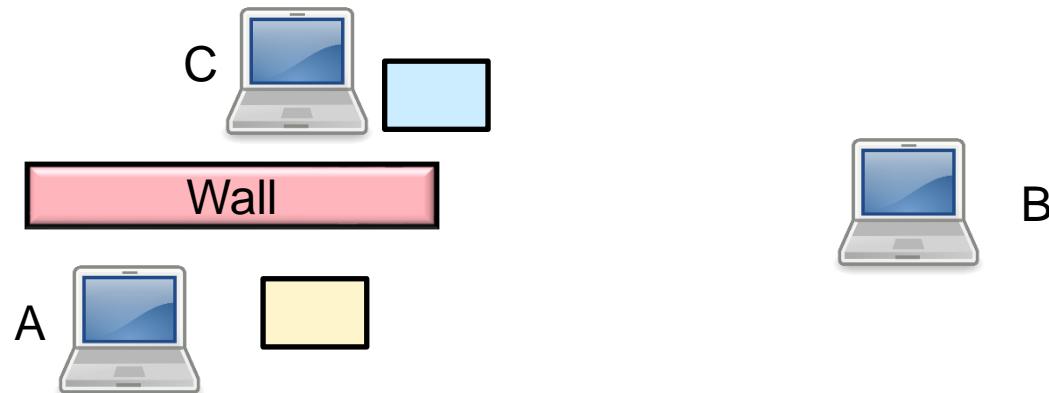
[Kurose2012]

Carrier Sense Multiple Access (CSMA)

- Listen before sending
- Send only if channel is idle
- Collisions can still happen (Hidden Node Problem)
- If collision, back-off for random delay and transmit again

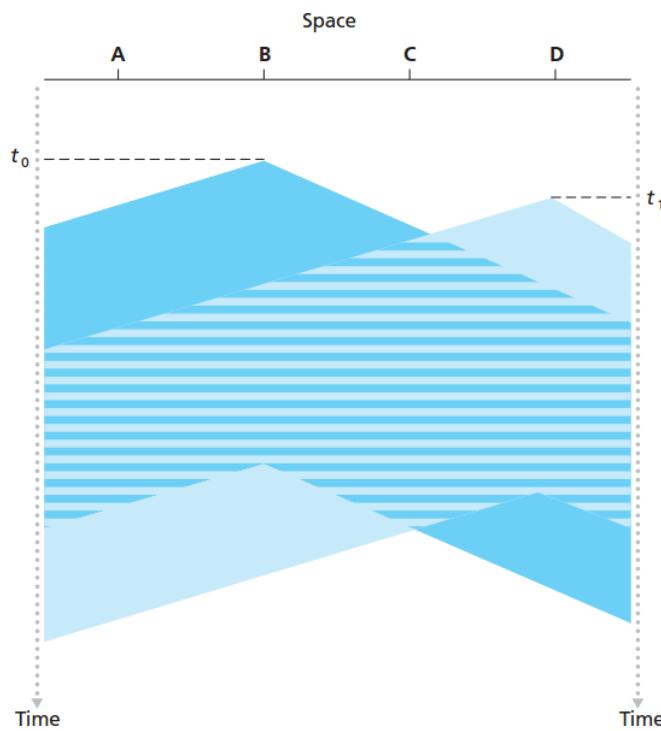


- Hidden Node Problem in Wireless Networks



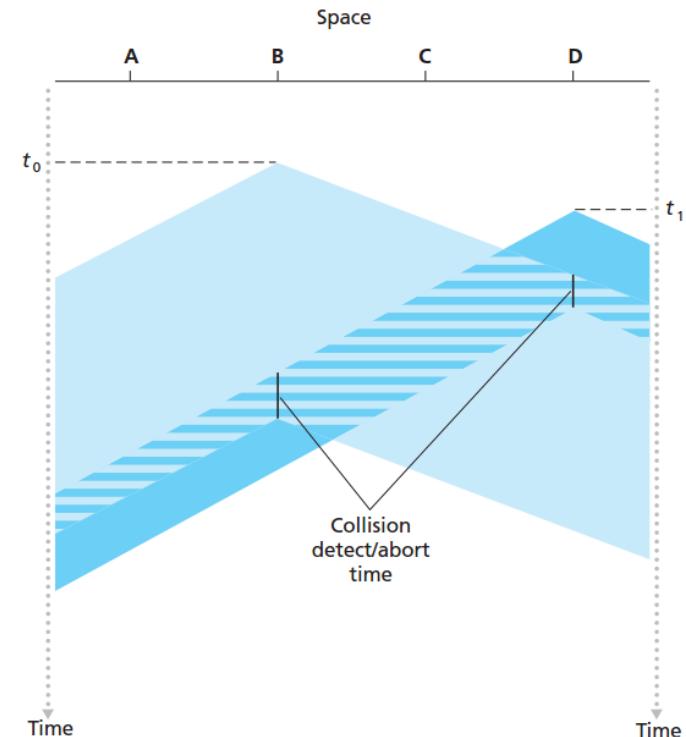
CSMA with collision detection (CD)

- Listen while transmitting!
- Stop transmitting as soon as collision is detected
- Wait for random duration before retry (binary exponential backoff)
- Improves CSMA performance at the cost of complexity
- Used in original Ethernet (wired LAN technology IEEE 802.3)



CSMA

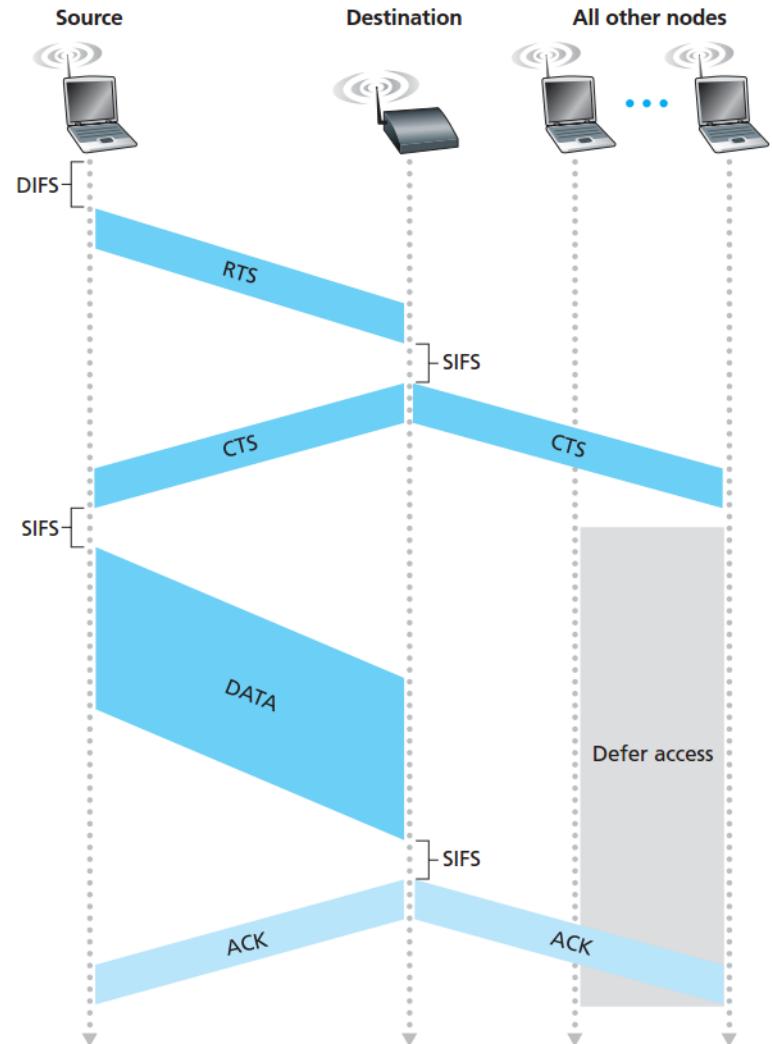
[Kurose2012]



CSMA/CD

CSMA with Collision Avoidance (CA)

- Use of ready to send (RTS) and clear to send (CTS)
 - In RTS/CTS access mode, prior to the data transmission the sending node will send a RTS packet to announce the upcoming transmission
 - When the destination node receives the RTS it will send a CTS packet after a short inter-frame space (SIFS) interval
 - Both the RTS and CTS packets are short control packets
- Used in most of the 802.11 (WLAN) technologies



Today's Class

Demand Assignment Protocols (or Taking Turn Protocols)

Motivation

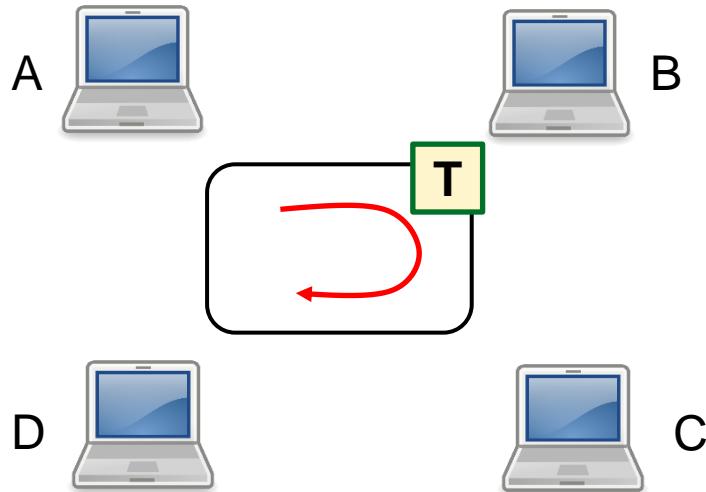
- Problem with channel partitioning
 - Inefficient at low load (idle subchannels)
- Problem with contention-based protocols
 - Inefficient at high loads (collisions)
- Taking turn protocols
 - Can improve efficiency of channel partitioning and have no collisions
 - Can potentially also offer guaranteed bandwidth, latency, etc.

Polling Protocol

- One of the nodes becomes master node
- Master node polls each node in round-robin fashion
- Node polled can transmit up to maximum number of frames
- Eliminates the collisions that plague random access protocols and empty slots in channel partitioning protocols
- Issue of single point failure, delay in polling and instructing to send
- Example: used in Bluetooth and 802.15 protocols

Token Passing (or Token Ring)

- A token is circulating in the ring and whichever node grabs that token will have right to transmit the data.
- This protocol provides fairness and eliminates collision
- Advantages: Decentralized and highly efficient
- Disadvantages: Node failure and node not releasing token
- Used in networks prior to Ethernet



Few other things!

Simplex and Duplexing Communications

- Simplex communication system: one device transmits, other listens
 - TV, FM, Surveillance monitors, wireless microphones
- Duplex communication system: both devices can transmit and receive
 - Most of the communication systems including cellphone, laptops, tablets

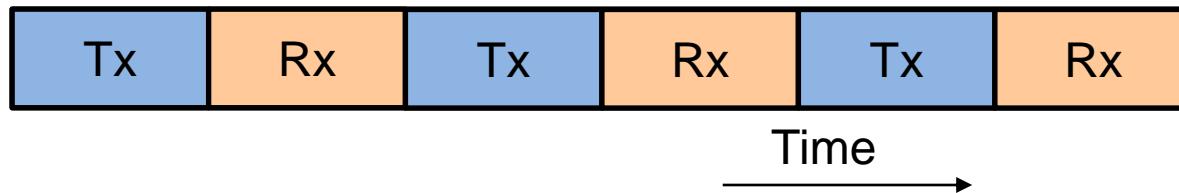
Types of Duplexing

- Half Duplex
 - Both parties cannot communicate simultaneously
 - Walkie-talkie (Push to talk button)
- Full Duplex
 - Both parties can talk simultaneously
 - Most of the communication devices

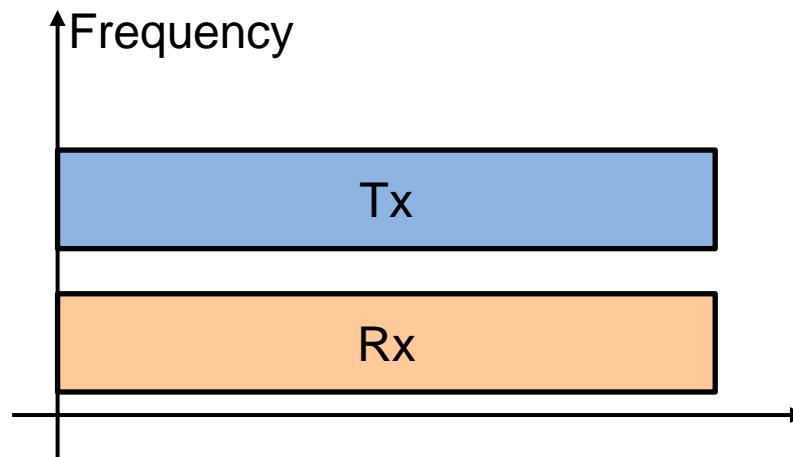
Duplexing Methods

- Methods used for dividing forward and reverse communication channels, they are called as duplexing methods such as

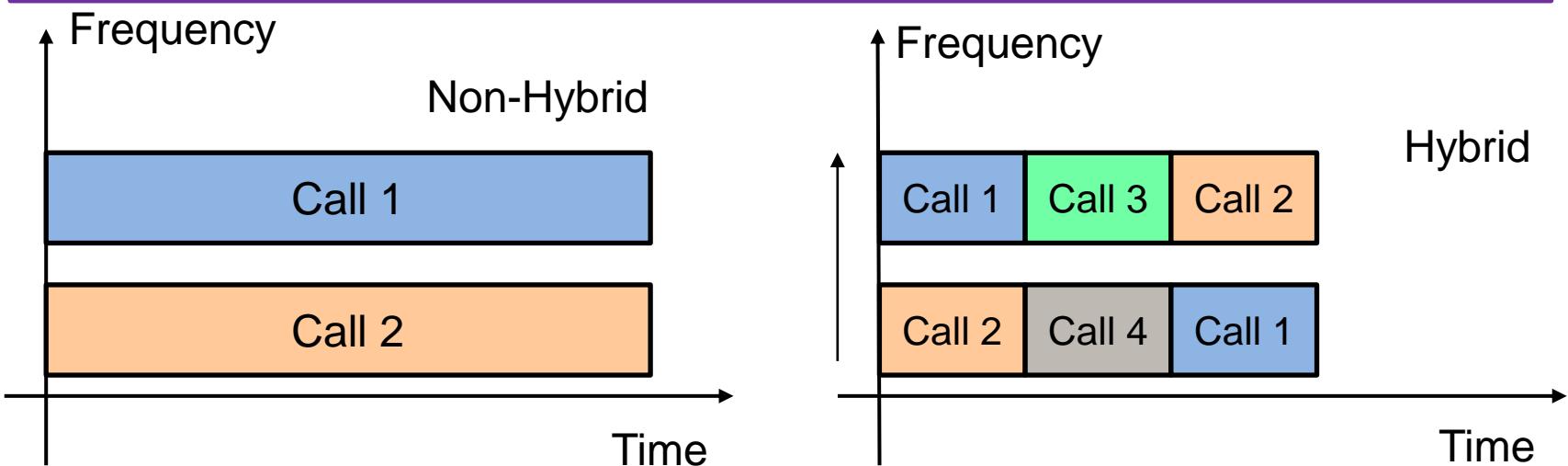
- Time division duplexing (TDD)
 - Half Duplex



- Frequency division duplexing (FDD)
 - Full Duplex

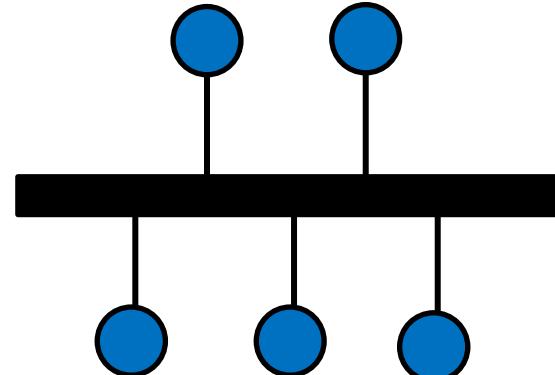


Hybrid Channel Access

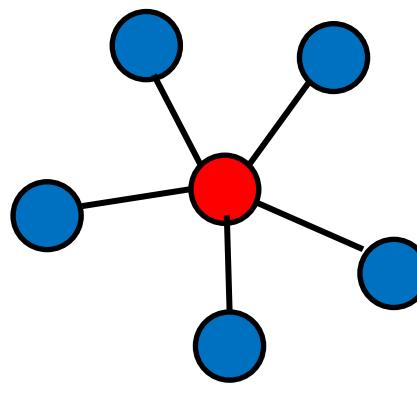


- The GSM cellular system combines the use of FDD to prevent interference between outward and return signals with FDMA and TDMA to allow multiple handsets in a single cell.
- Bluetooth packet mode communication combines frequency hopping for shared channel access among several private area networks in the same room with CSMA/CA for shared channel access inside a medium
- IEEE 802.11b WLAN are based on FDMA and DS-CDMA for avoiding interference among adjacent WLAN cells or access points

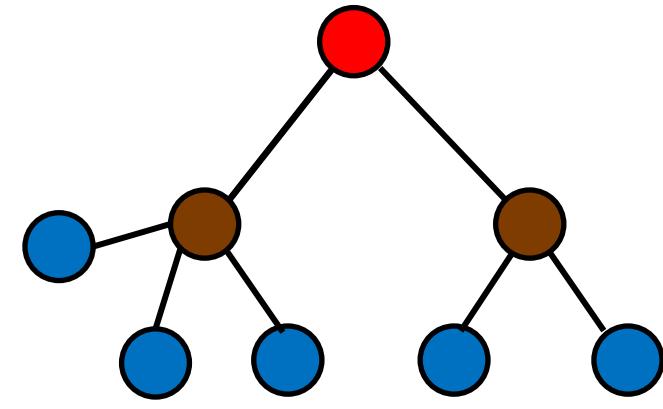
Network Topologies



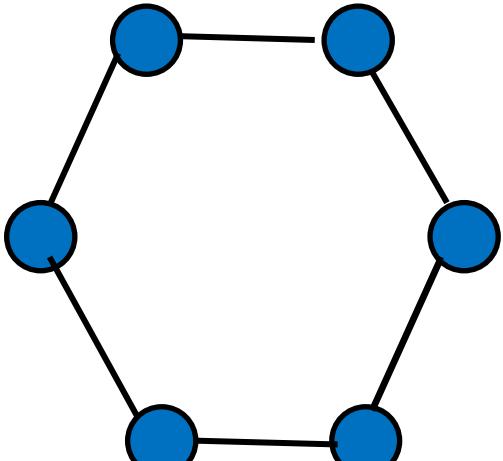
Bus



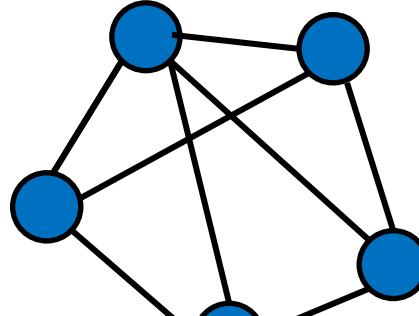
Star



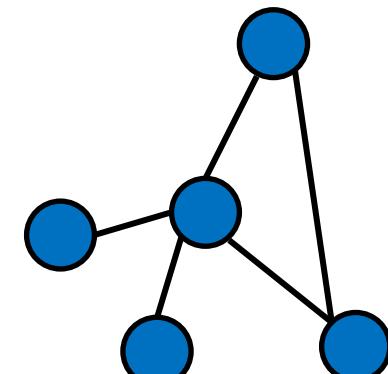
Tree



Ring



Mesh



Hybrid

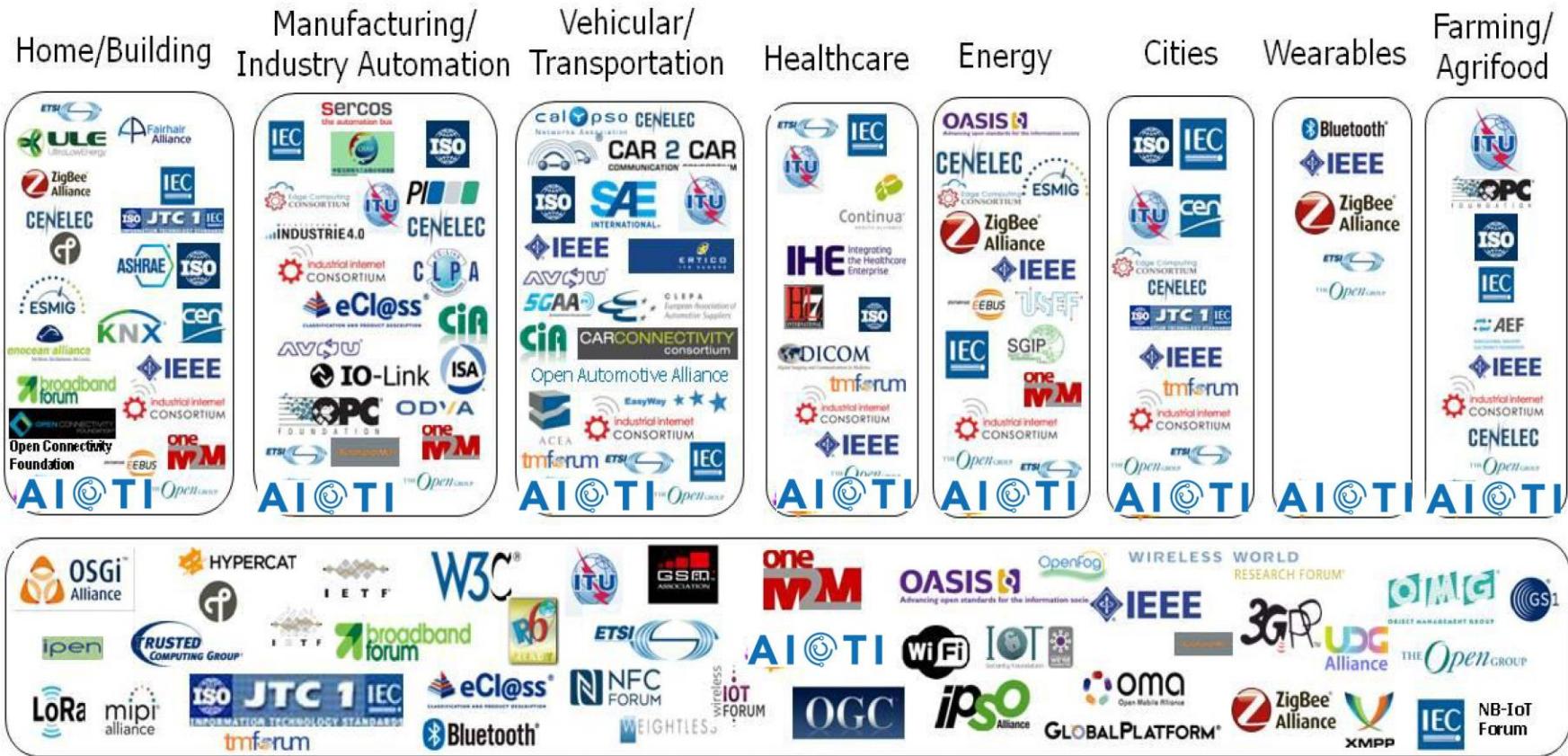
Issues in IoT from Communication Perspective

[Not an exhaustive list!]

- Low power consumption
- Support large number of devices with low data rates
- Coverage
- Quality of service
- Low cost
 - Network/Private (DIY)
 - Licensed/Unlicensed
- Privacy and security
- Standardization for interoperability between different vendors

Motivation for Interoperability

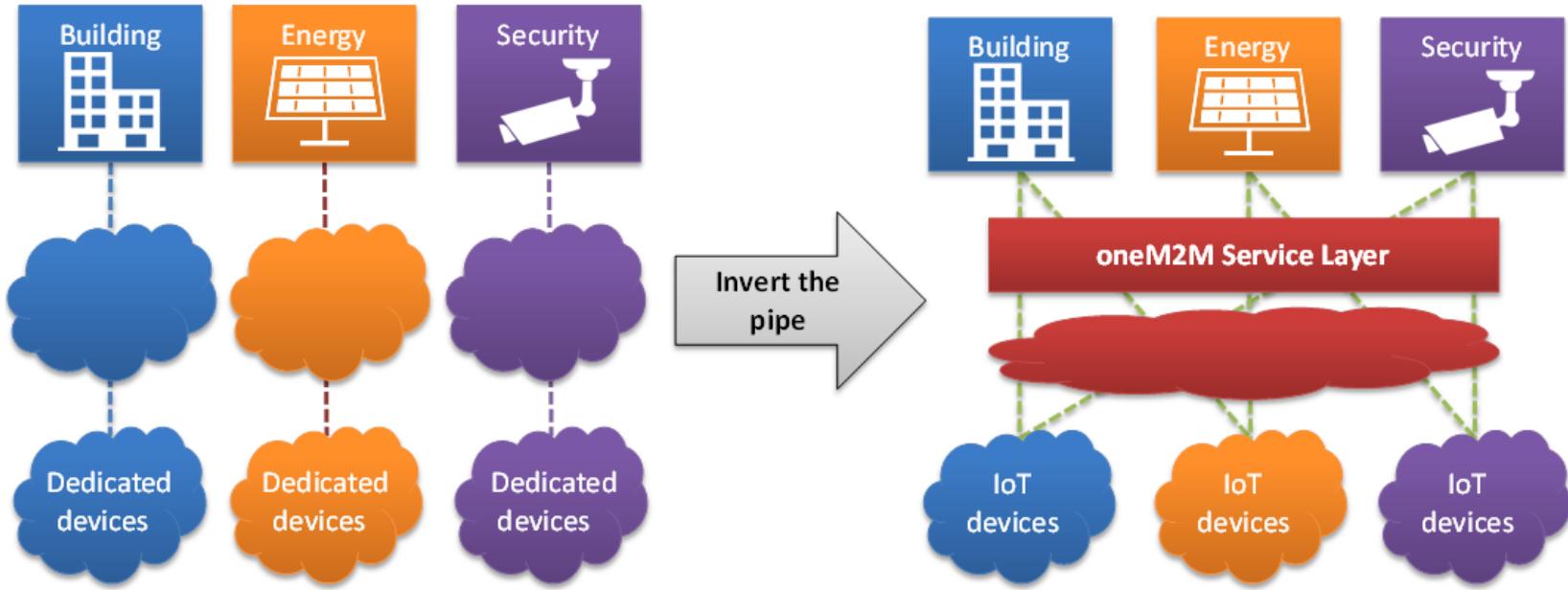
- Jungle of IoT standards



Horizontal/Telecommunication

Source: AIOTI WG3 (IoT Standardisation) – Release 2.8

oneM2M: interoperability standard



Without oneM2M

- Highly fragmented market with limited vendor-specific applications
- Reinventing the wheel: Same services developed again and again
- Each silo contains its own technologies without interoperability

With oneM2M

- End-to-end platform: common service capabilities layer
- Interoperability at the level of data and control exchanges via uniform APIs
- Seamless interaction between heterogeneous applications and devices

Factors contributing to energy waste/expense

- Energy consumption in transmission
 - Longer distances
 - Higher frequencies
 - More bandwidth
- Energy waste
 - Excessive overhead
 - Idle listening
 - Overhearing
 - Packet collisions and retransmissions

[Not exhaustive!]

Ways to Reduce Energy Waste/Consumption

- Reduced frequency/data rate/ bandwidth/ coverage
- Sleep
 - Low duty cycle
- Energy saving protocols
 - Schedule based (reduction in over-hearing and idle-listening)
 - Licensed spectrum; BLE
 - Contention based (less overhead and no need of synchronization)
 - Zigbee, WiFi
- Multihop and aggregation of data
- Signal processing
 - censoring, predictive filters
- Reduced overhead

Low Duty Cycle

Perform tasks

150mA
Average

Power consumption is averaged over the cycle (example current draw)

Sleep

2mA Average

Perform tasks

150mA
Average

<https://core-electronics.com.au/media/wysiwyg/tutorials/sam/example-duty-cycle.png>

Duty cycle in our paper 0.66% duty cycle (0.2 of 30ms).

Even with 99% reduction in data-transmissions, the life-time increased only by 3 times

A. Shastri, V. Jain, R. Singh, **S. Chaudhari**, S. Chouhan, S. Werner, "Improving the Accuracy of the Shewhart Test-based Data-Reduction Technique using Piggybacking," in *IEEE WF-IoT*, Ireland, Apr. 2019

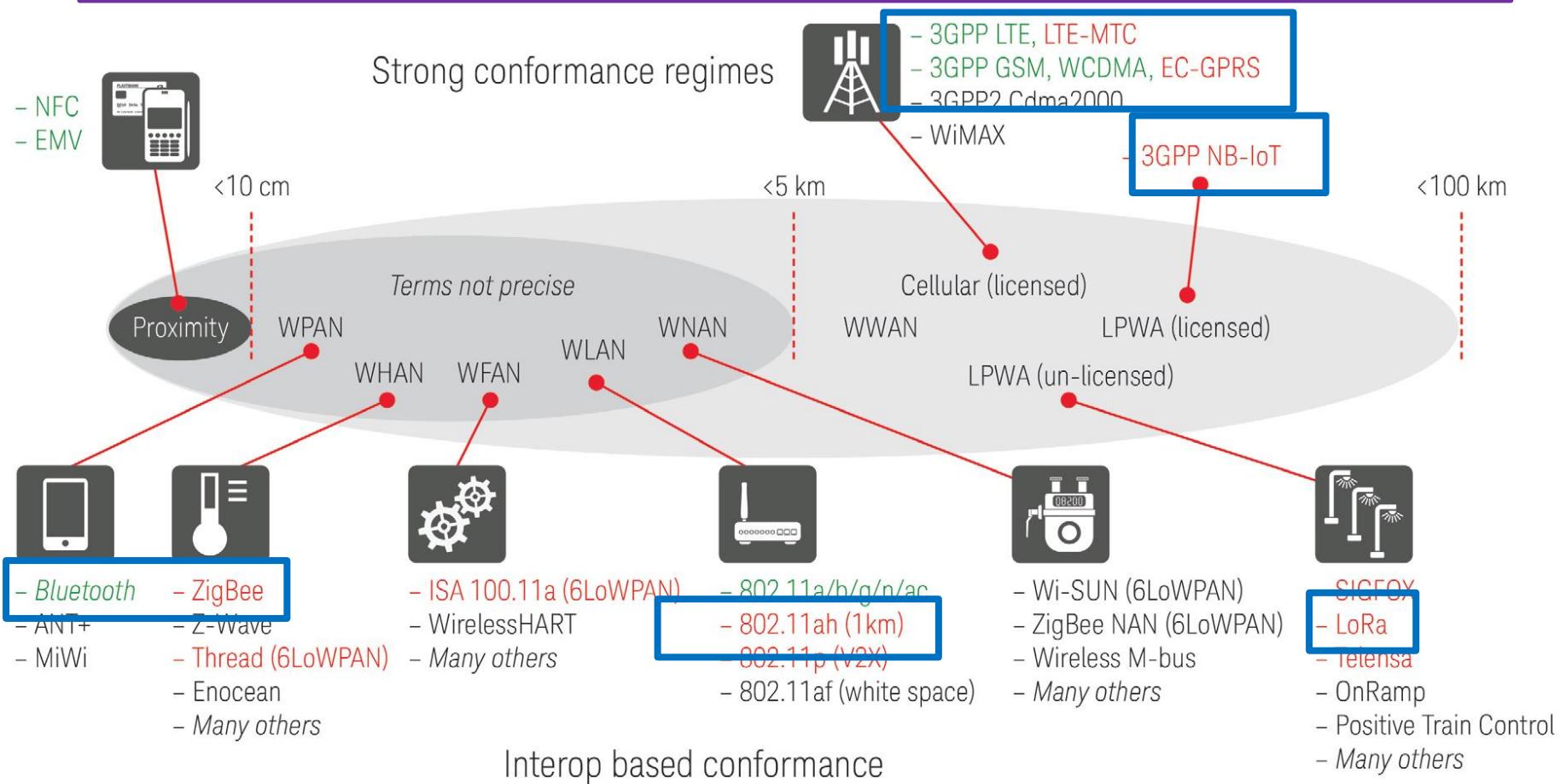
Questions?

Ungraded Quiz

- TV is a duplex system
 - True
 - False
- Lack of interoperability means
 - Vendor Lockin
 - Costly to built different verticals
 - Difficult to exchange data between different verticals
 - All of the above
- In which network topology, loops are allowed
 - Star
 - Cluster Tree
 - Mesh
 - None of the above

Communication Techniques for IoT

Communication Techniques for IoT



■ : > Billion units/year now
■ : Emerging

WPAN: Wireless Personal Area Network

WHAN: Wireless Home Area Network

WFAN: Wireless Field (or Factory) Area Network

WLAN: Wireless Local Area Network

WNAN: Wireless Neighborhood Area Network

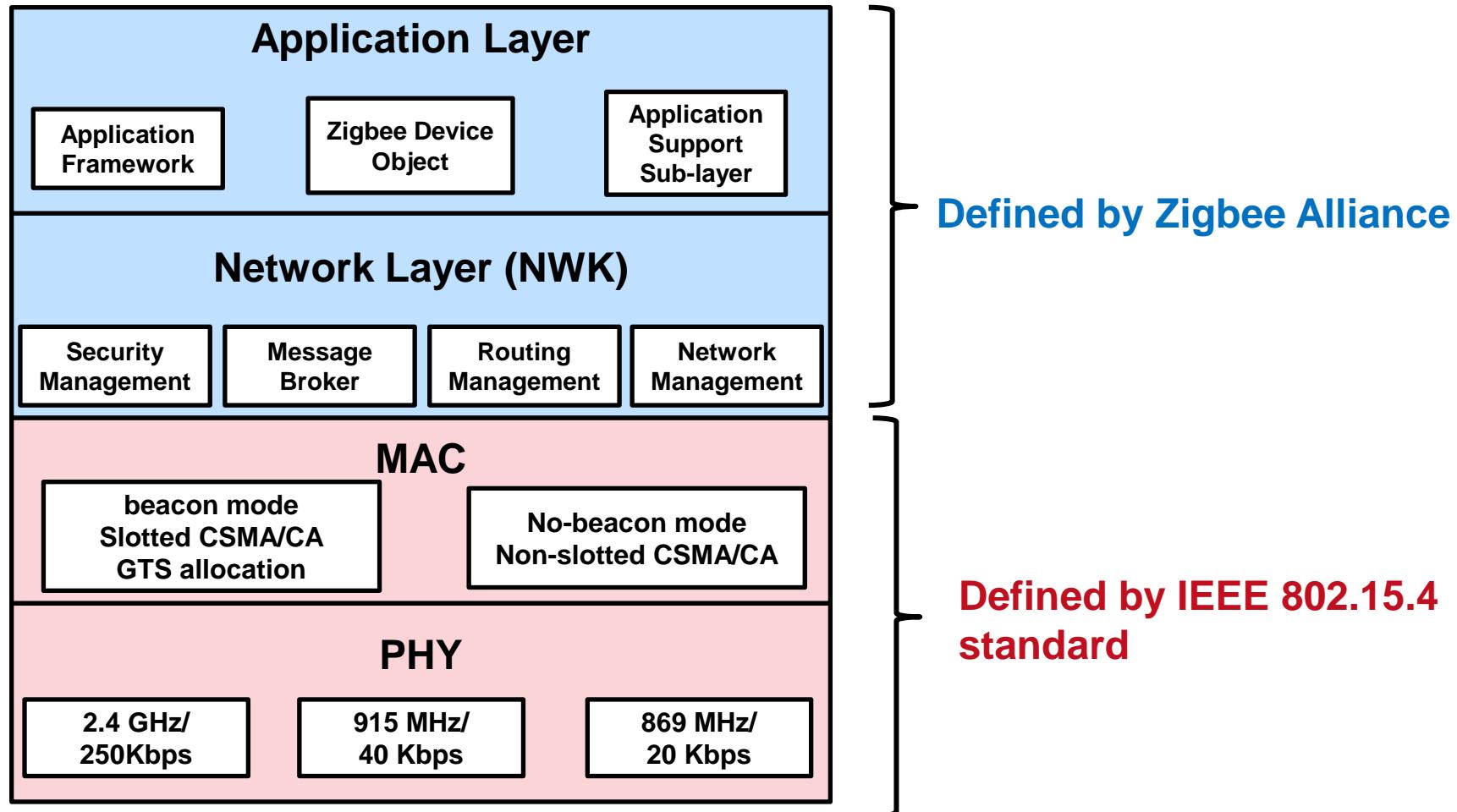
WWAN: Wireless Wide Area Network

LPWA: Low Power Wide Area

IEEE 802.15.4

Ref: K. Sohraby, D. Minoli, T. Znati, *Wireless Sensor Networks*, Wiley, 2007

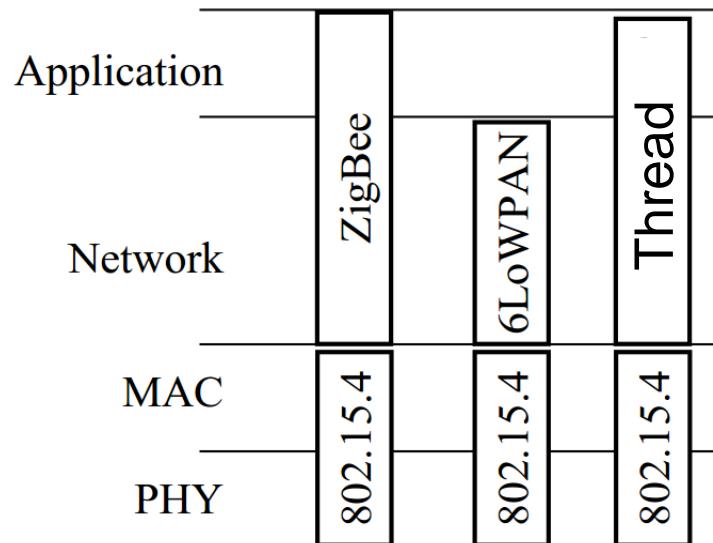
IEEE 802.15.4/Zigbee Protocol Stack



- Full protocol stack for low power, low rate and low cost wireless communications. Also applicable to Low rate WPAN – LR-WPAN.

IEEE 802.15.4

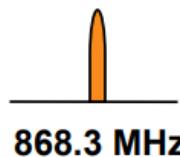
- IEEE 802.15.4 defines the operation of low-rate wireless personal area networks (LR-WPANs)
- Widely used in wireless sensor-network (WSN) applications
 - Vast number of industrial, home and medical applications
- It specifies the physical layer (PHY) and media access control (MAC) for LR-WPANs
- Does not have IP address
- Used by several “Internet of Things” protocols:
 - ZigBee, 6LowPAN, Thread, WiSuN etc.



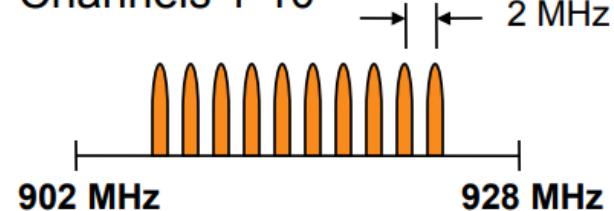
Physical Layer (PHY): Operating Frequency Bands

**868MHz/915MHz
PHY**

Channel 0

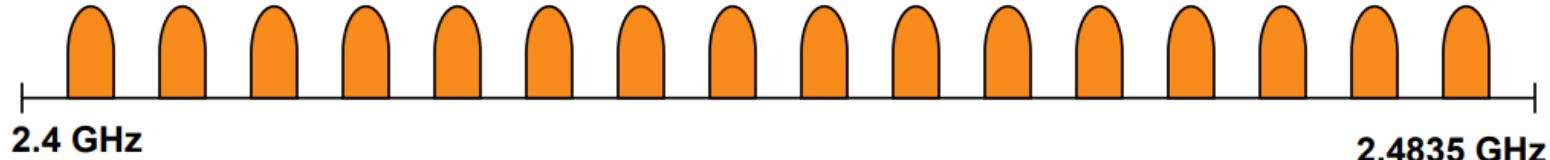


Channels 1-10



**2.4 GHz
PHY**

Channels 11-26



PHY: Frequency Bands Worldwide

Channel	Center Frequency (MHz)	Availability
868 MHz Band	0	868.3
915 MHz Band	1	906
	2	908
	3	910
	4	912
	5	914
	6	916
	7	918
	8	920
	9	922
	10	924
2.4 GHz Band	11	2405
	12	2410
	13	2415
	14	2420
	15	2425
	16	2430
	17	2435
	18	2440
	19	2445
	20	2450
	21	2455
	22	2460
	23	2465
	24	2470
	25	2475
	26	2480

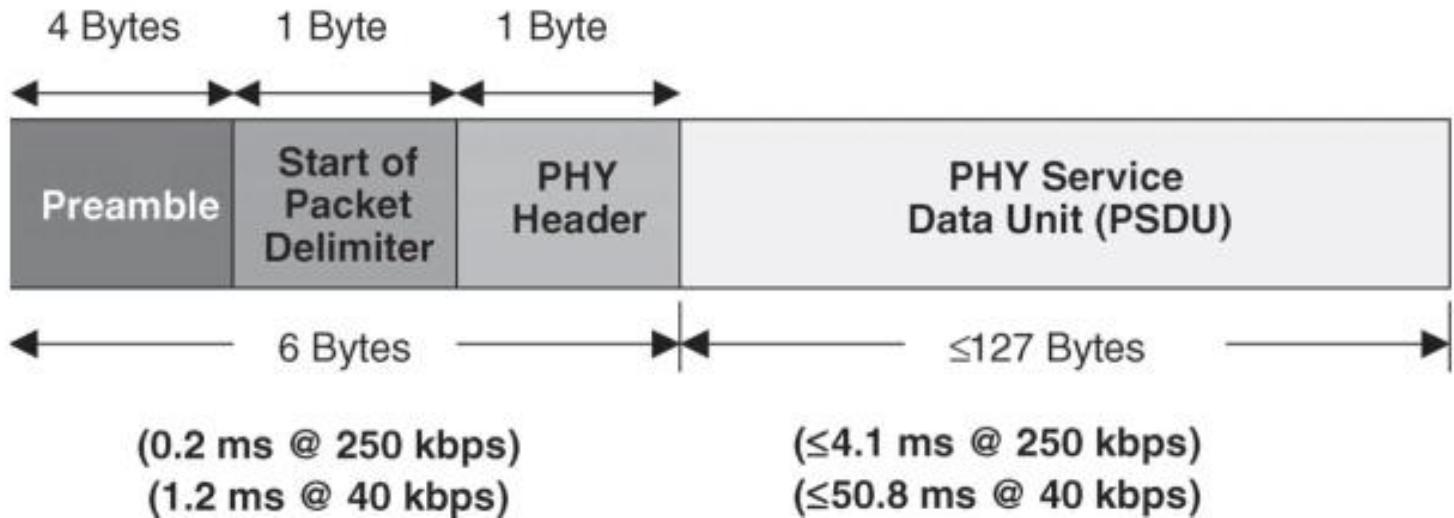
PHY: Modulation Parameters

Freq. band (MHz)	Spreading Parameters		Data Parameters		
	Chip rate (kchip/s)	Modulation	Bit rate (kbps)	Symbol rate (ksymbol/s)	Symbols
868	300	BPSK	20	20	Binary
915	600	BPSK	40	40	Binary
2400	2000	O-QPSK	250	62.5	16-ary

[Koubaa2007]

All bands are based on Direct sequence spread spectrum (DSSS),
a form of CDMA

PHY-layer packet structure



- Preamble -> Symbol synchronization
- Packet delimiter -> Frame synchronization
- PHY header: length of the PSDU
- PSDU can carry upto 127 bytes

Additional Tasks of PHY of IEEE 802.15.4

- **Activation and deactivation of the radio transceiver**
 - Three states: Transmitting, receiving and sleeping
- **Receiver energy detection**
 - No decoding or signal identification
 - Required to understand if the channel is busy or idle
- **Link quality indication**
 - Using energy or SNR estimation or both
- **Clear channel assessment**
 - Energy detection or carrier sense or both
- **Channel frequency selection**
 - 27 channels

Questions?

Communications & Controls in IoT

IEEE 802.15.4 and WLAN

Instructor: Sachin Chaudhari

Feb. 09, 2023



**INTERNATIONAL INSTITUTE OF
INFORMATION TECHNOLOGY**

HYDERABAD

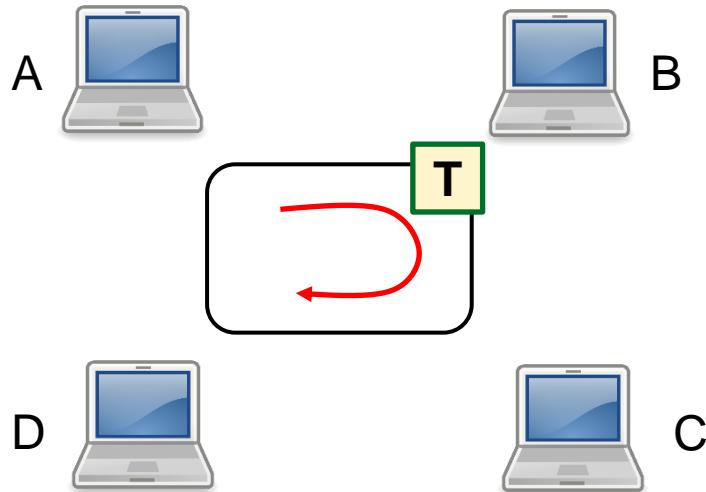
Recap: *Communication Techniques for IoT*

Polling Protocol

- One of the nodes becomes master node
- Master node polls each node in round-robin fashion
- Node polled can transmit up to maximum number of frames
- Eliminates the collisions that plague random access protocols and empty slots in channel partitioning protocols
- Issue of single point failure, delay in polling and instructing to send
- Example: used in Bluetooth and 802.15 protocols

Token Passing (or Token Ring)

- A token is circulating in the ring and whichever node grabs that token will have right to transmit the data.
- This protocol provides fairness and eliminates collision
- Advantages: Decentralized and highly efficient
- Disadvantages: Node failure and node not releasing token
- Used in networks prior to Ethernet



Simplex and Duplexing Communications

- Simplex communication system: one device transmits, other listens
 - TV, FM, Surveillance monitors, wireless microphones
- Duplex communication system: both devices can transmit and receive
 - Most of the communication systems including cellphone, laptops, tablets

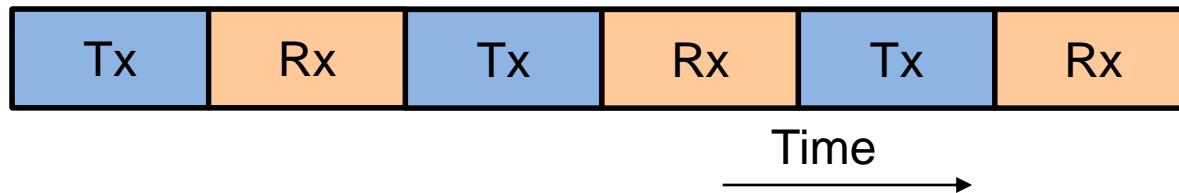
Types of Duplexing

- Half Duplex
 - Both parties cannot communicate simultaneously
 - Walkie-talkie (Push to talk button)
- Full Duplex
 - Both parties can talk simultaneously
 - Most of the communication devices

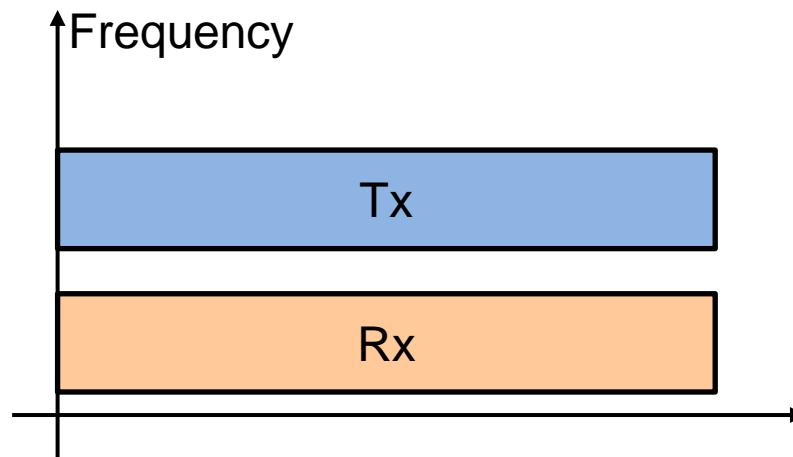
Duplexing Methods

- Methods used for dividing forward and reverse communication channels, they are called as duplexing methods such as

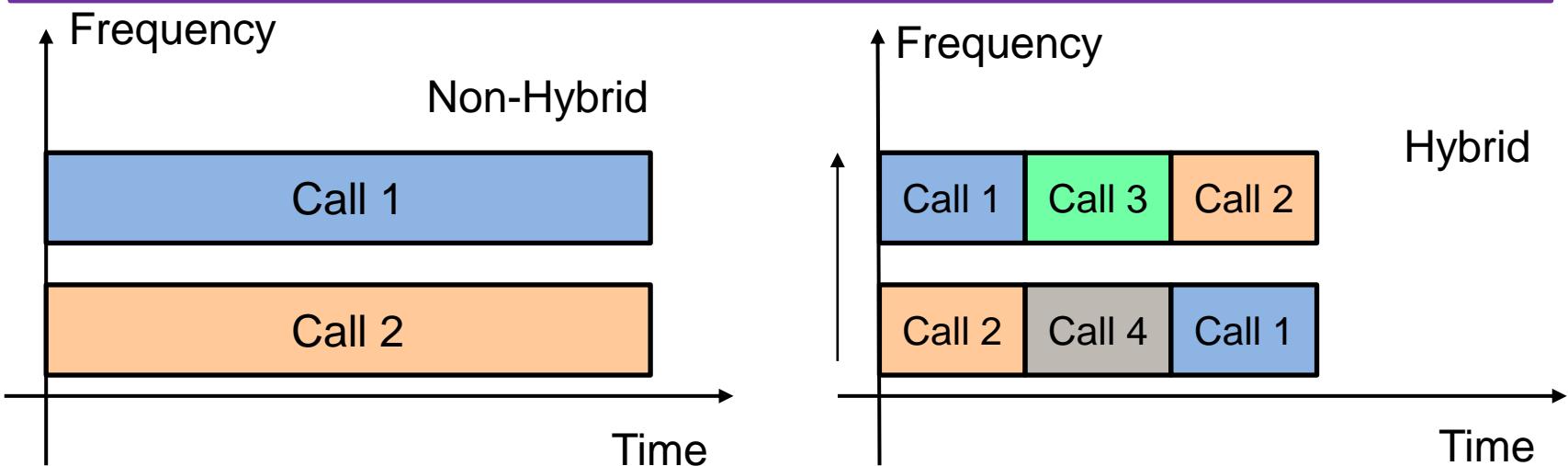
- Time division duplexing (TDD)
 - Half Duplex



- Frequency division duplexing (FDD)
 - Full Duplex

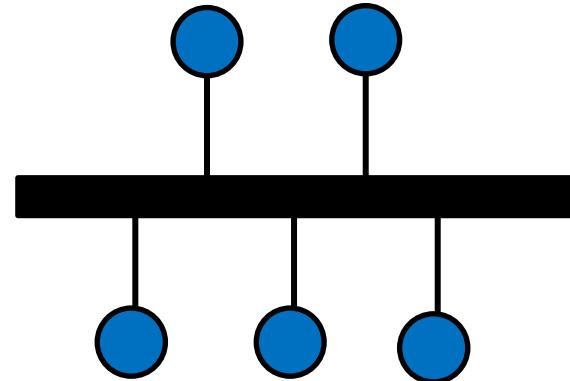


Hybrid Channel Access

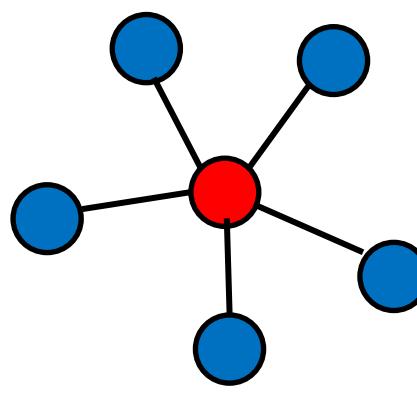


- The GSM cellular system combines the use of FDD to prevent interference between outward and return signals with FDMA and TDMA to allow multiple handsets in a single cell.
- Bluetooth packet mode communication combines frequency hopping for shared channel access among several private area networks in the same room with CSMA/CA for shared channel access inside a medium
- IEEE 802.11b WLAN are based on FDMA and DS-CDMA for avoiding interference among adjacent WLAN cells or access points

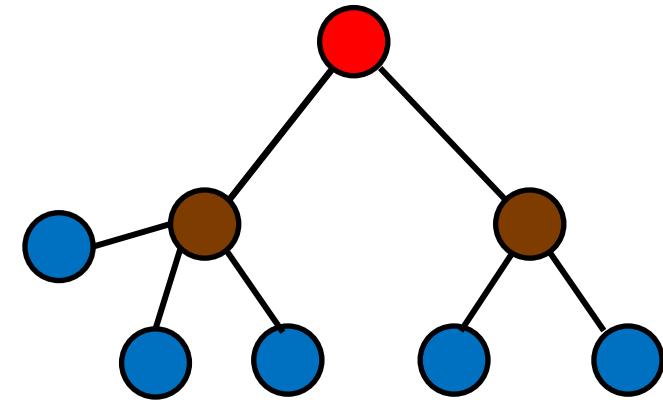
Network Topologies



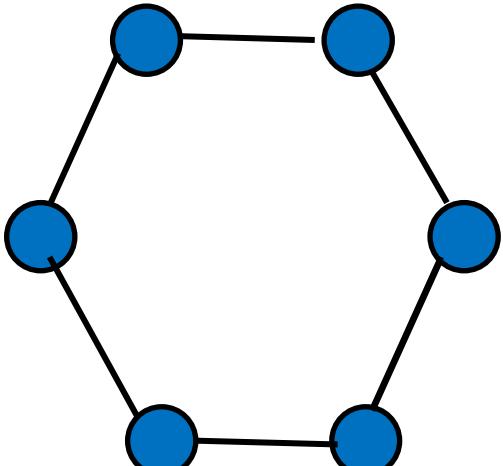
Bus



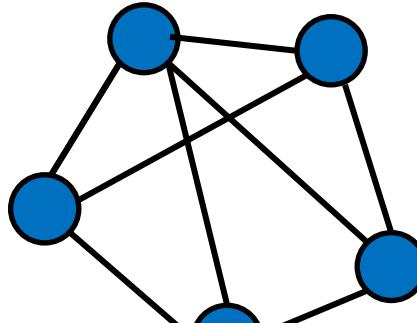
Star



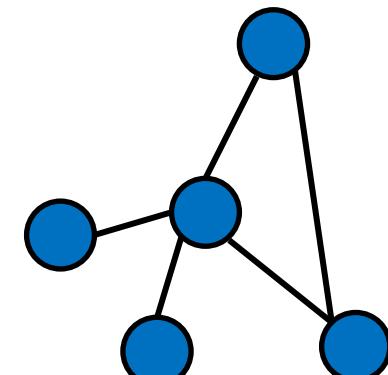
Tree



Ring



Mesh



Hybrid

Issues in IoT from Communication Perspective

[Not an exhaustive list!]

- Low power consumption
- Support large number of devices with low data rates
- Coverage
- Quality of service
- Low cost
 - Network/Private (DIY)
 - Licensed/Unlicensed
- Privacy and security
- Standardization for interoperability between different vendors

Factors contributing to energy waste/expense

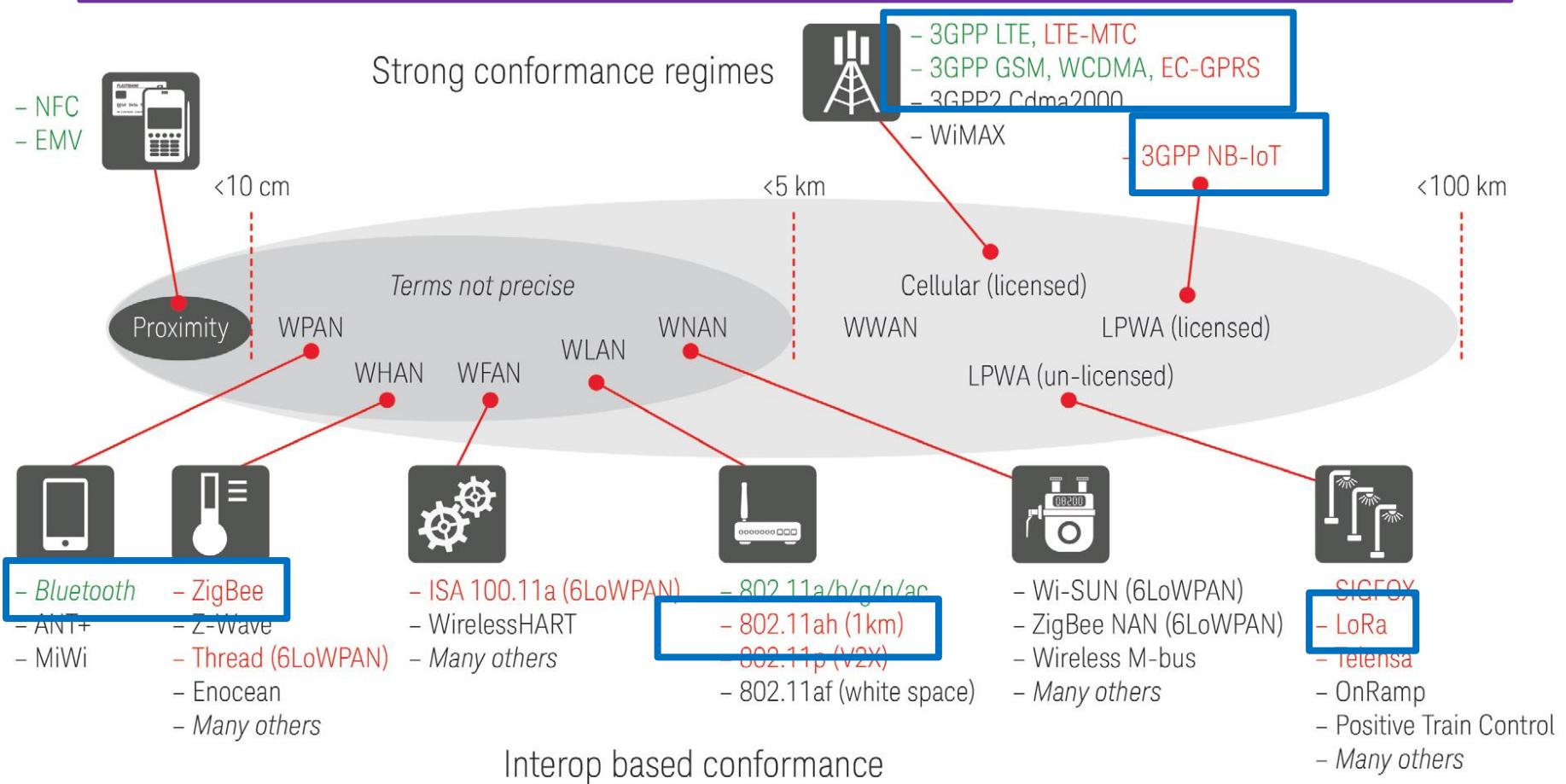
- Energy consumption in transmission
 - Longer distances
 - Higher frequencies
 - More bandwidth
- Energy waste
 - Excessive overhead
 - Idle listening
 - Overhearing
 - Packet collisions and retransmissions

[Not exhaustive!]

Ways to Reduce Energy Waste/Consumption

- Reduced frequency/data rate/ bandwidth/ coverage
- Sleep
 - Low duty cycle
- Energy saving protocols
 - Schedule based (reduction in over-hearing and idle-listening)
 - Licensed spectrum; BLE
 - Contention based (less overhead and no need of synchronization)
 - Zigbee, WiFi
- Multihop and aggregation of data
- Signal processing
 - censoring, predictive filters
- Reduced overhead

Communication Techniques for IoT



■ : > Billion units/year now
■ : Emerging

WPAN: Wireless Personal Area Network

WHAN: Wireless Home Area Network

WFAN: Wireless Field (or Factory) Area Network

WLAN: Wireless Local Area Network

WNAN: Wireless Neighborhood Area Network

WWAN: Wireless Wide Area Network

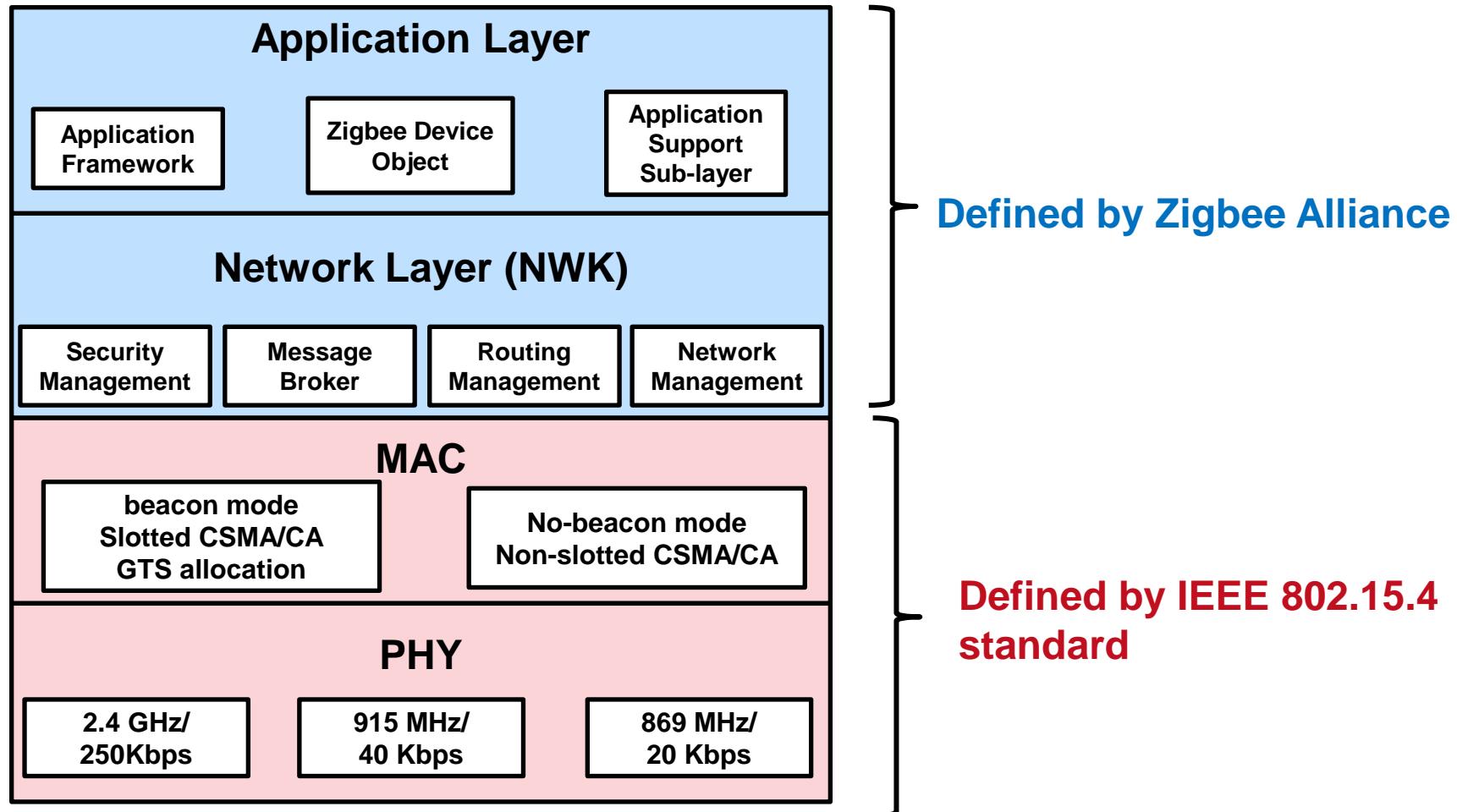
LPWA: Low Power Wide Area

Today's Class

IEEE 802.15.4

Ref: K. Sohraby, D. Minoli, T. Znati, *Wireless Sensor Networks*, Wiley, 2007

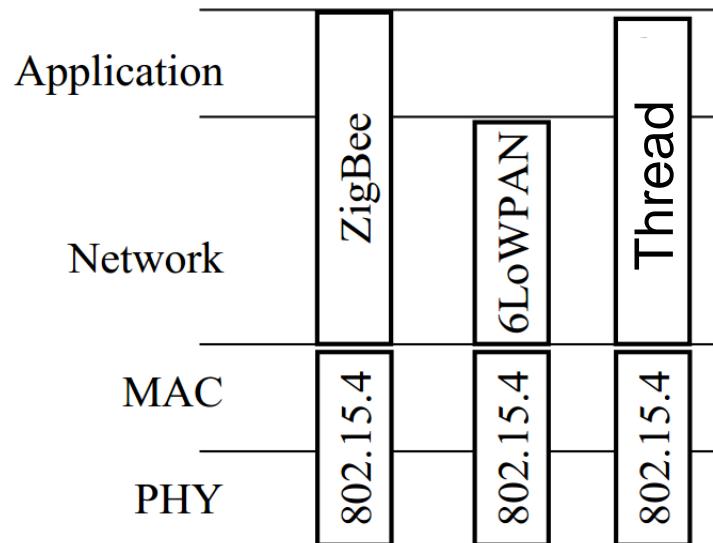
IEEE 802.15.4/Zigbee Protocol Stack



- Full protocol stack for low power, low rate and low cost wireless communications. Also applicable to Low rate WPAN – LR-WPAN.

IEEE 802.15.4

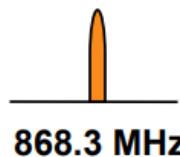
- IEEE 802.15.4 defines the operation of low-rate wireless personal area networks (LR-WPANs)
- Widely used in wireless sensor-network (WSN) applications
 - Vast number of industrial, home and medical applications
- It specifies the physical layer (PHY) and media access control (MAC) for LR-WPANs
- Does not have IP address
- Used by several “Internet of Things” protocols:
 - ZigBee, 6LowPAN, Thread, WiSuN etc.



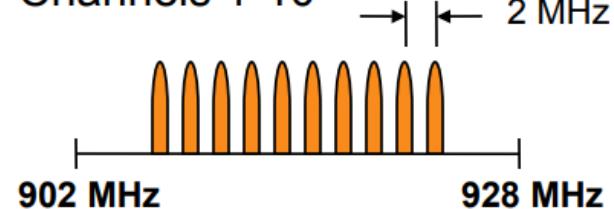
Physical Layer (PHY): Operating Frequency Bands

**868MHz/915MHz
PHY**

Channel 0

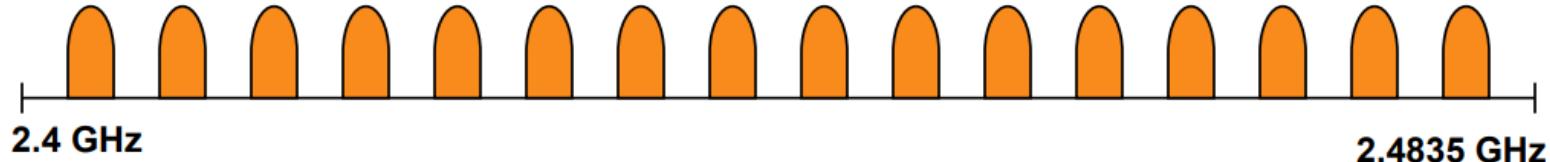


Channels 1-10



**2.4 GHz
PHY**

Channels 11-26



PHY: Frequency Bands Worldwide

Channel	Center Frequency (MHz)	Availability
868 MHz Band	0	868.3
915 MHz Band	1	906
	2	908
	3	910
	4	912
	5	914
	6	916
	7	918
	8	920
	9	922
	10	924
2.4 GHz Band	11	2405
	12	2410
	13	2415
	14	2420
	15	2425
	16	2430
	17	2435
	18	2440
	19	2445
	20	2450
	21	2455
	22	2460
	23	2465
	24	2470
	25	2475
	26	2480

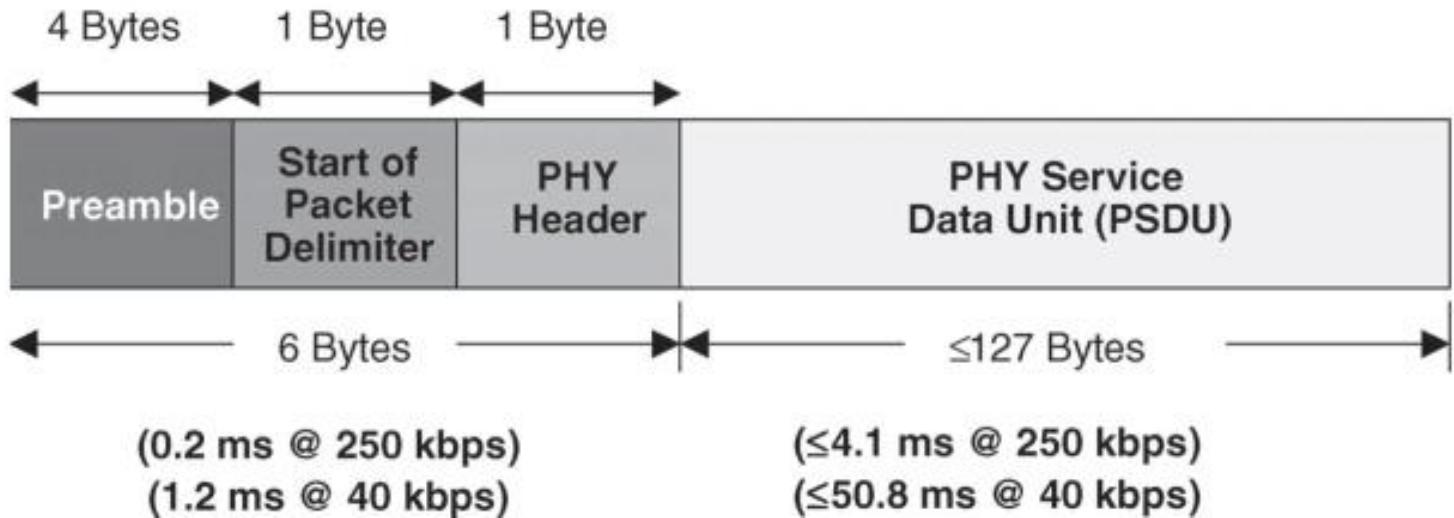
PHY: Modulation Parameters

Freq. band (MHz)	Spreading Parameters		Data Parameters		
	Chip rate (kchip/s)	Modulation	Bit rate (kbps)	Symbol rate (ksymbol/s)	Symbols
868	300	BPSK	20	20	Binary
915	600	BPSK	40	40	Binary
2400	2000	O-QPSK	250	62.5	16-ary

[Koubaa2007]

All bands are based on Direct sequence spread spectrum (DSSS),
a form of CDMA

PHY-layer packet structure



- Preamble -> Symbol synchronization
- Packet delimiter -> Frame synchronization
- PHY header: length of the PSDU
- PSDU can carry upto 127 bytes

Additional Tasks of PHY of IEEE 802.15.4

- **Activation and deactivation of the radio transceiver**
 - Three states: Transmitting, receiving and sleeping
- **Receiver energy detection**
 - No decoding or signal identification
 - Required to understand if the channel is busy or idle
- **Link quality indication**
 - Using energy or SNR estimation or both
- **Clear channel assessment**
 - Energy detection or carrier sense or both
- **Channel frequency selection**
 - 27 channels

MAC Layer features

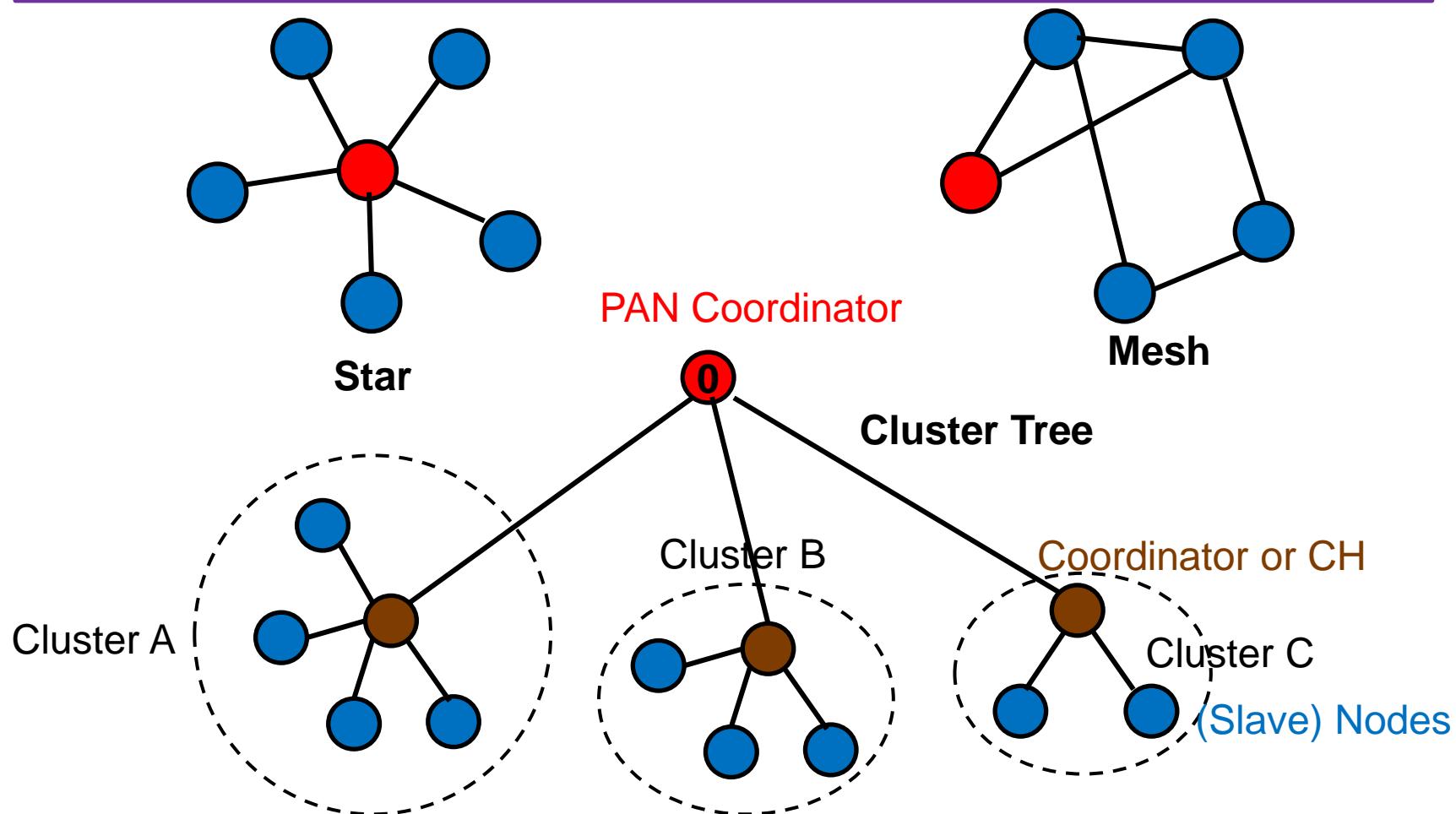
- Designed to support vast number of industrial and home applications for control and monitoring
- Enabling deployment of large number of devices with low cost and complexity
- Several features for flexible network configuration and low-power operation
 - Different topologies and network devices
 - Optional superframe structure with duty-cycle control
 - Both contention and scheduled based MAC protocols
 - Synchronized and non-synchronized operation
 - Efficient energy management
 - Adaptive sleep
 - Extended sleeping time
 - Flexible addressing scheme for large number of nodes

MAC Layer: *Device Types*

Two kind of devices in IEEE 802.15.4 based on complexity and capability

- Fully functional devices (FFD)
 - More resources
 - Multiple network responsibilities
- Reduced functionality devices (RFD)
 - Simple and low-cost device
 - Can only communicate with one FFD

Topologies: Zigbee (Network Layer)



- 16 bit addresses support 65536 devices in a PAN. For clusters, 255 clusters with 254 nodes each
- Self recovering ability

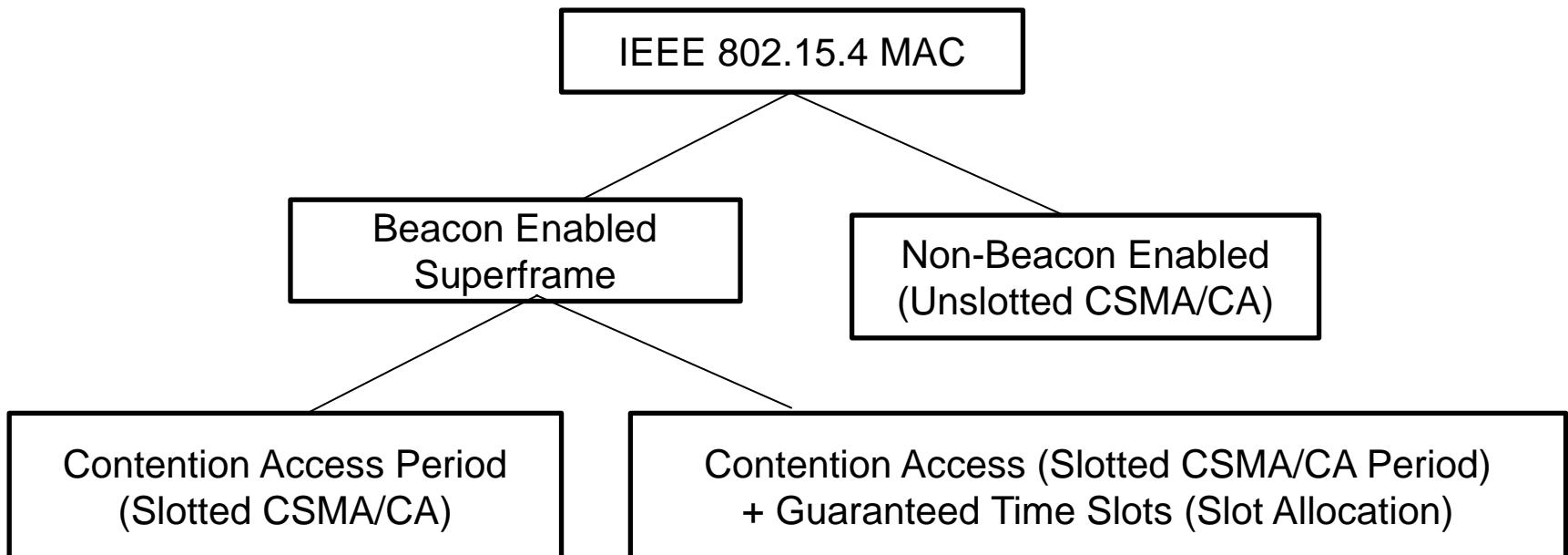
Zigbee Node Types

Zigbee defines three kinds of logical devices

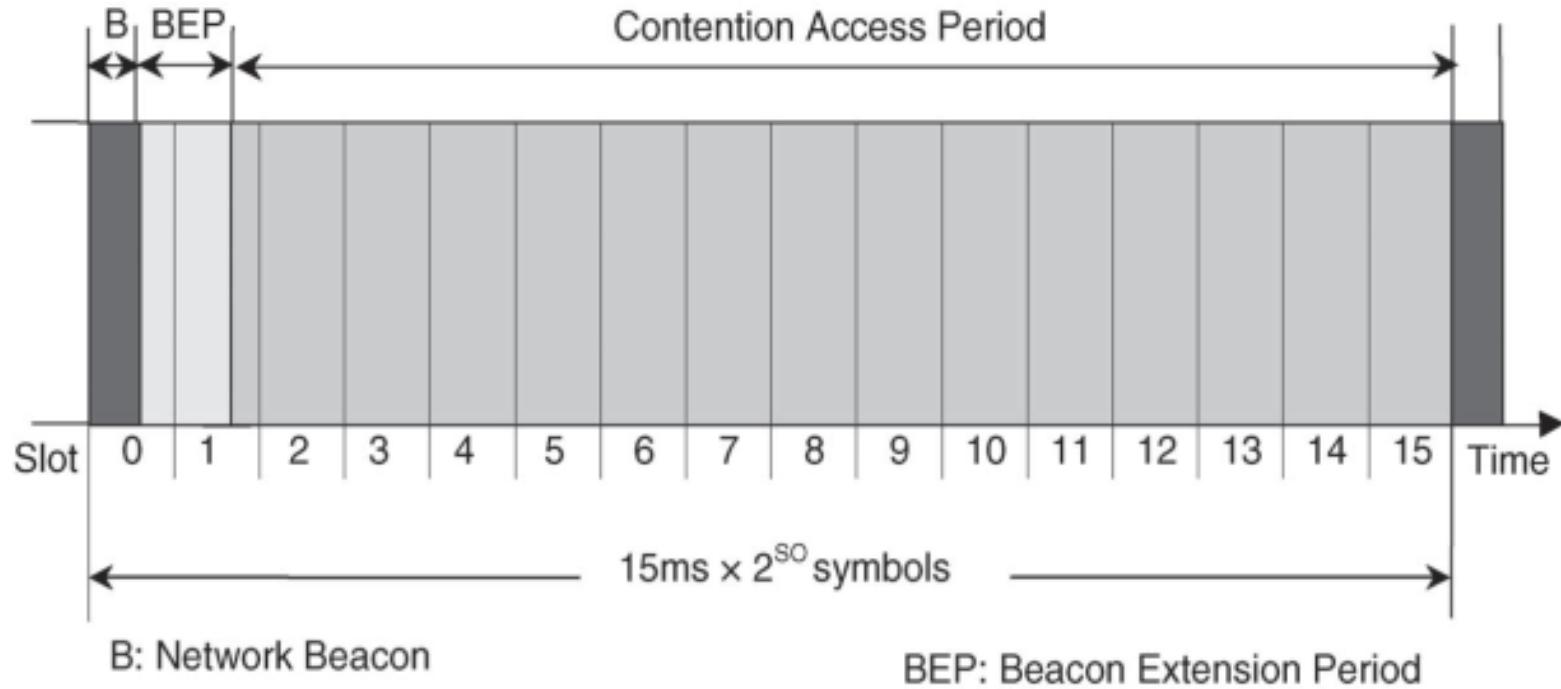
- **PAN coordinator or Master**
 - Principal controller of network
 - Managing list of all network devices or nodes
 - Identifies PAN and nodes associated with it
 - Provides global synchronization by transmitting beacon frames containing relevant information
- **Coordinator or cluster head (CH)**
 - Same functionalities as PAN coordinator locally in cluster
 - Managing association and disassociation of other nodes to PAN
 - Does not create its PAN
- **Simple (Slave) Nodes**
 - No coordination functionalities
- PAN Coordinator and CH are **FFD** while slave nodes are **RFD**

MAC layer functions

- Network association and disassociation
- Two modes of operation
 - Beaconing
 - Non-beaconing

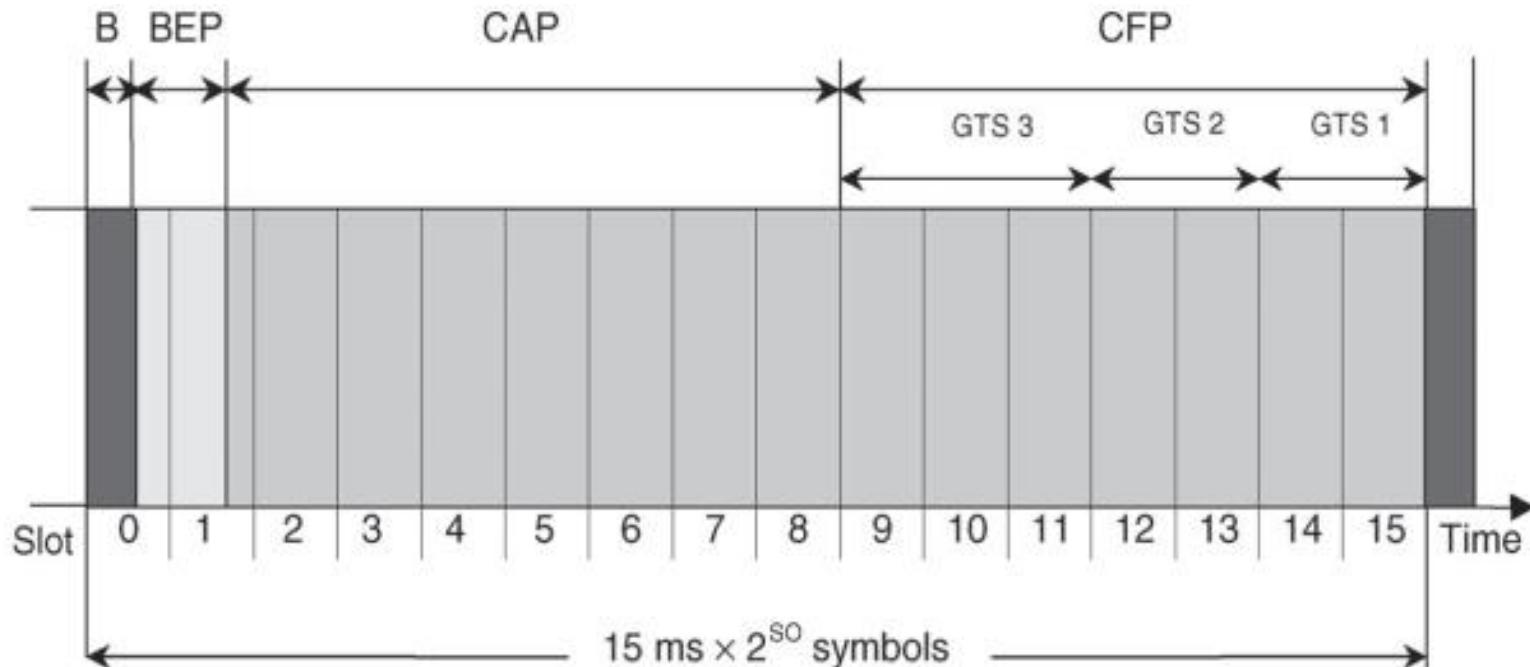


MAC: Superframe Structure



[Sohraby2007]

MAC: QoS Superframe Structure



CAP: Contention Access Period

GTS: Guaranteed Time Slot

SO: Superframe Order

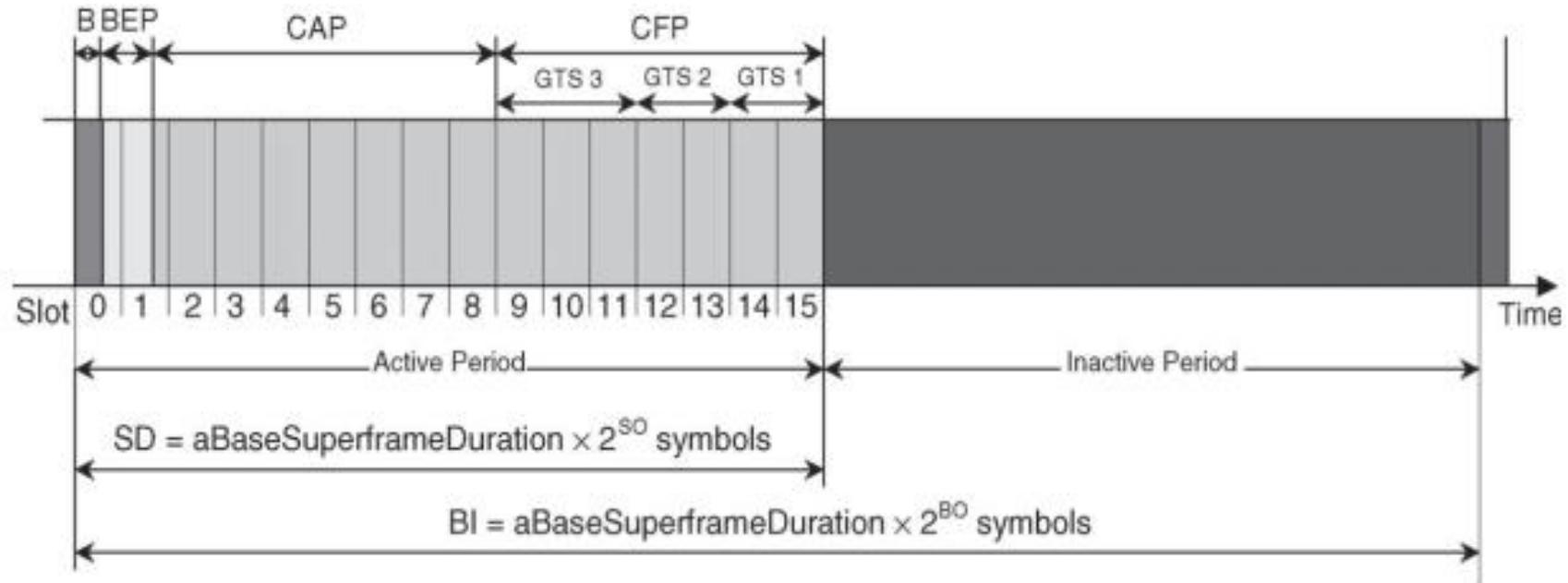
B: Network Beacon

BEP: Beacon Extension Period

CFP: Contention Free Period

[Sohraby2007]

MAC: Superframe Structure with Energy Saving



CAP: Contention Access Period
CFP: Contention Free Period
GTS: Guaranteed Time Slot

BO: Beacon Order
BI: Beacon Interval
SO: Superframe Order

B: Network Beacon
BEP: Beacon Extension Period
SD: Superframe Duration

SO and BO are MAC attributes, $0 \leq SO \leq BO \leq 14$.

[Sohraby2007]

General MAC frame format

Octets:	1	0/	0/2/	0/	0/2/	Variabl	2
Frame Control	Sequence number	Destinatio n PAN	Destinatio n	Source PAN Identifier	Source Address	Frame Payload	Frame Check Sequence
		Addressing				MAC Payload	MAC Footer
MAC							

Bits: 0-	3	4	5	6	7-	10-	12-	14-
Frame Type	Security Enabled	Frame Pending	Ack Request	Intra PAN	Reserved	Destination Addressing Mode	Reserve	Source Addressing Mode

Frame Type Value $b_0 \ b_1 \ b_2$	Description
0 0 0	Beacon
0 0 1	Data
0 1 0	Acknowledgement
0 1 1	MAC Command
1 0 0 - 1 1 1	Reserved

[Sohraby2007]

Frame Types

- Beacon
 - Transmitted periodically by PAN coordinator
 - Several purposes such as identifying the network and its structure, wake-up devices, synchronizing network operations
- Data
 - Payload up to 104 octets
 - Use of sequence number and frame sequence number field
- Acknowledgement
 - Receiver acknowledges reception of data
 - Successful or not
- Command
 - Control and configure devices remotely
 - Negotiation and communication with other nodes
 - Device association and disassociation, data request, beacon request, GTS requests

MAC: Types of traffic supported

- Periodic
 - temperature
- Intermittent
 - External impulse: pollution level exceeded
- Repetitive low-latency
 - Mouse, Security

IEEE 802.15.4 Versions

- Since the first version in 2003, new amendments are constantly being introduced.
- Modifications
 - New country specific (frequencies, regulation)
 - New application and network specific:
 - SUN: Smart utility meter monitoring
 - LECIM: Low Energy Critical Infrastructure Monitoring
 - RFID: Radio Frequency Identification
 - RCC: Railway Communications and Control
 - TVWS: TV White Space
 - Medical
 - New PHY specific
 - OFDM, ASK, FSK, QAM, GMSK, MSK, OOK
 - New Protocols
 - TSCH, Aloha, PCA

IEEE 802.15.4 Versions

Not in Syllabus for Exam

Classification	PHY	MAC	Revision
Versions			IEEE 802.15.4-2006
	IEEE 802.15.4a-2007	IEEE 802.15.4a-2007	
	IEEE 802.15.4c-2009		
	IEEE 802.15.4d-2009	IEEE 802.15.4d-2009	
			IEEE 802.15.4-2011
		IEEE 802.15.4e-2012	
	IEEE 802.15.4f-2012		
	IEEE 802.15.4g-2012	IEEE 802.15.4g-2012	
	IEEE 802.15.4j-2013		
		IEEE 802.15.4k-2013	
	IEEE 802.15.4m-2014	IEEE 802.15.4m-2014	
	IEEE 802.15.4p-2014	IEEE 802.15.4p-2014	
			IEEE 802.15.4-2015
	IEEE 802.15.4n-2016		
	IEEE 802.15.4q-2016		
	IEEE 802.15.4u-2016		
	IEEE 802.15.4t-2017		
	IEEE 802.15.4v-2017	IEEE 802.15.4v-2017	
	IEEE 802.15.4s-2018	IEEE 802.15.4s-2018	
	IEEE 802.15.4x-2019		

IEEE 802.15.4 Versions

Not in Syllabus for Exam

Versions	Date	Type of network	Max Data rate kb/s	Modulation Encoding	protocole used	Features
802.15.4	2003	LR-WPAN	250	BPSK O-QPSK O-QPSK	CSMA/CA	Ultra-low power consumption Low data rate, usage of security suite Very low-cost
802.15.4	2006	LR-WPAN	250	ASK O-QPSK BPSK	CSMA/CA	Improves usage of security suite Allowing synchronization of broadcast messages
802.15.4a	2007	LR-WPAN	1000	DQPSK DQPSK BPM-BPSK DQPSK	ALOHA	Using the same frequency channel simultaneously Precision ranging Support of long-range links
802.15.4c	2009	CWPAN	250	MPSK O-QPSK	-	-
802.15.4d	2009	LR-WPAN	100	BPSK GFSK GFSK	CSMA/CA	Coexistence of listen before talk Coexistence of transmission control Coexistence of duty cycle
802.15.4	2011	LR-WPAN	1000	See Sect. 4.5	CSMA/CA ALOHA	Editorial changes and not technical

IEEE 802.15.4 Versions

Not in Syllabus for Exam

Versions	Date	Type of network	Max Data rate kb/s	Modulation Encoding	protocole used	Features
802.15.4e	2012	Industrial LR-WPAN	–	–	DSME LLDN TSCH	QoS, Security Minimizing collisions Deterministic yet flexible bandwidth Interference avoidance Multi-channel, multi-superframe High reliability of the system
802.15.4f	2012	RFID	250	MSK OOK PPM FSK BPSK QPSK QAM O-QPSK	ALOHA	Multi-year battery life Reliable communications Precision location
802.15.4g	2012	SUN	800	O-QPSK	CCA	Interference avoidance Security
802.15.4j	2013	MBAN	250	BPSK O-QPSK	– –	Keeping a channelization scheme flexible
802.15.4k	2013	LECIM	–	FSK GFSK P-FSK P-GFSK	CSMA/CA PCA ALOHA PCA	Reduction of collision probability Good transmit power efficiency Higher sensitivity Priority Forward error correction QoS, security

IEEE 802.15.4 Versions

Not in Syllabus for Exam

Versions	Date	Type of network	Max kb/s	Data rate	Modulation Encoding	Protocol used	Features
802.15.4m	2014	TVWS	1638		FSK BPSK QPSK 16-QAM 64-QAM	-	Low energy mechanism Ranging performance enhancement
802.15.4p	2014	RCCN	36		GMSK C4FM QPSK $\frac{\pi}{4}$ DQPSK DPSK BPSK	CSMA/CA CSMA/CA PCA	Supporting fixed-to-fixed, fixed-to-mobile, and mobile-to-mobile communications
802.15.4	2015	SUN, TVWS MBAN RFID LECIM, RCC	1000		See Sect. 4.14	TSCH CCA TSCH CSMA CSMA/CA PCA ALOHA PCA	Editorial changes and not technical
802.15.4n	2016	CMB	500		O-QPSK GFSK	-	Medical information transmission

IEEE 802.15.4 Versions

Not in Syllabus for Exam

Versions	Date	Type of network	Max Data rate kb/s	Modulation Encoding	Protocol used	Features
802.15.4q	2016	—	Up to 1000	GFSK ASK	—	Reduction in energy consumption Higher data rates Further reduction in peak power Tradeoff between receiver complexity and performance
802.15.4u	2016	SUN	150	2-FSK	—	Used for broader unlicensed of power levels up to 4 W
802.15.4t	2017	—	2000	GMSK	—	High data-rate
802.15.4v	2017	SUN LECIM TVWS	300	O-QPSK FSK OFDM	—	Enabling the regional sub-GHz bands
802.15.4s	2018	—	—	—	—	Selection of the best available PAN
802.15.4x	2019	TVWS	Up to 2400	FSK O-QPSK OFDM	—	Efficient radio spectrum High data-rate

IEEE 802.15.4u-2016: India specific

PHY (MHz)	Frequency band (MHz)	Modulation	Data-rate (kb/s)	Number of channels
866	865-867	2-FSK mode 1	50	19
		2-FSK mode 2	100	10
		2-FSK mode 3	150	10

- Needed for M2M/IoT use cases in sub 1 GHz band in India
- Approved in Sept. 2016 as a third amendment to IEEE 802.15.4-2015
 - IEEE 802.15.4n-2016
 - IEEE 802.15.4q-2016
- Defines a new alternate SUN FSK PHY extension in the 866 MHz band

Zigbee Versions

- 2005 – Zigbee 2004 released
- 2006 – Zigbee 2006 released
- 2007 – Zigbee 2007 released (also known as Zigbee Pro)
- 2015 – Zigbee 3.0 version (with IP)
- 2019 – Zigbee Alliance merges into **Connectivity Standards Alliance**
 - Amazon, Apple, Google and Zigbee Alliance
 - Develop a new open standard for smart home device connectivity
 - Connected home over IP (CHIP) project
 - **Matter as home connectivity technology**
 - In addition to IEEE 802.15.4, Matter also supports Ethernet and WiFi

Zigbee Green Power

- Integrating battery-less (energy harvesting-based) or life-long battery-operated devices into the Zigbee network



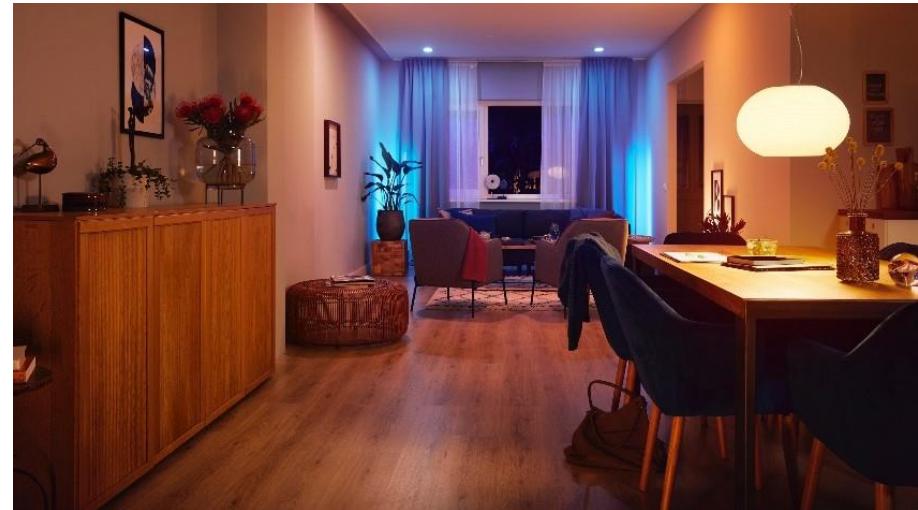
Sensors, open/close detectors, emergency buttons, industrial switches, ...



- (Light) switch: flipping the switch generates the energy for data-communication

Zigbee Use Case: Smart Lighting (Philips Hue)

- Benefits of Smart Lighting
 - Controlling lights automatically or remotely via app
 - Easily Dimmable
 - Energy efficient using LED bulbs
 - Can connect bulb to other devices in home such as camera, audio equipment, thermostat or home assistant
 - Configure the lights to mimic your presence when you are away
 - Mood lighting



Zigbee Use Cases



Amazon Echo Plus (2nd Gen+)

Samsung SmartThings

Philips Hue by Signify

IKEA



Xfinity by Comcast

Wink

Tuya



Lumi

Key IoT Features

Advantages

- Low power
 - Zigbee (20 mJ per hour)
 - Zigbee Pro (Green Power: 20 microJ per hour)
- Large coverage of 1Km in Sub-GHz band
 - Even more for boosted modules (3.2 km for Xbee)
- Easy to install and maintain (mesh, self-healing, self-organization)
- Reliable (mesh, multiple channels, demonstrated interference tolerance, automated retransmissions)
- Supports thousands of nodes
- Low cost (many suppliers)
- Long battery life (years on AA battery)
- Secure (AES 128 bit)

Source: Zigbee 3.0

Issues

- No mobility support, Scalability
- Less coverage area in 2.4 GHz band

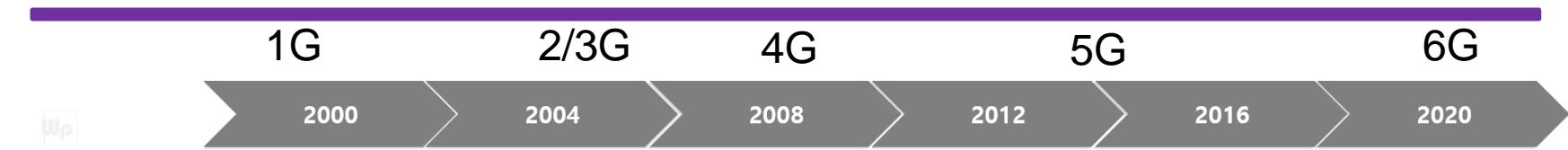
Questions?

WiFi: IEEE 802.11 family

WiFi: What's in a name?

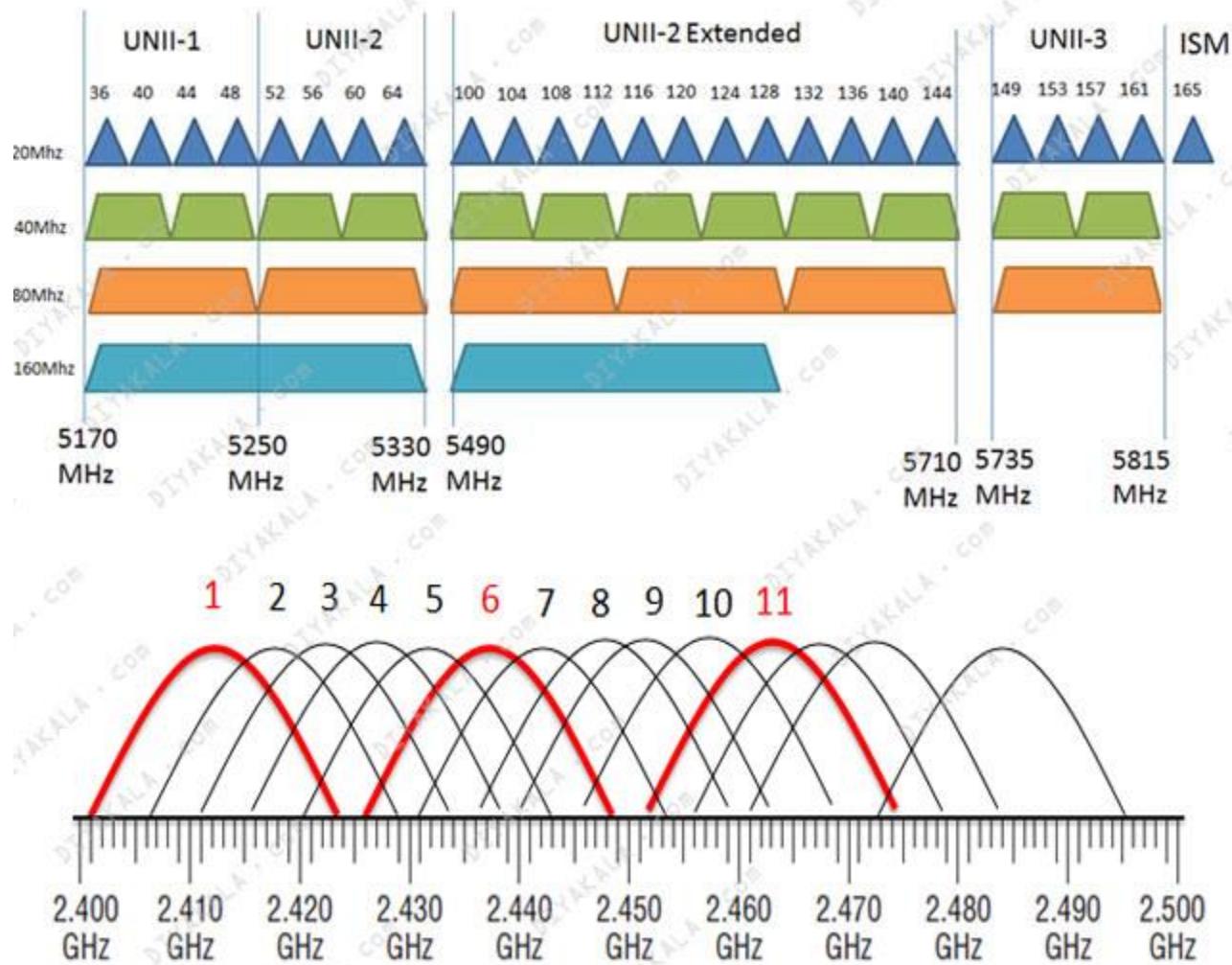
- WiFi is a short name for Wireless Fidelity
- On the lines of *Hi-Fi* (high fidelity), a term for high-quality audio technology
- *Fidelity is defined as the degree of exactness with which something is copied or reproduced.*
- This is also called Wireless Local Area Network (WLAN)

WLAN Standards



	1G	2/3G	4G	5G	6G	
Standard	11b	11a/g	11n	11ac (wave1)	11ac (wave2)	11ax
MCS	Spread Spectrum	OFDM			OFDM (OFDMA)	
Freq	2.4GHz	2.4GHz 5GHz			Same Freq (<7GHz)	
Bandwidth	20MHz	20MHz	+40MHz	+80MHz	+160MHz	Same BW (+320M)
Multiple Antenna			MIMO Beamforming		MU-MIMO (DL)	MU-MIMO (UL)
PHY Rate	11Mbps	54Mbps	600Mbps (40M,4SS)	1.7Gbps (80M,4SS)	6.7Gbps (160M,8SS)	9.6GHz (160M,8SS)
MAC	CSMA/CA in DCF	Security QoS	Aggregation			BSS Management

WLAN Bandwidths



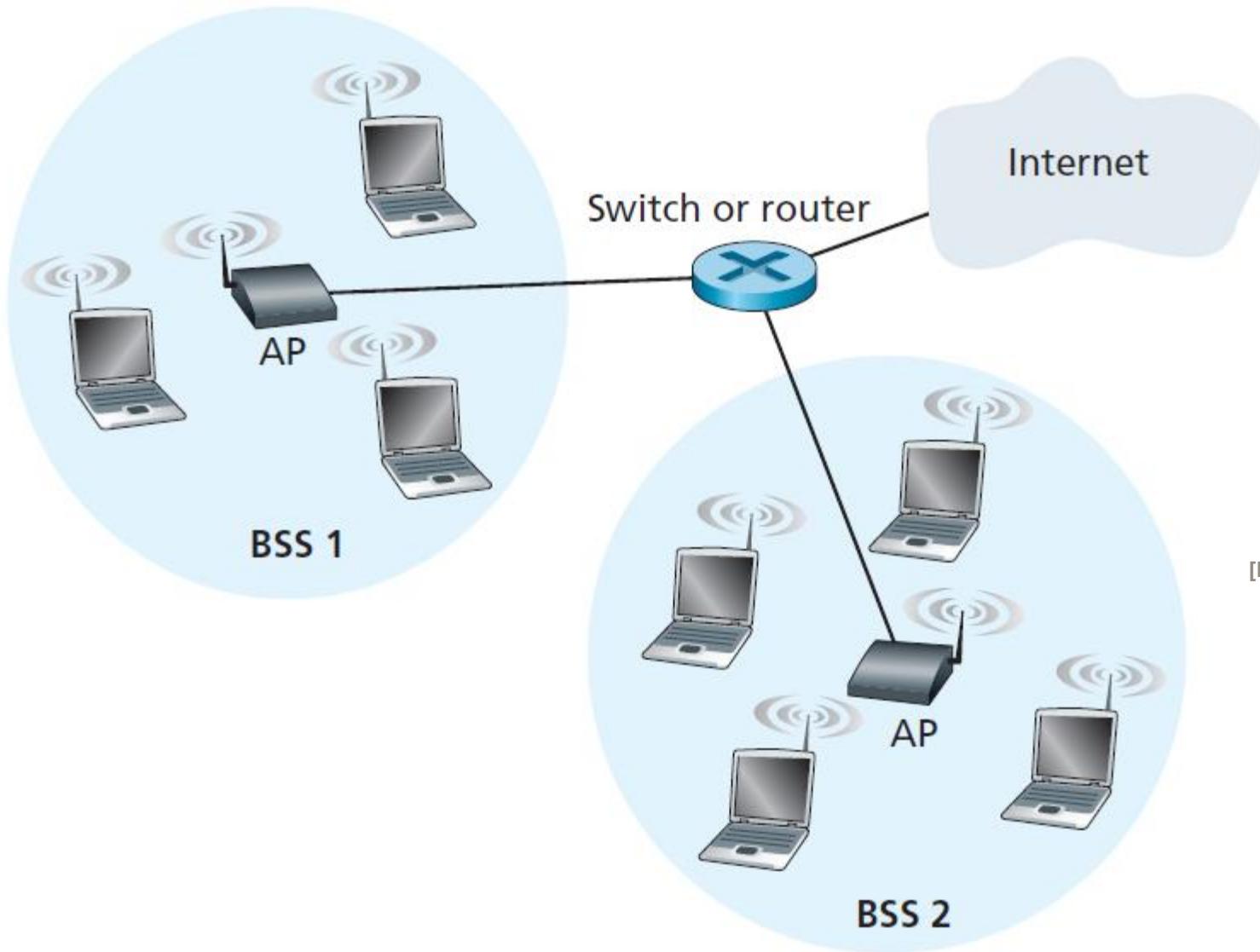
<https://commons.wikimedia.org/wiki/File:Frequency-bandwitch-wifi-camera-wireless-camera.jpg>

IEEE 802.11 Network Topologies

Nodes as **stations** and cluster head as **access point**

- Basic service set (BSS) or Star
- Extended service set (ESS) or cluster tree
- Independent basic service set (IBSS)
 - Ad-hoc = Mesh without access point
- Mesh basic service set (MBSS)
 - (wired or wireless) Mesh of cluster heads (Hybrid)

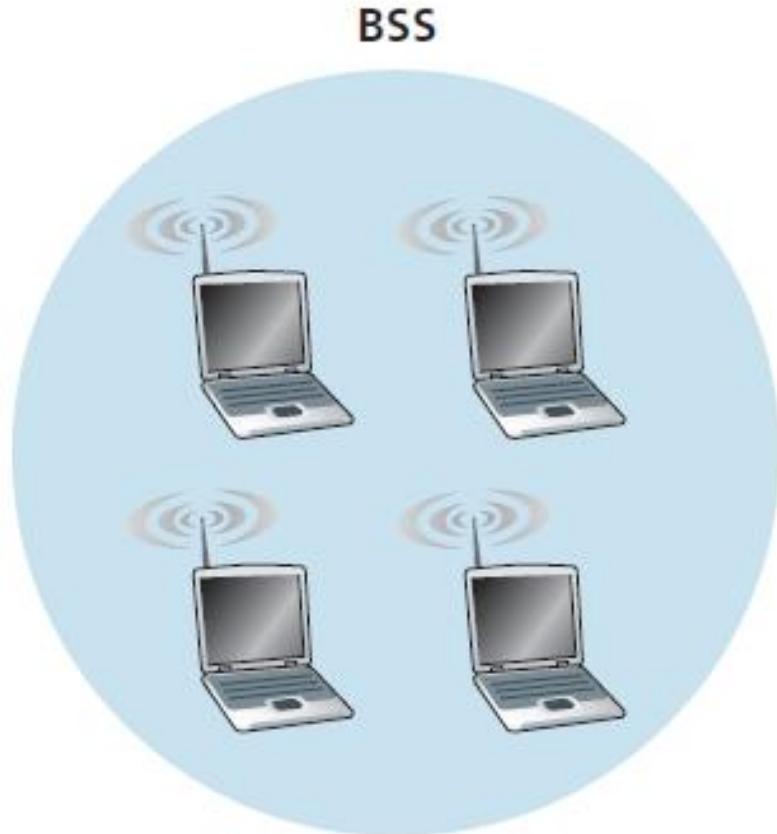
WLAN architecture: Infrastructure mode



[Kurose2012]

WLAN: Adhoc mode

- Also called WiFi Direct



MAC Frame Format

Frame (numbers indicate field length in bytes):

2	2	6	6	6	2	6	0-2312	4
Frame control	Duration	Address 1	Address 2	Address 3	Seq control	Address 4	Payload	CRC

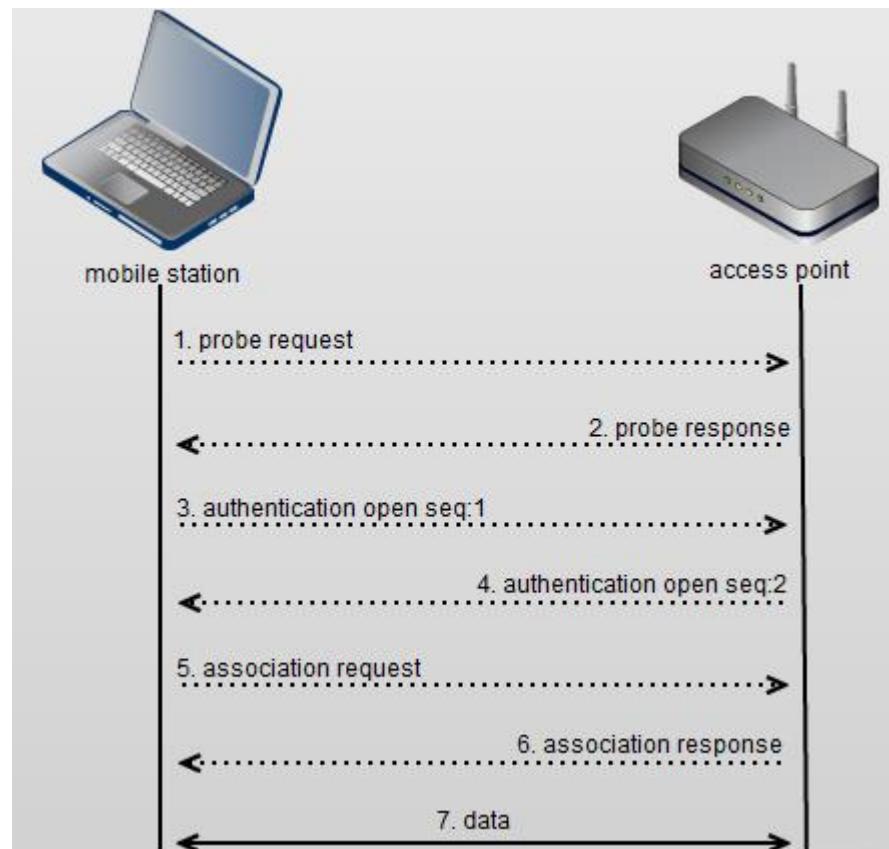
Frame control field expanded (numbers indicate field length in bits):

2	2	4	1	1	1	1	1	1	1	1
Protocol version	Type	Subtype	To AP	From AP	More frag	Retry	Power mgt	More data	WEP	Rsvd

- CRC: 32-bit Cyclic Redundancy Check
- WEP: Wired Equivalent Privacy

Association Process

- Three states
 - Not authenticated or associated
 - Authenticated but not associated
 - Authenticated and associated



How IEEE 802.11 adopted for IoT?

802.11ac (5G of WiFi) and 802.11ah (WiFi-Halow)

	802.11ac	802.11ah
Operating Bands	2.4 and 5 GHz	Sub 1-GHz
Spectrum available	100 + 150 MHz	26 MHz
Use Cases	Broadband wireless	Sensors and Meters Extended WiFi
Data Rate Requirement	20 Mbps - 3 Gbps	100 Kbps
Single Frame Size	Large (e.g., 1500 bytes)	Small (e.g., 100 bytes)
Traffic type	Video Streaming/ Large file transfer	Periodic packet transmission every few to tens minutes
Distance between devices	Up to 60 m	Up to 1 Km
Number of stations	3-20	8191
Location	Mostly indoor	Indoor and outdoor
Backward compatibility	Yes	No

PHY parameters for 802.11ah

- Use of orthogonal frequency division multiplexing (OFDM)
- Basically adapted a scaled-down version of 802.11ac
 - Bandwidths of 20-160 MHz to 2-16 MHz
 - Same number of subcarrier
 - Increased symbol duration

[Park2015]

Parameters	Supported Values
Channel Bandwidths	2, 4, 8, and 16 MHz
Modulation Schemes	BPSK, QPSK, 16-QAM, 64-QAM, 256-QAM
Code Rates	1/2 with 2 times repetition 1/2 , 2/3, 3/4 and 5/6 Convolution or low-density parity check (LDPC)
MIMO	Support up to 4 by 4
Data Rates	150 Kbps (1 MHz bandwidth, 1 spatial stream, BPSK, $\frac{1}{2}$ coding rate, repetition) to 347 Mbps (16 MHz bandwidth, 4 spatial streams, 256 QAM, $\frac{5}{6}$ coding rate)

Link Budget Comparison

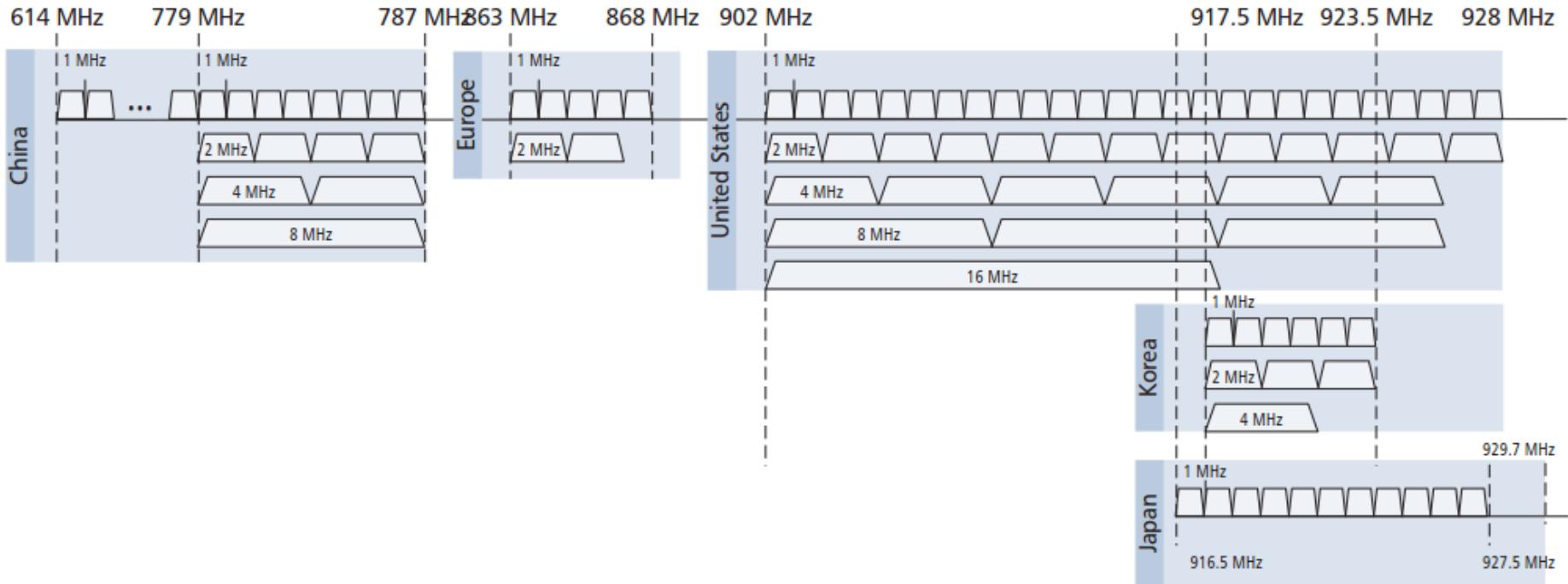
Parameters	Link budget enhancements of 900 MHz 802.11ah over 2.4 GHz 802.11n
Free space path loss	+8.5 dB
Noise bandwidth	+10 dB
Sub-total link budget gain	+18.5 dB
1 MHz channel width	+3 dB
Repetition coding	+3 dB
Total link budget gain	+24.5 dB

[Park2015]

Low Power and Low Cost Support for Indoor Sensors:

This can reduce the transmit energy consumption and also lower the cost of an 802.11ah radio of a small sensor device.

Frequency Bands in Different Countries



[Park2015]

802.11ah MAC features

- Hierarchical association identifier
- Access scheme: Hybrid Coordination Function (HCF)
- Optional Restricted Access Window (RAW)
- Increased sleep time
- Target wake-up time
- Bidirectional transmission opportunity
- Short MAC frame
- Null data packet for ACK
- Synchronization frame operation
- And few more!

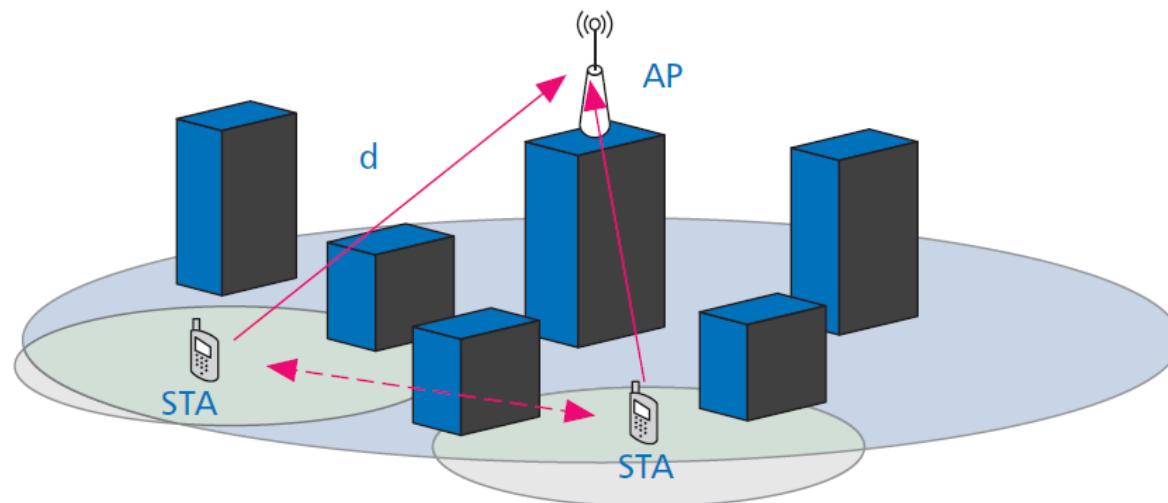
Hybrid Coordination Function (HCF)

- 802.11
 - CSMA/CA
- 802.11ah
 - HCF controlled channel access
 - Polling based for infrastructure based networks
 - Guarantees Quality of Service
 - Enhanced distributed channel access (EDCA)
 - EDCA is extension of CSMA/CA that tries to implement service differentiation by classifying the traffic into different categories with different priorities

[Gonzalez2016]

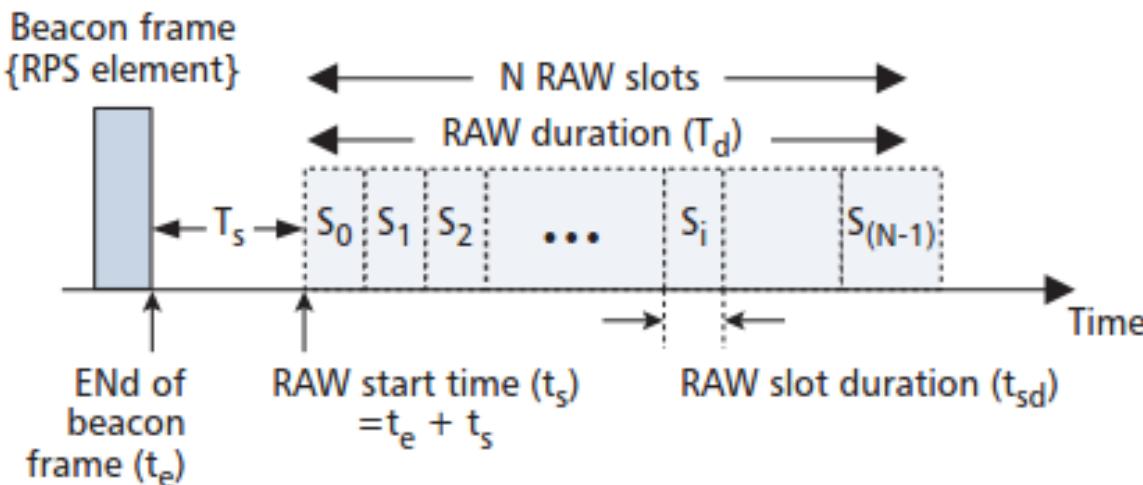
Restricted Access Window (RAW)

- Issue with supporting 1 Km range outdoors
 - Access point for outdoor applications are installed on top while users are near grounds
 - High path loss and Shadowing
 - Severe hidden node problem between several APs supporting thousands of nodes
 - Several collisions and subsequent retransmissions cause energy consumption
- RAW minimizes the issue



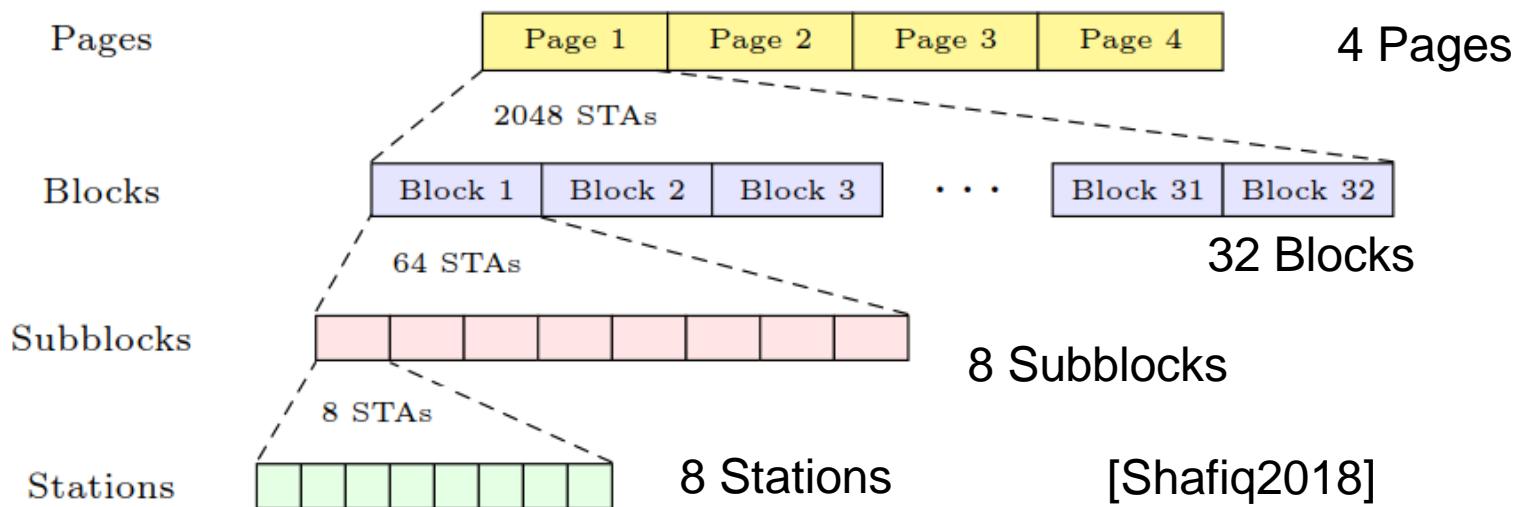
Restricted Access Window (RAW)

- Station or node grouping mechanism
- New optional contention channel access scheme
 - Combination of TDMA and CSMA/CA
- In the time window frame is divided into RAW slots
 - maximum of 64 slots
- Nodes are divided into groups and only members of a particular group have access to that time slot
- Reduces collisions, improve channel efficiency, and allows an increasing number of users



Hierarchical Association IDentifier (AID)

- Legacy 802.11 supports 2007 nodes (or associated stations) per access point
- 802.11ah uses a novel hierarchical AID
 - New AID consists of 13 bits and can support 8191 nodes
 - Four levels: Page, block, sub-block, and station-index in sub-block
 - This structure can be used to group stations based on similar characteristics such as traffic, pattern, location, battery levels, etc.



Increased Sleep Time

- In 802.11, the max sleep time is 18 hours without getting disassociated with the AP
- For 802.11ah, the max sleep time is redefined such that the station can sleep for approximately 5.2 years

Target Wakeup Time (TWT)

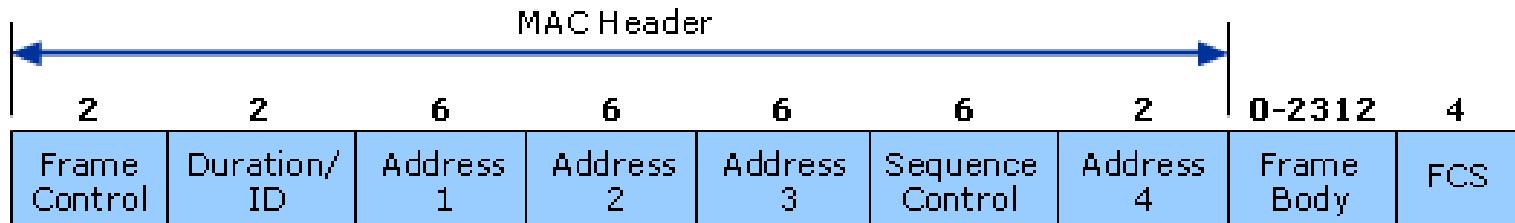
- 802.11
 - An AP buffers data destined for a station while the station is in sleep state
 - The station periodically wakes up at beacon transmission times and receives a beacon to see if there is any buffered data at the AP based on the information in traffic indication map (TIM)
 - If TIM indicates that there is data buffered at AP, it sends a PS-Poll frame to the AP to indicate the station is awake and is ready to receive the buffered data
 - The AP needs time to find the buffered data and has to contend for the medium: this indefinite latency makes the station consume energy waiting for the buffered data
- 802.11ah
 - Uses target wake-up time between an AP and a station so that the AP knows when the station will be awake
 - Removes the processing time and medium access latency

Bidirectional Transmission Opportunity

- Bidirectional transmission (BDT) allows an AP and a station to exchange one or more uplink and downlink packets in one TXOP.
- Packets are separated by short inter frame space (SIFS)
- In the BDT procedure, a station uses the More Data bit in the SIGNAL field of PHY preamble of a packet to indicate whether the station has more data to transmit following the current packet transmission.
- This reduces the number of contention-based channel accesses, improves channel efficiency by minimizing the number of frame exchanges required for uplink and downlink data frames, and enables stations to extend battery lifetime by keeping Awake-times short.

Short MAC Frame

- In 802.11n, the MAC header can be 30 bytes long



[https://technet.microsoft.com/en-us/library/cc757419\(v=ws.10\).aspx](https://technet.microsoft.com/en-us/library/cc757419(v=ws.10).aspx)

- For IoT application transmitting only 50 bytes of data infrequently, this is big overhead
- In 802.11ah, the length of header is reduced to 12 bytes

802.11ax (6G of WiFi)

- Convergence of high data rates and IoT applications
- Smarter access points for improved outdoor coverage with longer guard intervals
- Target Wake-up Time
- BSS coloring to reduce interference
- Only on 5 GHz
- Comparison with 802.11ac
 - 6 times speed, 7 times battery life with TWT, 4 times range
 - Support much more than 7 devices
- OFDMA instead of OFDM
- MU-MIMO
- 1024 QAM and 160 MHz bandwidth to give multi-giga bit data rates

Key IoT Features (802.11ah)

- High data rates
 - Can handle diverse range of applications including camera
- Longer range than traditional WiFi
- Scalable to thousands of nodes
- Widely used

Issues

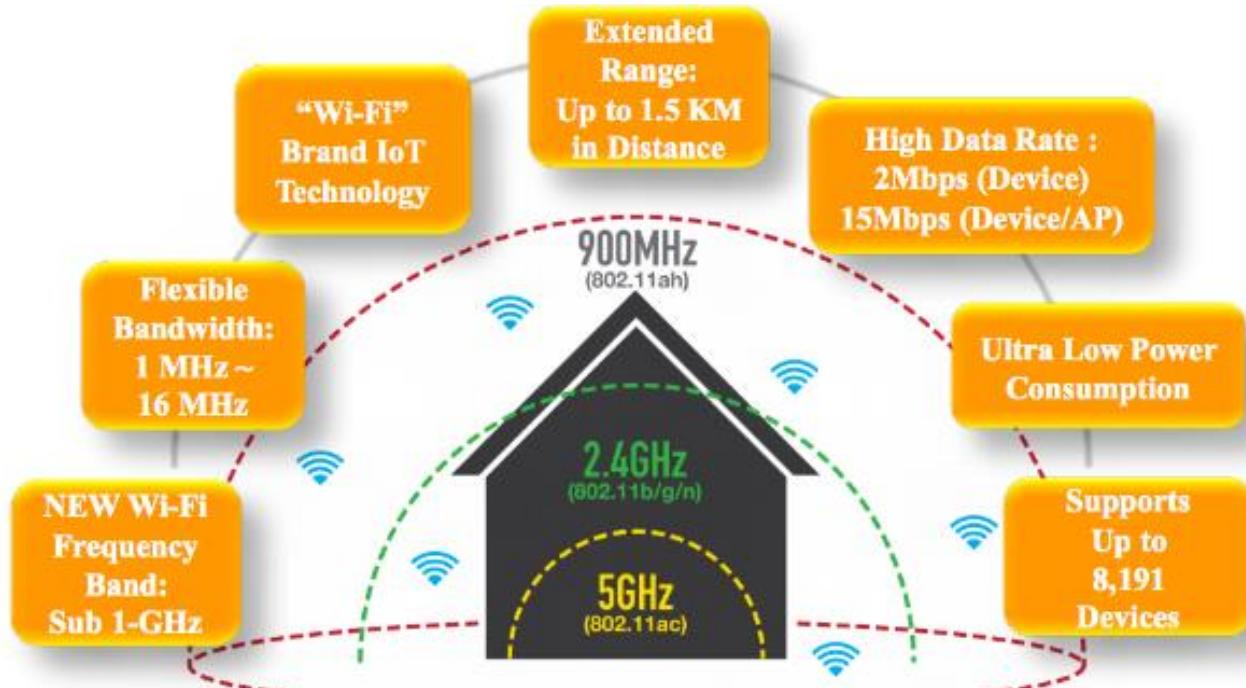
- Most of the world is using 2.4 GHz
 - Problem for 802.11ah
- 802.11ah available, but products are hardly there
 - Mostly using 802.11b/g/n
- Security
- High power consumption
- Roaming

WiFi Halow products

- [Adapt-IP](#)
- [Alfa wireless](#)
- [Methods2Business](#)
- [Newratek / Newracom](#)
- [Palma Ceia SemiDesign](#)
- [Huge-IC](#)
- [Silex Technology's SX-NEWAH](#)

Links embedded

Example of SX-NEWAH



NEWRACOM © 2017 NEWRACOM INC.

<https://www.silextechnology.com/connectivity-solutions/embedded-wireless/sx-newah>

Questions?

References

- Perry Lea, *Internet of Things for Architect*, Packt Publication, 2018
- [Gonzalez]
- [Kurose2012] J. Kurose and K. Ross, *Computer Networking*, Pearson, 2012
- [Park 2015] M. Park, “IEEE 802.11ah: Sub 1-GHz License Exempt Operation for the Internet of Things,” *IEEE Communications Magazine*, September 2015
- [Tian2021] L. Tian et al., “WiFi HaloW for the Internet of Things: An up-to-date survey on IEEE 802.11ah research,” *Journal of Network and Computer Applications*, 2021

**That's all for today!
Thank You!**

Communications & Controls in IoT

LoRaWAN and Cellular Technologies

Instructor: Sachin Chaudhari

Feb. 13, 2023



**INTERNATIONAL INSTITUTE OF
INFORMATION TECHNOLOGY**

HYDERABAD

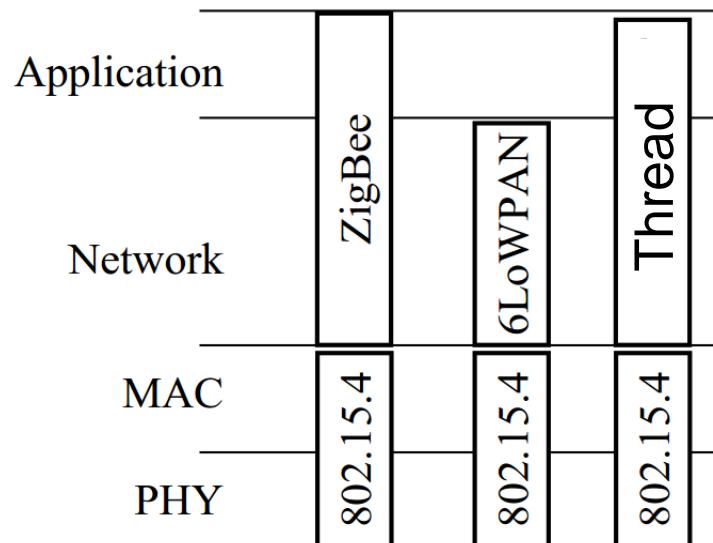
Recap

IEEE 802.15.4

Ref: K. Sohraby, D. Minoli, T. Znati, *Wireless Sensor Networks*, Wiley, 2007

IEEE 802.15.4

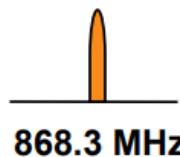
- IEEE 802.15.4 defines the operation of low-rate wireless personal area networks (LR-WPANs)
- Widely used in wireless sensor-network (WSN) applications
 - Vast number of industrial, home and medical applications
- It specifies the physical layer (PHY) and media access control (MAC) for LR-WPANs
- Does not have IP address
- Used by several “Internet of Things” protocols:
 - ZigBee, 6LowPAN, Thread, WiSuN etc.



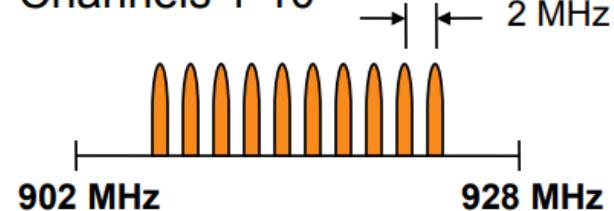
Physical Layer (PHY): Operating Frequency Bands

**868MHz/915MHz
PHY**

Channel 0

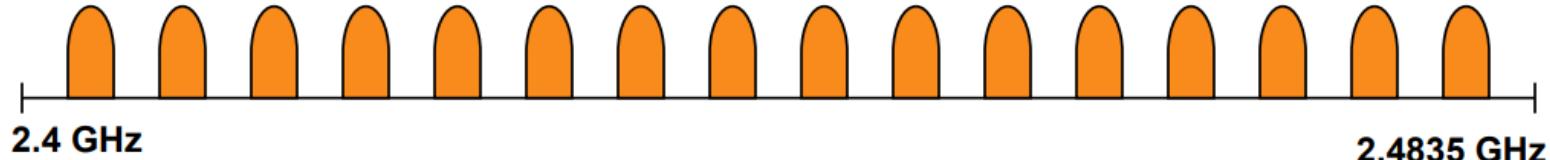


Channels 1-10



**2.4 GHz
PHY**

Channels 11-26



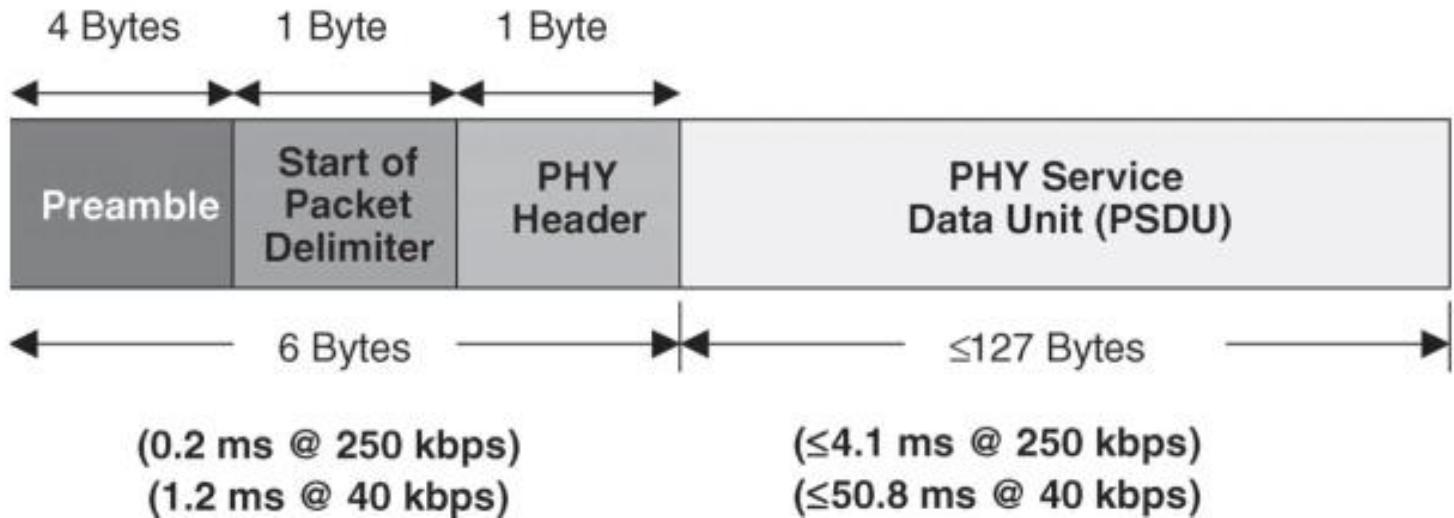
PHY: Modulation Parameters

Freq. band (MHz)	Spreading Parameters		Data Parameters		
	Chip rate (kchip/s)	Modulation	Bit rate (kbps)	Symbol rate (ksymbol/s)	Symbols
868	300	BPSK	20	20	Binary
915	600	BPSK	40	40	Binary
2400	2000	O-QPSK	250	62.5	16-ary

[Koubaa2007]

All bands are based on Direct sequence spread spectrum (DSSS),
a form of CDMA

PHY-layer packet structure



- Preamble -> Symbol synchronization
- Packet delimiter -> Frame synchronization
- PHY header: length of the PSDU
- PSDU can carry upto 127 bytes

MAC Layer features

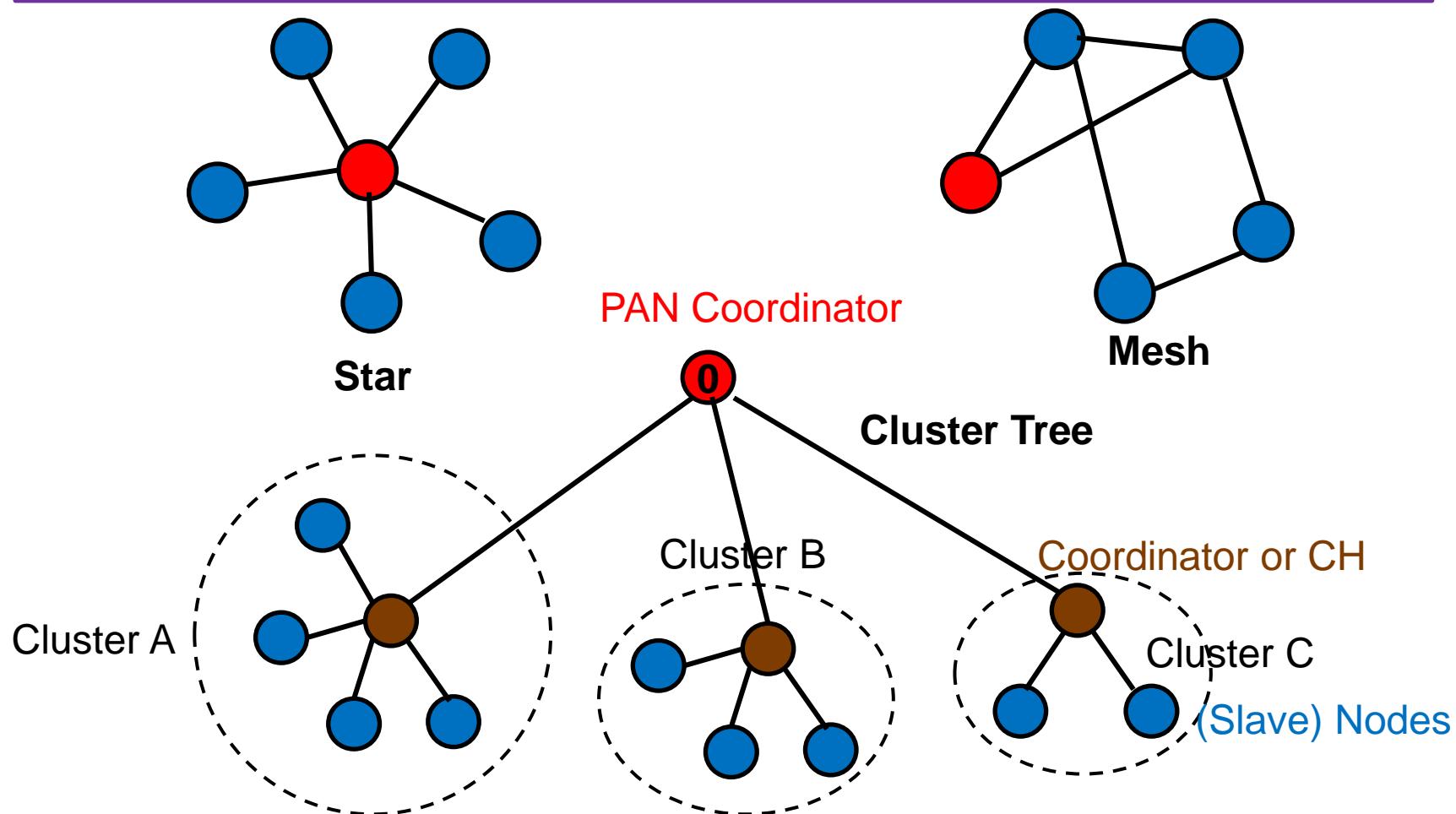
- Designed to support vast number of industrial and home applications for control and monitoring
- Enabling deployment of large number of devices with low cost and complexity
- Several features for flexible network configuration and low-power operation
 - Different topologies and network devices
 - Optional superframe structure with duty-cycle control
 - Both contention and scheduled based MAC protocols
 - Synchronized and non-synchronized operation
 - Efficient energy management
 - Adaptive sleep
 - Extended sleeping time
 - Flexible addressing scheme for large number of nodes

MAC Layer: *Device Types*

Two kind of devices in IEEE 802.15.4 based on complexity and capability

- Fully functional devices (FFD)
 - More resources
 - Multiple network responsibilities
- Reduced functionality devices (RFD)
 - Simple and low-cost device
 - Can only communicate with one FFD

Topologies: Zigbee (Network Layer)



- 16 bit addresses support 65536 devices in a PAN. For clusters, 255 clusters with 254 nodes each
- Self recovering ability

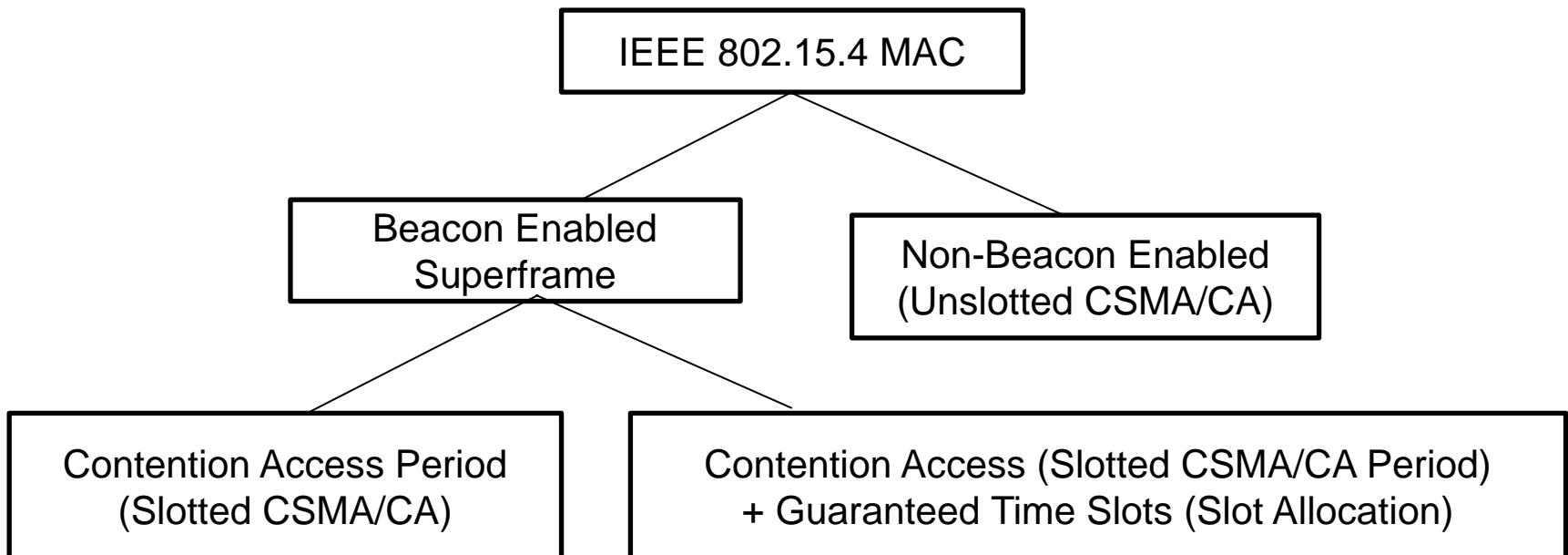
Zigbee Node Types

Zigbee defines three kinds of logical devices

- **PAN coordinator or Master**
 - Principal controller of network
 - Managing list of all network devices or nodes
 - Identifies PAN and nodes associated with it
 - Provides global synchronization by transmitting beacon frames containing relevant information
- **Coordinator or cluster head (CH)**
 - Same functionalities as PAN coordinator locally in cluster
 - Managing association and disassociation of other nodes to PAN
 - Does not create its PAN
- **Simple (Slave) Nodes**
 - No coordination functionalities
- PAN Coordinator and CH are **FFD** while slave nodes are **RFD**

MAC layer functions

- Network association and disassociation
- Two modes of operation
 - Beaconing
 - Non-beaconing



IEEE 802.15.4 Versions

- Since the first version in 2003, new amendments are constantly being introduced.
- Modifications
 - New country specific (frequencies, regulation)
 - New application and network specific:
 - SUN: Smart utility meter monitoring
 - LECIM: Low Energy Critical Infrastructure Monitoring
 - RFID: Radio Frequency Identification
 - RCC: Railway Communications and Control
 - TVWS: TV White Space
 - Medical
 - New PHY specific
 - OFDM, ASK, FSK, QAM, GMSK, MSK, OOK
 - New Protocols
 - TSCH, Aloha, PCA

IEEE 802.15.4u-2016: India specific

PHY (MHz)	Frequency band (MHz)	Modulation	Data-rate (kb/s)	Number of channels
866	865-867	2-FSK mode 1	50	19
		2-FSK mode 2	100	10
		2-FSK mode 3	150	10

- Needed for M2M/IoT use cases in sub 1 GHz band in India
- Approved in Sept. 2016 as a third amendment to IEEE 802.15.4-2015
 - IEEE 802.15.4n-2016
 - IEEE 802.15.4q-2016
- Defines a new alternate SUN FSK PHY extension in the 866 MHz band

Zigbee Versions

- 2005 – Zigbee 2004 released
- 2006 – Zigbee 2006 released
- 2007 – Zigbee 2007 released (also known as Zigbee Pro)
- 2015 – Zigbee 3.0 version (with IP)
- 2019 – Zigbee Alliance merges into **Connectivity Standards Alliance**
 - Amazon, Apple, Google and Zigbee Alliance
 - Develop a new open standard for smart home device connectivity
 - Connected home over IP (CHIP) project
 - **Matter as home connectivity technology**
 - In addition to IEEE 802.15.4, Matter also supports Ethernet and WiFi

Zigbee Green Power

- Integrating battery-less (energy harvesting-based) or life-long battery-operated devices into the Zigbee network



Sensors, open/close detectors, emergency buttons, industrial switches, ...



- (Light) switch: flipping the switch generates the energy for data-communication

Key IoT Features

Advantages

- Low power
 - Zigbee (20 mJ per hour)
 - Zigbee Pro (Green Power: 20 microJ per hour)
- Large coverage of 1Km in Sub-GHz band
 - Even more for boosted modules (3.2 km for Xbee)
- Easy to install and maintain (mesh, self-healing, self-organization)
- Reliable (mesh, multiple channels, demonstrated interference tolerance, automated retransmissions)
- Supports thousands of nodes
- Low cost (many suppliers)
- Long battery life (years on AA battery)
- Secure (AES 128 bit)

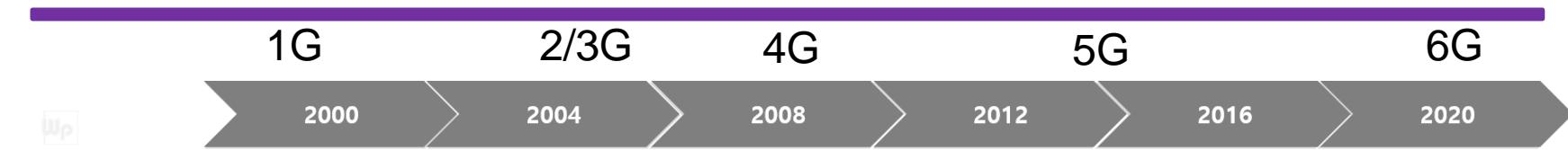
Source: Zigbee 3.0

Issues

- No mobility support, Scalability
- Less coverage area in 2.4 GHz band

WiFi: IEEE 802.11 family

WLAN Standards



	1G	2/3G	4G	5G	6G	
Standard	11b	11a/g	11n	11ac (wave1)	11ac (wave2)	11ax
MCS	Spread Spectrum	OFDM			OFDM (OFDMA)	
Freq	2.4GHz	2.4GHz 5GHz			Same Freq (<7GHz)	
Bandwidth	20MHz	20MHz	+40MHz	+80MHz	+160MHz	Same BW (+320M)
Multiple Antenna			MIMO Beamforming		MU-MIMO (DL)	MU-MIMO (UL)
PHY Rate	11Mbps	54Mbps	600Mbps (40M,4SS)	1.7Gbps (80M,4SS)	6.7Gbps (160M,8SS)	9.6GHz (160M,8SS)
MAC	CSMA/CA in DCF	Security QoS	Aggregation			BSS Management

802.11ac (5G of WiFi) and 802.11ah (WiFi-Halow)

	802.11ac	802.11ah
Operating Bands	2.4 and 5 GHz	Sub 1-GHz
Spectrum available	100 + 150 MHz	26 MHz
Use Cases	Broadband wireless	Sensors and Meters Extended WiFi
Data Rate Requirement	20 Mbps - 3 Gbps	100 Kbps
Single Frame Size	Large (e.g., 1500 bytes)	Small (e.g., 100 bytes)
Traffic type	Video Streaming/ Large file transfer	Periodic packet transmission every few to tens minutes
Distance between devices	Up to 60 m	Up to 1 Km
Number of stations	3-20	8191
Location	Mostly indoor	Indoor and outdoor
Backward compatibility	Yes	No

PHY parameters for 802.11ah

- Use of orthogonal frequency division multiplexing (OFDM)
- Basically adapted a scaled-down version of 802.11ac
 - Bandwidths of 20-160 MHz to 2-16 MHz
 - Same number of subcarrier
 - Increased symbol duration

[Park2015]

Parameters	Supported Values
Channel Bandwidths	2, 4, 8, and 16 MHz
Modulation Schemes	BPSK, QPSK, 16-QAM, 64-QAM, 256-QAM
Code Rates	1/2 with 2 times repetition 1/2 , 2/3, 3/4 and 5/6 Convolution or low-density parity check (LDPC)
MIMO	Support up to 4 by 4
Data Rates	150 Kbps (1 MHz bandwidth, 1 spatial stream, BPSK, $\frac{1}{2}$ coding rate, repetition) to 347 Mbps (16 MHz bandwidth, 4 spatial streams, 256 QAM, $\frac{5}{6}$ coding rate)

Link Budget Comparison

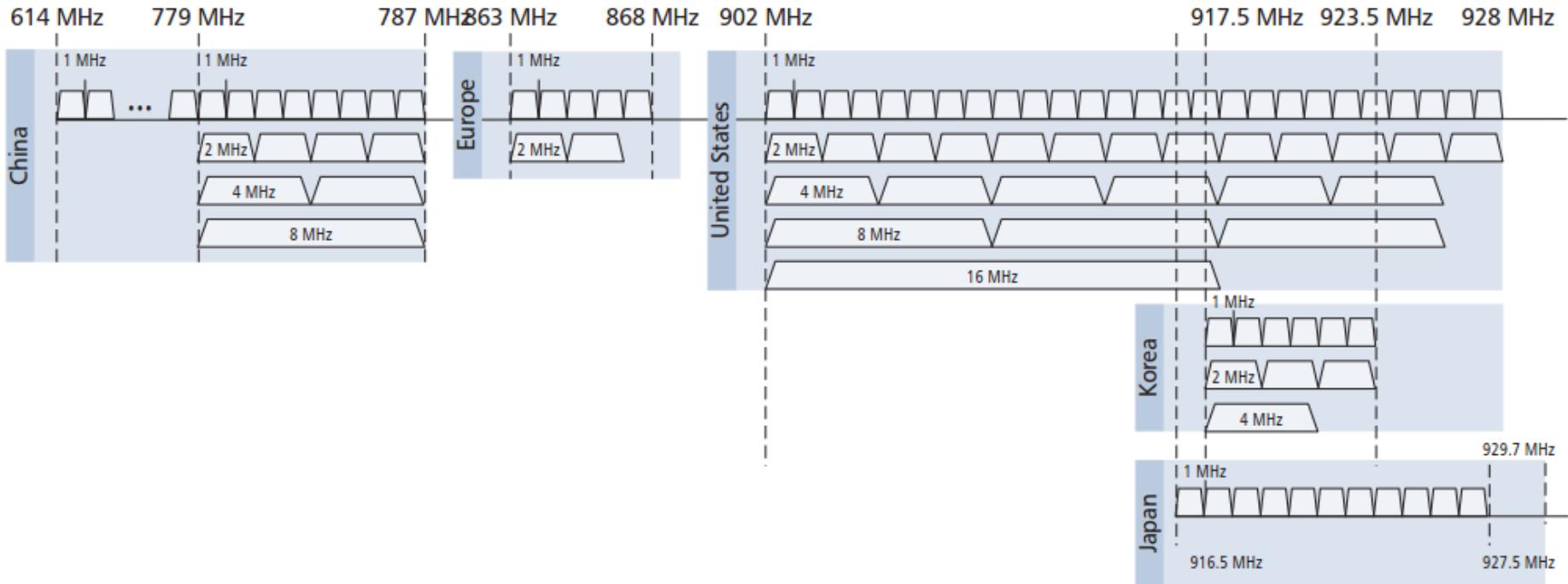
Parameters	Link budget enhancements of 900 MHz 802.11ah over 2.4 GHz 802.11n
Free space path loss	+8.5 dB
Noise bandwidth	+10 dB
Sub-total link budget gain	+18.5 dB
1 MHz channel width	+3 dB
Repetition coding	+3 dB
Total link budget gain	+24.5 dB

[Park2015]

Low Power and Low Cost Support for Indoor Sensors:

This can reduce the transmit energy consumption and also lower the cost of an 802.11ah radio of a small sensor device.

Frequency Bands in Different Countries



[Park2015]

IEEE 802.11 Network Topologies

Nodes as **stations** and cluster head as **access point**

- Basic service set (BSS) or Star
- Extended service set (ESS) or cluster tree
- Independent basic service set (IBSS)
 - Ad-hoc = Mesh without access point
- Mesh basic service set (MBSS)
 - (wired or wireless) Mesh of cluster heads (Hybrid)

802.11ah MAC features

- Hierarchical association identifier
- Access scheme: Hybrid Coordination Function (HCF)
- Optional Restricted Access Window (RAW)
- Increased sleep time
- Target wake-up time
- Bidirectional transmission opportunity
- Short MAC frame
- Null data packet for ACK
- Synchronization frame operation
- And few more!

802.11ax (6G of WiFi)

- Convergence of high data rates and IoT applications
- Smarter access points for improved outdoor coverage with longer guard intervals
- Target Wake-up Time
- BSS coloring to reduce interference
- Only on 5 GHz
- Comparison with 802.11ac
 - 6 times speed, 7 times battery life with TWT, 4 times range
 - Support much more than 7 devices
- OFDMA instead of OFDM
- MU-MIMO
- 1024 QAM and 160 MHz bandwidth to give multi-giga bit data rates

Key IoT Features (802.11ah)

- High data rates
 - Can handle diverse range of applications including camera
- Longer range
- Scalable to thousands of nodes
- Widely used

Issues

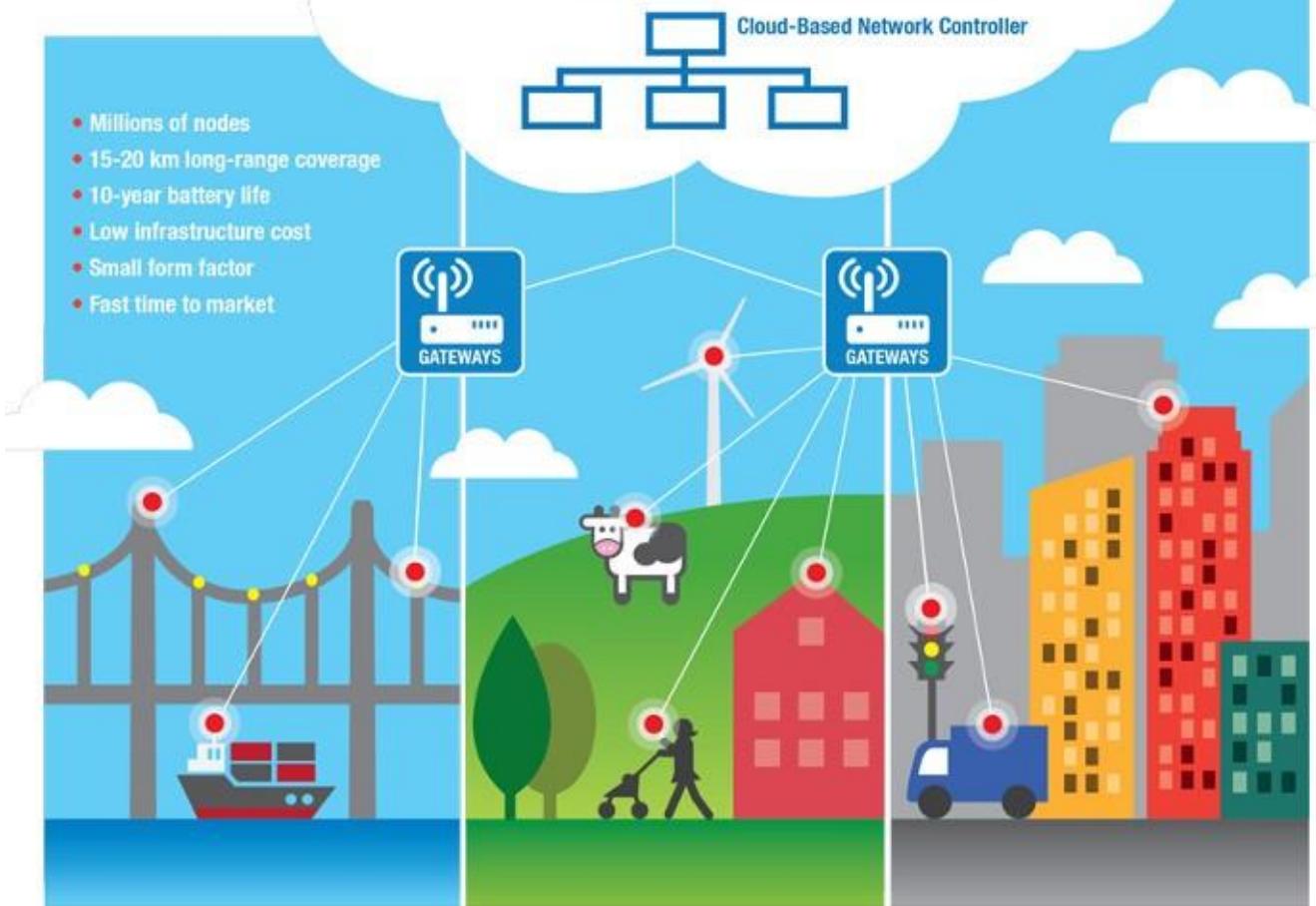
- Most of the world is using 2.4 GHz
 - Problem for 802.11ah
 - Problem for 802.11ax
- 802.11ah available, but products are hardly there
 - Mostly using 802.11b/g/n
- Security
- High power consumption
- Roaming

Today's Class

LoRa and LoRaWAN

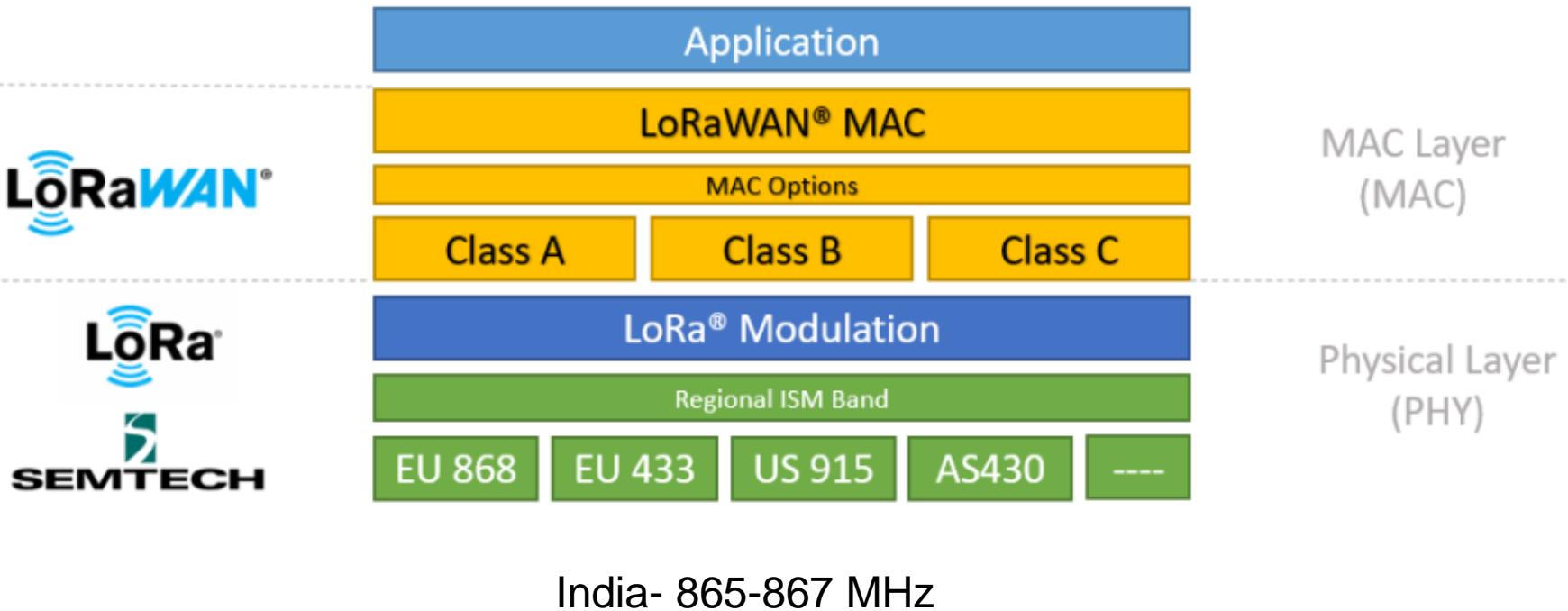
LoRaWAN

LoRa™ End-Node Solution For Long Range and Low Power IoT Networks.



Star of Stars Topology
Data Rate: 0.3-50 kbps
Range: Few Kms

LoRaWAN Technology Stack



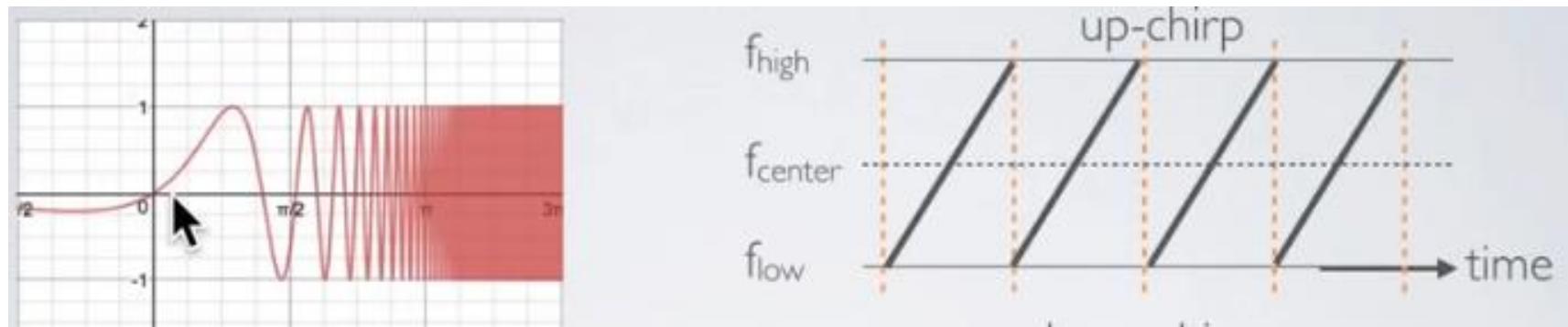
India- 865-867 MHz

- LoRa is a proprietary RF modulation technology at PHY
- Created by Cycleo (acquired by Semtech)
- LoRaWAN is technology stack on top of LoRa
 - LoRaWAN alliance of more than 500 companies
 - Semtech a founding member

LoRa (PHY)

LoRa: Modulation

- A proprietary spread spectrum technique derived out of *Chirp Spread Spectrum*
- Chirp spread spectrum (CSS) is a spread spectrum technique that uses wideband linear frequency modulated chirp pulses to encode information
- Chirp
 - A *sinusoidal signal of frequency increase or decrease over time, often with a polynomial expression for the relationship between time and frequency*



<https://www.youtube.com/watch?v=r84GMLeiqg8>

LoRa: Frequency and BW

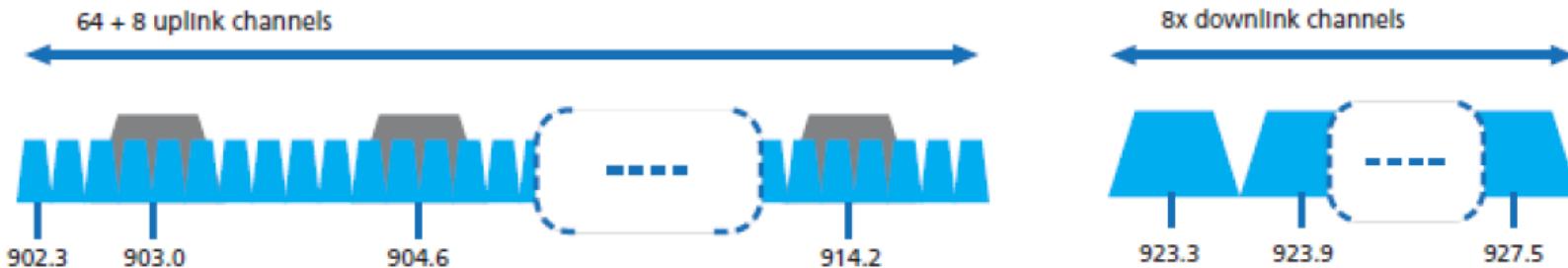
- Unlicensed Sub-GHz frequencies

- Europe
 - 433.05-434.79 MHz
 - 863-870 MHz
 - Australia: 915–928 MHz
 - North America: 902–928 MHz
 - India: 865–867 MHz
 - Southeast Asia: 433.05-434.79 MHz

https://lora-alliance.org/wp-content/uploads/2019/11/rp_2-1.0.0_final_release.pdf

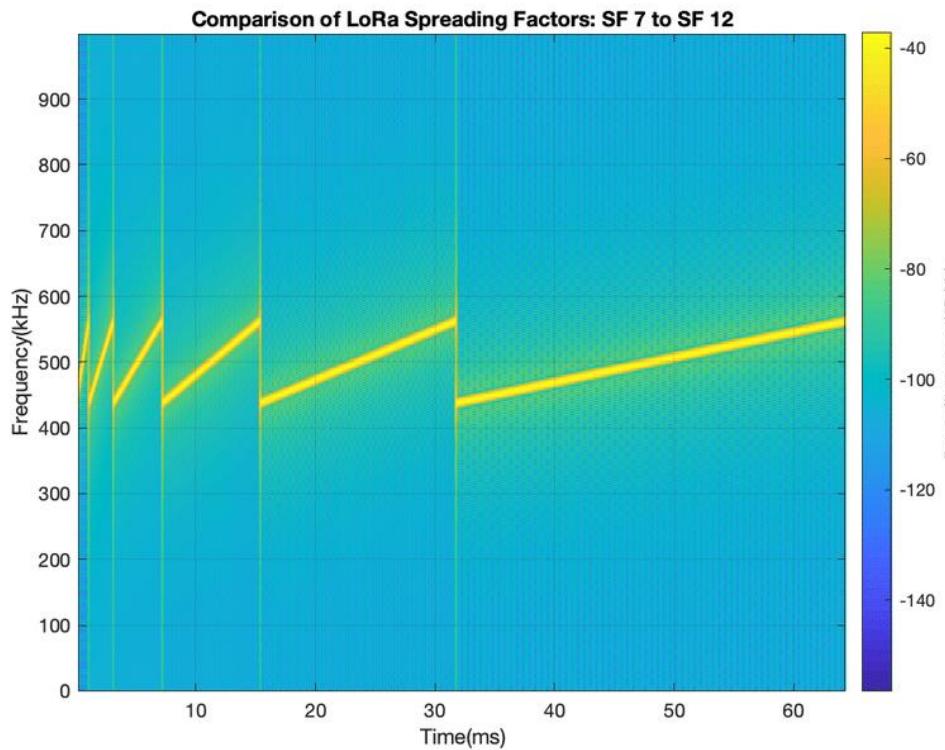
- Fixed bandwidths

- Uplink: 125 KHz channels or 500 KHz
 - Downlink: 500 KHz



Spreading Factor

- Processing gain by multiplying by spreading code
 - Increase in frequency component
- Six frequency spreading factors (SFs) are possible: 7-12
- Symbol rate is given by $R = BW/2^{SF}$
 - When spreading factor increases from n to $n+1$, the symbol duration ($T = 1/R$) doubles
- Tradeoff
 - Larger SF
- Adaptive
 - End device selects SF



ansmit

should use higher

Spreading Factor Orthogonality

- LoRa signals in same frequency bands with different SFs are orthogonal
- Two packets with the same SF in same slots will collide
 - However, if one of the two packets is stronger by six dB, it will survive.

LoRa Modulation Characteristics

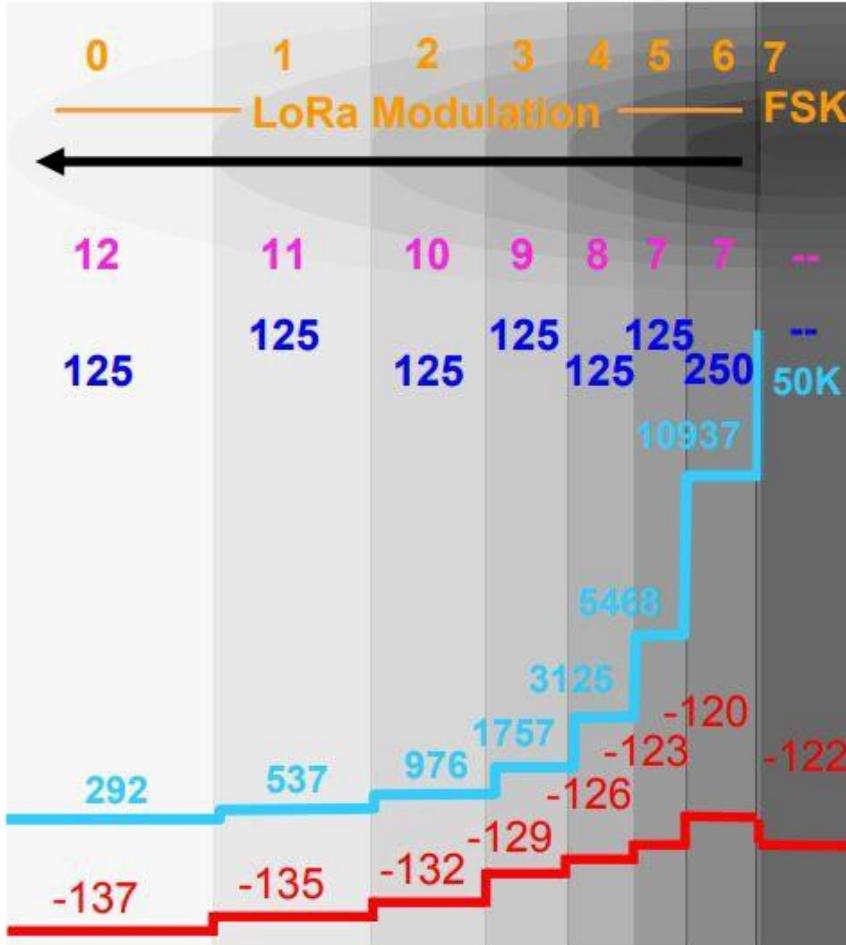
Data Rate (DR)	Spreading Factor (SF)	Channel Frequency	Uplink or Downlink	Bitrate (Bits/Sec)	Maximum User Payload Size (Bytes)
0	SF10	125 kHz	Uplink	980	11
1	SF9	125 kHz	Uplink	1,760	53
2	SF8	125 kHz	Uplink	3,125	125
3	SF7	125 kHz	Uplink	5,470	242
4	SF8	500 kHz	Uplink	12,500	242
5 – 7					
8	SF12	500 kHz	Downlink	980	53
9	SF11	500 kHz	Downlink	1,760	129
10	SF10	500 kHz	Downlink	3,125	242
11	SF9	500 kHz	Downlink	5,470	242
12	SF8	500 kHz	Downlink	12,500	242
13	SF8	500 kHz	Downlink	21,900	242

LoRa Modulation Characteristics

Modulation	Spreading factor	Bandwidth [kHz]	Maximum application throughput per channel [bps]	Maximum application layer throughput per end device per channel [bps]		
				10% duty cycle ¹	1% duty cycle ²	0.1% duty cycle ³
LoRa	12	125	146.1	14.61	1.46	0.15
LoRa	11	125	261.4	26.14	2.61	0.26
LoRa	10	125	584.2	58.42	5.84	0.58
LoRa	9	125	1359.2	135.92	13.59	1.36
LoRa	8	125	2738.1	273.81	27.38	2.74
LoRa	7	125	4844.7	484.47	48.45	4.84
LoRa	7	250	9689.3	968.93	96.89	9.69
GFSK	-	150	45660.4	1851.6 ⁴	456.6	45.66

https://www.researchgate.net/figure/Maximum-throughput-per-LoRaWAN-end-device-per-channel_tbl2_315119434

Spreading factor: Tradeoff



Data Rate (DR)
Range
Spreading Factor (SF)
Bandwidth (BW) (kHz)
Bitrate (BR) (bps)

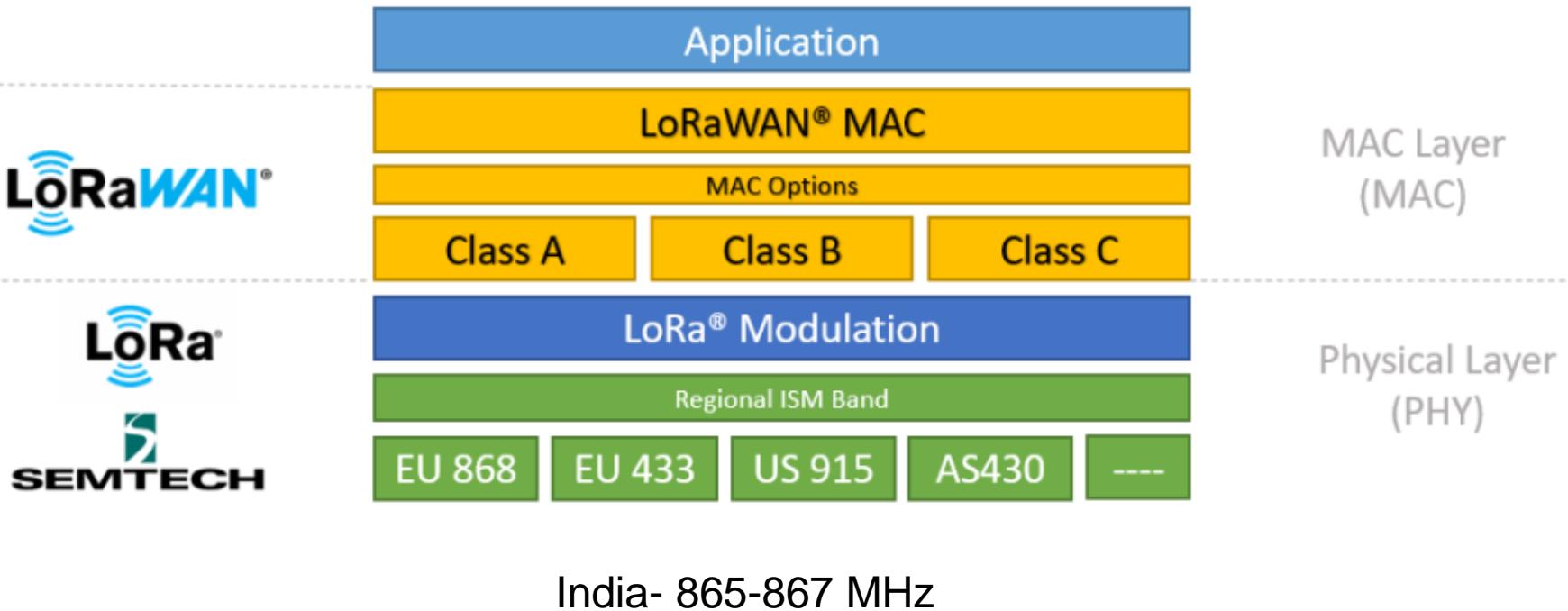
Receive Sensitivity (dBm)

Spreading Factor

Spreading Factor (For UL at 125 KHz)	Bit Rate	Range (Depends on Terrain)	Time on Air for an 11-byte payload
SF10	980 bps	8 km	371 ms
SF9	1760 bps	6 km	185 ms
SF8	3125 bps	4 km	103 ms
SF7	5470 bps	2 km	61 ms

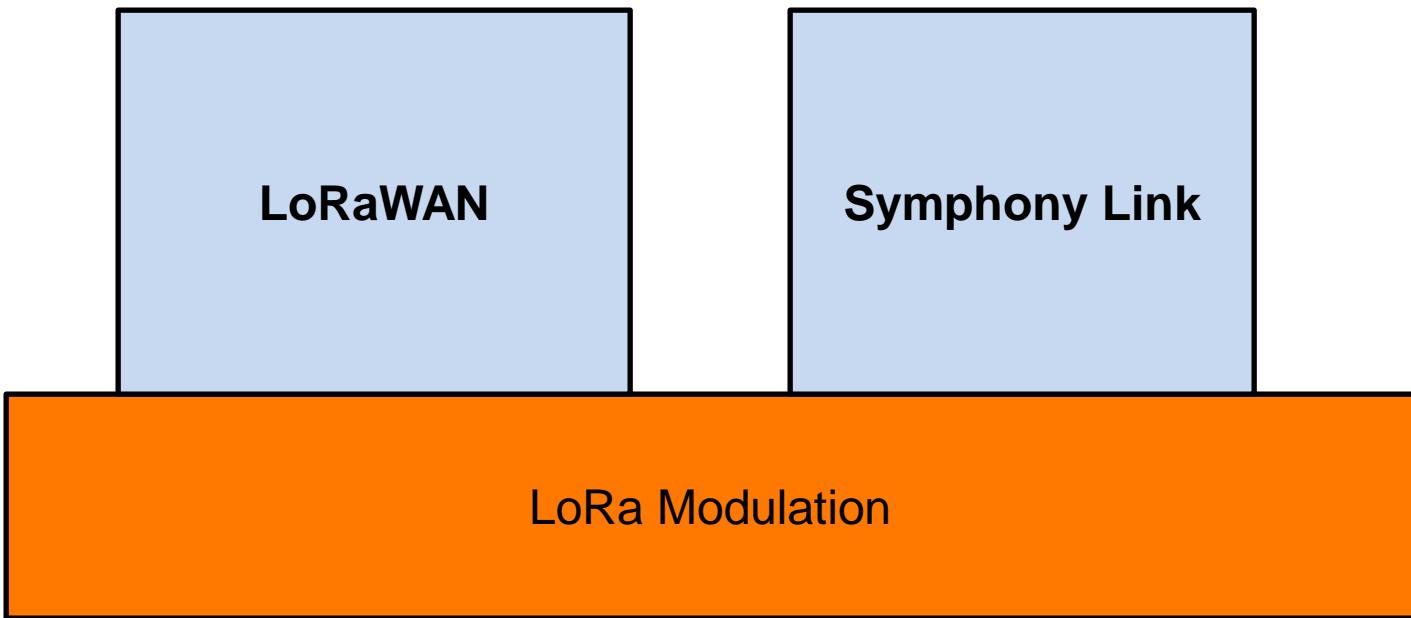
LoRaWAN

LoRaWAN Technology Stack

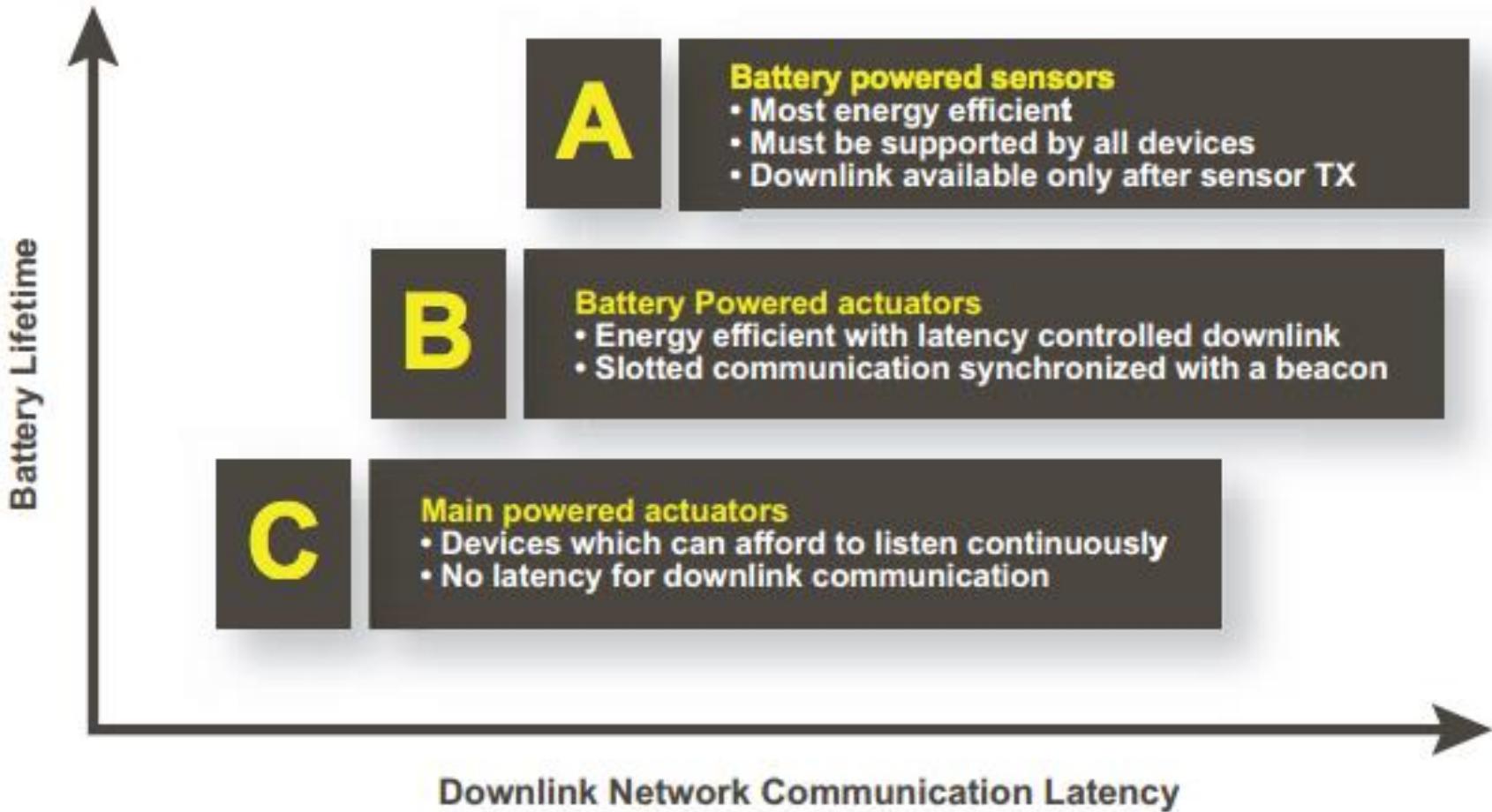


- LoRa is a proprietary RF modulation technology at PHY
- Created by Cycleo (acquired by Semtech)
- LoRaWAN is technology stack on top of LoRa
 - LoRaWAN alliance of more than 500 companies
 - Semtech a founding member

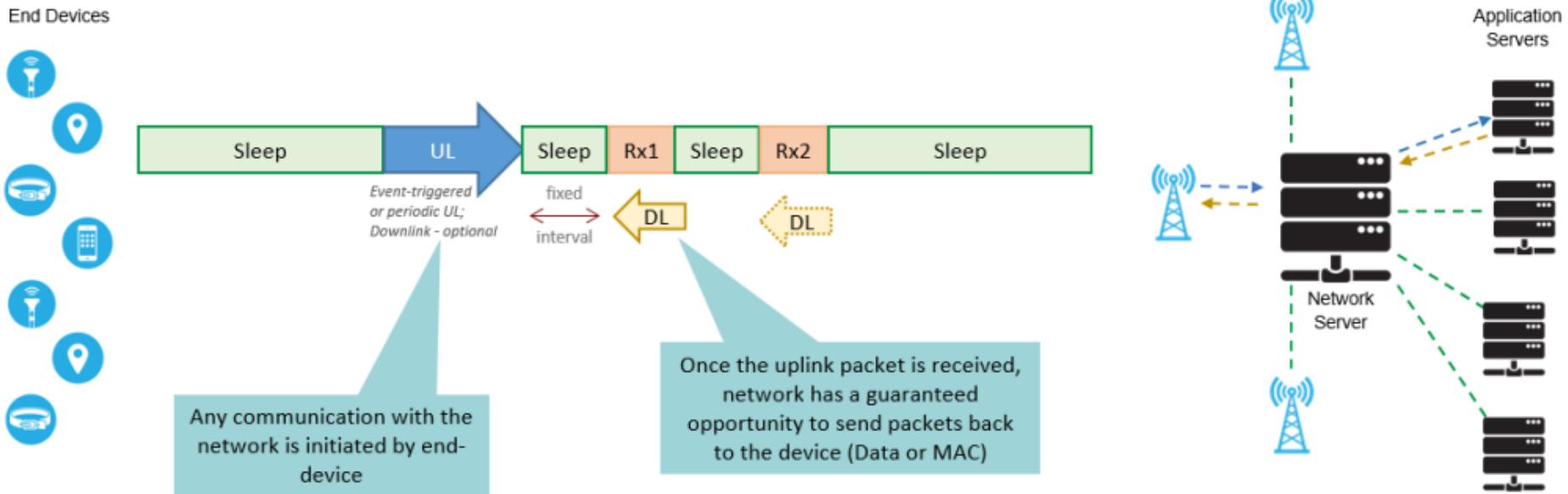
Other solutions possible with LoRa



LoRaWAN classes



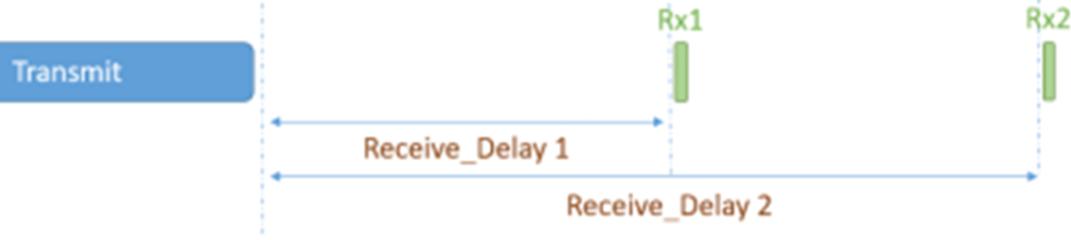
Class A operation



- Downlink only after uplink transmission
- Any communication is initiated by end-device
- Most energy efficient
- All devices should support this
- Default mode
- No way application can wakeup the end-device
 - Serious latency issues
 - Not suitable for actuators

Class A operation

Receive Windows: Nothing is received



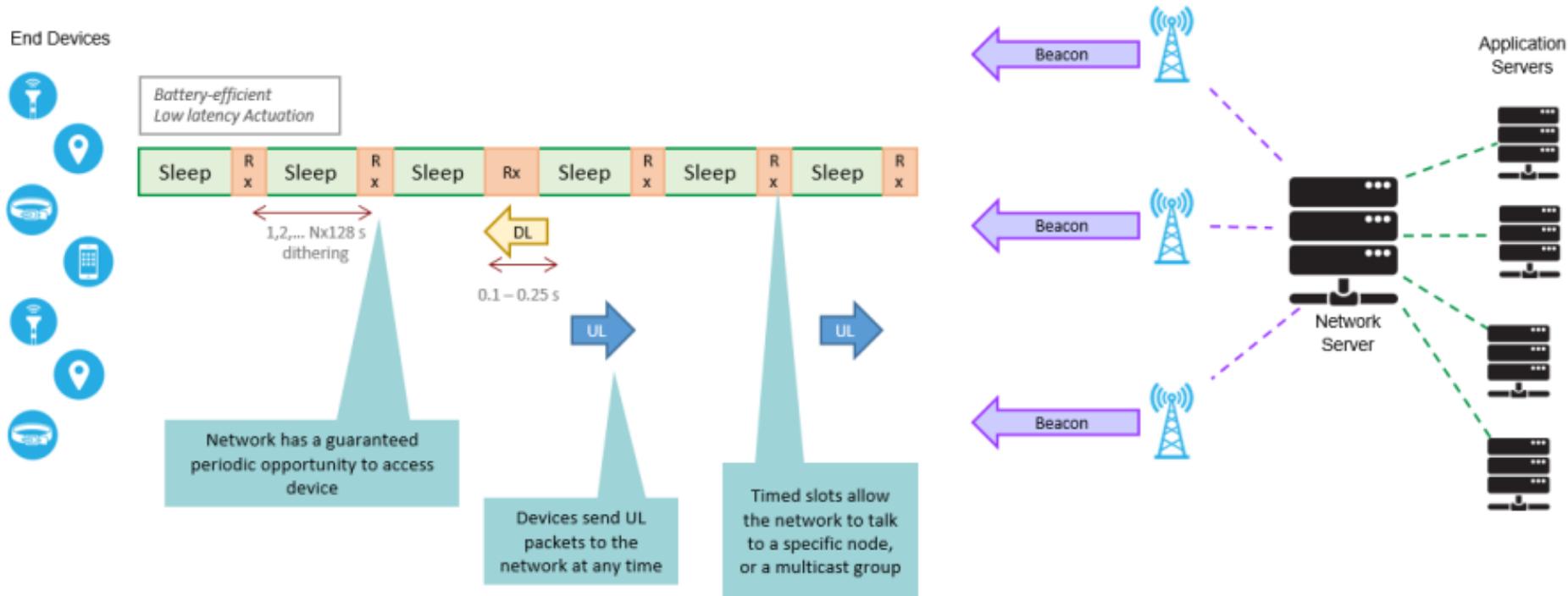
Receive Windows: Packet received in Rx1 window



Receive Windows: Packet is received in Rx2 window

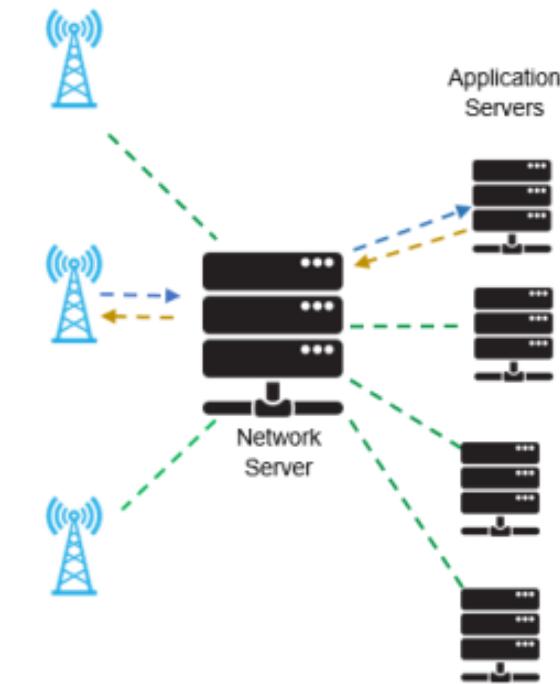
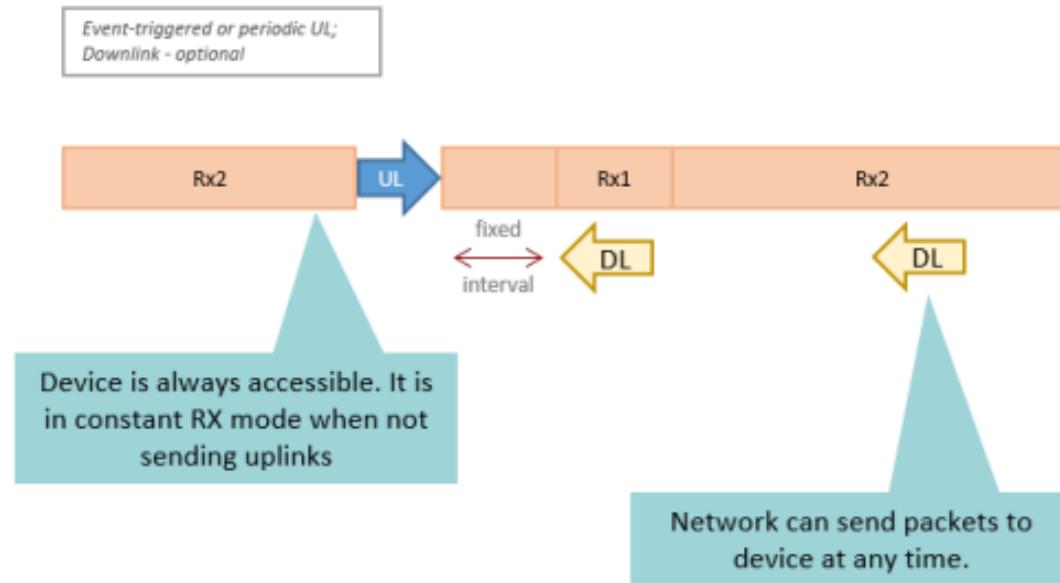


Class B operation



- Fixed time slots for an end device to receive downlinks
- Beacon required to synchronize the nodes
- Gateways need GPS timing source
- Beacon interval of 128 s (675 beacons in day)
- Suitable for battery powered actuators

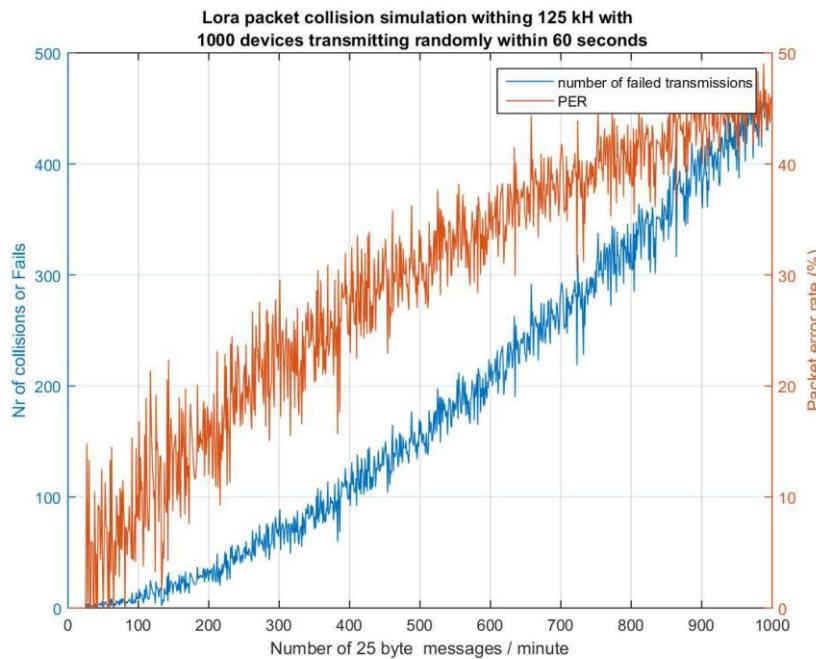
Class C Operation



- Class C devices are always ON
 - Streetlights, electrical meters
- Devices are always listening for downlink messages, unless they are transmitting an uplink
 - Lowest latency for communication from the server to an end device.

LoRaWAN

- Gateways listen on 8 frequencies on all spreading factor
- Collision prevented by maximum duty cycle



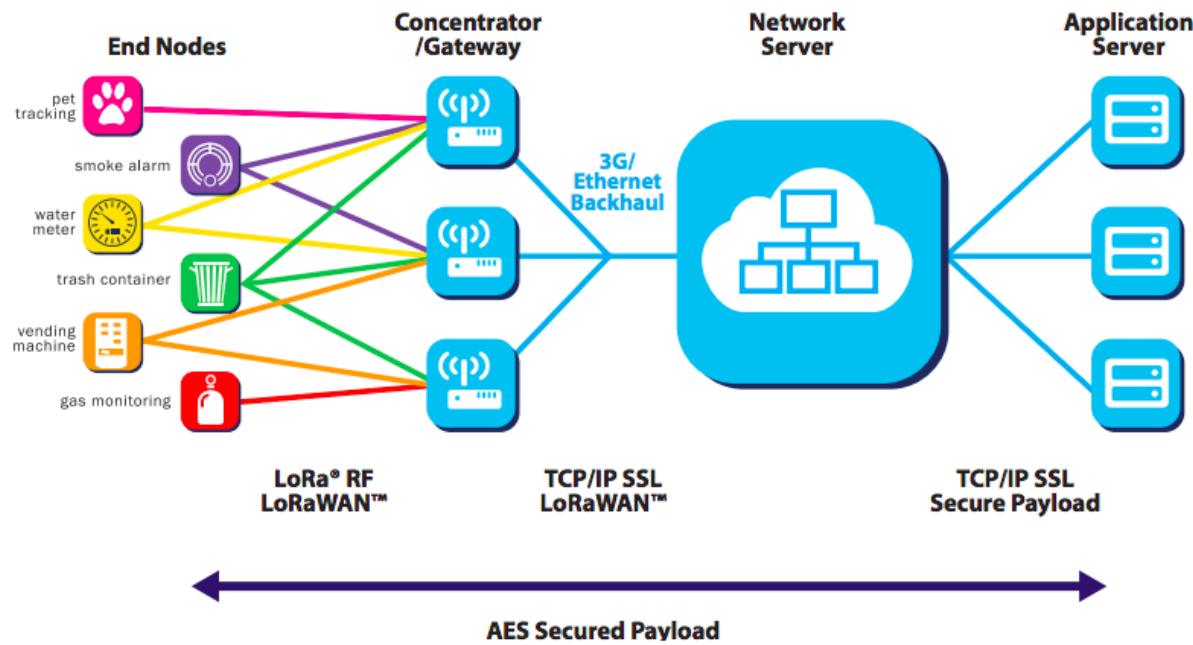
Source: <https://sites.google.com/a/wesdec.be/mwelyn/lpwan>

Capacity of network

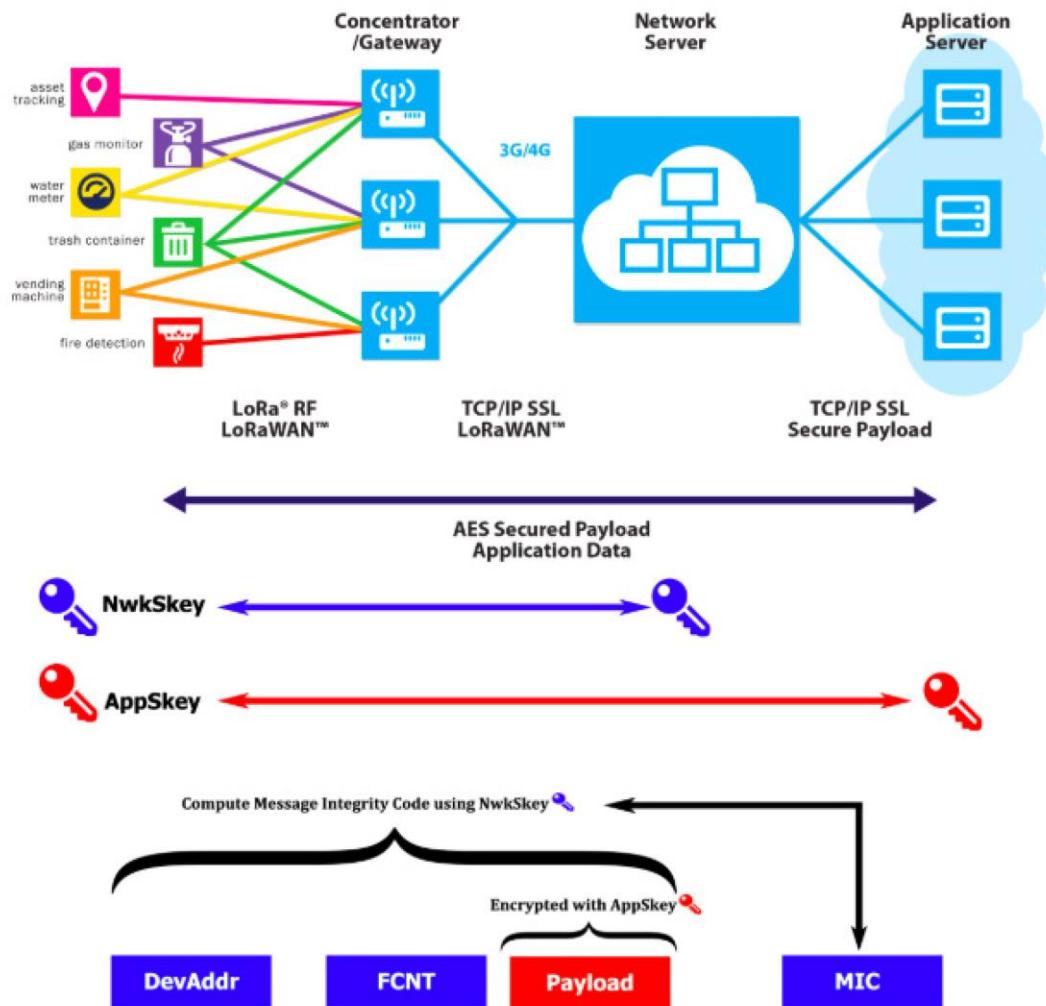
- LoRaWAN network can support millions of message
- A single gateway of 8 channels can support 1.5 Million messages over a day
 - If each device sends data every hour, a gateway can support 60,000 devices
- Add new gateway for more capacity and coverage
- Alternatively, we can use 16- or 64-channel gateway
- 64-channel is only used outdoors while others can be used indoors as well

LoRaWAN

- LoRaWAN is a software layer above LoRa
 - Pure Aloha (18.4% efficiency) + CSS
 - Dumb Gateways and Smart Server: *Filters data at server*
 - Makes transmission reliable by allowing retransmissions
 - Transposes data on IP network
 - Adds security as LoRa does not have security



LoRaWAN security



NwkSkey to guarantee the message integrity from the device to LoRa server

AppSkey to used for end to end AES-128 bit encryption from device to application server

IoT Features: *Advantages*

- Designed for majority of IoT applications
- Low powered sensors (Battery life of 2-5 years)
 - Class A and B
- Wide coverage area up to 15 Kms
- Low Costs
 - free(unlicensed) frequencies
- One gateway can support thousands of end devices
- Simple Architecture
- Security: a layer of security for the network and one for the application with AES encryption.
- Localization without GPS
- Roaming
- LoRa Alliance: 500+ members companies including IBM and Cisco

IoT Features: *Disadvantages*

- Payload limited to 100 bytes
- High latency (actuators are not possible)
- Low data rates
 - Does not support voice or video
- Low duty cycles (1% in EU)
- Interference issues
 - Unlicensed frequency for other technology users
 - Crowding of LoRaWAN gateways increase interference
 - High packet error rate
- Cost in terms of cloud-based servers for network and applications
 - Things Network, LoRIoT
- Needs fair amount of development work
 - DIY
 - Not a complete protocol stack

IoT Features: *Disadvantages*

- Not for continuous or real-time monitoring and actuations (most of low-latency industry cases)
 - High latency (actuators are not possible)
 - High packet error rate
 - Low data rates
 - Low duty cycles

Few Use Cases

- Utility monitoring
 - Water, Electricity, Gas
- Environment monitoring
 - Air Pollution, Water Quality, Soil Detection
- Animal Tracking
- Farming
- Smart building
 - Temperature, humidity sensors
- Smart Cities
 - Street Lights, Parking, Dustbins

References

- P. Lea, *Internet of Things for Architect*, Packt, 2018
- Semtech, “LoRa and LoRaWAN: A Technical Overview,” Feb. 11, 2020
 - [Online: https://lora-developers.semtech.com/uploads/documents/files/LoRa_and_LoRaWAN-A_Tech_Overview-Downloadable.pdf]

Questions?

Ungraded Quiz

1. Arrange in the order of lower to higher latency? ABC, CBA, CAB, BAC
2. Data rates increase with increasing spreading factor?
Yes, No, Does not depend of SF
3. LoRaWAN can be used for low-latency industrial IoT applications? Yes, No

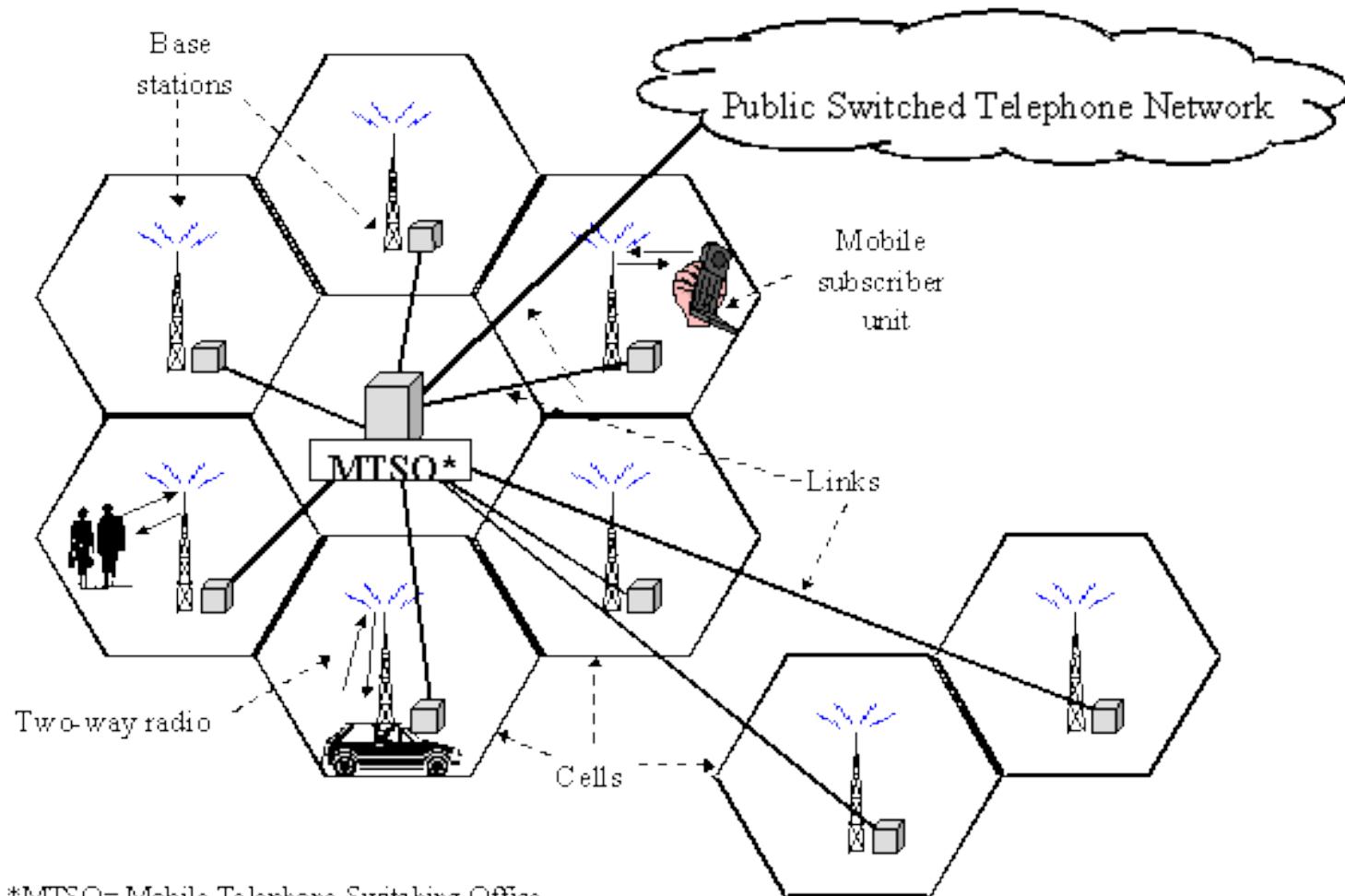
Cellular Technologies: 2G, 3G, 4G, 5G

Cellular Technologies

- Licensed
 - 2G, 3G, 4G
 - LTE-MTC: Cat-1, Cat-0, LTE-CatM1, NB-IoT, 5G
 - low-power, long-range applications
- Unlicensed
 - Example: Sigfox, LoRa, Weightless
 - +ves: Extremely low-power, low data rate, long coverage
 - -ves: Unlicensed band, deployment should exist in region of interest, gateways,

Cellular Architecture: an example of 2G

<https://www.itu.int/osg/spu/ni/3G/technology/index.html>



Good introduction to cellular communication: https://www.youtube.com/watch?v=1JZG9x_VOwA

Spectrum Chart in India

- 2G: GSM 900 / GSM 1800
- 3G: UMTS 900, UMTS 2100
- 4G: LTE 850, 1800, 2100, 2300, 2500

<https://www.gsmarena.com/network-bands.php3?sCountry=INDIA>

Exact allocation:

<https://dot.gov.in/sites/default/files/NFAP%202018.pdf?download=1>

Spectrum Chart in India

				Uplink Frequency (MHz)	Downlink Frequency (MHz)
700 MHz	FDD	x2	B28 - n28	703-748 MHz	758-803 MHz
800 MHz	FDD	x2	B5	824-844 MHz	869-889 MHz
ISM Band	Industrial, Scientific and Medical - License Free Band. *T&C Apply. 200 kHz carrier bandwidth. Currently used in India for LoRa etc.				
900 MHz	FDD	x2	B8	890-915 MHz	935-960 MHz
1800 MHz	FDD	x2	B3	1710-1785 MHz	1805-1880 MHz
2100 MHz	FDD	x2	B1	1920-1980 MHz	2110-2170 MHz
2300 MHz	TDD		B40		2300-2400 MHz

Cellular Spectrum Chart in India

Circle	AP	AS	BH	DL	GU	HA	HP	JK	KA	KE	KO	MP	MH	MU	NE	OD	PU	RA	TN	UPE	WB	AGG.	
Band	Type Telco	A	C	C	M	A	B	C	C	A	B	M	B	A	M	C	C	B	B	A	B	B	Telco AGG.
B5 800MHz	AIRTEL	3.75			1.25		1.25						2.5	2.5									11.25
	JIO	3.75	5	5	7.5	8.75	5	5	5	7.5	7.5	8.75	5	3.75	5	5	5	7.5	7.5	7.5	6.25	7.5	137.5
	RCOMM	1.25	5	5	1.25	2.5	5	5	5	1.25	1.25	1.25	5	1.25	5	5	5	2.5	1.25	1.25	3.75	1.25	66.25
	TSP AGG.	8.75	10	10	10	11.25	11.25	10	10	8.75	8.75	10	10	7.5	12.5	10	10	10	8.75	8.75	10	8.75	
	BAL	13.75	2.5	12.5	12.5	6.25	10	10	2.5	13.75	13.75	12.5	12.5	15	10	2.5	11.25	11.25	12.5	13.75	12.5	12.5	236.25
	CB AGG.																						
B8 900MHz	AIRCEL		4.4						4.4							4.4							13.2
	AIRTEL	9	8	7.8	6			7.4	6.2	8.8		7			5	8.8	7.4	10	6	6.2	6.2	6.6	116.4
	BSNL	6.2	6.2	6.2	6.2	6.2	6.2	6.2	8	6.2	6.2	6.2	6.2	6.2	6.2	6.2	6.2	6.2	6.2	6.2	6.2	138.2	
	JIO																						0
	RCOMM							5					5										10
	Vi	5			10	11	12.2		5	12.4	7	7.4	14	11		5	5.6	6.4	6.2	5.6	6.6		141.6
	TSP AGG.	20.2	18.6	14	22.2	17.2	18.4	18.6	18.6	20	18.6	20.2	18.6	20.2	22.2	19.4	18.6	21.8	18.6	18.6	18	19.4	
	BAL	3	4.6	9.2	1	3	0.2	4.6	0	3.2	4.6	3	4.6	3	1	3.8	4.6	0	0	17	5.2	4.6	81.4
B3 1800MHz	AIRCEL	4.4	1.8	6.2	4.4			1.8	4.4	4.4	4.4			4.4	1.8	4.4	4.4	6	10	6.2	5.6		74.6
	AIRTEL	21.4	15.45	15.2	7	16.2	16.2	10.2	5	8.8	11.2	9	17	23.2	20.2	10	11.8	10	10	8	12.8	6.2	283.05
	BSNL	3.8	3.8				3.8	3.8		3.8	3.8		3.8			3.8	3.8			3.8	3.8	1.8	47.4
	JIO	5.8	5.4	5	5.4	6	5	10.4	10	5	5	10	6.4	5	6.6	6.4	5	5.2	10	6.8	6.4	10.6	146.4
	RCOMM						0.6	5	0.6					0.6	5	5	0.6						17.4
	Vi	11	25	17.8	18.6	20.8	15.8	15.6	17	19	20	15	18.6	12.4	14.6	25.8	17	21.2	16.2	12.4	14.8	23.4	
	TSP AGG.	46.4	51.45	44.2	35.4	43	41.4	40	38.8	41.6	44.4	38.4	45.8	40.6	46.4	52.8	47	41.4	42.2	41	44	47.6	
	BAL	12.6	3	8.6	25.4	17.8	19.4	19	14	21	16.4	14.4	15	22.2	15.6	0	5.8	19.4	16.8	17.6	15	5.2	323.6

<https://telecomtalk.info/india-spectrum-data-sheet/134245/>

Cat-1, Cat-0, eMTC, NB-IoT and EC-GSM-IoT

LTE-M or LTE-MTC or LTE Cat M1

	LTE Cat 1	LTE Cat 0	LTE Cat M1 (eMTC)	LTE Cat NB1 (NB-IoT)	EC-GSM-IoT
3GPP Release	Release 8	Release 12	Release 13	Release 13	Release 13
Downlink Peak Rate	10 Mbit/s	1 Mbit/s	1 Mbit/s	250 kbit/s	474 kbit/s (EDGE) 2 Mbit/s (EGPRS2B)
Uplink Peak Rate	5 Mbit/s	1 Mbit/s	1 Mbit/s	250 kbit/s (multi-tone) 20 kbit/s (single-tone)	474 kbit/s (EDGE) 2 Mbit/s (EGPRS2B)
Latency	50–100ms	not deployed	10ms–15ms	1.6s–10s	700ms–2s
Number of Antennas	2	1	1	1	1–2
Duplex Mode	Full Duplex	Full or Half Duplex	Full or Half Duplex	Half Duplex	Half Duplex
Device Receive Bandwidth	1.08 – 18 MHz	1.08 – 18 MHz	1.08 MHz	180 kHz	200 kHz
Receiver Chains	2 (MIMO)	1 (SISO)	1 (SISO)	1 (SISO)	1–2
Device Transmit Power	23 dBm	23 dBm	20 / 23 dBm	20 / 23 dBm	23 / 33 dBm

LTE-M and NB-IoT enhanced in Release 14

http://www.3gpp.org/images/articleimages/iot_summary_large.jpg

Cat-1, Cat-0, and EC-GSM-IoT

- Cat-1
 - IoT support in LTE 3G (Release 8)
 - Already standardized
 - Premium IoT applications
- Cat-0
 - IoT support in LTE-A 4G (Release 12)
 - Optimized cost as compared to Cat-1
- EC-GSM
 - IoT support in GSM networks
 - Extended Coverage

Cat-M1 (eMTC)

- IoT support in LTE-A 4G (Release 13)
- Compatible with existing LTE network (only software upgrade)
- 1.4 MHz bandwidth
- Supports mobility and voice-over LTE (VoLTE)
- Asset tracking and wearables

NB-IoT or Cat-M2

- Competing against Sigfox, LoRa (Release 13)
- Support of massive number of low throughput devices, ultra-low device cost, low device power consumption and optimized network architecture
- Improved indoor coverage
 - Power boosting
 - Repetition
- flexible spectrum: in-band and guard band in LTE; standalone deployment; GSM re-farming possible
- Not backward compatible with other 3G/4G devices
- No mobility support and not suitable for latency low applications
- LPWAN applications like smart metering

NB-IoT deployment

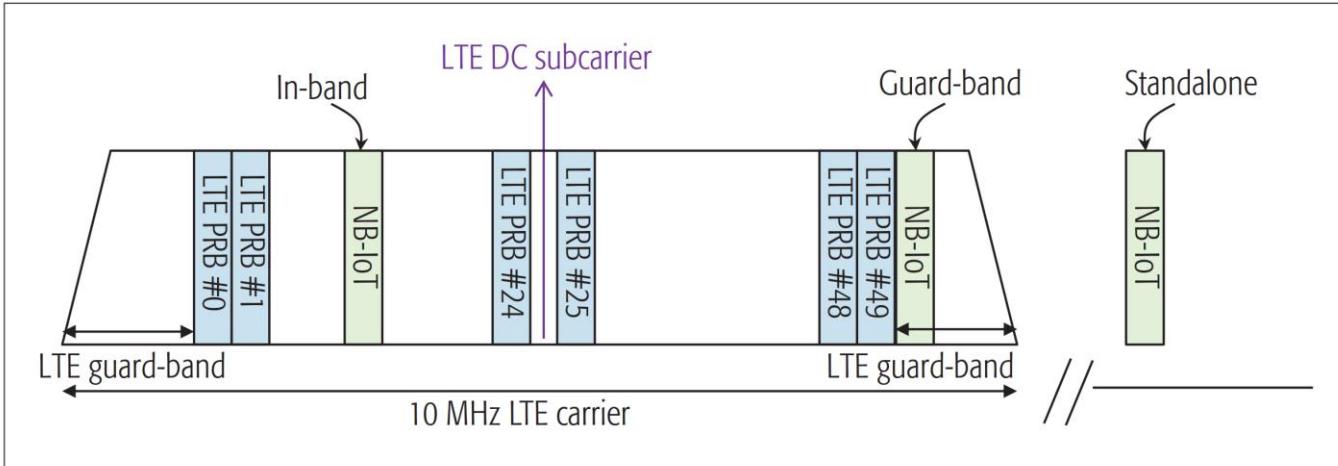


Figure 1. Examples of NB-IoT stand-alone deployment and LTE in-band and guard-band deployments.

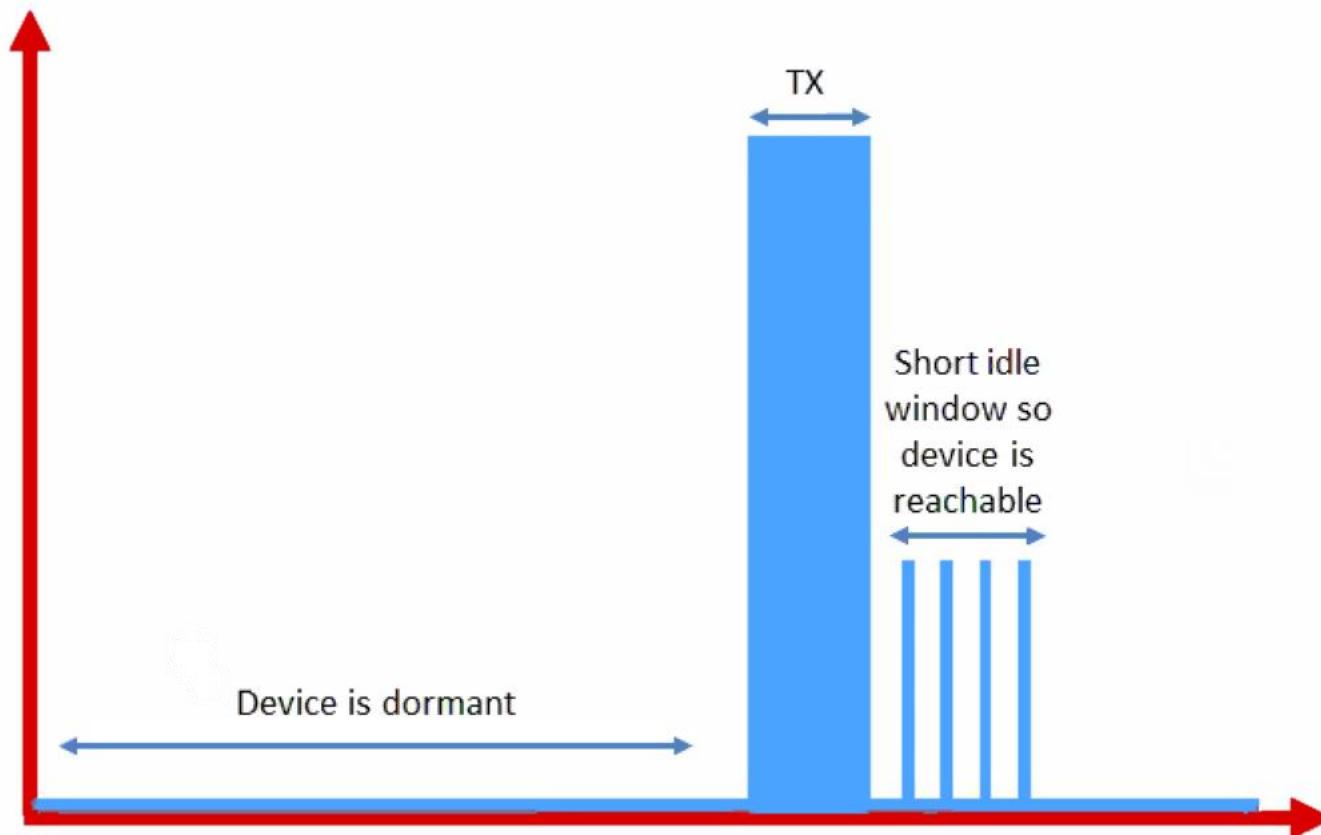
<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7876968>

Few Power Saving Features

- Power Saving Mode
- Extended Discontinuous Reception

Power Saving Mode

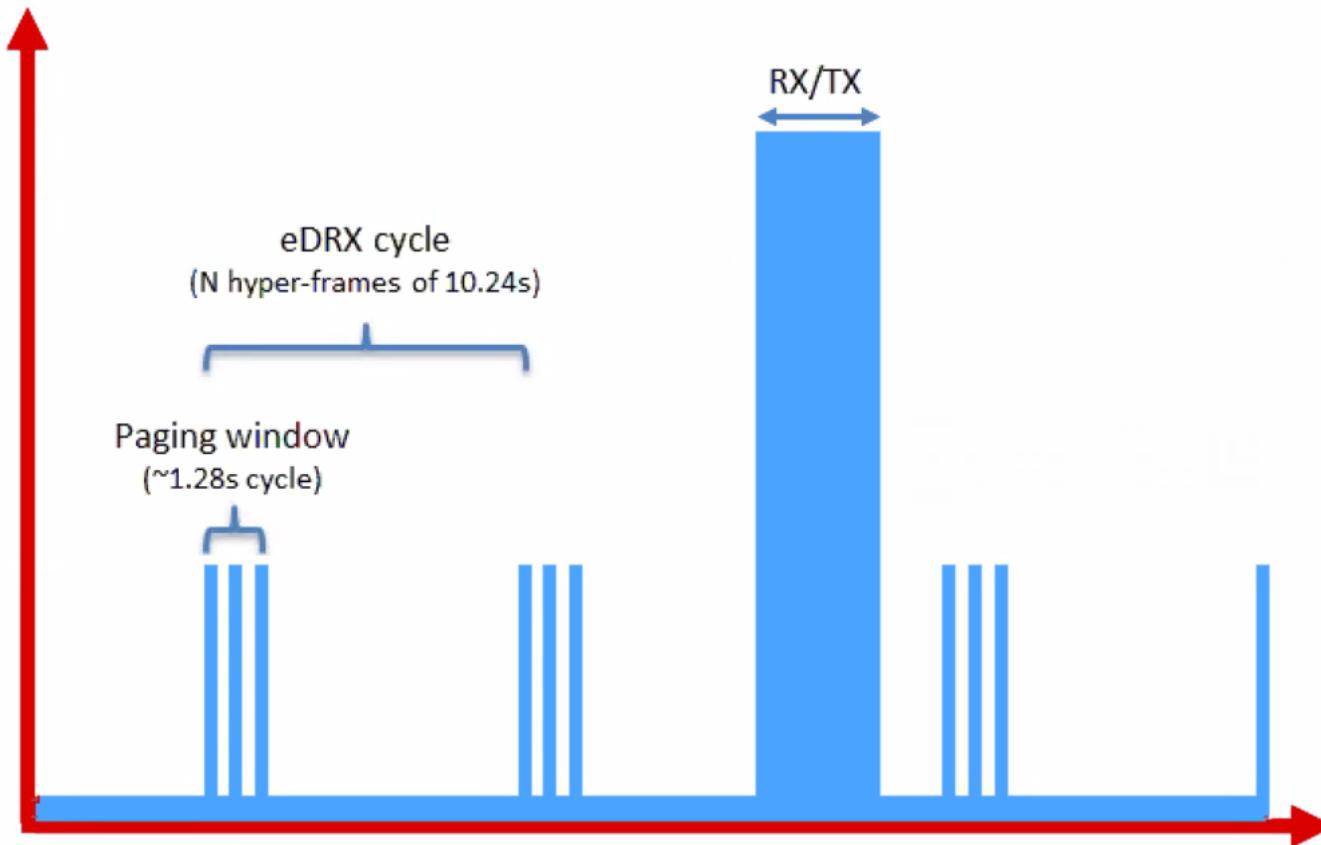
PSM allows LTE-M devices to go idle without having to re-join the network when they wake up.



<https://www.link-labs.com/blog/lte-e-drx-psm-explained-for-lte-m1>

Extended Discontinuous Reception (eDRX)

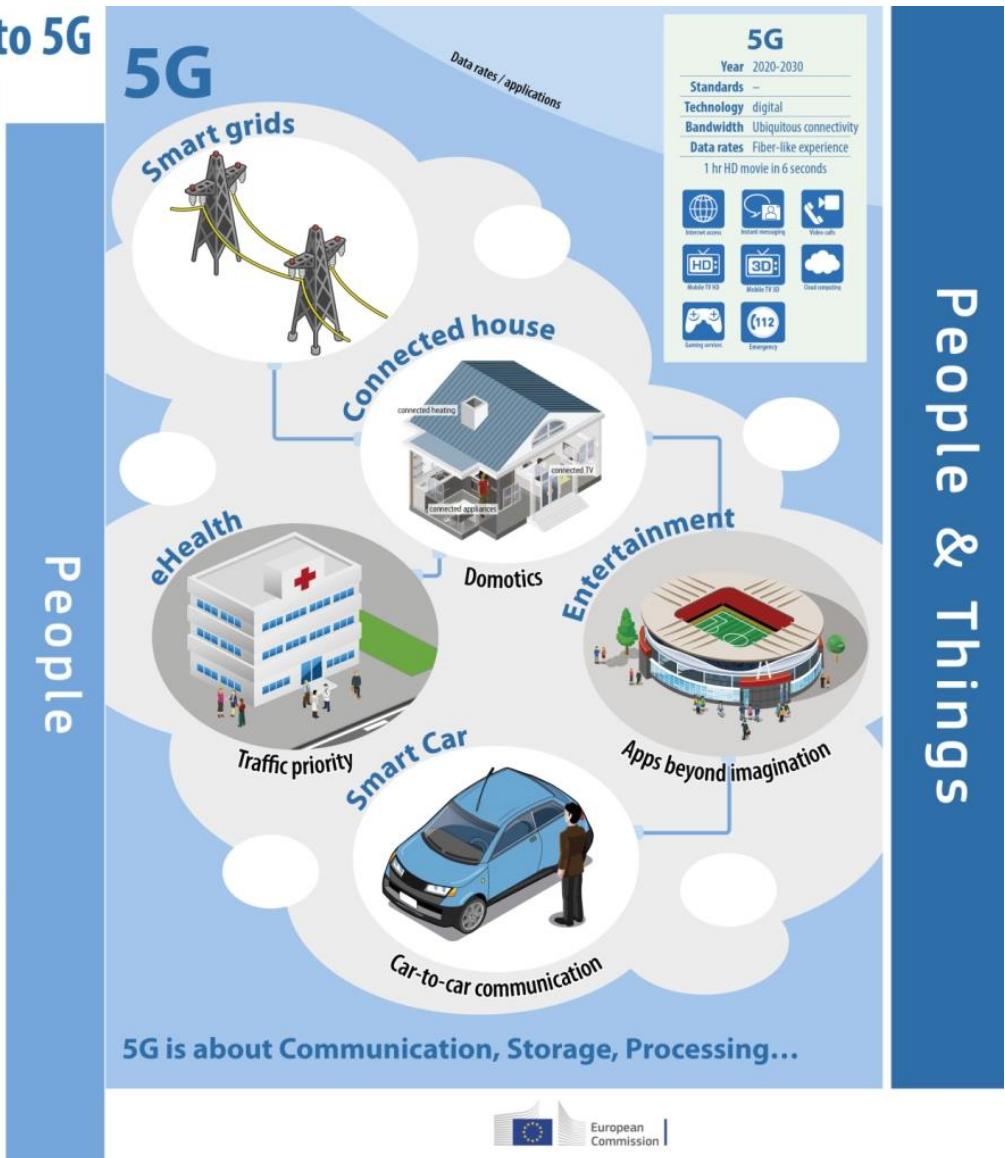
Node can skip several paging cycles



1G to 5G

Mobile communications: from 1G to 5G

Generation	Device	Specifications
1G		<p>1G</p> <p>Year: early 80s Standards: AMPS, TACS Technology: Analog Bandwidth: – Data rates: –</p>
2G		<p>2G</p> <p>Year: 1991 Standards: GSM, GPRS, EDGE Technology: Digital Bandwidth: Narrow Band Data rates: < 80 - 100 Kbit/s</p>
3G		<p>3G</p> <p>Year: 2001 Standards: UMTS / HSPA Technology: digital Bandwidth: Broad Band Data rates: up to 2 Mbit/s</p>
4G		<p>4G</p> <p>Year: 2010 Standards: LTE, LTE Advanced Technology: digital Bandwidth: Mobile Broad Band Data rates: xDSL-like experience 1 hr HD movie in 6 minutes</p>



IoT in 5G Era



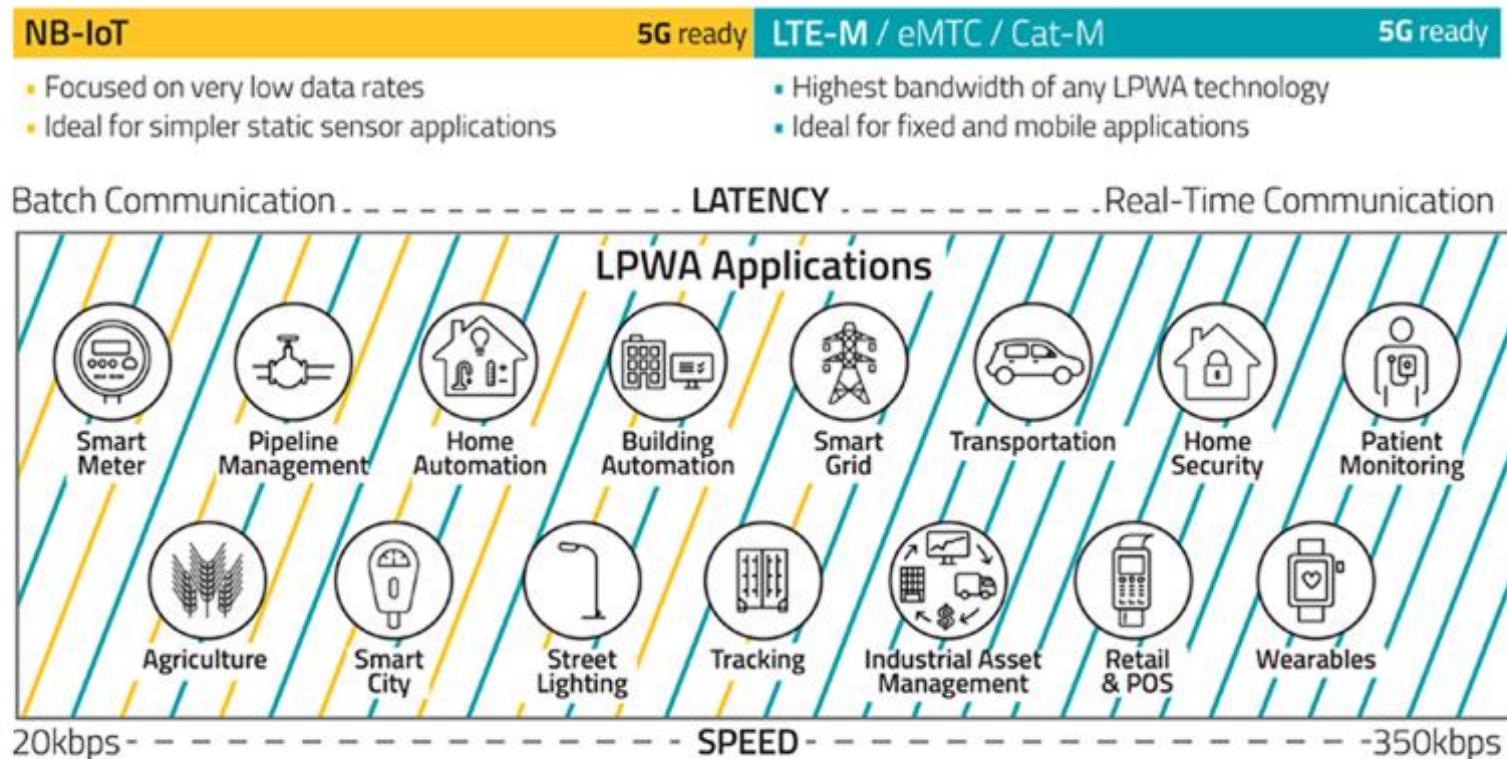
- **Mobile IoT/Massive IoT/LPWA:** improved network coverage, long device operational lifetime and a high density of connections. This is also known as mMTC (Massive MTC)
- **Enhanced Mobile Broadband:** improved performance and a more seamless user experience accessing multimedia content for human-centric
- **Critical Communications:** high performance, ultra-reliable, low latency industrial IoT and mission critical applications. This is also known as Critical IoT, URLLC (Ultra Reliable Low Latency Communications)

Source: GSMA

LTE-M and NB-IoT=Massive IoT

IoT in 5G era

Two Leading LPWA Technologies



<https://www.iotforall.com/cellular-iot-explained-nb-iot-vs-lte-m/>

Key IoT features

Advantages

- Quality of service
 - Licensed band
 - Low latency
- Ubiquitous
- Security: SIM card protection + AES 256 bit (best)
- Great coverage: 2 km (LTE), 10 km (NB-IoT), 35 km (GSM)
- Global mobility and roaming support
- Scalable
- Connected even during power failure

Issues

- Cost: License, Capex and Opex, subscription
- High power
- Not possible to make your own network

References

- [Kurose2012] J. Kurose and K. Ross, *Computer Networking, 5th edition*, Pearson, 2012
- [Sohraby2007] K. Sohraby, D. Minoli, and T. Znati, *Wireless Sensor Networks: Technology, Protocols, and Applications*, Wiley, 2007
- [Koubaa2007] Anis Koubaa, Mário Alves and Eduardo Tovar, Time Sensitive IEEE 802.15.4 Protocol, *Sensor Networks and Configuration*, pp. 19-49
- [Townsend2014] K. Townsend, C. Cufi, Akiba, R. Davison, *Getting Started With BLE*, O'Reilly, May 2014
- [Gonzalez2016] V. Gonzalez et. Al. “IEEE 802.11ah: A Technology to face the IoT Challenge,” *Sensors*, 2016
- [Park2015] M. Park, “IEEE 802.11ah: Sub-1-GHz license-exempt operation for the internet of things,” *IEEE Communications Magazine*, September 2015.
- [Dohler2016] M. Dohler, et. al. “Internets of Things in 5G Era: Enablers, Architecture, and Business Models,” *IEEE Journal on Selected Areas in Communications*, Vol. 34, No. 3, March 2016
- [Mekki2019] K. Mekki, E. Bajic, F. Chaxel, and F. Meyer, “A comparative study of LPWAN technologies for large scale IoT deployment,” *Science Direct, ICT express* 5 2019.
- [IEEE802.11ax] Broadcom, *White paper on IEEE 802.11ax*, 17 Oct. 2018

**That's all for today!
Thank You!**

Communications & Controls in IoT

WiSUN and BLE

Instructor: Sachin Chaudhari

Feb. 20, 2023



**INTERNATIONAL INSTITUTE OF
INFORMATION TECHNOLOGY**

HYDERABAD

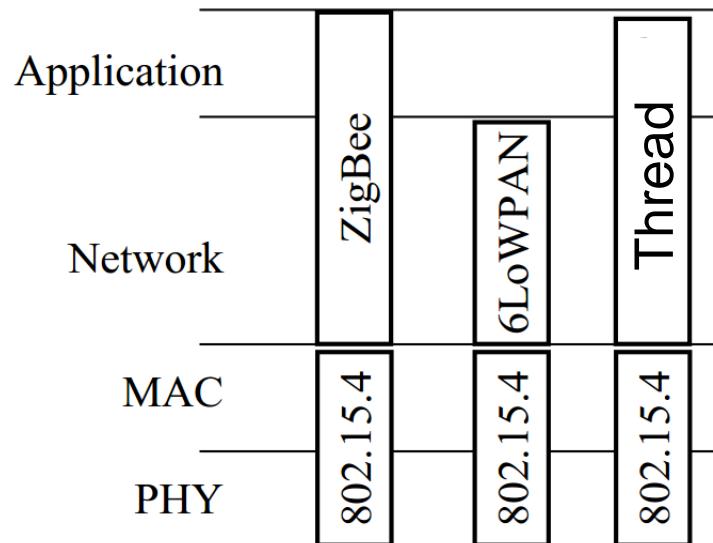
Recap

IEEE 802.15.4

Ref: K. Sohraby, D. Minoli, T. Znati, *Wireless Sensor Networks*, Wiley, 2007

IEEE 802.15.4

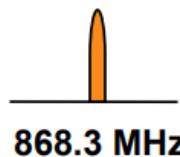
- IEEE 802.15.4 defines the operation of low-rate wireless personal area networks (LR-WPANs)
- Widely used in wireless sensor-network (WSN) applications
 - Vast number of industrial, home and medical applications
- It specifies the physical layer (PHY) and media access control (MAC) for LR-WPANs
- Does not have IP address
- Used by several “Internet of Things” protocols:
 - ZigBee, 6LowPAN, Thread, WiSuN etc.



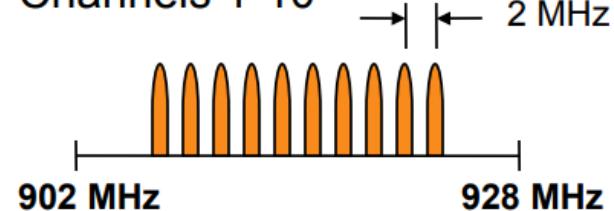
Physical Layer (PHY): Operating Frequency Bands

**868MHz/915MHz
PHY**

Channel 0

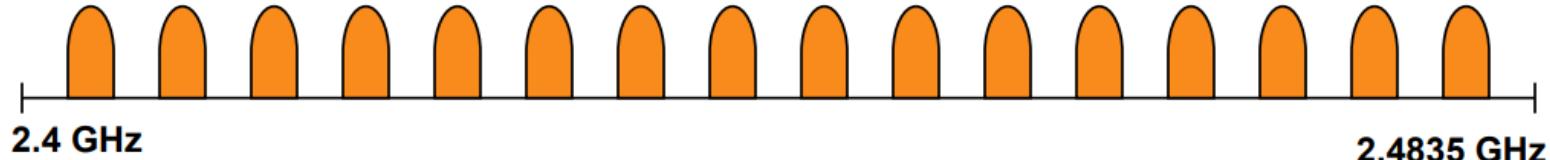


Channels 1-10



**2.4 GHz
PHY**

Channels 11-26



PHY: Modulation Parameters

Freq. band (MHz)	Spreading Parameters		Data Parameters		
	Chip rate (kchip/s)	Modulation	Bit rate (kbps)	Symbol rate (ksymbol/s)	Symbols
868	300	BPSK	20	20	Binary
915	600	BPSK	40	40	Binary
2400	2000	O-QPSK	250	62.5	16-ary

[Koubaa2007]

All bands are based on Direct sequence spread spectrum (DSSS)

Additional Tasks of PHY of IEEE 802.15.4

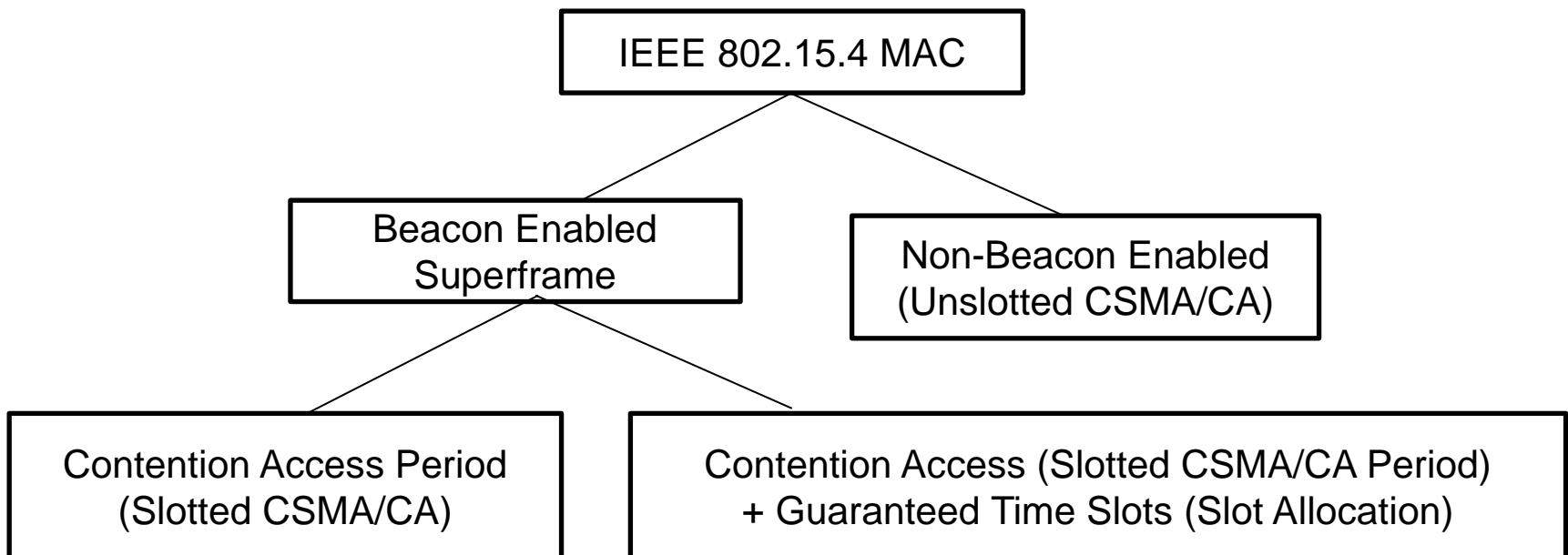
- **Activation and deactivation of the radio transceiver**
 - Three states: Transmitting, receiving and sleeping
- **Receiver energy detection**
 - No decoding or signal identification
 - Required to understand if the channel is busy or idle
- **Link quality indication**
 - Using energy or SNR estimation or both
- **Clear channel assessment**
 - Energy detection or carrier sense or both
- **Channel frequency selection**
 - 27 channels

MAC Layer features

- Designed to support vast number of industrial and home applications for control and monitoring
- Enabling deployment of large number of devices with low cost and complexity
- Several features for flexible network configuration and low-power operation
 - Different topologies and network devices
 - Optional superframe structure with duty-cycle control
 - Both contention and scheduled based MAC protocols
 - Synchronized and non-synchronized operation
 - Efficient energy management
 - Adaptive sleep
 - Extended sleeping time
 - Flexible addressing scheme for large number of nodes

MAC layer functions

- Network association and disassociation
- Two modes of operation
 - Beaconing
 - Non-beaconing



IEEE 802.15.4 Versions

- Since the first version in 2003, new amendments are constantly being introduced.
- Modifications
 - New country specific (frequencies, regulation)
 - New application and network specific:
 - SUN: Smart utility meter monitoring
 - LECIM: Low Energy Critical Infrastructure Monitoring
 - RFID: Radio Frequency Identification
 - RCC: Railway Communications and Control
 - TVWS: TV White Space
 - Medical
 - New PHY specific
 - OFDM, ASK, FSK, QAM, GMSK, MSK, OOK
 - New Protocols
 - TSCH, Aloha, PCA

IEEE 802.15.4 Versions

Not in Syllabus for Exam

Versions	Date	Type of network	Max Data rate kb/s	Modulation Encoding	protocole used	Features
802.15.4e	2012	Industrial LR-WPAN	–	–	DSME LLDN TSCH	QoS, Security Minimizing collisions Deterministic yet flexible bandwidth Interference avoidance Multi-channel, multi-superframe High reliability of the system
802.15.4f	2012	RFID	250	MSK OOK PPM FSK BPSK QPSK QAM O-QPSK	ALOHA	Multi-year battery life Reliable communications Precision location
802.15.4g	2012	SUN	800	O-QPSK	CCA	Interference avoidance Security
802.15.4j	2013	MBAN	250	BPSK O-QPSK	– –	Keeping a channelization scheme flexible
802.15.4k	2013	LECIM	–	FSK GFSK P-FSK P-GFSK	CSMA/CA PCA ALOHA PCA	Reduction of collision probability Good transmit power efficiency Higher sensitivity Priority Forward error correction QoS, security

Today's Class

Wireless Smart Ubiquitous Network (WiSUN) Field Area Networks (FAN)



Good References

- <https://www.wi-sun.org/>
- https://www.wi-sun.org/wp-content/uploads/Wi-SUN-Alliance-Comparing_IoT_Networks-r1.pdf

WiSUN Field Area Networks (FAN)



Field Area Network (FAN) provides connectivity to a large number of devices spread throughout a given geographic area

WiSUN-FAN

- An open specification based on the **IEEE 802.15.4g-2012** specification, as well as other IEEE 802 and IETF standards
- Application focus
 - smart cities
 - smart utilities
 - smart lighting
 - Mostly competing against LoRaWAN and NB-IoT
- Wi-SUN Alliance
 - Established in 2011
 - Manages the specifications and certifications to ensure interoperability
 - 300+ members
 - 100 million + devices worldwide
 - 150+ WiSUN certified products

IEEE 802.15.4g-2012: WiSUN

- Amendment made for Smart Metering Utility Networks (SUN)
- PHY
 - Support **outdoor**, low-data rate and wireless applications under multiple regulatory domain
 - **Multiple PHY** layers for targeting different markets and applications
 - Three alternate PHY provided
 - Multi-rate and multi-regional FSK (MR-FSK)
 - Good transmit power efficiency
 - Multi-rate and multi-regional OFDM (MR-OFDM)
 - Higher data rates in multipath fading channels
 - Multi-rate and multi-regional OQPSK (MR-OQPSK)
 - Same characteristics as previous versions

WiSUN Features

- Both star and mesh topologies are possible
 - Powerful use of mesh topology
 - One cluster of 5000 sensor nodes in 1 Km range
- Unlicensed band operation
- Leverages IPv6
- Can be designed for frequent communications
 - Every 10 secs
- Low latency of about 0.02-1 secs
- Data rates of upto 300 Kbps
- Coverage of 4 Km point-to-point using 1W non-directional antenna
- Security at multiple levels
 - Native public-key infrastructure (PKI) integration providing security certification capabilities for each device
- Networks designed for long lifecycle
 - Low power design (15-20 year battery target)
 - Backward and forward compatibility with different generations

WiSUN: Mesh topology

- Self-forming, self-healing
 - Easy to add new devices
 - Robustness against outages or node-failures
- Unlike Zigbee, designed for a much greater scale and several hops (around 30)
- With sufficient density, very robust against blockages and outages
- Coverage gaps can be filled with additional devices
- Star topology may encounter urban canyons and coverage gaps in urban environments
 - Electric meters cannot move to find reception

Star Networks

STAR NETWORKS (WI-FI, Cellular, LPWAN)



MESH NETWORKS

99%+ Reliability
Ubiquitous Coverage



Comparison between Mesh and Star

	Mesh	Cellular (Star)
No. of devices	Many in same location	Few (<10) in one location
Coverage	Cellular coverage not there	Coverage available
Device communication	Device talk more frequently to neighbors	Devices mostly talk to cloud
Communication Frequency	More frequent	Less frequent

Use Case: London Streetlights



WiSun Lighting the City of London <https://www.youtube.com/watch?v=3nQDSqx3S3w>

Use Case: London Streetlighting

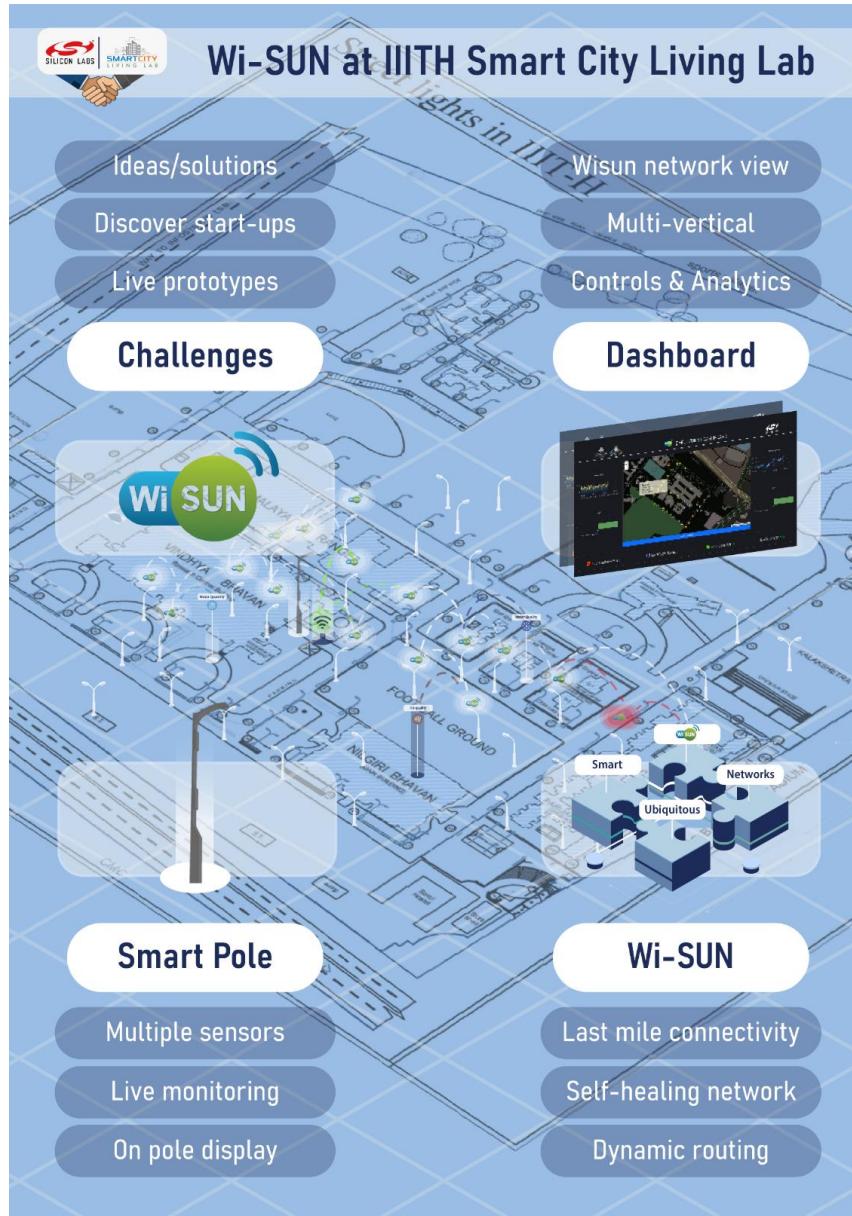
- Dynamic population in downtown
 - 9000 residents but 4,50,000 in peak times
- Narrow streets, alleys, tall buildings, and hidden areas
 - Unique and historical feel for London
 - Creates large urban canyons, where the tallest buildings sometimes block out the horizon
- LEDs and a Central Management System (CMS)
 - **Set the scene of its historic assets**
 - Use of the tunable settings of digital lighting that are not so easily achievable using analog lights
 - Reduction of maintenance and energy consumption costs
 - automated brightness adjustment to match lighting levels to environmental conditions, as well as to vehicle, bicycle, and pedestrian traffic.

Use Case: London Streetlighting

- Use of open standards so that any third-party devices can be integrated
 - traffic and parking monitoring, occupancy sensing, environmental monitoring, asset management, and lighting control
- Itron deployed 12,000 lights in two-years supported by 10 gateways
- UrbanControl's software-based security offerings to comply with the City's stringent requirements.

Use Case: IIITH Streetlighting

Wi-SUN Deployment in Collaboration With Silicon Labs



- ❖ RF Mesh – Enables Last Mile Connectivity for IOT Data
- ❖ Control through Dashboard
- ❖ Self Healing Network
- ❖ Interconnectivity through oneM2M

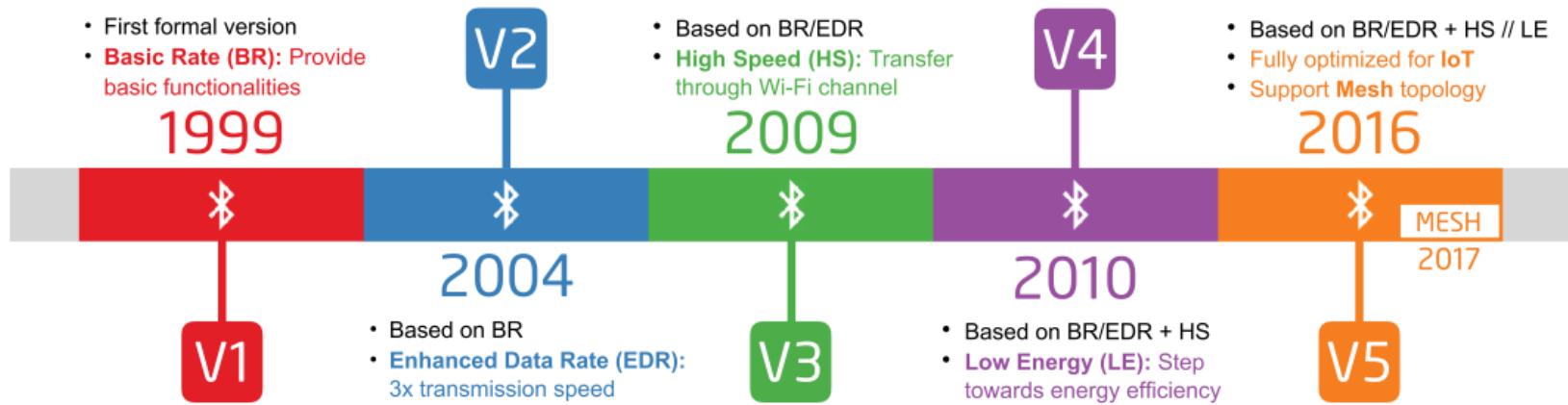
Questions?

Bluetooth

Good References

- [Lea2018] P. Lea, *Internet of Things for Architects*, Packt, 2018
- [Yin2019] J. Yin et. al, “A Survey on Bluetooth 5.0 and Mesh: New Milestones of IoT,” ACM Trans. on Sensor Networks 15(3):1-29, May 2019
- <https://www.bluetooth.com/>

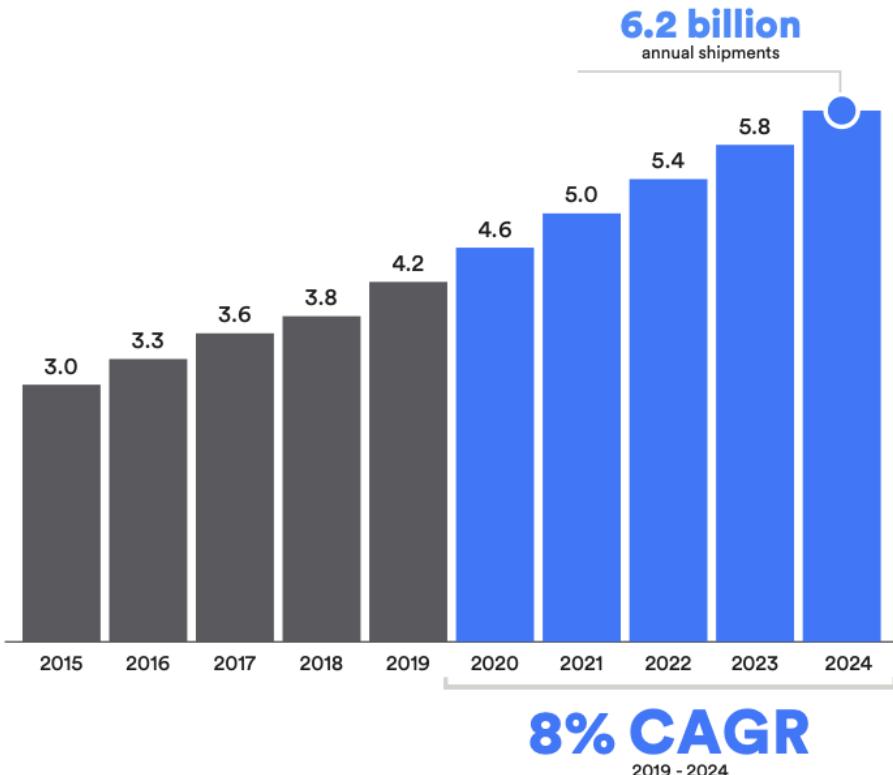
Bluetooth Versions



Bluetooth devices shipped every year

Total Annual Bluetooth® Device Shipments

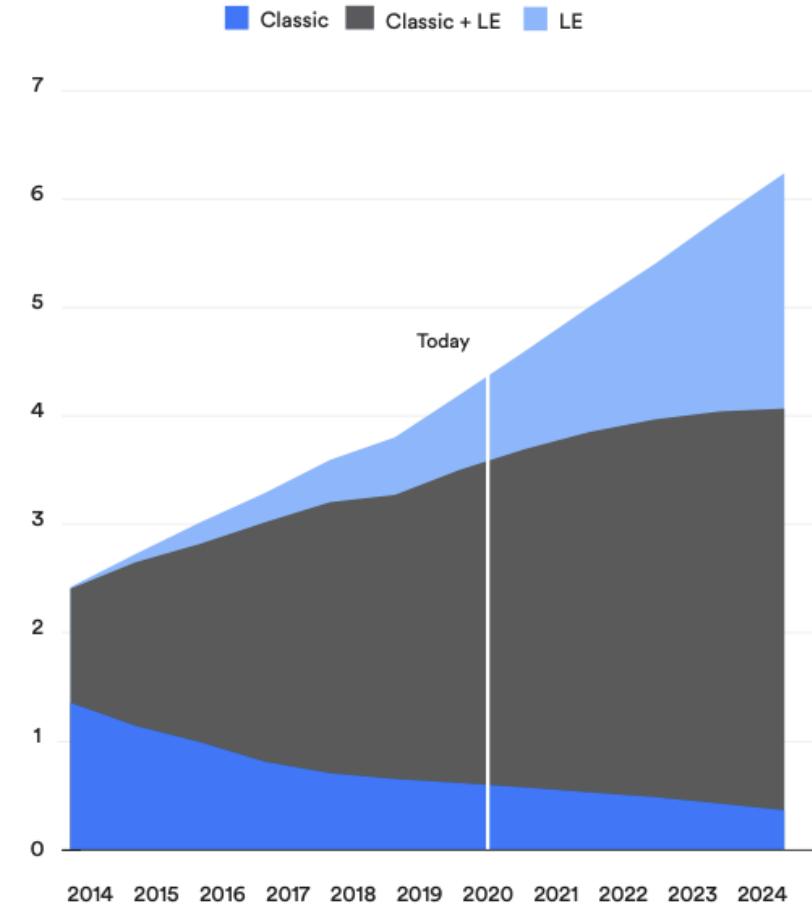
numbers in billions



Source: ABI Research, 2020

Annual Bluetooth® Device Shipments by Radio Version

numbers in billions



Source: ABI Research, 2020

Outline

- Bluetooth
- Bluetooth Low Energy (BLE)

Classic Bluetooth



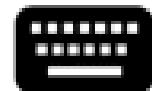
 **Bluetooth®**



Bluetooth Low Energy (BLE)



 **Bluetooth®**
SMART



Wireless devices streaming
rich content like data, video,
and audio

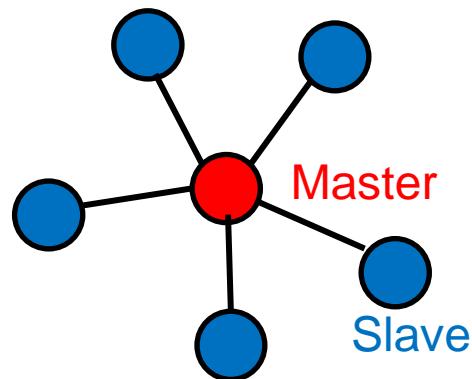
Sensor devices sending
small bits of data, using very
little energy

[Signils]

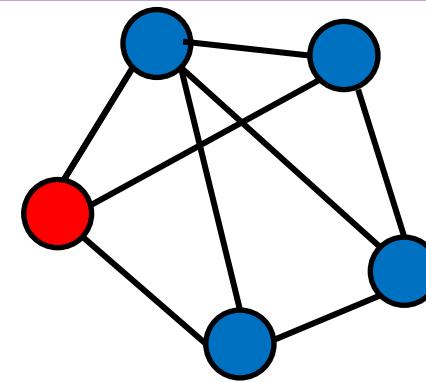
Introduction

- Bluetooth is a standard wire-replacement communications protocol primarily designed for low-power consumption, with a short range based on low-cost transceiver microchips in each device. [wiki]
- Applications
 - Stream audio in devices including headsets and mobile phones, home stereos, MP3 players
 - Transfer data (meeting schedules, phone numbers), audio, graphic images and video from one device to the other provided they are Bluetooth compliant
- Earlier IEEE 802.15.1 standard (Now maintained by Bluetooth SIG)
- Why not use WiFi?

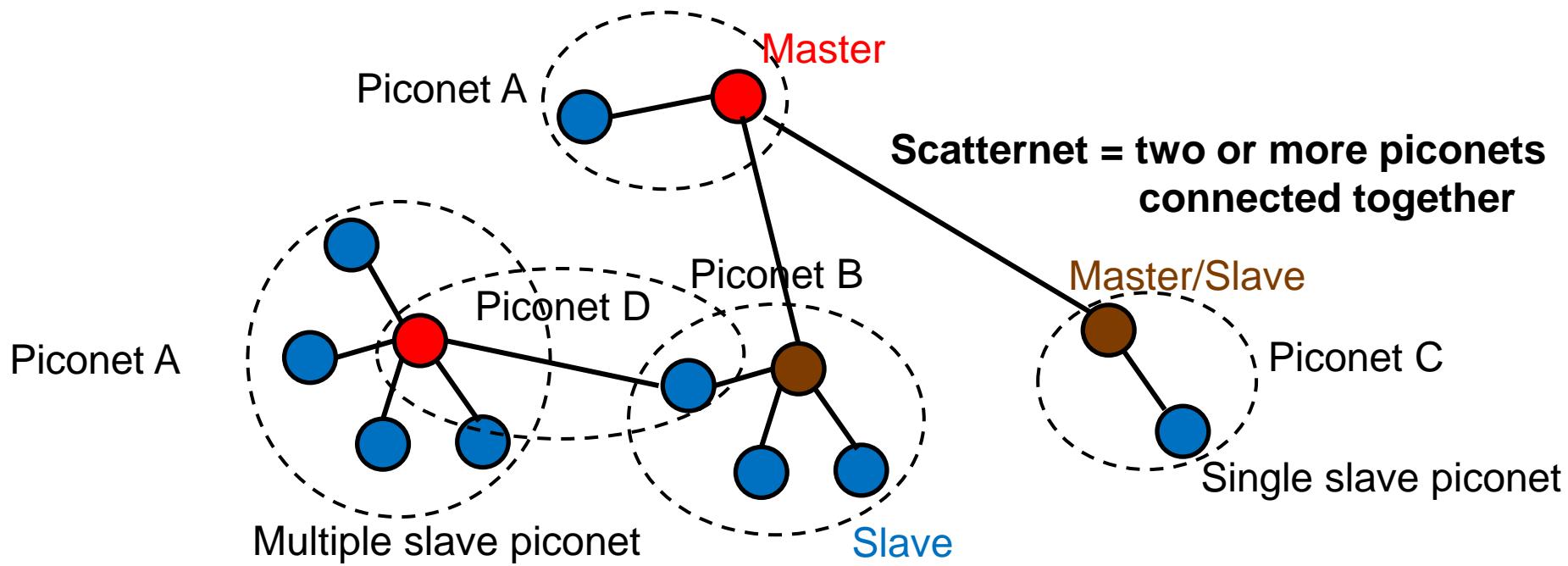
Network Topologies: Piconet and Scatternet



Piconet (Star)



Mesh Not Possible!



Physical Layer

Specification	Features Supported
RF Frequency	2.4 GHz
Transmit power	1 mW (min), 100 mW (max)
Data rate	1 Mbps
Distance	100 m (max)
RF bandwidth	220 KHz to 1MHz
Number of channels	23 (min) to 79 (max)
Topology	Up to 7 links in star configuration
Hopping rate	1600 hops per second
Access type	FH-TDD-TDMA

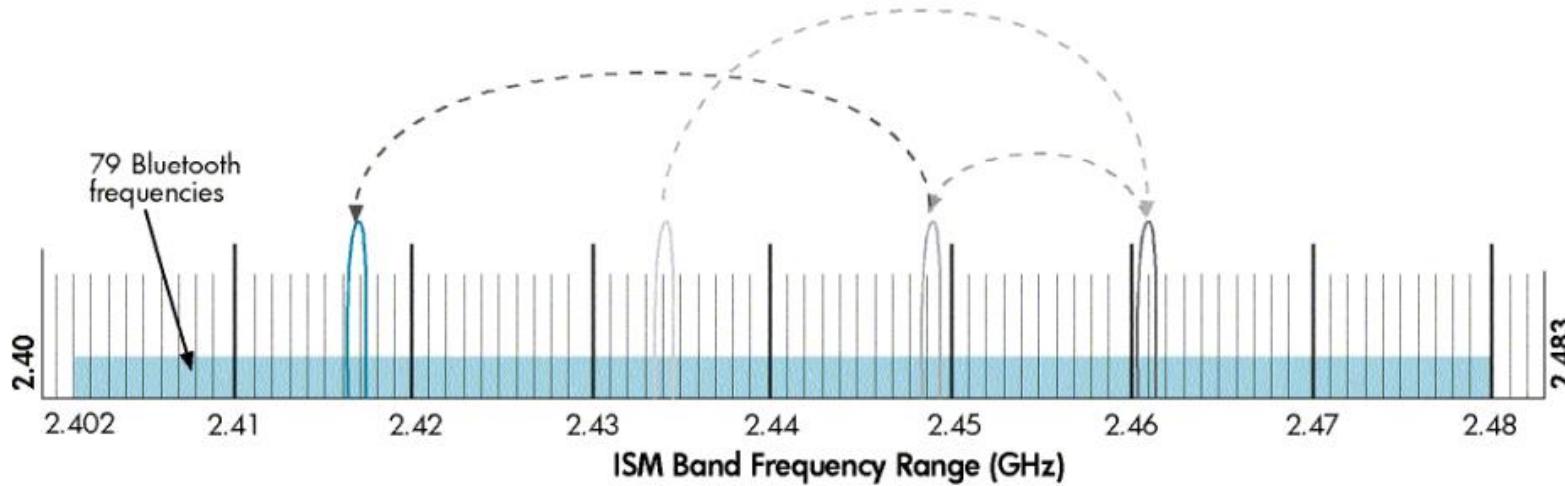
Frequency Allocation

- Bluetooth operates at 2.4 GHz ISM band.
- Following table defines Bluetooth frequencies used across the world.

Region	Frequency Range	RF Channels
In US, Europe, and rest of the world	2.4 to 2.4835 GHz	$f = 2.402 + n \text{ MHz}$ (n=0 to 78)
Japan	2.471 to 2.497 GHz	$f = 2.473 + n \text{ MHz}$ (n=0 to 22)
Spain	2.445 to 2.475 GHz	$f = 2.449 + n \text{ MHz}$ (n=0 to 22)
France	2.4465 to 2.4835 GHz	$f = 2.454 + n \text{ MHz}$ (n=0 to 22)

Frequency Hopping

- Bluetooth devices hop between frequencies up to 1600 times per second or every slot of 625 microsecs
- This is primarily to minimize eavesdropping and interference from other networks that use the 2.4 GHz ISM bands
- The transmitter and receiver exchange a data packet at one frequency, and then they hop to another frequency to exchange another packet. They repeat this process until all the data is transmitted



<http://h10032.www1.hp.com/ctg/Manual/c00186949.pdf>

Example of PN Sequence: 2, 5, 19, 31, 78, 43, 65, 7, 2, 5, 19,

Time Division Multiplexing

- Basic unit of operation is slot of 625 microsecs.
- In pre-connection stage (inquiry/page/scan), Tx and Rx can occur in half slots
- In connection state, Tx and Rx can occur in multiple slots: 1,3,5

Power Classes

Class number	Max Output Power (dBm)	Max Output Power (mW)	Max. Range
Class 1	20 dBm	100 mW	100 m
Class 2	4 dBm	2.5 mW	10 m
Class 3	0 dBm	1 mW	10 cm

Modulation Formats

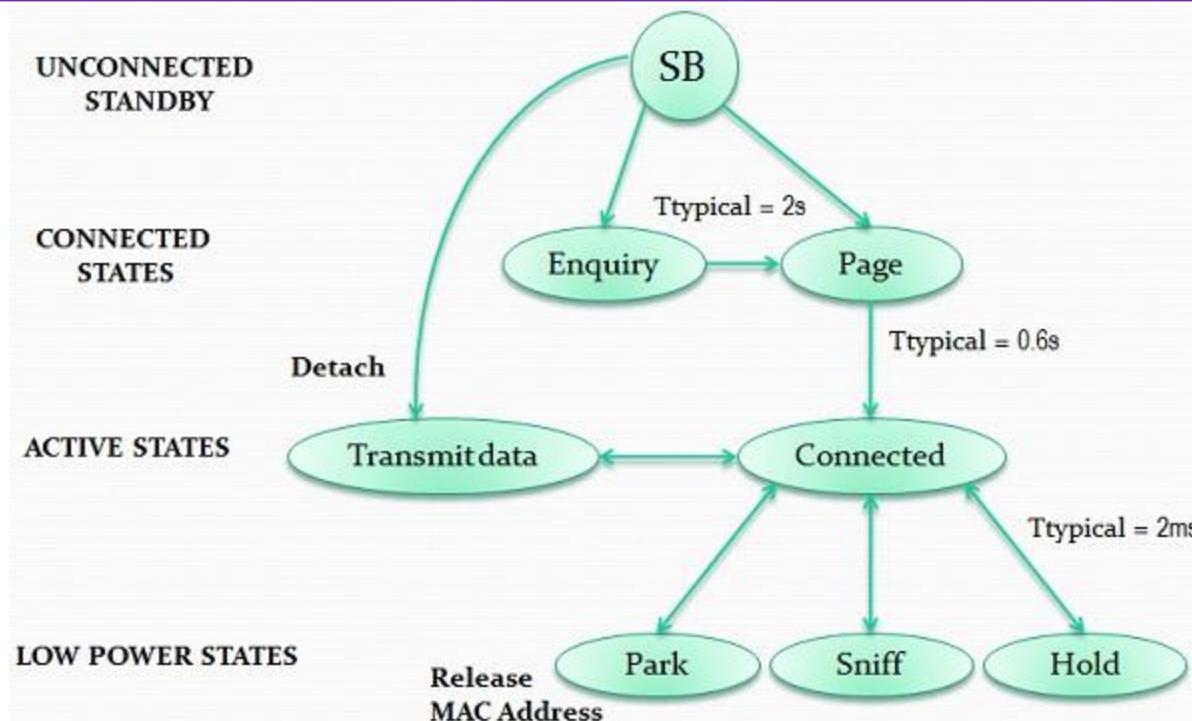
- The first version of the Bluetooth system used a **Gaussian Frequency Shift Keying (GFSK) modulation** types which was robust and reliable but not very efficient.
- Bluetooth systems from version 2.0 and above can use either **GFSK, pi/4 QPSK or D8PSK**.

Bluetooth Device Address



- 48 bit IEEE MAC address (BD_ADDR)
 - Lower address part (LAP) of 24 bits
 - Upper address part (UAP) of 8 bits
 - Non-significant address part (NAP) of 16 bits
- Globally unique!
- LAP and UAP of master are also used to determine frequency hopping sequence
- NAP is also used for encryption
- Bluetooth device may not have IP address!

States



- The connection establishment in Bluetooth is done in two phases- **Inquiry and page**
- Active devices are allocated a 3-bit **AMA (Active Member Address)**, Parked devices are assigned an 8-bit **PMA (Parked Member Address)**, Standby devices do not need an address.

States

- Inquiry:
 - If two Bluetooth devices know absolutely nothing about each other, one must run an inquiry to try to **discover** the other. One device sends out the inquiry request, and any device listening for such a request will respond with its address, and possibly its name and other information.
- Paging
 - Paging is the process of forming a connection between two Bluetooth devices. Before this connection can be initiated, each device needs to know the address of the other (found in the inquiry process).
- Connection
 - After a device has completed the paging process, it enters the connection state. While connected, a device can either be actively participating or it can be put into a low power sleep mode

States..

- Connection States
 - Active mode
 - Listens for packet from master
 - Processes all packets from master
 - Sniff Mode
 - Listens to piconet less frequently in this mode
 - Device will sleep and only listen for transmissions at a set interval (e.g. every 100ms).
 - Hold Mode
 - Device temporarily sleeps for a defined period and then returns back to active mode when that interval has passed.
 - The master can command a slave device to hold.
 - Park Mode
 - Park is the deepest of sleep modes. A master can command a slave to “park”, and that slave will become inactive until the master tells it to wake back up
 - Releases AM_ADDR

Bonding Pairing

- Bonded devices **automatically establish a connection** whenever they're close enough through two step process
- Pairing
 - Requires an authentication mechanism
 - Share information such as addresses, names, and profiles
 - Create temporary common security encryption key
- Bonding
 - information is stored in non-volatile memory
 - Creation and storage of permanent security keys

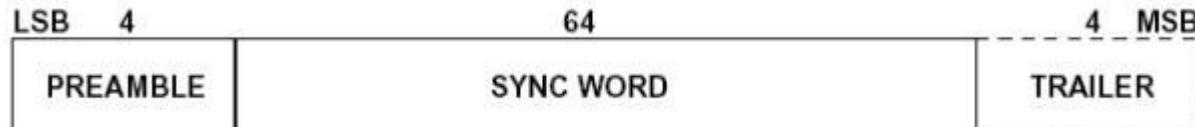
Packet Format



- Packet can contain
 - Access code
 - Access code and Header
 - Access code + Header + Payload

Packet Format

- Access code



- Based on master's identity and master's system clock
- Used for synchronization and identification
- All packets in a piconet use the same access code
- Packet Header



- Used for error correction, retransmission, and flow control information
- AM_ADDR
 - 3-bit slave address
 - Temporary assigned while the slave is active and specific to piconet
 - Same in both directions
 - All zeros for broadcast

Bluetooth Low Energy (BLE)

Bluetooth Versions

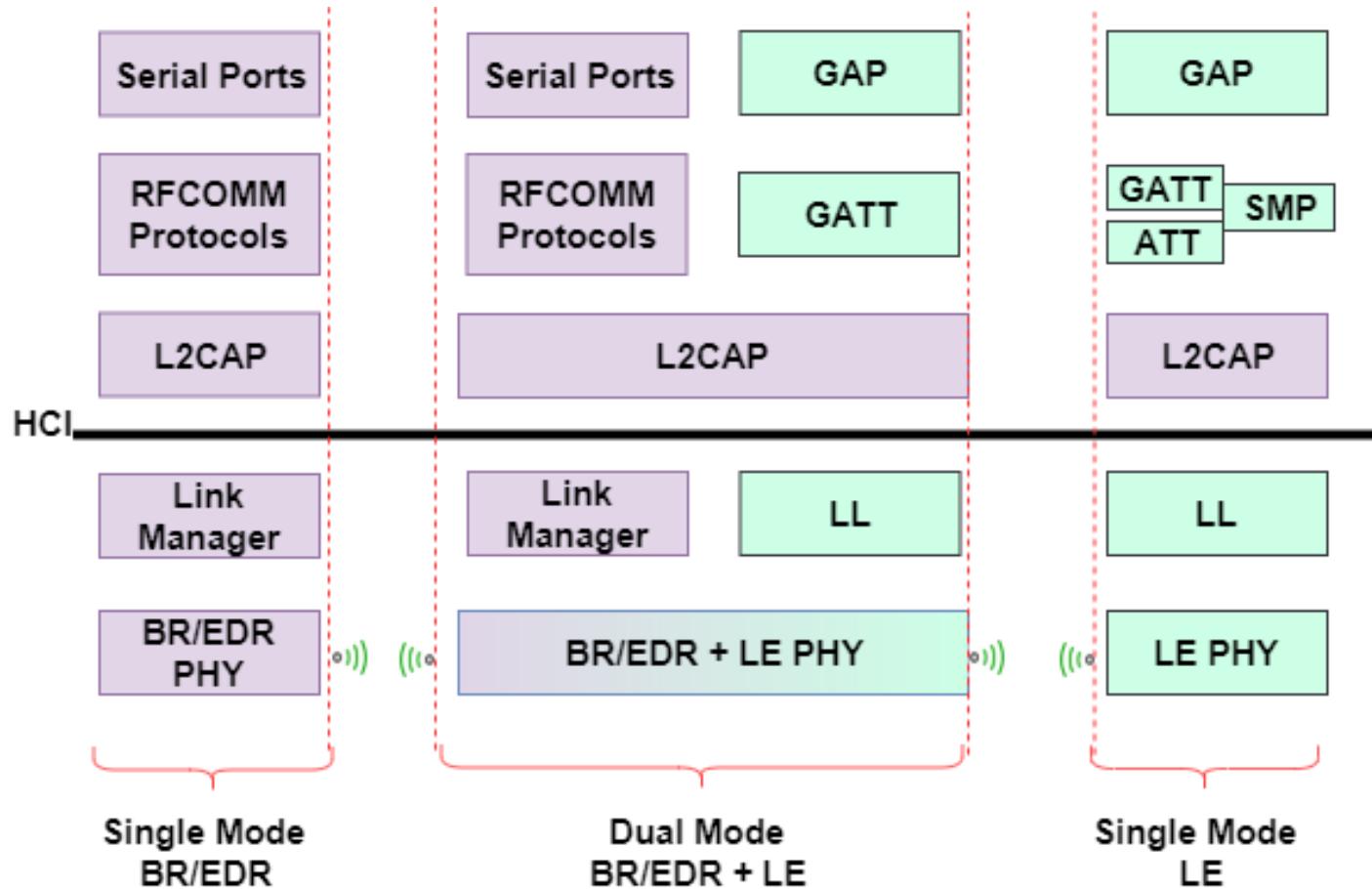
- Maintained by Bluetooth SIG
- **Bluetooth v1.2**
 - The v1.x releases laid the groundwork for future versions
 - Supported data rates of up to 1 Mbps (**Basic Rate (BR)**) using GFSK
 - 10 meter maximum range.
- **Bluetooth v2.1 + Enhanced Data Rate (EDR)**
 - The 2.x versions of Bluetooth introduced **EDR**
 - Data rates up to 3 Mbps using pi/4 QPSK and 8 DPSK
 - **Secure simple pairing (SSP)**
- **Bluetooth v3.0 + High Speed (HS)**
 - 24 Mbps speed with WiFi offloading in “+HS” versions
- **Bluetooth v4.0, v4.1, v4.2**
- **Bluetooth v5.0 (Dec. 2016)**
 - 2 * data rate, 4 * range of BLE, 8 * broadcasting capacity
 - Backward compatible with classic as well as BLE
 - V5.1 in Jan. 2019 and V5.2 in Dec. 2019 (contains **LE Audio**), v5.3 (July 2021), v5.4 (Feb. 2023)

Bluetooth 4.0 and BLE

- **Bluetooth v4.0 and Bluetooth Low Energy**
 - Bluetooth 4.0 split the Bluetooth specification into three categories: **classic**, **high-speed**, and **low-energy**. Classic and high speed are backward compatible to Bluetooth versions v2.1+EDR and v3.0+HS respectively.
 - BLE is **Bluetooth Smart**
 - BLE + v4.0 classic is Bluetooth Smart Ready (can connect to both classic as well BLE)
- **Bluetooth low energy (BLE)**
 - BLE is a massive overhaul of the Bluetooth specifications, aimed at very low power applications. It sacrifices range (50m instead of 100m) and data throughput (0.27 Mbps instead of 0.7-2.1 Mbps) for a significant savings in power consumption.
 - Throughput is different than over the air data rate
 - Earlier versions could not carry voice. Only suited for intermittent data transmissions (no continuous data transmission)
 - V5 has LE Audio
 - Not backward compatible with Bluetooth classic
- **Bluetooth Mesh (2017)**

BLE protocol stack

Not in Syllabus for Exam



Applications

- Smart home automation
 - Lighting
 - Thermostats
 - Door locks
 - Refrigerators
 - Smoke detectors
- Wearables
 - Sports and fitness bands
 - Smart watches
- Medical devices
 - Glucometer, Heart Rate Monitor
- Peripherals
 - Keyboards
 - Mouse
- Many more

Bluetooth Low Energy (BLE)



Sensor devices sending
small bits of data, using very
little energy

Applications



Smart bulbs



Wireless Keyboards



Smart Wearables



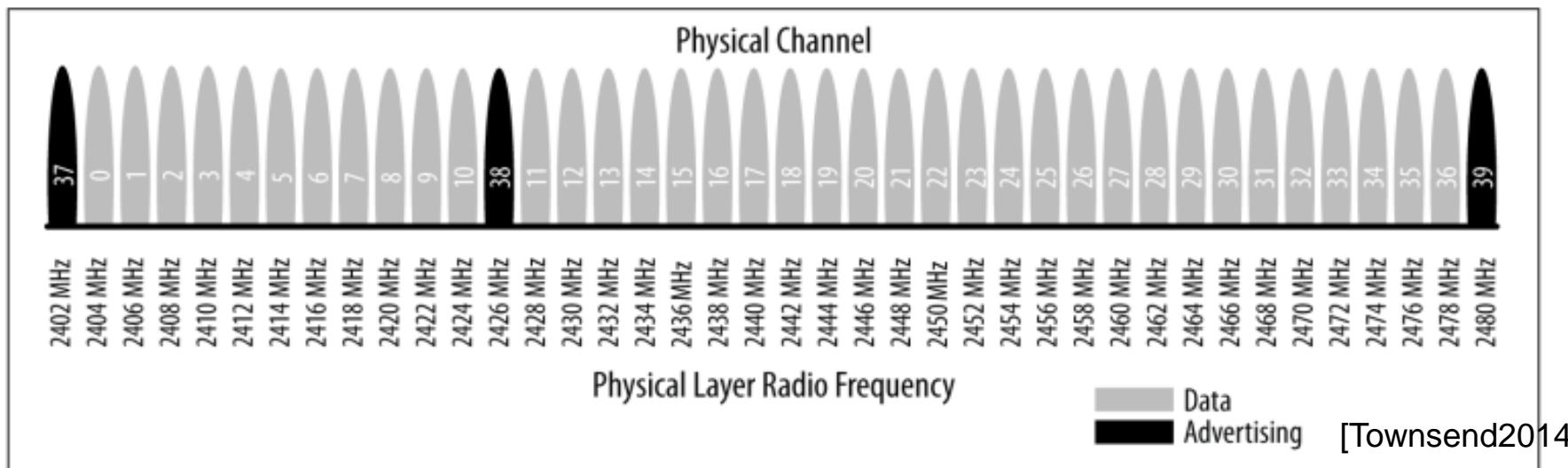
Smart Doorlocks



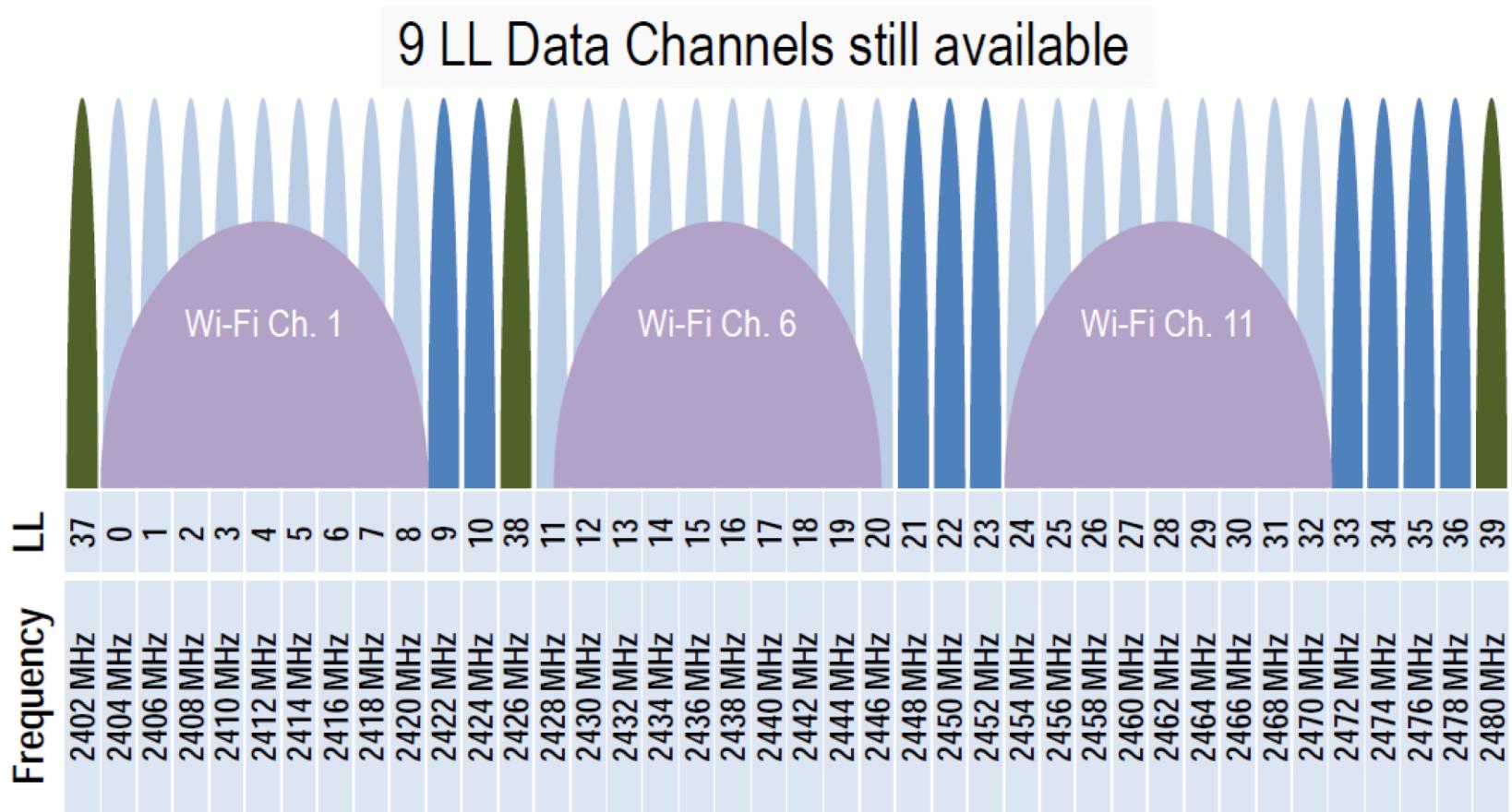
Batteryless
BLE Switches

PHY Channels

- BLE operates in 2.4 GHz ISM band
 - In US, Europe, and rest of the world: 2.4 to 2.4835 GHz
 - Japan: 2.471 to 2.497 GHz
 - Spain: 2.445 to 2.475 GHz
 - France: 2.4465 to 2.4835 GHz
- BLE uses 40 channels of 2 MHz each
 - Channels are used for connection data and the last three channels (37-39) are used as advertising channels to set up connections and send broadcast data
 - Adaptive frequency hopping to avoid interference

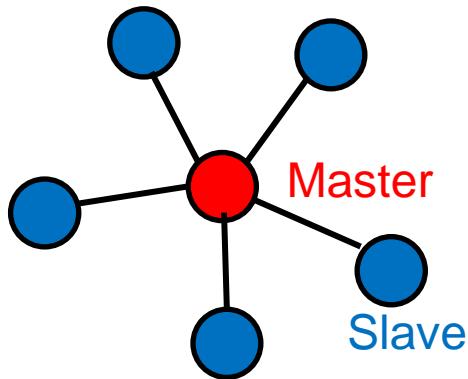


Advertising channels avoid popular WiFi Channels

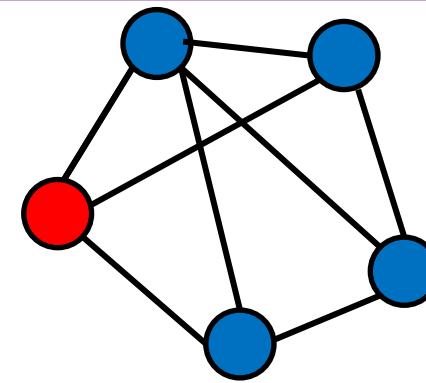


In Bluetooth 5.0, the other 37 channels can also be used as secondary advertising!

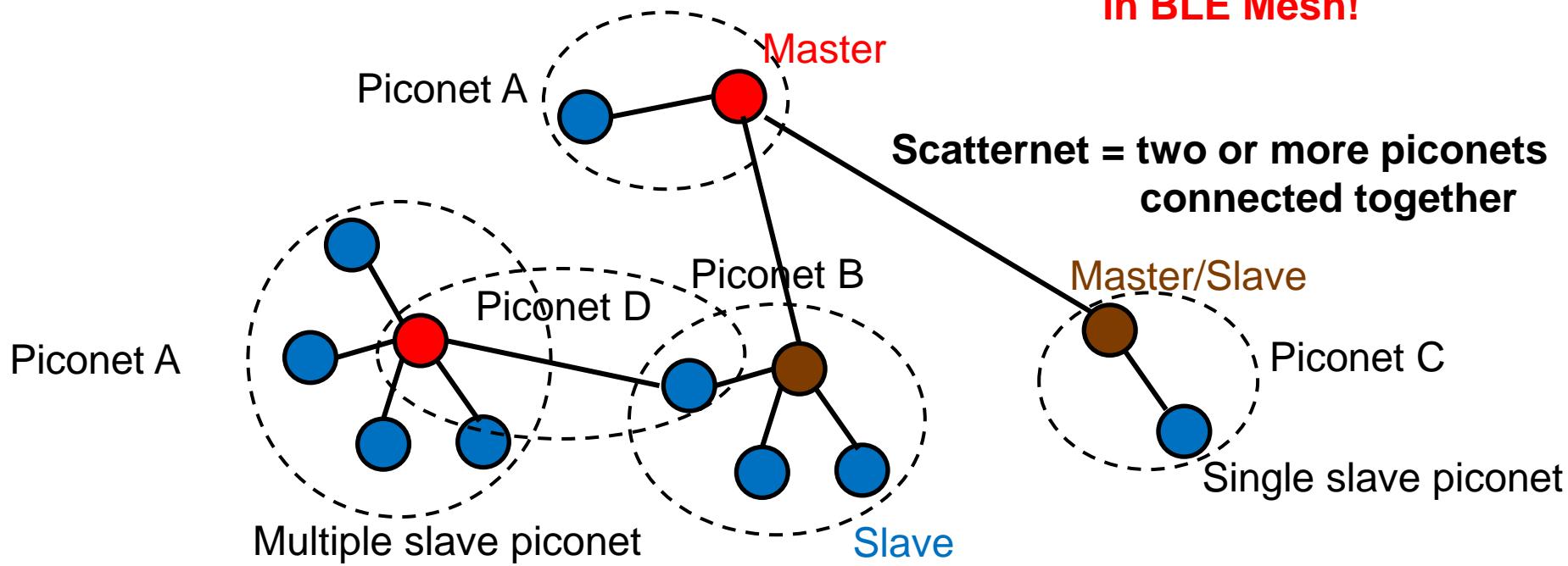
Network Topologies: Piconet and Scatternet



Piconet (Star)



Mesh now possible
in BLE Mesh!

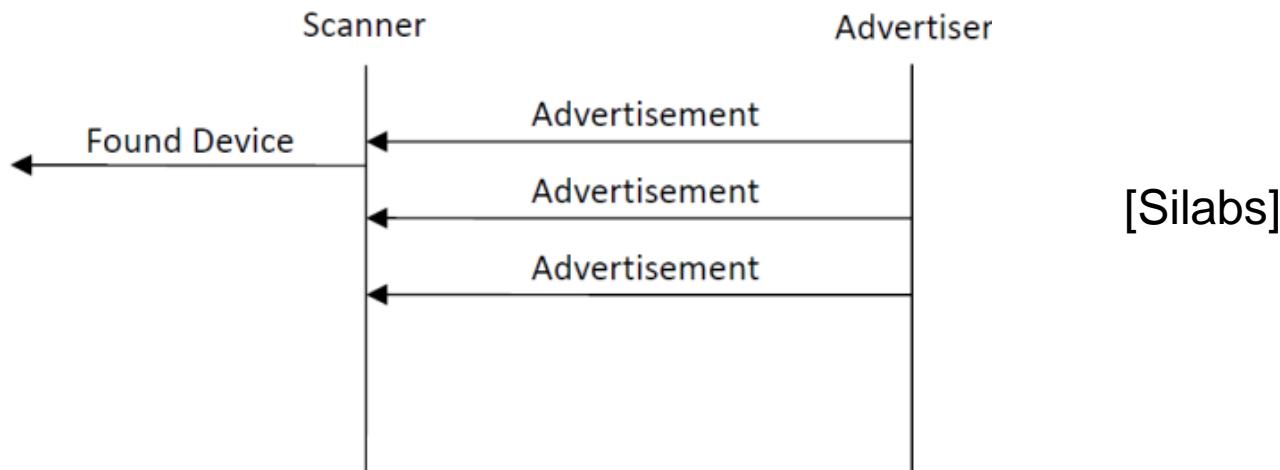


Link Layer Operations

- Advertising
 - Scanning
 - Connection
-
- BLE has only one packet format and two types of packets (advertising and data packets), which simplifies the protocol stack implementation immensely.

Advertising

- Purpose
 - Enables devices to broadcast their presence
 - Broadcast data for applications that do not need the overhead of a full connection establishment
 - Discover devices and to connect to them
- Advertising device broadcasts packets on one or multiple advertising channels which remote devices can pick

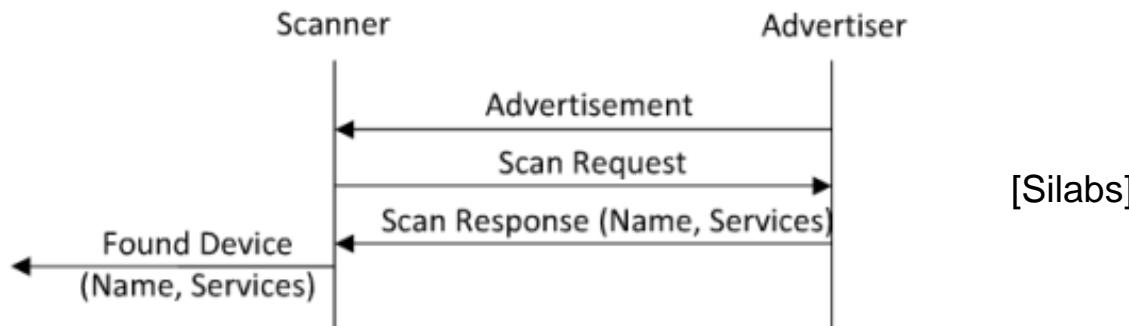


Scanning

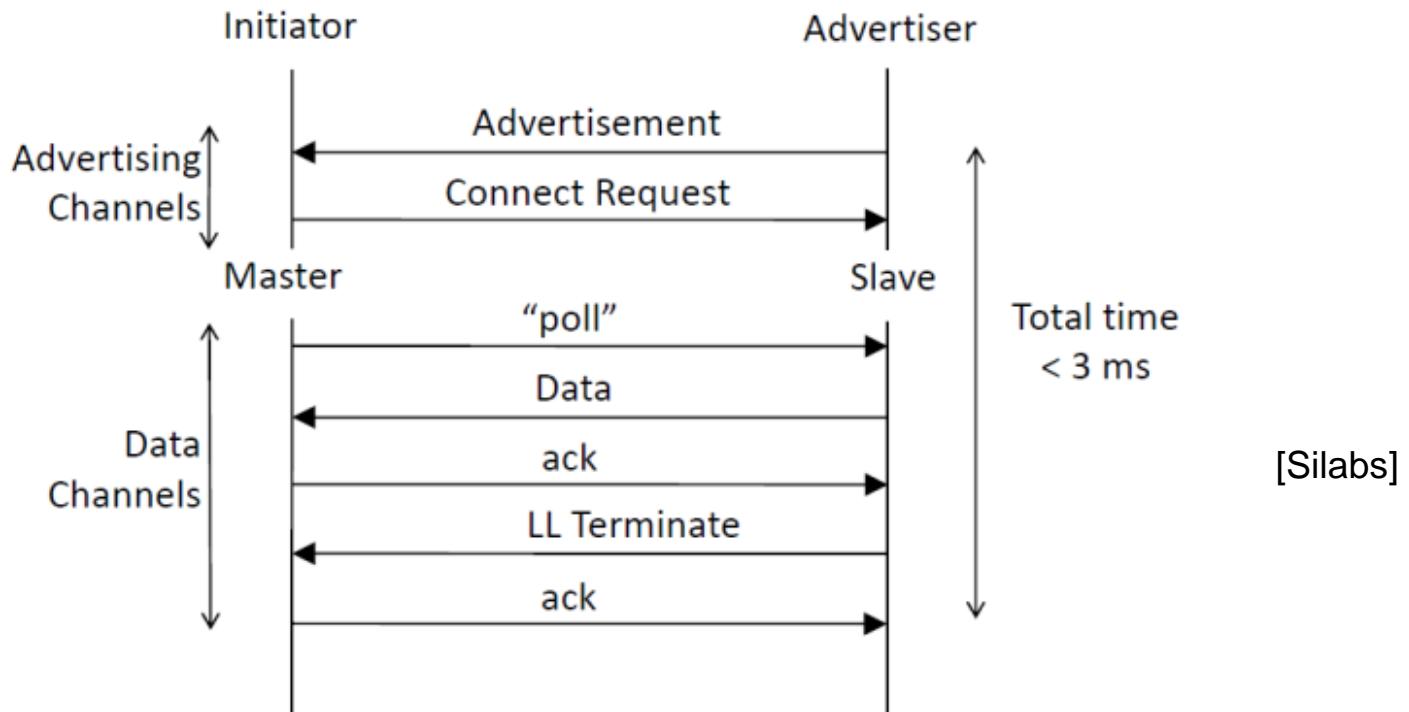
- Passive scanning
 - Device simply listens to advertisement
 - Scans through the advertisement channels in round robin fashion



- Active scanning
 - In addition to basic scanning, requests for more information

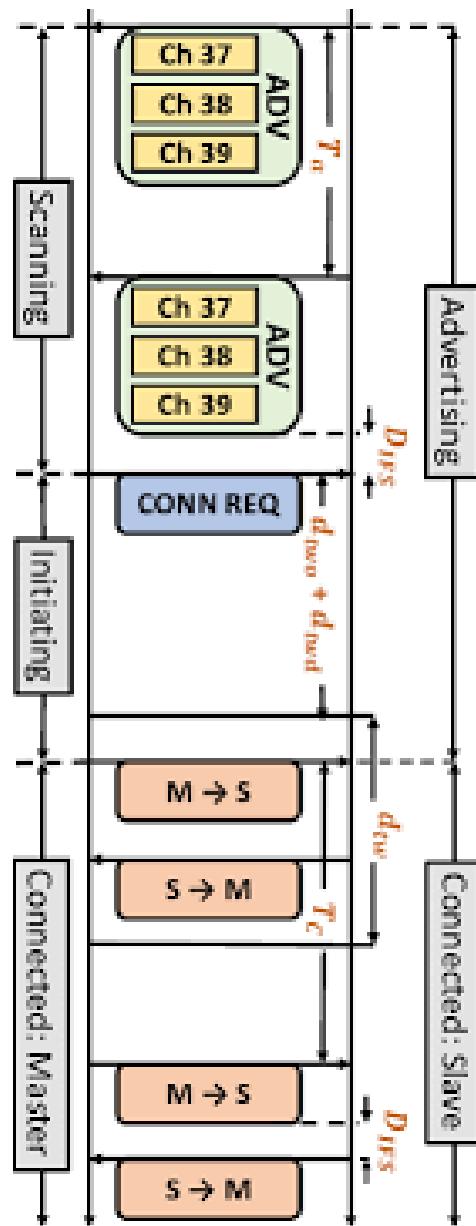


Connection Establishment

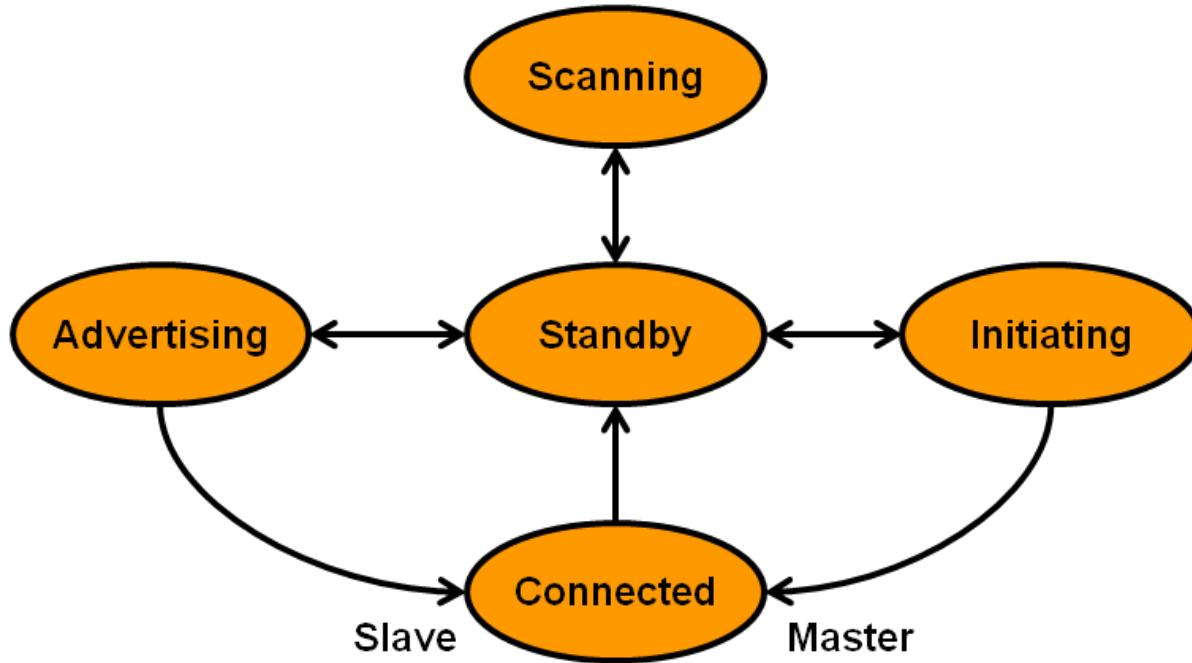


- Reliable data transfer: use of CRC, acknowledgments and retransmissions
- Adaptive frequency hopping

Connection Establishment



Link Layer States



Bluetooth 5 enhancements

- New physical layers
 - LE Coded PHY
 - Longer range transmissions with no increase in power
 - LE 2M PHY
 - Higher data rates with 2M symbols per second
- Extended advertising
- Improved frequency hopping
 - New channel selection algorithm 2 (CSA2)
- Slot availability mask (SAM)
 - For BR/EDR

Not in Syllabus for Exam

BLE 5: New PHY

Not in Syllabus for Exam

	LE 1M	LE Coded S=2	LE Coded S=8	LE 2M
Symbol Rate	1 Ms/s	1 Ms/s	1 Ms/s	2 Ms/s
Protocol Data Rate	1 Mbit/s	500 Kbit/s	125 Kbit/s	2 Mbit/s
Approximate Max. Application Data Rate	800 kbps	400 kbps	100 kbps	1400 kbps
Error Detection	CRC	CRC	CRC	CRC
Error Correction	NONE	FEC	FEC	NONE
Range Multiplier (approx.)	1	2	4	0.8
Requirement	Mandatory	Optional	Optional	Optional

Comparison between classic and BLE

Bluetooth® Classic

Solution Areas



AUDIO STREAMING DATA TRANSFER

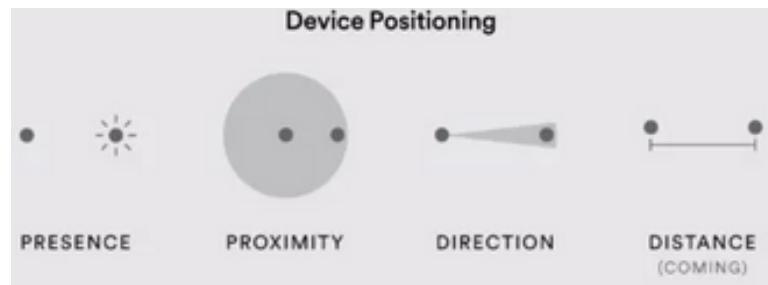
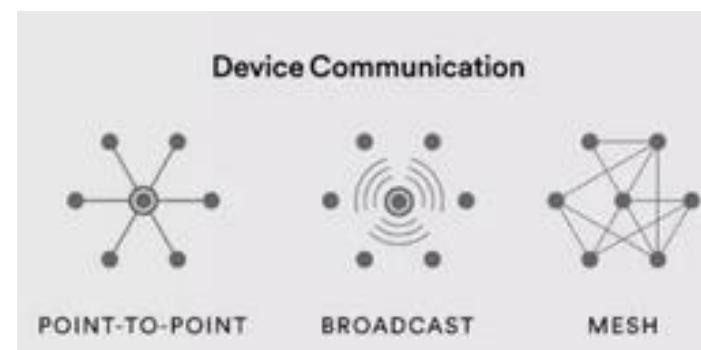


Bluetooth® Low Energy

Solution Areas



AUDIO STREAMING DATA TRANSFER LOCATION SERVICES DEVICE NETWORKS



	Bluetooth Low Energy (LE)	Bluetooth Classic
Frequency Band	2.4GHz ISM Band (2.402 – 2.480 GHz Utilized)	2.4GHz ISM Band (2.402 – 2.480 GHz Utilized)
Channels	40 channels with 2 MHz spacing (3 advertising channels/37 data channels)	79 channels with 1 MHz spacing
Channel Usage	Frequency-Hopping Spread Spectrum (FHSS)	Frequency-Hopping Spread Spectrum (FHSS)
Modulation	GFSK	GFSK, $\pi/4$ DQPSK, 8DPSK
Data Rate	LE 2M PHY: 2 Mb/s LE 1M PHY: 1 Mb/s LE Coded PHY (S=2): 500 Kb/s LE Coded PHY (S=8): 125 Kb/s	EDR PHY (8DPSK): 3 Mb/s EDR PHY ($\pi/4$ DQPSK): 2 Mb/s BR PHY (GFSK): 1 Mb/s
Tx Power*	≤ 100 mW (+20 dBm)	≤ 100 mW (+20 dBm)
Rx Sensitivity	LE 2M PHY: ≤ -70 dBm LE 1M PHY: ≤ -70 dBm LE Coded PHY (S=2): ≤ -75 dBm LE Coded PHY (S=8): ≤ -82 dBm	≤ -70 dBm
Data Transports	Asynchronous Connection-oriented Isochronous Connection-oriented Asynchronous Connectionless Synchronous Connectionless Isochronous Connectionless	Asynchronous Connection-oriented Synchronous Connection-oriented
Communication Topologies	Point-to-Point (including piconet) Broadcast Mesh	Point-to-Point (including piconet)
Positioning Features	Presence (Advertising) Proximity (RSSI) Direction (AoA/AoD) Distance (Coming)	None Not in Syllabus for Exam

BLE Beacons

- Advertise some information on a periodic basis
 - Channels 37, 38, 39
- Only one way transmissions
 - Cannot connect or pair with other devices
 - Connection will stop advertising and no other device will be able to listen to them
 - Beacons cannot track user
 - Only the installed app can
- Beacons can transmit calibrated RSSI at one meter
- Period can be in the order of 100ms

Three use cases for beaconing

- Static point of interest
 - Distributing messages at the point of interest
 - Retail, Museum
 - Mobile payment through PoS
- Broadcasting telemetry data
 - In Eddystone, telemetry data regarding the beacon like battery level, time since power-on, advertisement count
- Indoor localization and geolocation services
 - Indoor positioning system
 - Key finder
 - Asset tracking and logistics

Key IoT features of BLE

Advantages

- Low power: years on button battery
- Small size and low cost (License free)
- Ubiquitous
- Connectivity to mobile phones
- Security: AES 128 bit

Issues

- Low range
- Low data rates: No voice or video capability
 - Now possible on low quality: <https://www.youtube.com/watch?v=eqAFVi16I5E>
- Roaming
- Security
 - Wearables, Smart locks: <https://www.tomsguide.com/us/bluetooth-lock-hacks-defcon2016.news-23129.html>

BLE Use Case: COVID-19 management



Reducing the Risk of COVID-19 Transmission Using Philips' Biosensor BX100 and Cassia's Bluetooth Gateways

- <https://www.cassianetworks.com/wp-content/uploads/2020/09/Philips-and-Cassia-Networks-Case-Study.pdf>
- Deployed in a major Dutch hospital
- 40 BLE paired simultaneously
- 300 meters coverage in open space for the AP
- Continuous patient monitoring

BLE Use Case: Avia Smart Lock

- Multipoint locks
- Home kit integration
- Uses BL654 BLE module
- Bluetooth 5
- AES-256 encryption with Diffie-Hellman pairing
- ARM Cryptocell™-310 cryptographic accelerator
 - several high-level cryptographic functions that are key to strong wireless security, such as true random number generation, hashing functions, and public key cryptography.



<https://www.lairdconnect.com/resources/case-studies/mightons-avia-smartlock-provides-unparalleled-security-laird-connectivitys-family-bluetooth-5>

BLE Use Case: Smart Hospital Wayfinding



- Digital wayfinding at an NYC hospital complex
 - <https://www.pointr.tech/blog/smart-hospital-wayfinding-cisco-dna-spaces>
- Use of blue-dots for accurate position (<1m) in a big complex of multi-floor building

Good References

[Lea2018] P. Lea, *Internet of Things for Architects*, Packt, 2018

[Yin2019] J. Yin et. al, “A Survey on Bluetooth 5.0 and Mesh: New Milestones of IoT,” *ACM Trans. on Sensor Networks* 15(3):1-29, May 2019

[Bluetooth5Pt1] Bluetooth Core Specification ver5.1, Bluetooth SIG, Jan. 2019

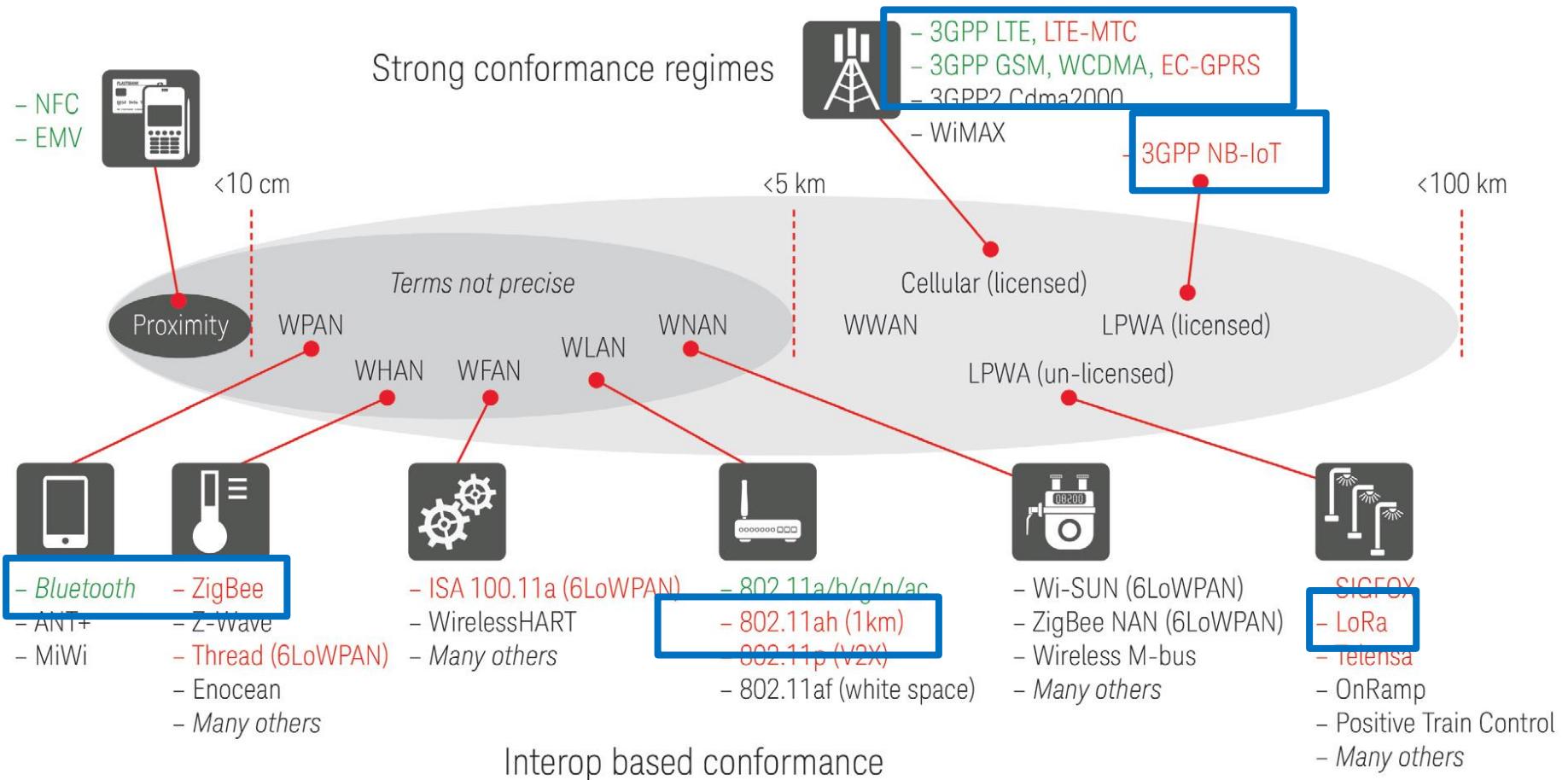
[Bluetooth5] M. Wooley, “Bluetooth Core Specification Version 5.0 Feature Enhancements”, *Bluetooth Resources*, Sept. 2021

[Signils] Online, <https://www.signils.com/whats-the-difference-between-classic-bluetooth-and-bluetooth-low-energy/> Accessed: 30 Dec 2021

Questions?

Comparison of Different Communication Techniques for IoT

Communication Techniques for IoT



■ : > Billion units/year now
■ : Emerging

WPAN: Wireless Personal Area Network

WHAN: Wireless Home Area Network

WFAN: Wireless Field (or Factory) Area Network

WLAN: Wireless Local Area Network

WNAN: Wireless Neighborhood Area Network

WWAN: Wireless Wide Area Network

LPWA: Low Power Wide Area

Issues in IoT from Communication Perspective

[Not an exhaustive list!]

- Low power consumption
- Support a large number of devices with low data rates
 - Overhead
 - Network setup, management and maintenance
- Coverage
- Quality of service
- Low cost
 - Network/Private (DIY)
 - Licensed/Unlicensed
- Privacy and security
- Standardization for interoperability between different vendors

Key IoT Features (802.11ah)

- Highest data rates
 - Can handle diverse range of applications including camera
- Longer range (1 Km)
- Scalable to thousands of nodes
- WiFi family widely used

Issues

- Most of the world is using 2.4 GHz
- Very few 802.11ah products available
 - Mostly using 802.11b/g/n
- Security (*Popularity*)
- High power consumption
- Roaming

Key IoT Features (Cellular)

Advantages

- **Quality of service**
 - Licensed band, Low latency
- **Ubiquitous**
- **Present on Phone, Convergence**
- **Security: SIM card protection + AES 256 bit (best)**
- Great coverage: 2 km (LTE), 10 km (NB-IoT), 35 km (GSM)
- **Global mobility and roaming support**
- Scalable
- Connected even during power failure

Issues

- Cost: License, Capex and Opex, subscription
- High power
- Not possible to make your own network

Key IoT Features (LoRaWAN)

- Designed for majority of IoT applications
- **Low powered sensors (Battery life of 2-5 years)**
 - Class A and B
- **Wide coverage area up to 15 Kms**
- **Low Costs**
 - free(unlicensed) frequencies
- One gateway can support thousands of end devices
- **Simple Architecture**
- Security: a layer of security for the network and one for the application with AES encryption.
- **Localization without GPS**
- **Roaming**
- LoRa Alliance: 500+ members companies including IBM and Cisco

IoT Features: *Disadvantages (LoRaWAN)*

- **Payload limited to 100 bytes**
- High latency (actuators are not possible)
- Low data rates
 - Does not support voice or video
- **Low duty cycles (1% in EU)**
- Interference issues
 - Unlicensed frequency for other technology users
 - Crowding of LoRaWAN gateways increase interference
 - High packet error rate
- Cost in terms of cloud-based servers for network and applications
 - Things Network, LoRIoT
- Needs fair amount of development work
 - DIY; Not a complete protocol stack
- Not for continuous or real-time monitoring and actuations (most of industry cases)
 - High latency (actuators are not possible), High packet error rate, Low data rates, Low duty cycles

Key IoT Features (BLE)

Advantages

- **Low power: years on button battery**
- **Small size and low cost (License free)**
- **Ubiquitous**
- **Connectivity to mobile phones**
- Decent Security (AES 128 bit)

Issues

- Low range
- Low data rates: No voice or video capability
 - Now possible on low quality: <https://www.youtube.com/watch?v=eqAFVi16l5E>
- Roaming
- Security
 - Wearables, Smart locks: <https://www.tomsguide.com/us/bluetooth-lock-hacks-defcon2016.news-23129.html>

Key IoT Features (IEEE 802.15.4)

Advantages

- **Low power**
- Large coverage of 1 Km in Sub-GHz band
 - Even more for boosted modules (3.2 km for Xbee)
- Easy to install and maintain (mesh, self-healing, self-organization)
- **Reliable (mesh, multiple channels, demonstrated interference tolerance, automated retransmissions)**
- Supports thousands of nodes
- **Low cost (many suppliers)**
- Long battery life (years on AA battery)
- Decent Security (AES 128 bit)

Issues

- No mobility support, Scalability
- Less coverage area in 2.4 GHz band
- Not part of mobile phones

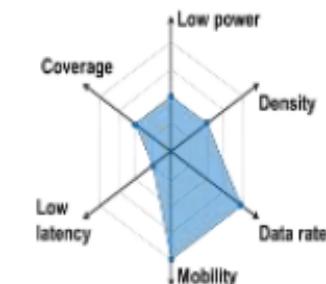
WiSUN Features

- Both star and mesh topologies are possible
 - Powerful use of mesh topology
 - One cluster of 5000 sensor nodes in 1 Km range
- Unlicensed band operation
- Leverages IPv6
- Can be designed for frequent communications
 - Every 10 secs
- Low latency of about 0.02-1 secs
- Data rates of upto 300 Kbps
- Coverage of 4 Km point-to-point using 1W non-directional antenna
- Security at multiple levels
 - Native public-key infrastructure (PKI) integration providing security certification capabilities for each device
- Networks designed for long lifecycle
 - Low power design (15-20 year battery target)
 - Backward and forward compatibility with different generations

	IEEE 802.15.4	IEEE 802.11ah	BLE	NB-IoT	LoRaWAN
Bands	868, 915 MHz 2.4 GHz (ISM)	900 MHz (ISM)	2.4 GHz (ISM)	Between 400- 2200 MHz (Licensed Frequency)	868/915/433 MHz (ISM)
Topology	Star, Mesh, Cluster Tree	Star, Mesh, Cluster-Tree	Piconet, Scatternet, Mesh	Star	Star of Stars
Max. Range	1 Km	1 Km	100 m	10 Km in one cell	10 Km in one cell
Max.Power	100 mW	100 mW	100 mW	200 mW	25 mW
DIY	Yes	Yes	Yes	No	Yes
Max. No. of Nodes per cell	65536	8192	Unlimited	100K	50 K
Modulation	BPSK, O-QPSK + DSSS	OFDM	GFSK	QPSK	CSS
Channel Access	Slotted/Unslotte d CSMA/CA + GTS	CSMA/CA + GTS + RAW	FH-TDD-TDMA Polling	HD-FDD TDMA+FDMA	Aloha
Data rates	20, 40 and 250 Kbps	150 Kbps- 347 Mbps	0.2-2 Mbps	20-250 Kbps	0.3-50 kbps
Power Saving	Sleep-Wake Schedule	Sub-1GHz TWT, Sleep-Wake, BDTO, Short MAC	Sniff, Hold, Park	PSM, low bandwidth, low duty cycle, eDRX	Target Application

One size does not fit all!

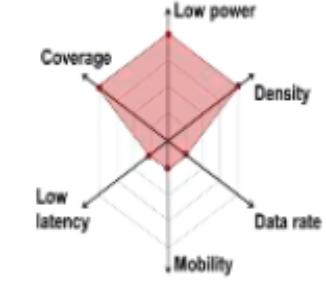
Requirement



~ Few Mbps

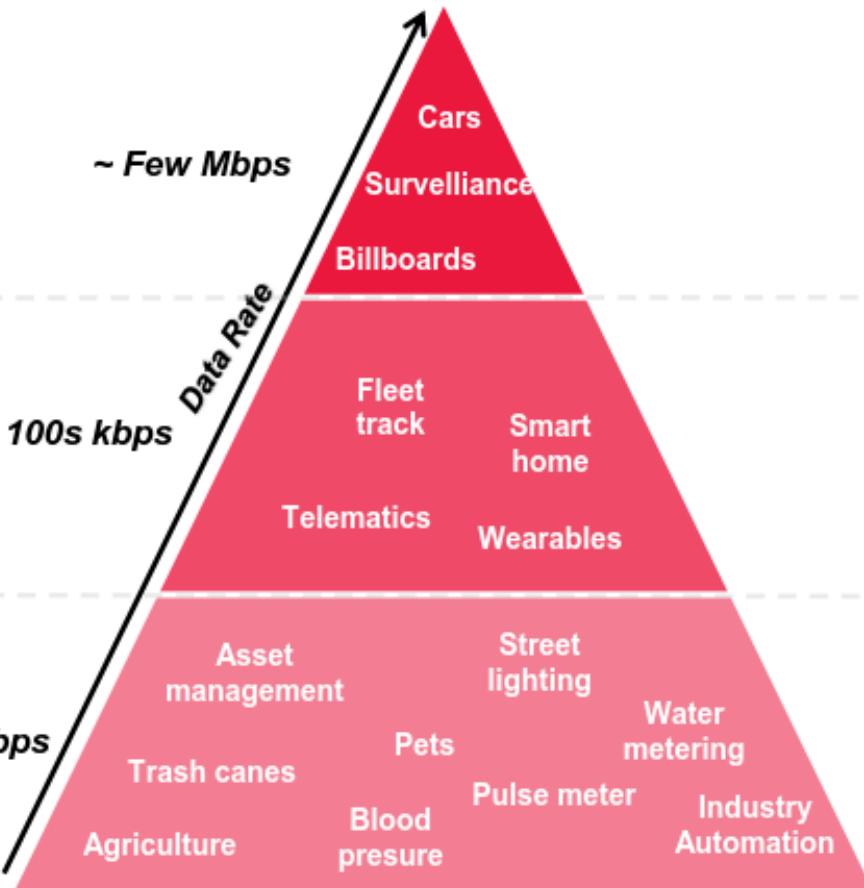


~ 100s kbps



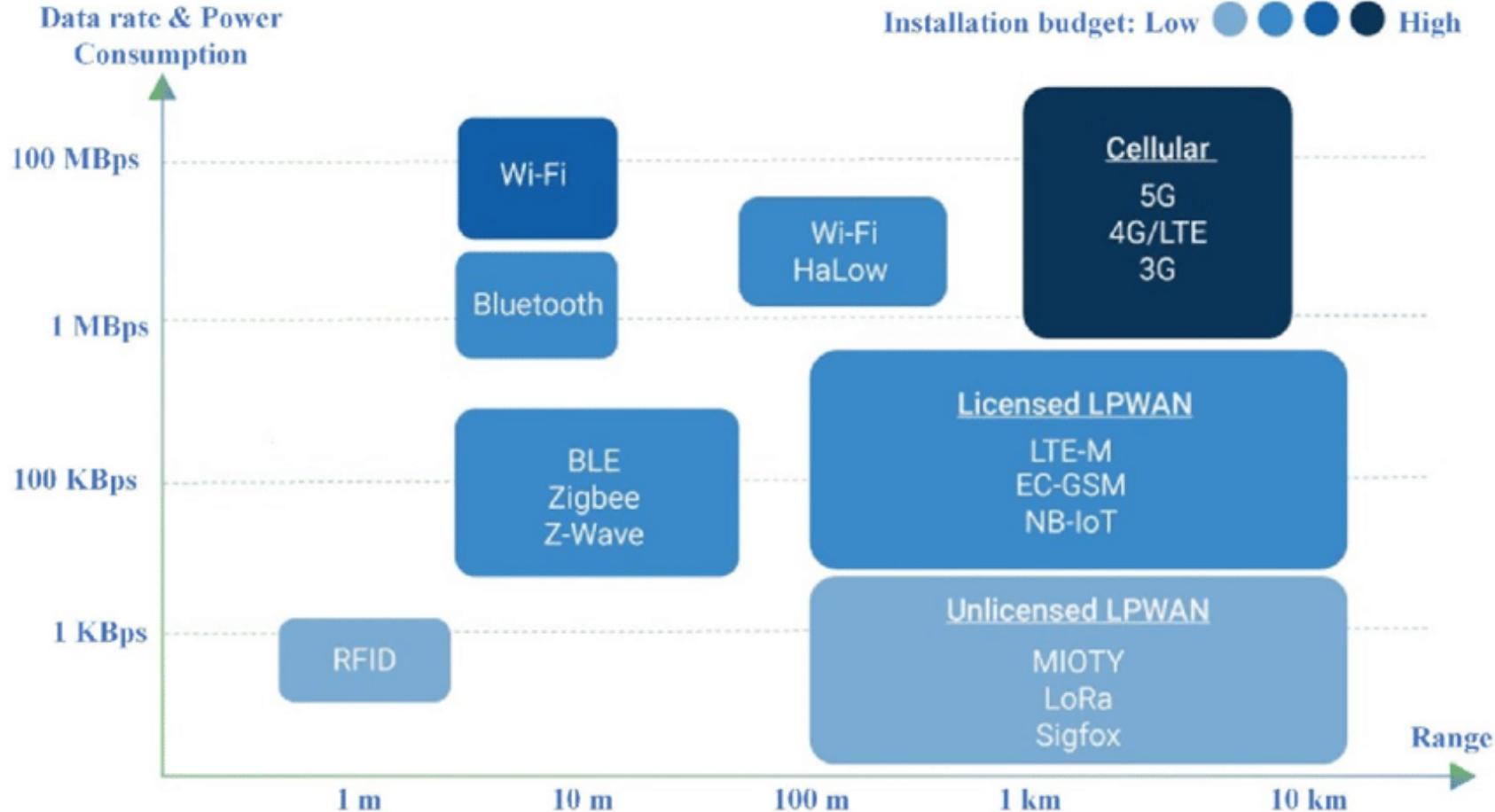
~ 10s kbps

Devices per



- IoT is a very broad field with too diverse requirements!
- Different applications have different requirements !

Competing and Complementing



<https://industrytoday.com/best-uses-of-wireless-iot-communication-technology/>

IoT market is diverse and big to accommodate several technologies!

Comparison

- Current trends
 - Indoor: WiFi, BLE, IEEE 802.15.4
 - Outdoor: Cellular, LoRaWAN, WiSUN, WiFi (Halow)
- On-mobile advantage
 - Cellular, WiFi, and BLE
- Lead advantage
 - WiFi, BLE, and LoRaWAN
- Ecosystem advantage
 - Cellular
 - Convergence of applications

Use Cases (Personal Assessment)

- Smart Metering
 - All Technologies
- Agriculture
 - LoRaWAN and then Cellular
- Mining
 - LoRaWAN and then Cellular
- Asset Tracking
 - Indoors or Jobsites
 - Since long time, RFID and Barcode
 - Bluetooth tags for construction tools
 - Outdoors (Vehicles, machinery, containers, tools):
 - Mostly cellular: <https://www.hologram.io/blog/10-best-iot-asset-tracking-systems>
 - LoRaWan based GPS tracker (Tata), <https://iot.tatacommunications.com/product/gps-tracker>
- Wearables
 - All Technologies
 - Bluetooth dominant and cellular

Few More Use Cases

- Smart Home
 - Metering: All technologies
 - Lighting: BLE, WiFi, Cellular, WiSUN
 - Signify has smart bulbs based on BLE, WiFi, and Zigbee
- Smart City
 - Street lighting: WiSUN (London example), LoRaWAN (Bordeaux), Cellular IoT (Australia)
 - Parking
 - Waste Management
- Forest: LoRaWAN

Smart Metering

- WiFi:
 - [Acrel](#)
 - [AmiciSmart3](#)
 - [Kigg](#)
- LoRaWAN:
 - [Acrel](#)
- Cellular: NB-IoT, LTE-M
 - [Digikey Link](#)
 - [Acrel](#) (4G)
 - [Sierra Wireless](#) (NB-IoT and LTE-M)
 - [Kigg](#)
- BLE:
 - [Kigg](#)
- IEEE 802.15.4
 - [Kigg](#)
- PLC
 - [Kigg](#)

Mines

LoRa Technology
Enabling Smarter
Mining Solutions
from Transco
Industries



www.semtech.com

**Semtech's LoRa Technology Enabling Smarter Mining Solutions
from Transco Industries Inc.**

Small and durable LoRa-enabled sensors monitor data in real-time to prevent costly conveyor damage and fires

<https://www.semtech.com/company/press/semtechs-lora-technology-enabling-smarter-mining-solutions-from-transco-industries-inc>

5G for healthcare

- 5G (not NB-IoT and LTE-M) is going to make these applications wide-spread
 - Telemedicine
 - Large file transfer
 - Improving augmented / virtual reality
 - Real time monitoring

<https://www.business.att.com/learn/updates/how-5g-will-transform-the-healthcare-industry.html>

References

- [Kurose2012] J. Kurose and K. Ross, *Computer Networking, 5th edition*, Pearson, 2012
- [Sohraby2007] K. Sohraby, D. Minoli, and T. Znati, *Wireless Sensor Networks: Technology, Protocols, and Applications*, Wiley, 2007
- [Koubaa2007] Anis Koubaa, Mário Alves and Eduardo Tovar, Time Sensitive IEEE 802.15.4 Protocol, *Sensor Networks and Configuration*, pp. 19-49
- [Townsend2014] K. Townsend, C. Cufi, Akiba, R. Davison, *Getting Started With BLE*, O'Reilly, May 2014
- [Gonzalez2016] V. Gonzalez et. Al. “IEEE 802.11ah: A Technology to face the IoT Challenge,” *Sensors*, 2016
- [Park2015] M. Park, “IEEE 802.11ah: Sub-1-GHz license-exempt operation for the internet of things,” *IEEE Communications Magazine*, September 2015.
- [Dohler2016] M. Dohler, et. al. “Internets of Things in 5G Era: Enablers, Architecture, and Business Models,” *IEEE Journal on Selected Areas in Communications*, Vol. 34, No. 3, March 2016
- [Mekki2019] K. Mekki, E. Bajic, F. Chaxel, and F. Meyer, “A comparative study of LPWAN technologies for large scale IoT deployment,” *Science Direct, ICT express* 5 2019.
- [IEEE802.11ax] Broadcom, *White paper on IEEE 802.11ax*, 17 Oct. 2018

**That's all for today!
Thank You!**

Communications & Controls in IoT

Connectivity Technologies

Instructor: Sachin Chaudhari

22 Sept. 2020



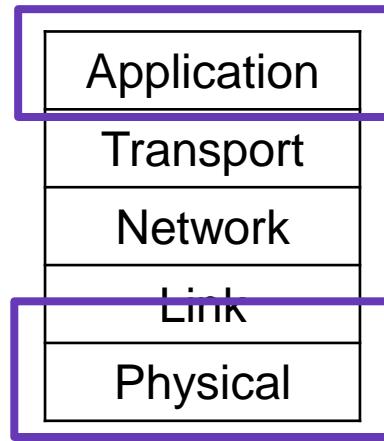
**INTERNATIONAL INSTITUTE OF
INFORMATION TECHNOLOGY**

HYDERABAD

Focus in this course



Seven-layer
Open Systems Interconnection
(OSI) model



Five-layer
Internet Protocol
stack

Application
MAC and PHY

Application Layer Protocols for IoT

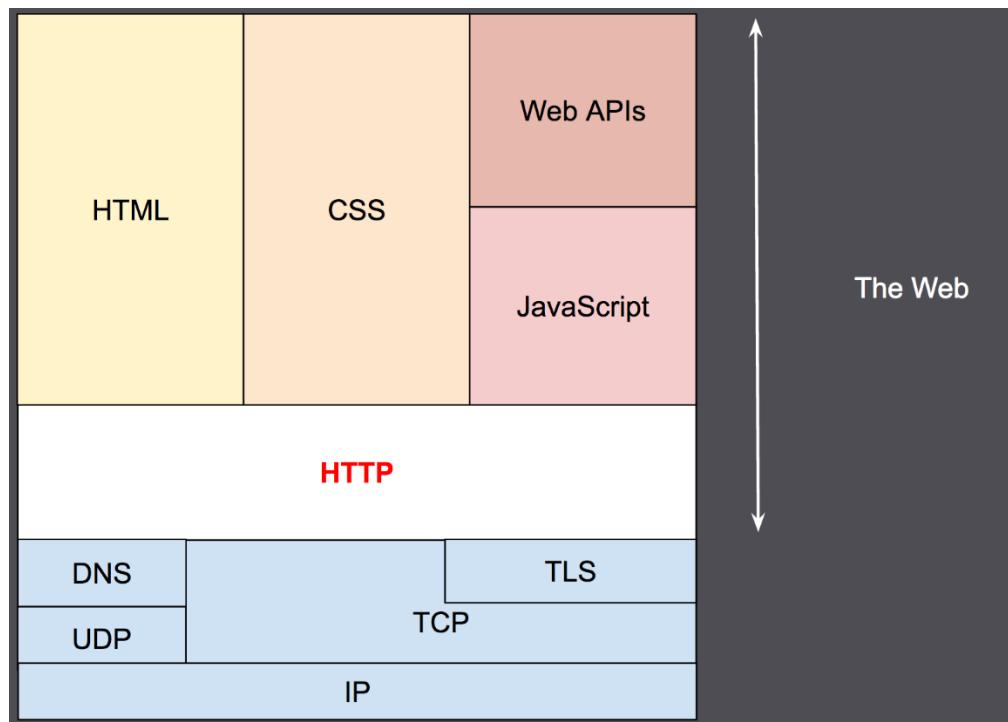
- Hypertext transport protocol (HTTP)
- Message Queue telemetry transport (MQTT)
- Constrained application protocol (CoAP)
- Advanced Message Queuing protocol (AMQP)
- More...

References

- [Kurose2012] J. Kurose and K. Ross, *Computer Networking*, 5th edition, Pearson, 2012
- [Lea2018] P. Lea, *Internet of Things for Architects*, Packt, 2018.
- [StevensMQTT] <http://www.steves-internet-guide.com/mqtt-works/>

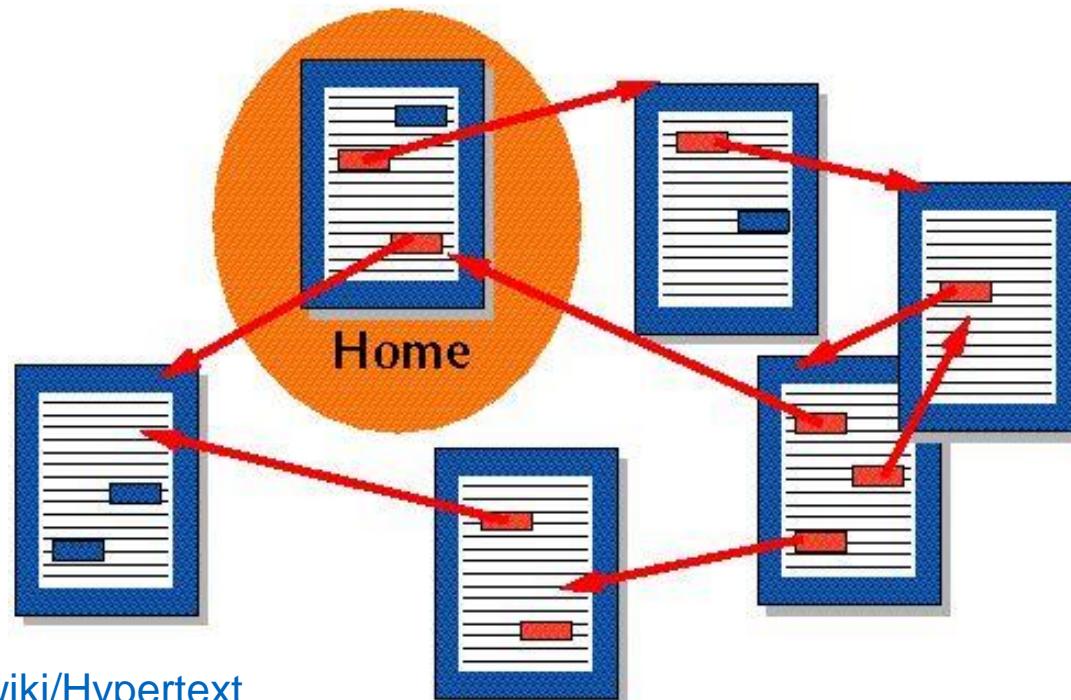
HTTP

- **Hyper Text Transport Protocol (HTTP)** is a protocol which allows the fetching of resources, such as HTML documents.
- It is the foundation of any data exchange on the web



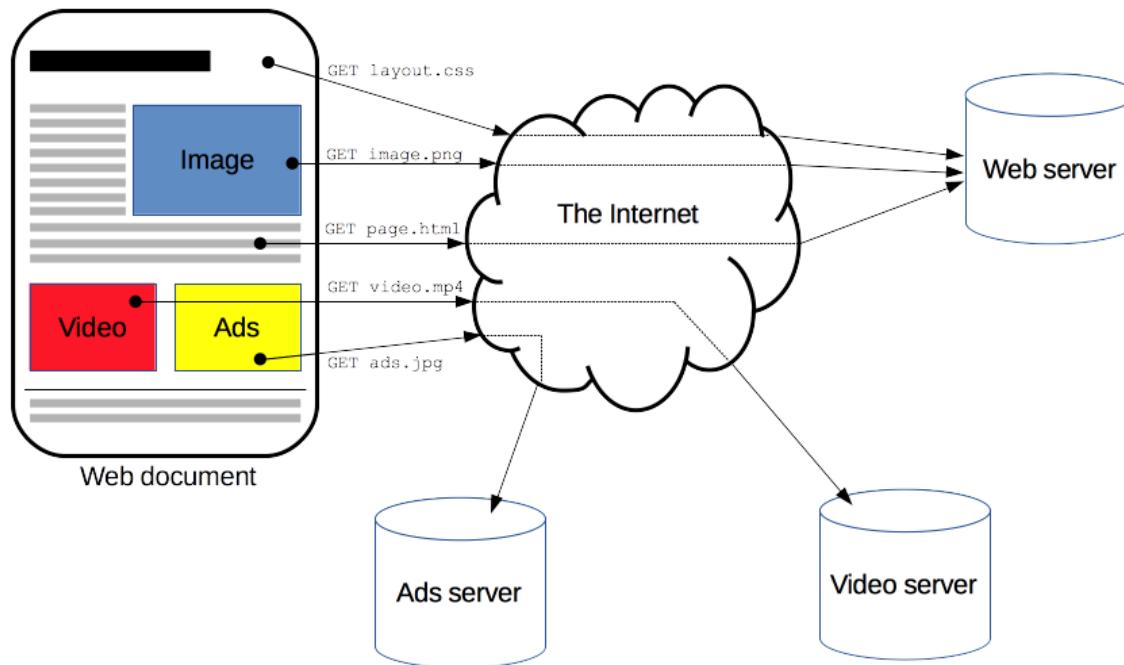
What is hypertext?

- Hypertext is text displayed on a computer display or other electronic devices with references to other text that the reader can immediately access.
- Hypertext documents are interconnected by hyperlinks, which are typically activated by a mouse click, keypress set or by touching the screen.

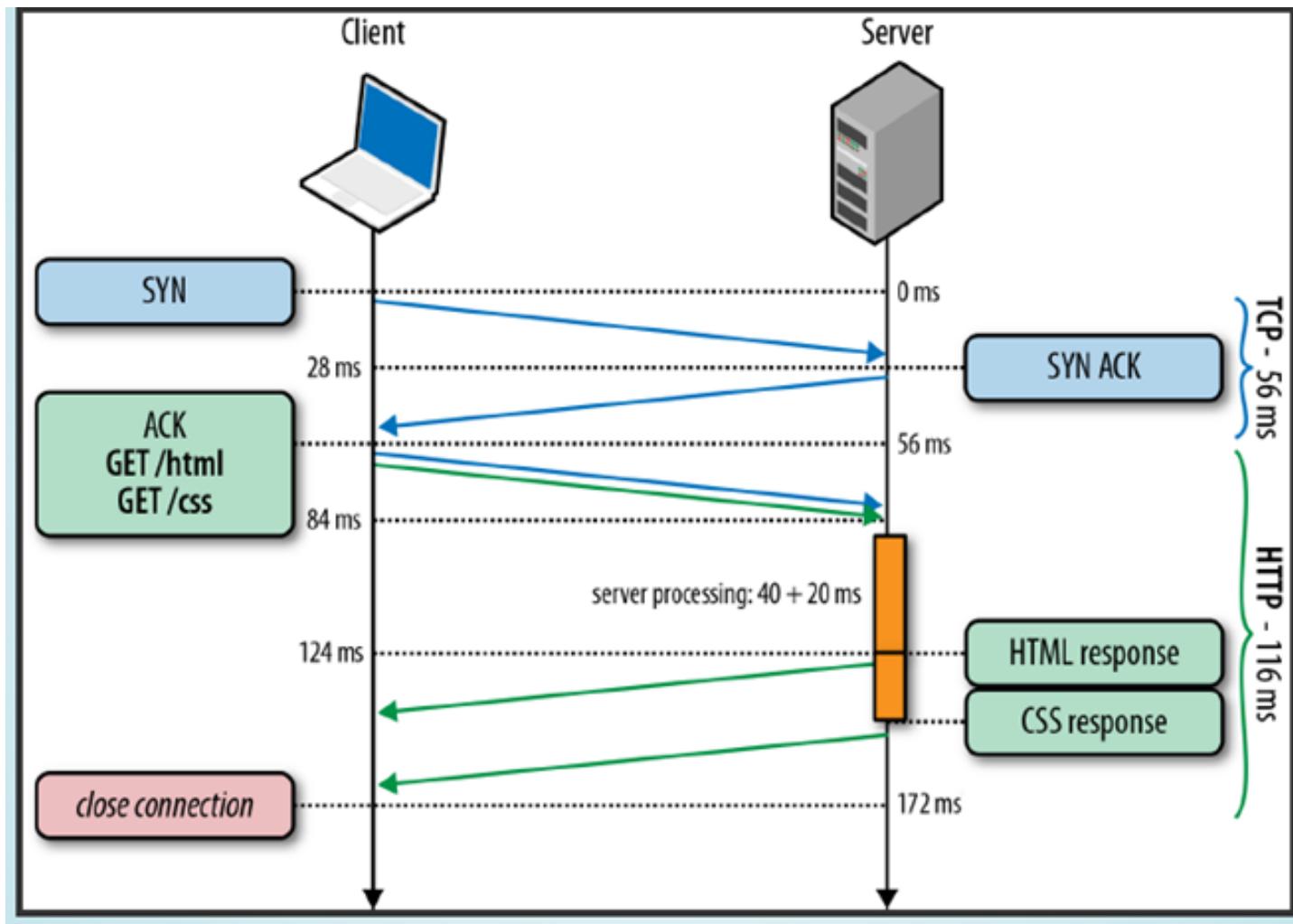


HTTP: Web Document Structure

- A complete document is reconstructed from the different sub-documents fetched, for instance text, layout description, images, videos, scripts, and more.



HTTP client-server model



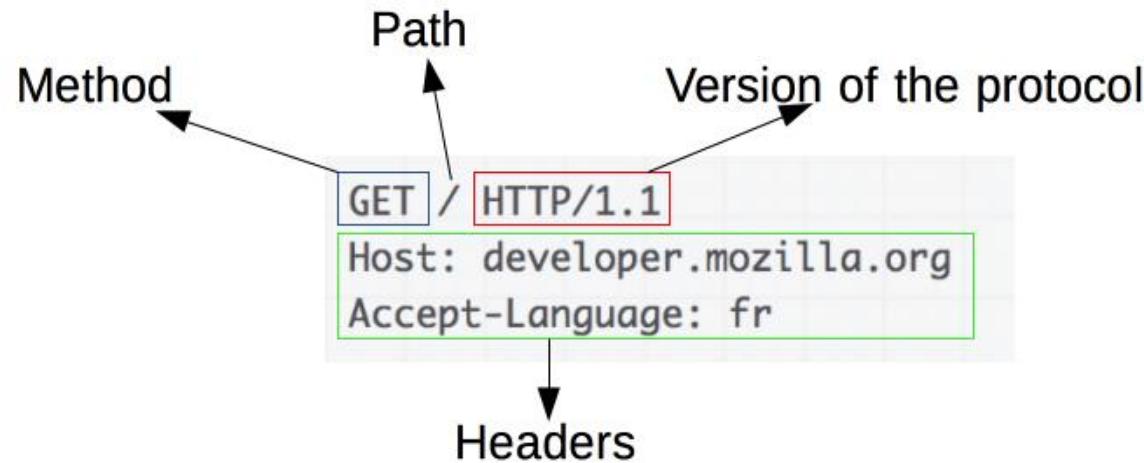
HTTP: client server model

- Distributed application structure that partition tasks between the providers called *servers* and service-requesters called as *clients*
- A server host runs one or more server programs
- A client does not share any of its resources
- Clients and servers exchange messages in a request-response messaging pattern
 - The client sends a request, and the server returns a response
- Client initiates the connection
- Servers waits for incoming requests
- Port number 80 or 8080
- Examples of applications that use the client–server model: email and internet

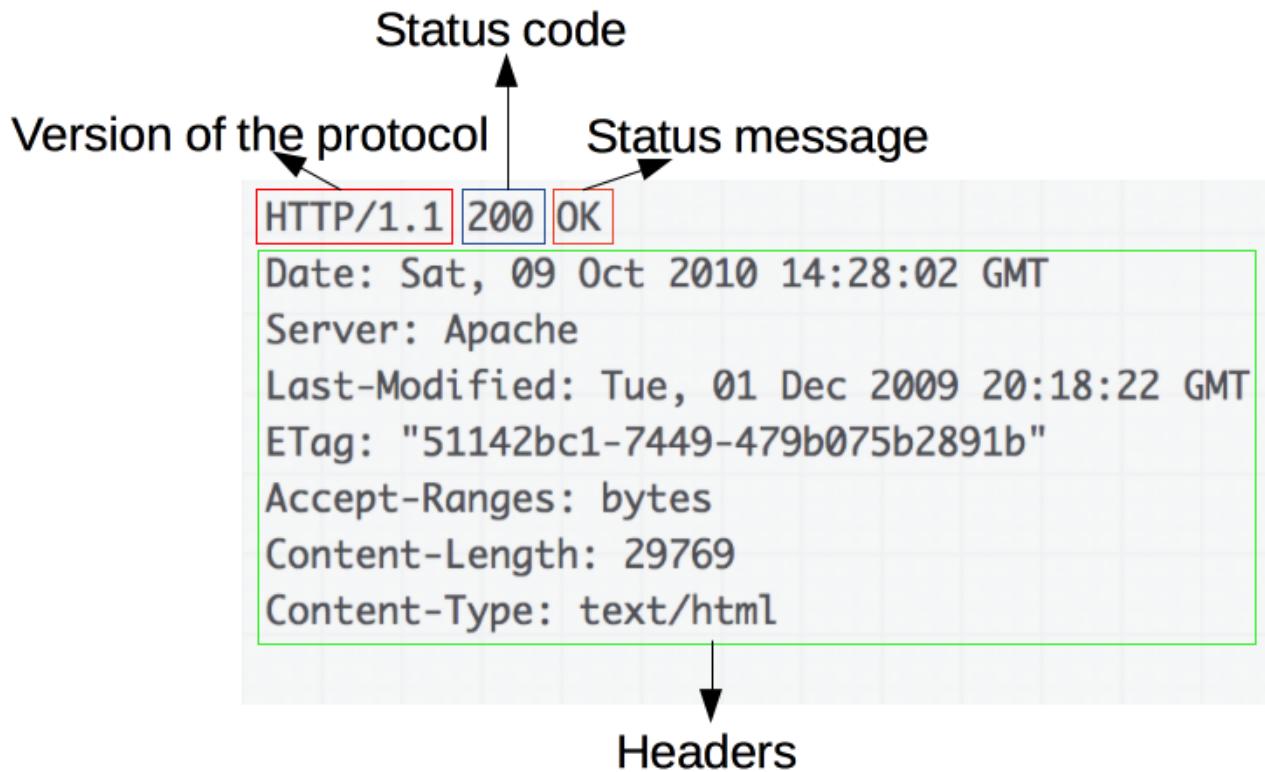
HTTP and TCP

- HTTP data rides above the TCP protocol, which guarantees reliability of delivery, and breaks down large data requests and responses into network-manageable chunks.
- TCP is a “connection” oriented protocol, which means when a client starts a dialogue with a server the TCP protocol will open a connection, over which the HTTP data will be reliably transferred, and when the dialogue is complete that connection should be closed.

HTTP Request: Example



HTTP Response: Example



HTTP Request Message Types

- **GET**
 - Requesting an object
 - Mostly used
- **POST**
 - Form filling
- **HEAD**
 - Similar to GET
 - Empty response
 - Used for debugging
- **PUT**
 - Upload an object
- **DELETE**
 - Deletes the content

Advantages of HTTP

- Simple and human readable
- Extensible
 - HTTP Headers make this protocol easy to extend and experiment with
- Stateless but not sessionless
 - there is no link between two requests being successively carried out on the same connection
 - while the core of HTTP itself is stateless, HTTP cookies allow the use of stateful sessions

Issues

- One-to-one communication
 - In most of the IoT applications, large number of sensors may want to push the data to the server at the same time
- Uni-Directional
 - In the case of IoT applications, we may need to send data in both directions
- Synchronous request-response
 - After requesting a resource to the server, the client has to wait for the server to respond
 - IoT sensors are small devices with very limited computing resources and hence cannot work efficiently in a synchronous manner
 - All the widely used IoT protocols are based on asynchronous model.

Issues

- Not designed for event-based communication
 - Most of the IoT applications are event based
 - Example: temperature based turning off a switch
- Scalability
 - HTTP connections utilize high system resources
 - As more sensor devices are added in the network, the load on the server increases
- High Power Consumption
 - Since HTTP utilizes heavy system resources as explained above, this also leads to heavy power consumption

HTTP has severe limitations for IOT applications. Many advanced application-layer protocols(MQTT, AMQP, CoAP) have been developed to overcome these limitations.

MQTT

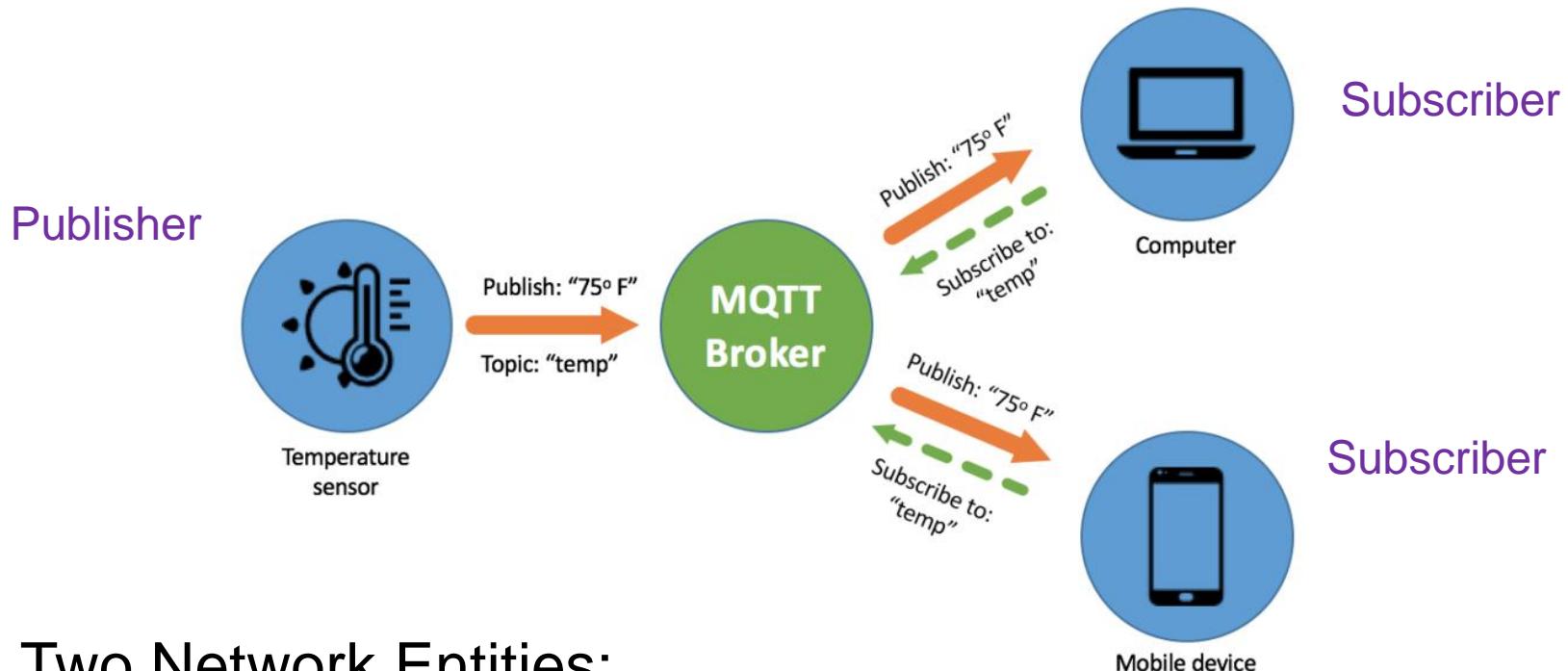
What is MQTT?

- Message Queuing Telemetry Transport is a simple messaging protocol for constrained devices and low-bandwidth
- Based on a publish-subscribe model
- Port number 1883
- Applications using MQTT
 - Facebook messenger (some aspects)
 - Amazon IoT
 - Microsoft Azure IoT Hub
 - Node-red supports MQTT with TLS

History of MQTT

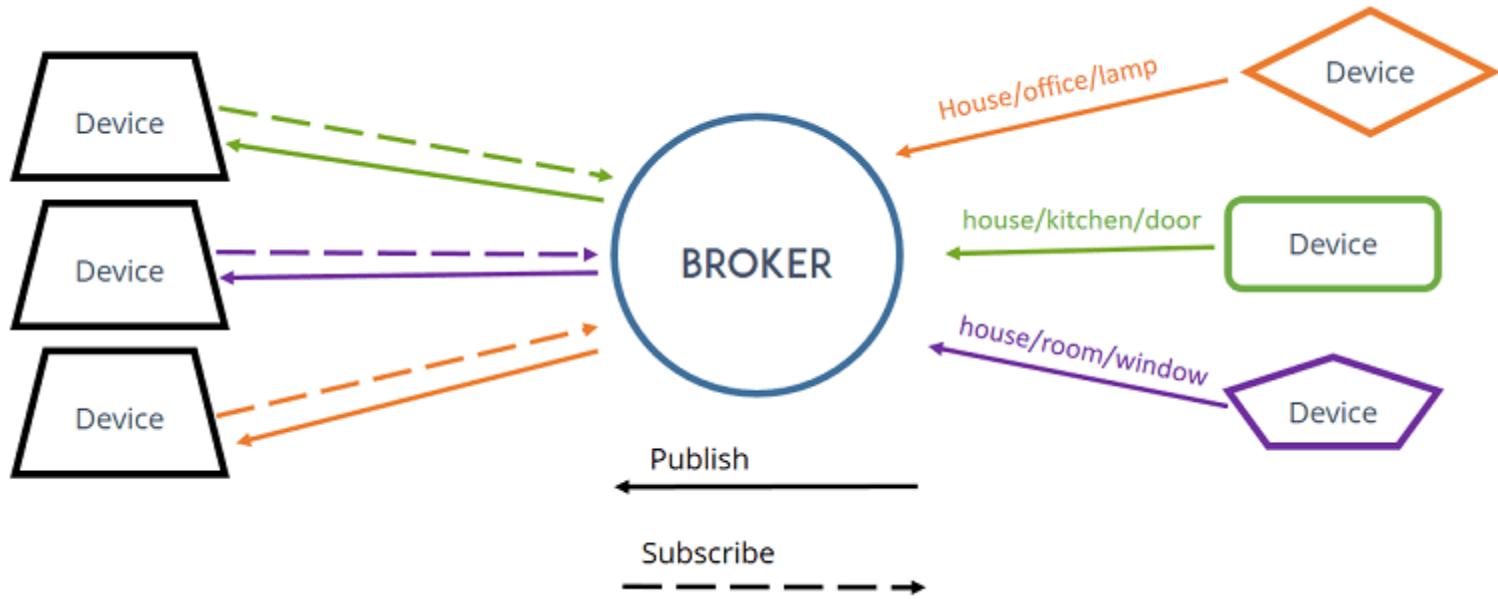
- Designed by IBM and Eurotech in 1999
 - Monitor an oil pipeline through the desert
 - Connected through extremely expensive satellite link
- MQ part is a misnomer
 - No message queues
 - Came from IBM MQSeries product line

Publish Subscribe Model



- Two Network Entities:
 - Broker: Server which receives all messages from the clients (ex. sensors) and routes them to other clients (ex. computer)
 - Client: Any device running MQTT library and connected to broker
 - Publishers: These clients send data to the broker
 - Subscribers: These clients read data from the broker
 - Clients can become subscribers (for monitoring) and publisher (for actuation) as well

Broker



Example of brokers: HiveMQ, Mosquito

<https://randomnerdtutorials.com/what-is-mqtt-and-how-it-works/>

Broker

- Filter messages based on topic, and then distribute them to subscribers.
- Manages and tracks all client connection states, including security credentials and certificates
- In general, do not store any messages
 - Can retain message based on flag
- Cloud-managed MQTT broker can take millions of messages per hour and support thousands of publishers

Publish Subscribe Model

- Clients do not have addresses like in email systems, and messages are not sent to clients
- Messages are published to a broker on a topic
- A client can receive these messages by subscribing to that topic on the same broker
- There is no direct connection between a publisher and subscriber
- All clients can publish (broadcast) and subscribe (receive)
- Data agnostic
 - Unlike HTTP (document), MQTT transfers any data

Topics

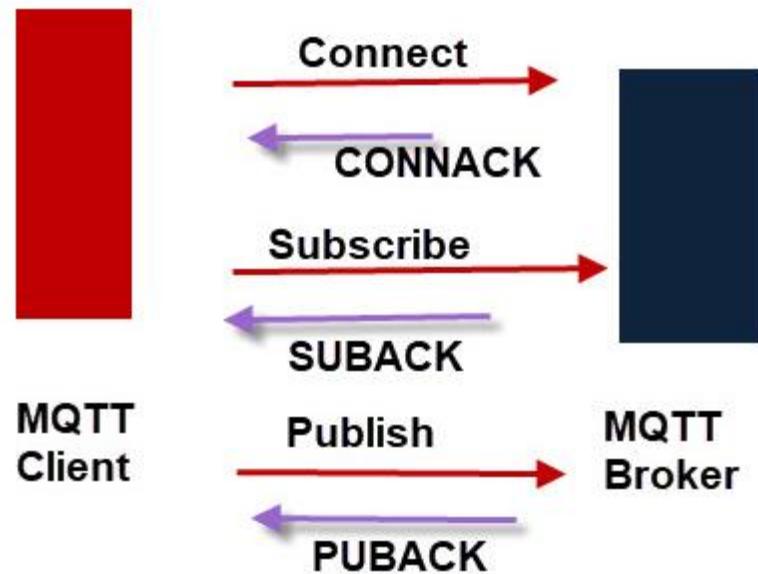
- Topics are the way you register interest for incoming messages or how you specify where you want to publish the message.
 - Topics are like channels



Message Types

- Connect
- Publish
- Subscribe

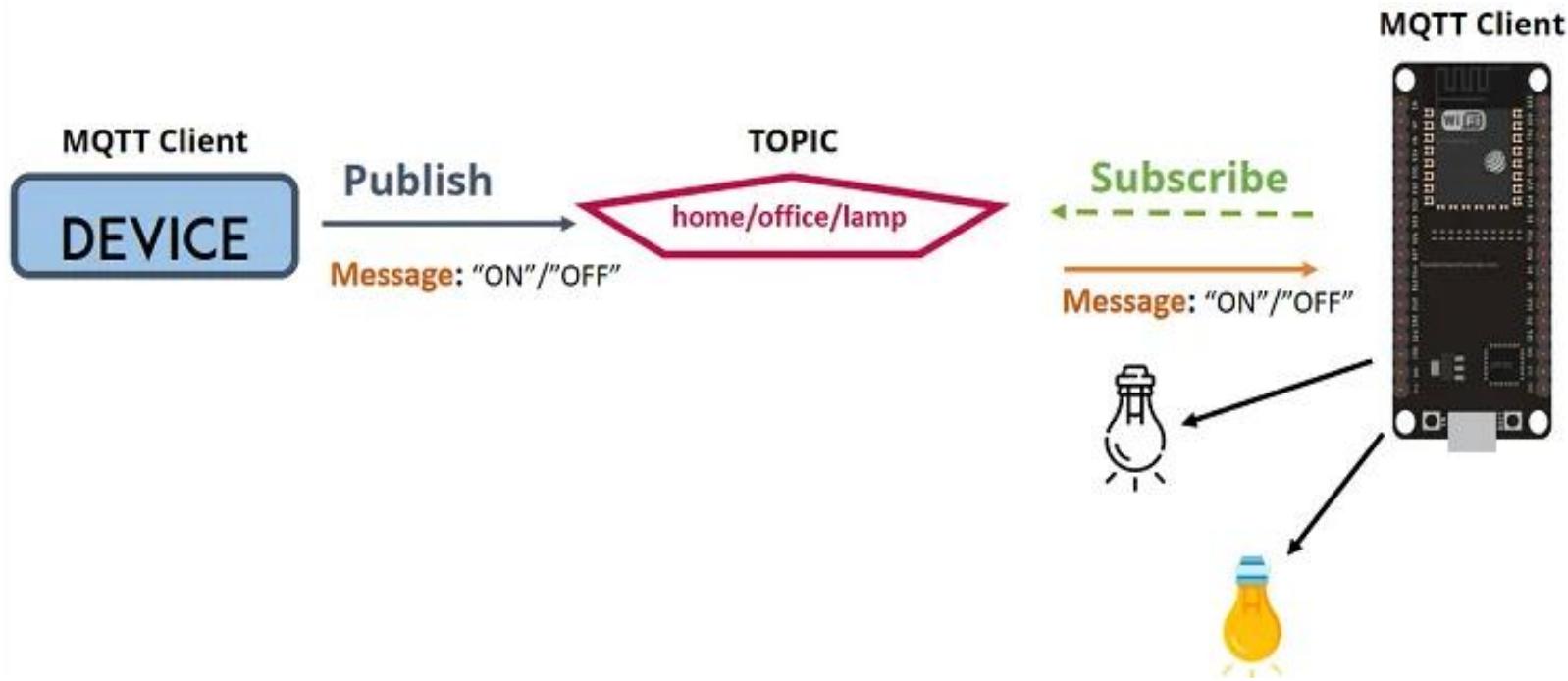
MQTT Message Flow



- MQTT is a command-response protocol. Each command is acknowledged
- You cannot publish or subscribe unless you are connected

<http://www.steves-internet-guide.com/mqtt-works/>

Example



Quality of Service Levels

- Three levels in order of increasing order of overhead:
 - At most once (fire and forget).
 - At least once (acknowledged delivery).
 - Exactly once using a two-level handshake (assured delivery).

Advantages

- Broker decouples publishers from consumers
 - More secure
 - No need of knowing the addresses of the publishers
- Publish-Subscribe model is time-invariant
 - Not one-to-one: Possible to send messages to multiple clients.
 - Highly scalable

TCP or UDP

- MQTT relies on the TCP protocol for data transmission.
 - Minimal control message from 2-bytes which can carry about 256 MB of data if needed
- A variant, MQTT-SN, is used over other transports such as UDP or Bluetooth/Zigbee

Security

- MQTT sends connection credentials in plain text format and does not include any measures for security or authentication
 - Can be provided by the underlying TCP transport
- Secure MQTT (SMQTT)
 - Uses light weight attribute based encryption
 - Broadcast encryption feature which does encryption of one message and delivers the same to multiple nodes
 - Algorithm is divided into four parts viz. setup, encryption, publish and decryption

<https://www.rfwireless-world.com/Terminology/MQTT-vs-SMQTT.html>

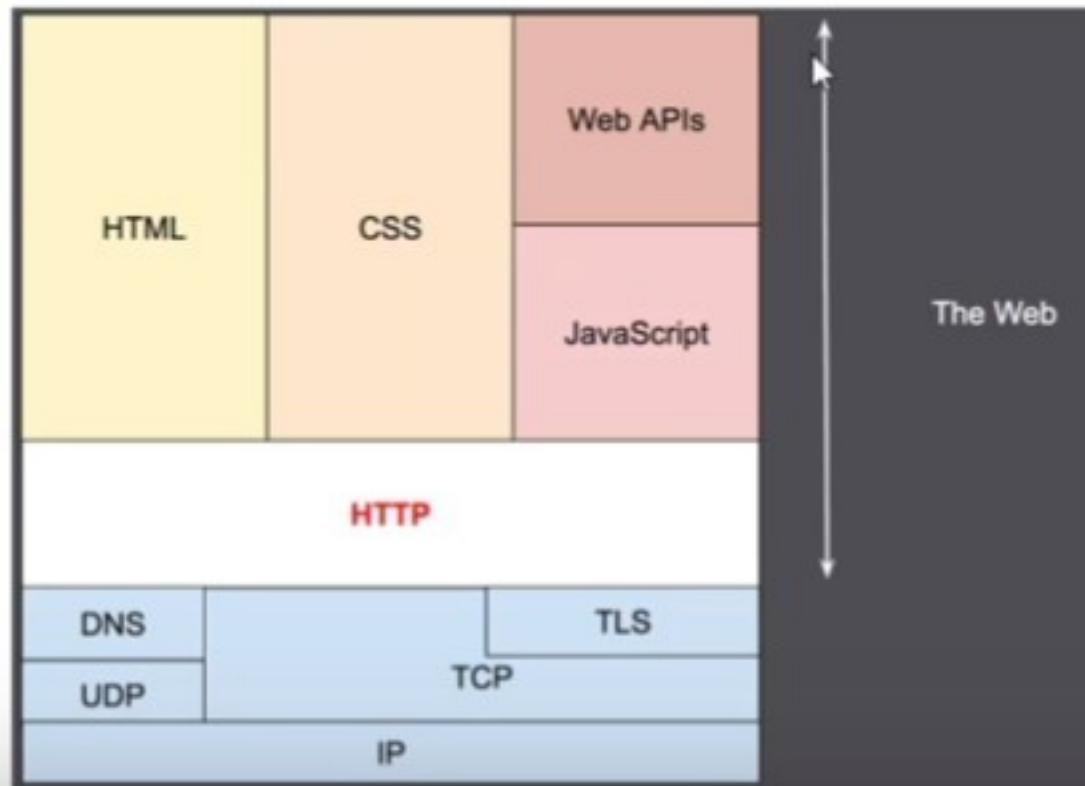
Thank You and Good Luck!

Application Layer Protocols for IoT

- Hypertext transport protocol (HTTP)
- Message Queue telemetry transport (MQTT)
- Constrained application protocol (CoAP)
- Advanced Message Queuing protocol (AMQP)
- More... 

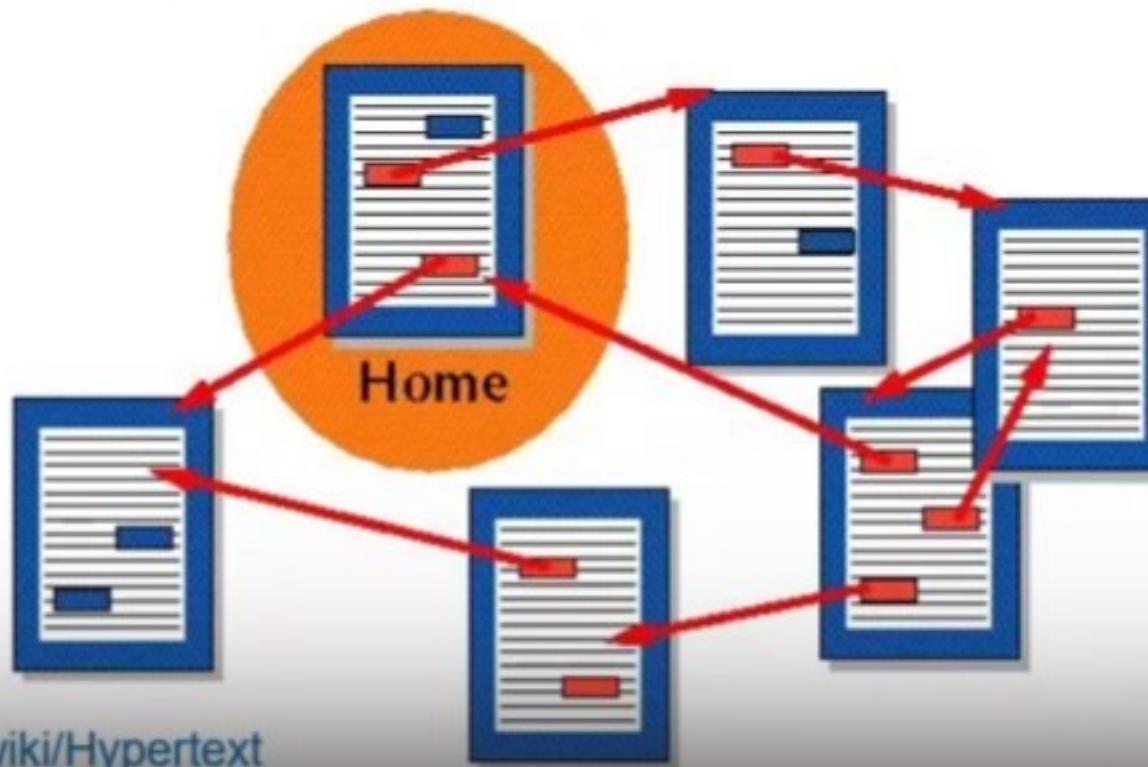
HTTP

- **Hyper Text Transport Protocol (HTTP)** is a protocol which allows the fetching of resources, such as HTML documents.
- It is the foundation of any data exchange on the web



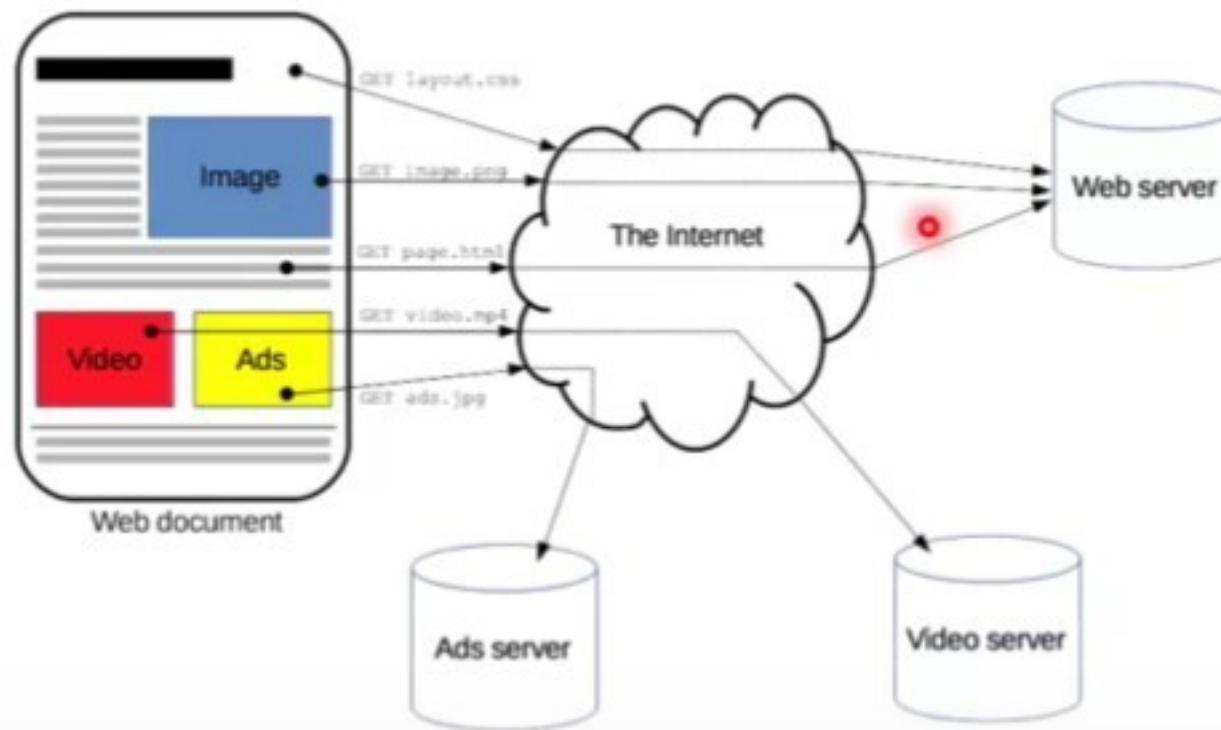
What is hypertext?

- Hypertext is text displayed on a computer display or other electronic devices with references to other text that the reader can immediately access.
- Hypertext documents are interconnected by hyperlinks, which are typically activated by a mouse click, keypress set or by touching the screen.

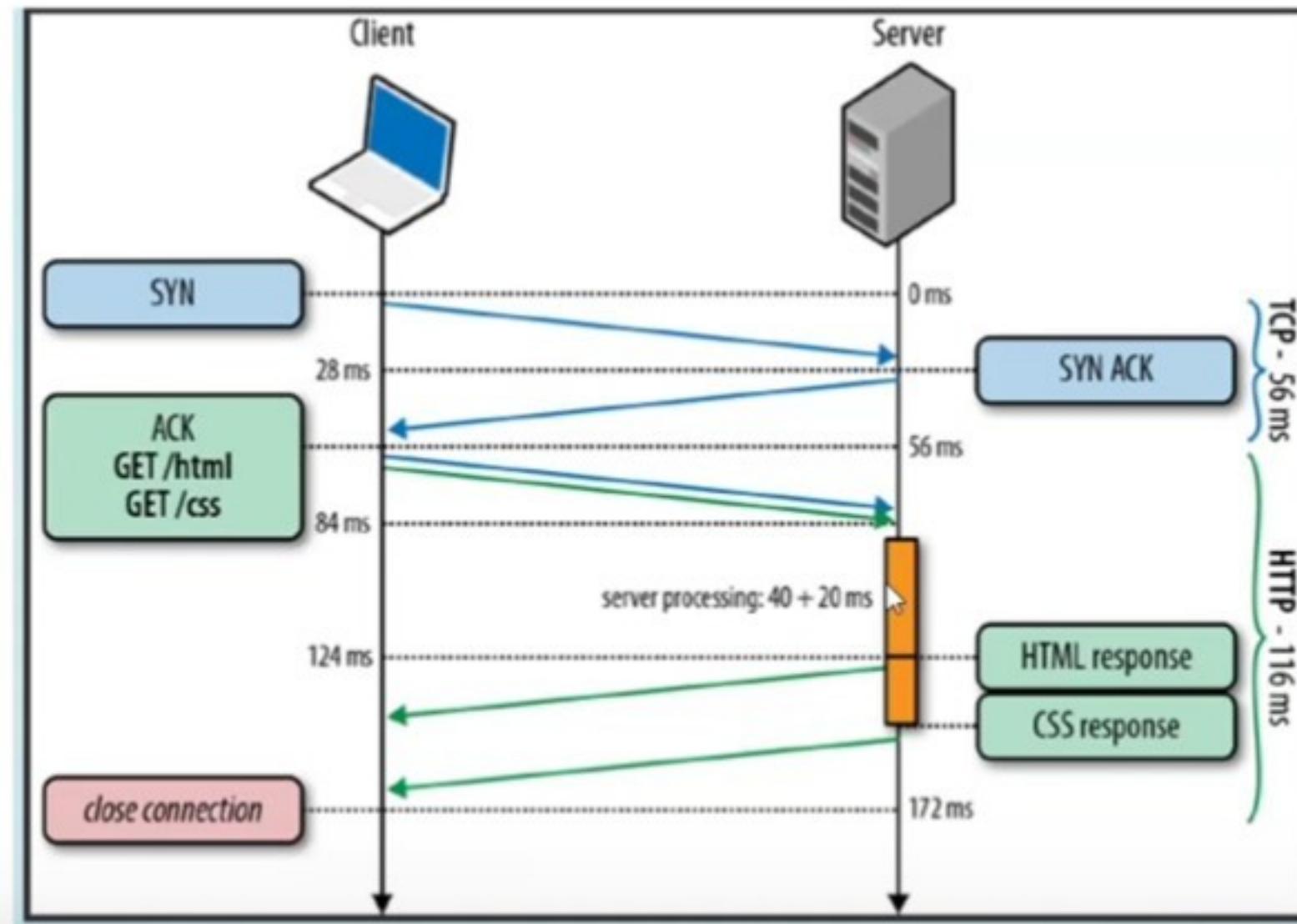


HTTP: Web Document Structure

- A complete document is reconstructed from the different sub-documents fetched, for instance text, layout description, images, videos, scripts, and more.



HTTP client-server model



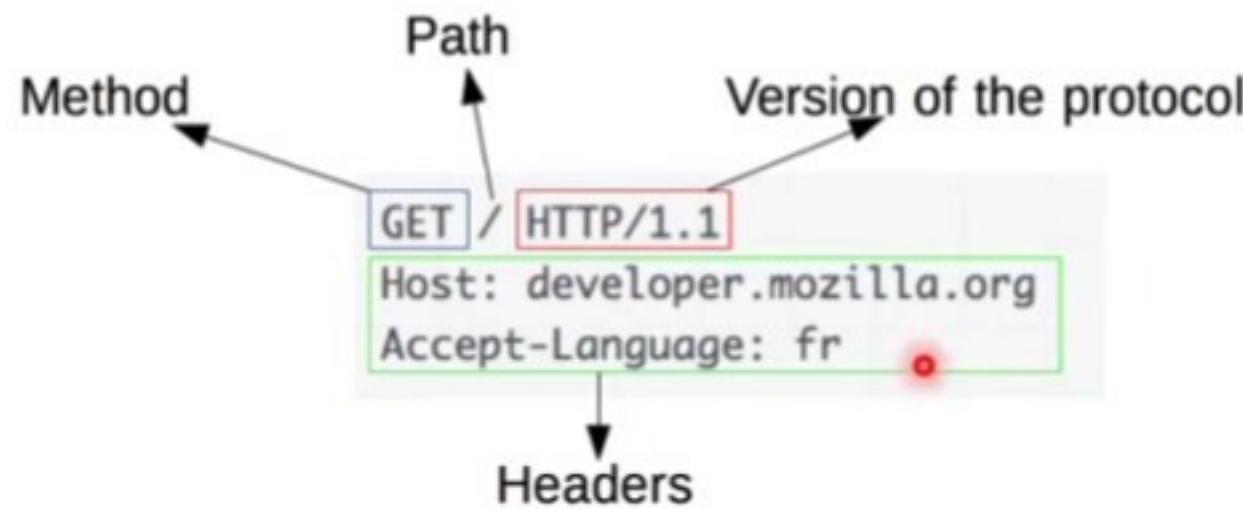
HTTP: client server model

- Distributed application structure that partition tasks between the providers called *servers* and service-requesters called as *clients*
- A server host runs one or more server programs
- A client does not share any of its resources
- Clients and servers exchange messages in a request-response messaging pattern
 - The client sends a request, and the server returns a response
- Client initiates the connection
- Servers waits for incoming requests
- Port number 80 or 8080
- Examples of applications that use the client-server model: email and internet

HTTP and TCP

- HTTP data rides above the TCP protocol, which guarantees reliability of delivery, and breaks down large data requests and responses into network-manageable chunks.
- TCP is a “connection” oriented protocol, which means when a client starts a dialogue with a server the TCP protocol will open a connection, over which the HTTP data will be reliably transferred, and when the dialogue is complete that connection should be closed.

HTTP Request: Example



HTTP Response: Example

Diagram illustrating the structure of an HTTP response:

The response consists of:

- Status code**: 200
- Version of the protocol**: HTTP/1.1
- Status message**: OK
- Headers**:
 - Date: Sat, 09 Oct 2010 14:28:02 GMT
 - Server: Apache
 - Last-Modified: Tue, 01 Dec 2009 20:18:22 GMT
 - ETag: "51142bc1-7449-479b075b2891b"
 - Accept-Ranges: bytes
 - Content-Length: 29769
 - Content-Type: text/html

HTTP Request Message Types

- GET
 - Requesting an object
 - Mostly used
- POST
 - Form filling
- HEAD
 - Similar to GET
 - Empty response
 - Used for debugging
- PUT
 - Upload an object
- DELETE
 - Deletes the content

Advantages of HTTP

- Simple and human readable
- Extensible
 - HTTP Headers make this protocol easy to extend and experiment with
- Stateless but not sessionless
 - there is no link between two requests being successively carried out on the same connection
 - while the core of HTTP itself is stateless, HTTP cookies allow the use of stateful sessions

Issues

- One-to-one communication
 - In most of the IoT applications, large number of sensors may want to push the data to the server at the same time
- Uni-Directional
 - In the case of IoT applications, we may need to send data in both directions
- Synchronous request-response
 - After requesting a resource to the server, the client has to wait for the server to respond
 - IoT sensors are small devices with very limited computing resources and hence cannot work efficiently in a synchronous manner
 - All the widely used IoT protocols are based on asynchronous model.

Issues

- Not designed for event-based communication
 - Most of the IoT applications are event based
 - Example: temperature based turning off a switch
- Scalability
 - HTTP connections utilize high system resources
 - As more sensor devices are added in the network, the load on the server increases
- High Power Consumption
 - Since HTTP utilizes heavy system resources as explained above, this also leads to heavy power consumption



Issues

- Not designed for event-based communication
 - Most of the IoT applications are event based
 - Example: temperature based turning off a switch
- Scalability
 - HTTP connections utilize high system resources
 - As more sensor devices are added in the network, the load on the server increases
- High Power Consumption
 - Since HTTP utilizes heavy system resources as explained above, this also leads to heavy power consumption



HTTP has severe limitations for IOT applications. Many advanced application-layer protocols(MQTT, AMQP, CoAP) have been developed to overcome these limitations.

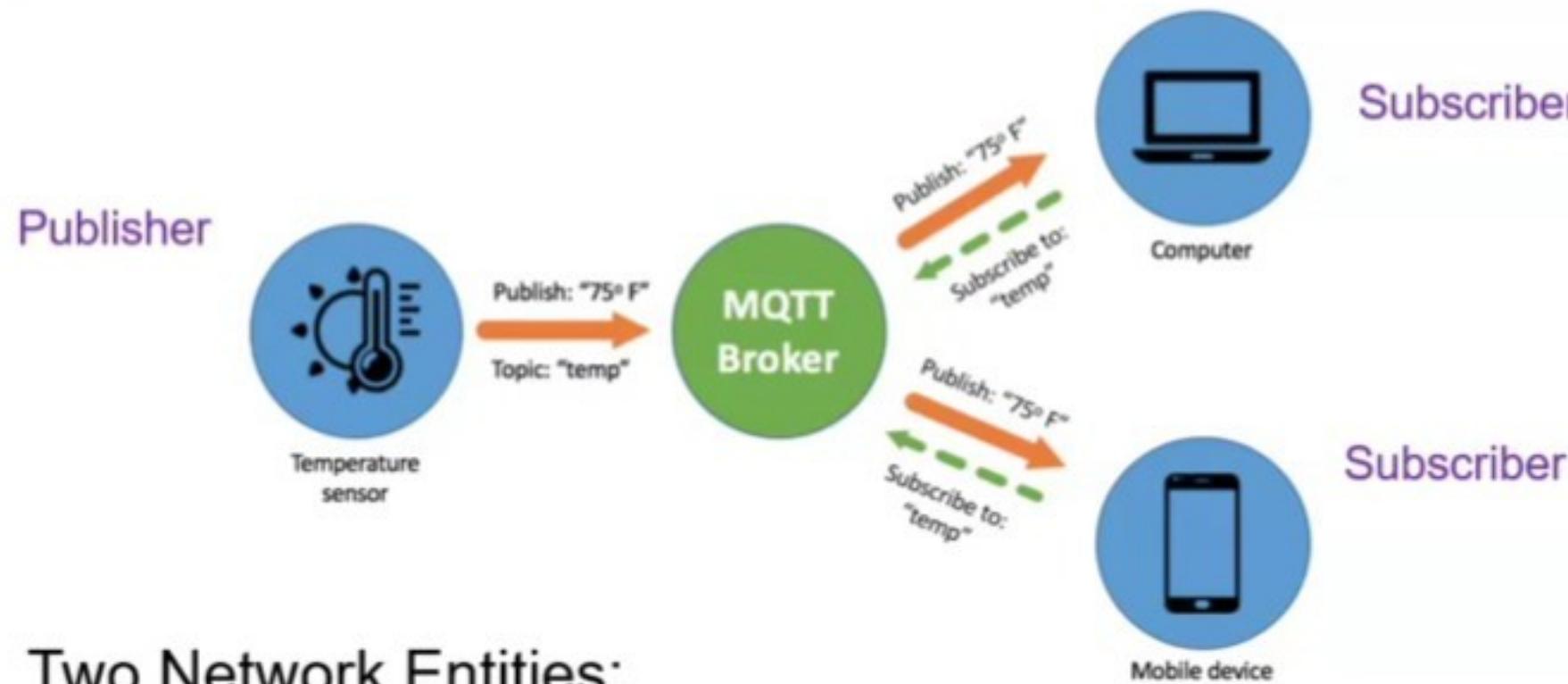
What is MQTT?

- Message Queuing Telemetry Transport is a simple messaging protocol for constrained devices and low-bandwidth
- Based on a publish-subscribe model
- Port number 1883
- Applications using MQTT
 - Facebook messenger (some aspects)
 - Amazon IoT
 - Microsoft Azure IoT Hub
 - Node-red supports MQTT with TLS

History of MQTT

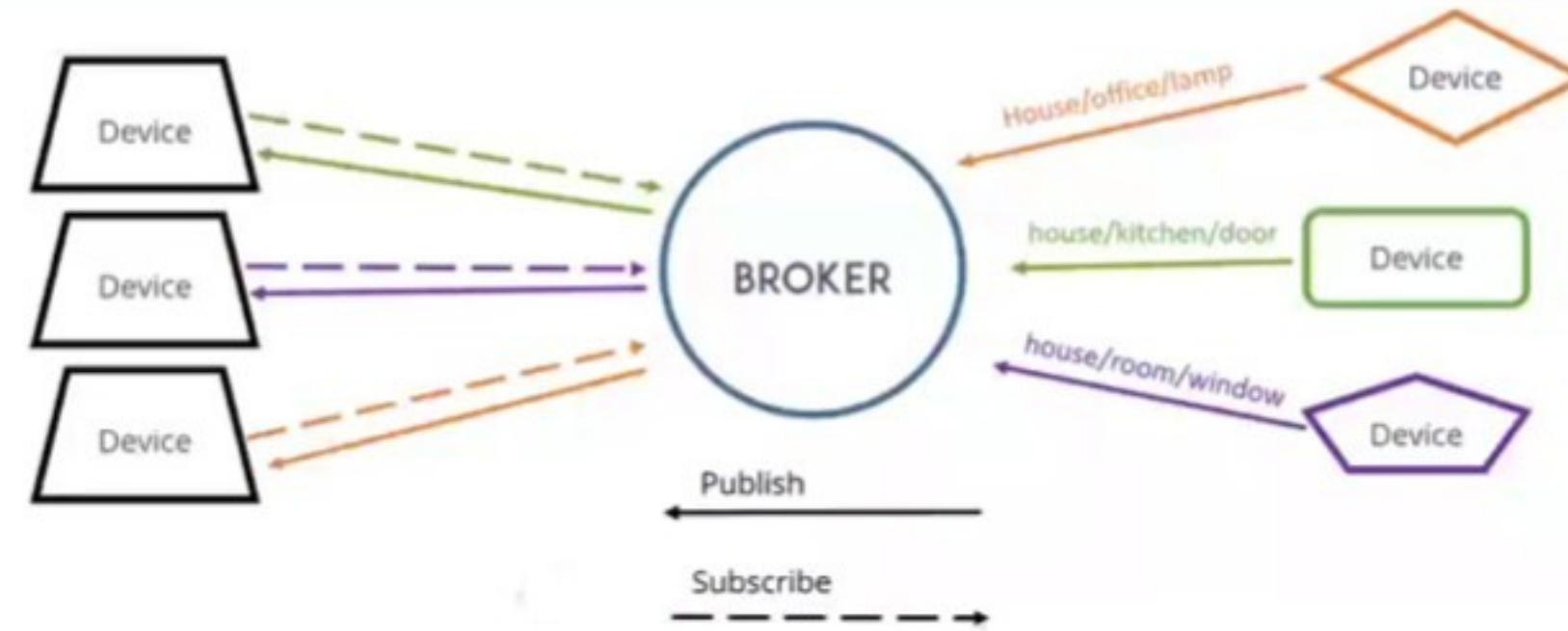
- Designed by IBM and Eurotech in 1999
 - Monitor an oil pipeline through the desert
 - Connected through extremely expensive satellite link
- MQ part is a misnomer
 - No message queues
 - Came from IBM MQSeries product line

Publish Subscribe Model



- Two Network Entities:
 - Broker: Server which receives all messages from the clients (ex. sensors) and routes them to other clients (ex. computer)
 - Client: Any device running MQTT library and connected to broker

Broker



Example of brokers: HiveMQ, Mosquito



<https://randomnerdtutorials.com/what-is-mqtt-and-how-it-works/>

Broker

- Filter messages based on topic, and then distribute them to subscribers.
- Manages and tracks all client connection states, including security credentials and certificates
- In general, do not store any messages
 - Can retain message based on flag
- Cloud-managed MQTT broker can take millions of messages per hour and support thousands of publishers

Publish Subscribe Model

- Clients do not have addresses like in email systems, and messages are not sent to clients
- Messages are published to a broker on a topic
- A client can receive these messages by subscribing to that topic on the same broker
- There is no direct connection between a publisher and subscriber
- All clients can publish (broadcast) and subscribe (receive)
- Data agnostic
 - Unlike HTTP (document), MQTT transfers any data

Topics

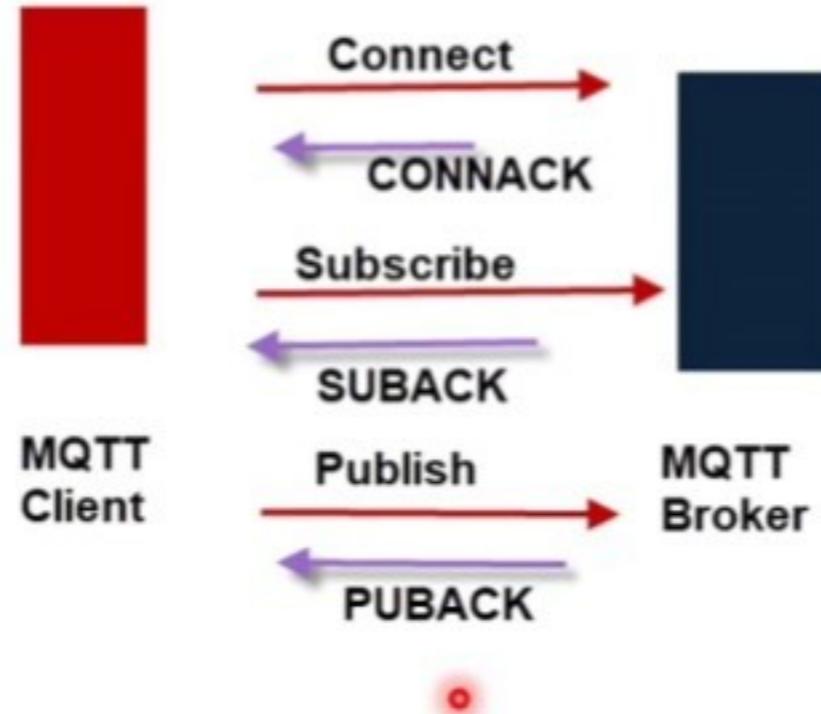
- Topics are the way you register interest for incoming messages or how you specify where you want to publish the message.
 - Topics are like channels



Message Types

- Connect
- Publish
- Subscribe

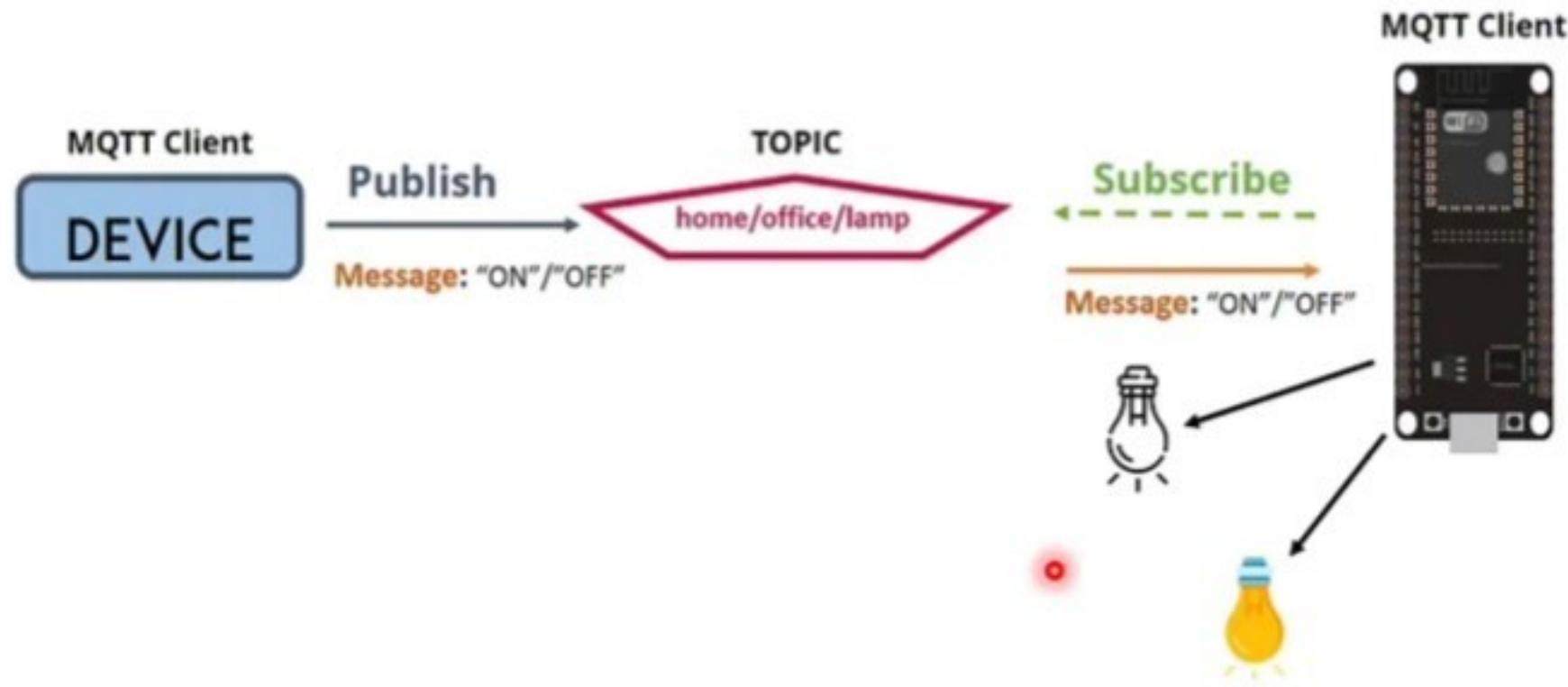
MQTT Message Flow



- MQTT is a command-response protocol. Each command is acknowledged
- You cannot publish or subscribe unless you are connected

<http://www.steves-internet-guide.com/mqtt-works/>

Example



Quality of Service Levels

- Three levels in order of increasing order of overhead:
 - At most once (fire and forget).
 - At least once (acknowledged delivery).
 - Exactly once using a two-level handshake (assured delivery).



Advantages

- Broker decouples publishers from consumers
 - More secure
 - No need of knowing the addresses of the publishers
- Publish-Subscribe model is time-invariant
 - Not one-to-one: Possible to send messages to multiple clients.
 - Highly scalable

TCP or UDP

- MQTT relies on the TCP protocol for data transmission.
 - Minimal control message from 2-bytes which can carry about 256 MB of data if needed
- A variant, MQTT-SN, is used over other transports such as UDP or Bluetooth/Zigbee

Security

- MQTT sends connection credentials in plain text format and does not include any measures for security or authentication
 - Can be provided by the underlying TCP transport
<https://en.wikipedia.org/wiki/MQTT>
- Secure MQTT (SMQTT)
 - Uses light weight attribute based encryption
 - Broadcast encryption feature which does encryption of one message and delivers the same to multiple nodes
 - Algorithm is divided into four parts viz. setup, encryption, publish and decryption

<https://www.rfwireless-world.com/Terminology/MQTT-vs-SMQTT.html>