

41: بحث Low Ram: نسخه Yun در آردینو نسخه ای است که از یک OS لینوکسی ساپورت میکند. نسخه light از لینوکس است. البته اگر توی forum ها بگردیم میبینیم برخی از دولوپر ها برای بقیه آردینو ها هم درست کردن ولی تنها در Yun است که به طور رسمی این کار انجام شده. (در غیر رسمی ممکنه کندی داشته باشیم)

42: از لحاظ برنامه نویسی هرچند آردینو به اسم تحت عنوان AL (Arduino Language) داره، اما اساسش C و C++ است که ما با کامپیوتر مینویسیم و با USB در برد پروگرم کنیم. امکان برنامه نویسی به کمک جاوا اسکریپت هم موجود است.

43: در مورد Connectivity: پین هایی که به ما اجازه I/O را میدهند: به دو دسته دیجیتال و آنالوگ داریم. در دیجیتال به صورت بیت با برد تبادل داده صورت میگیرد. در آنالوگ یک سیگنال پیوسته مثل speech داریم که خروجی میکروفون یک اختلاف پتانسیل آنالوگ است.

در آنالوگ رگوله میکنیم و در برد ذخیره میکنیم. یا برعکس وقتی میخوایم speaker داشته باشیم، یا آنالوگش رو میده یا دیجیتالش رو میده به D2A و آنالوگش رو بهمون تحویل میده.

به پین های I/O میگویند Low-level چون باید مشخص باشه جزئیاتش پین به پین چطوری باشه.

44: میان برد های Yun، Arduino و MKR1000 مستقیم به اینترنت بدون هیچ Shield به اینترنت وصل میشن. در بقیه باید به Shield قرار بدیم. Yun اینترنت هم داره ولی MKR1000 فقط وای فای رو پشتیبانی میکنه. علت این که GSM اینا هم توی shield میذارن اینه که وقتی میخوایم تو به فاصله دور در حد 1 الی 2 کیلومتر تبادل داده کنیم، دیگه wifi و اینترنت به درد نمیخوره و GSM باید استفاده بشه. البته انتخاب های دیگه هم هست که جلوتر توضیح داده میشه (چون GSM برای تلفن ها استفاده میشه و برای برد ها که باتری ضعیفی دارن سودمند نیست)

:47

Board name	Onboard connectivity	GPIO pins	Analog input	USB ports	ICSP header	Other hardware interfaces
Uno	–	14	6	1	1	SPI, UART, I2C/TWI
Mega2560	–	54	16	1	1	SPI, 4 UART, I2C/TWI
Due	–	40	12	2	1	SPI, 4 UART, I2C, 2 TWI
Yún	Ethernet, Wi-Fi	20	12	2	1	SPI, UART, I2C/TWI
101	BLE	14	6	1	1	SPI, UART, I2C/TWI
MKR1000	IEEE 802.11b/g/n	8	7	1	–	SPI, UART, I2C

GPIO: General Purpose هستن و بسته به تعداد آنالوگ و دیجیتال در برد های مختلف متفاوت است. این پین ها مثل SPI و UART و I2C و .. میباشند که هرکدام استاندارد خودشون رو دارن.

اگر بخوایم خارج برد تبادل داده کنیم، میریم سراغ Wifi و اینترنت و ...

از USB Port ها جهت پروگرم کردن، تبادل داده بین برد و کامپیوتر و هم جای باتری استفاده بشه.

48: \* وقتی میگی بحثای Serial Communication منظور ارتباطیه که نزدیک ماعه. هرچند میشه یک USB رو خیلی سیم طولانی درست کرد ولی استاندارد نیست.

49: از نظر سنسور های Onboard (سنسور هایی که در خود برد هست)، ما نگاهمون عموماً اینطوره که یک سنسور رو جدا بگیریم بذاریم روی برد. ولی آیا چیزی در خودش وجود داره؟ بله. نسخه های مثل Genuino 101 تنها بردی از آردینو است که سنسور هایی مثل ژيروسکوپ را دارد.

50: از لحاظ Security Feature هرچند بسیار مهمه در تمرکز درس نیست. در این بحث همون بحث که بین PC و موبایل است، این که PC رمزگذاری های سنگین تری دارد، به همون نسبت بین موبایل و دستگاه های IoT هم موجوده. اما باید بهش توجه بشه چون اگر نشه وارد خیلی از کاربرد ها نمیشه شد.

در لایه اپلیکیشن پروتکل های CoAP و MQTT اهمیت دارد.

-----پایان آردینو

51: از خانواده SoC دو تا برد دیگه هم گفته میشه. یکی از این ها ESP هستش که 2 نسخه داره. یکیش 8266 است که به بیشتر به خاطر ارتباط Wireless شناخته شده است. که یک TCP/IP رو خودش داخل خودش داره و نیازی به نوشتن جزئیاتش دیگه نیست. پس ارتباطات شبکه ای رو بسیار ساده میکنه.

این برد میتونه به سری اپلیکیشن هایی رو روش پروگرم کرد که دیگه نیازی به اضافه کردن ماژول اضافه وای فای نداریم.

52: نسخه دیگه که پیشرفته تره هست ESP 32 هست. این نسخه میشه گفت Next-generation قبلی هست و توانایی پردازش های بیشتری می باشد و رمزگذاری های بهتری رو میشه باهاش انجام داد.

53: یک برد دیگه که نسخه هاش نوی ایران کمتره Particle Electron عه، قابلیت معروفش ارتباطات Cellularع. میشه سیمکارت گذاشت. از لحاظ کاربردی علاوه بر این که مثلاً به خط لوله داشته باشیم و نیاز به ارتباطات از راه دور هستیم، میشه به کاربرتش در جاهایی که Mobility داریم اشاره کرد.

البته خب توی آردینو هم میشه Shield گذاشت که GSM داشته باشه ولی در این Built-in قرار داره.

بدی این برد نیز استفاده از سیم کارت و در نتیجه مصرف انرژی زیاد است. علاوه بر اون هزینه زیاد استفاده از آن است.

استفاده ازش: مثلاً داخل ماشین میتونیم بذاریم اگه دزدیدن بره تو حالت تماس انتقال داده کنه.

یک استفادش میتونه این باشه سیمکارت رو بذاریم داخلش بعد بذاریم ایران بعد بریم خارج و وقتی کسی زنگ زد این برد با اینترنت زنگ بزنه به ما توی خارج و جوابشو بدیم و اینطوری کاهش هزینه بشه.

:54

microcontrollers		
	Particle Electron	Espressif
	12 Analog in	21
	2 Analog out	14
PWM	30 Digital – 13 PWM	3.1
	3.3V	3.1

خروجی آنالوگ بهمون کمک میکنه به جایی که D2A استفاده کنیم خودش خروجی Analog میده و برای اسپیکر راحت تریم.

--- پایان دسته اول (SoC)

یه موقع میخوایم proof of concept کنیم، مثلاً میخوایم یه برد طراحی کنیم که وقتی یکی میره خارج سیمکارتو بذاره ایران و ارزان جواب بده. وقتی میخوایم ببینیم کار میکنه نگران قیمتش نیستیم و میبینیم کار میکنه. بعد حالا میخوایم تولید انبوه کنیم. اون موقع 1 دلار ارزون درآمدن هم مهمه. چون 2 دلار گرون بشه توی مقیاس 100 هزار دستگاه میشه 200 هزار دلار.

SBC ها یک سطح بالاتر هستند و تمام قابلیت های کامپیوتر در آن نهفته است. یک تفاوت مهم با SoC این است که در SBC داخل خودش OS هم داره.

مزیت هاش از قبلم گفته شده بود سخت افزارش اثبات شده است، استفاده ازش آسونه، آگه توی SoC یک اشتباهی کنیم ممکنه بسوزه ولی اینجا خودش هندل میکنه. Customizable هستش و میشه بهش چیز چیز اضافه کرد. یک source بیشتر نداره و Time to market اش خیلی سریع تر از SoC عه.

از نظر هزینه اگر بخوایم تولید انبوه کنیم، ممکنه احساس بشه نباید بریم سراغ SBC ها چون گرونن (مگر در شرایط خاص). ولی برای Prototyping خوبه. برای SoC نیاز به دانش بیشتری داریم تا SBC چون SoC به صورت Low-level تره

58: خوبی Raspberry Pi اینه که Compatible عه و میتونه با برد های قبلیش ارتباط برقرار کنه.

چون HDMI داره میتونیم به مانیتور وصلش کنیم و خروجی هایش رو ببینیم. با این که مانیتور خودش از یه جا دیگه انرژی میگیره ولی این که سیگنال چون داریم بهش میفرستیم همین سیگنال فرستادنش انرژی زیادی رو میگیره.

در این جلسه ادامه بحث در مورد معرفی پلتفرم های سخت افزاری برای کاربرد IoT می باشد.

در جلسه پیش Raspberry Pi رو دیدیم به عنوان SBC دیدیم کاربرد بیشتری نسبت به SoC مثل آردینو رو دارد.

60: Raspberry Pi دیگر Open Source نیست، مدل هاش از یه شرکت انگلیسی درست میشه. لایسنس اشو به 2 تا شرکت هم داده البته.

حدود 8 گیگ برای اپلیکیشن های Raspberry Pi فضا دارد. پس میشه در مقایسه با SoC برنامه های خیلی سنگین تری رو اجرا کرد.

63: پردازنده های Raspberry Pi به صورت Arm می باشد. دارای GPU هم برای پردازش های گرافیکی سنگین داراست.

64: در بین Raspberry Pi، که 1 تا 4 دارد. خودش B+ اینا هم داره که نسخه های پردازشی متفاوتی دارند.

65: برای مصرف توان کمتر، Mode های مختلفی دارد:

- Run: استفاده از تمام قابلیت ها
- Standby: اصل پردازشگر روشن
- Shutdown: کلا خاموش
- Dormant mode: مشابه standby با حداقل feature استفاده میکند.

در SoC ها با این که این همه mode نداریم توان خیلی کمتری داریم.

چون قطعات زیادی به Raspberry Pi وصل میشوند، مهم است وقتی که کاری باهاشون نداریم Disconnect کنیم. این زمانی اهمیت داره که Mobility داریم و به برق شهر مستقیم وصل نیستیم.

66: از لحاظ مصرف توان، نسبت به SoC میبینیم مصرف توانش بالاتره. عدد هایی که اینجا میبینیم ممکنه در شرایط آزمایشگاهی متفاوت بدست بیاریم ولی نسبت هاش درسته.

67: چون مصرف توان SBC بالاچه باید یک سری مواردی برای مدیریت توان داشته باشیم که 5 تاشو معرفی کرده

\* وقتی که کاربرد هامون real-time هستش این که بین مود های مختلف switch کنیم به تاخیر میلی ثانیه ای ایجاد میشه و باید حواسمون باشه تو کاربرد های realtime استفاده نشه از این mode ها

69: در مورد بحث OS، میشه از نسخه های مختلف لینوکس بالا آورد که در forum ها موجوده و یکی از مهم ترین تفاوت های SoC و SBC رو میتون همین بالا آوردن OS در SBC اشاره کرد.

70: از لحاظ زبان های برنامه نویسی طیف وسیعی از برنامه نویسی میشه استفاده کرد هرچند دیفالتش پایتون و اسکریپت هستش

71: در مورد connectivity، نسخه های اولیه ethernet رو onboard نداشت ولی به صورت shield میشد اضافه کرد.

73: نکته قابل توجه جدول اینه که CSI داره و میتونه دوربین بهش وصل بشه که کاربرتش اینه که مثلا یکی جلو دوربین وصل وایستاد به طور Real time بیاد face detection و بعد به یک actuator بگه مثلا در رو باز کنه.

74: در مورد بحث sensor ها هم تو خودش مثلا سنسور دما رو داره میشه ازش خوند ولی عموما باید از بیرون وصل کنیم.

75: در مورد Security میشه الگوریتم های پیشرفته تری نسبت به SoC داشته باشیم.

در مورد Raspberry نسبت به SoC میشه گفت High level تر هستش و بدون دانش خاصی میتونیم راحت تر کار کنیم و بیشتر درگیر الگوریتم میشیم تا سخت افزار. بعضی مواقع ولی ما میخوایم از یک دیواس در یک منطقه ای با به مدت زیادی بدون برق مستقیم کار کنیم اون موقع باید بریم سراغ SoC ها چون هزینه کمتری داره و توان کمتری مصرف میکنه.

76-78: BeagleBone هم یک SBC هستش ولی نسبت به Raspberry Pi از لحاظ پردازش های مالی مدیا و گرافیک ضعیف تره.

81: مشاهده میکنیم SoC ها GPU ندارن. از لحاظ سرعت کلاک و حافظه مشاهده میکنیم SBC خیلی بهتره و SBC پروتکل های ارتباطی رو تو خودش داره و SoC باید با شیلد وصل بشه و تنوع زبان در SBC بیشتره.

82: چه موقع میریم سراغ آردینو؟ موقی که یک سری Task های تکراری چه به صورت Periodic چه Event-driven که ساده هستند رو میخوایم انجام بدیم میریم سراغ Arduino. ولی وقتی Multi task داریم مثلا ویدیو بگیریم بعد Authentication بشه و بعد با Actuator یک کاری انجام بده بعد Cloud هم بره ذخیره شه دیگه اون موقع میریم سراغ SBC

84: قبل این که میخوایم ببینیم SoC بریم یا SBC باید نیازمندی هارو از لحاظ مصرف توان و ... بررسی کنیم بعد ببینیم کدومش رو انتخاب کنیم. مثلا Wifi داریم ولی توان کمی میخوایم بهتره Arduino باشه با شیلد تا این که SBC باشه.

87: اینجا یک نظرسنجی از افراد مختلف رو داریم میبینیم ولی خب این نمیشه مرجعی باشه.

88: اینجا نظر افراد رو میبینیم که کدوم پارامتر ها مهم هستش.

Community ecosystem: یک جایی که مشکلا گزارش داده میشه و در محل development مشکلاتی که قبلا پیش اومده رو ببینیم و حل کنیم.

این از نظر عموم هستش ولی ممکنه ما تو به کاربردی باشیم که Multimedia برامون اهمیت اول رو داشته باشه.

89: بعد این که Requirement engineering کردیم حالا یک سری موارد هست در داخل IoT که میایم رو کاغذ میاریم بررسی میکنیم ببینیم سراغ کدوم برد بریم. یک سری معیار های دیگه هم داریم مثلا این که میشه addon اضافه کرد یا نه، OS، داره یا نه و .. که در انتخاب برد تاثیر داره.

101: در مورد OS هایی که وجود داره، معلوم نیست هنوز کدوم OS در IoT بازار رو بره.

چرا OS مهمه؟ چون وقتی کار به جایی میرسه که امنیت مهم میشه، اون موقع OS نقش پر اهمیتی داره.

102: دو نوع High-end و Low-end داریم:

- High-end: بر پایه لینوکس که بر SBC و SoC استفاده میشه
- Low-end: بر پایه لینوکس نیستن و در SoC استفاده میشه.

108: عموما کاربرد های IoT، Battery Based هستن و Capacity توش اهمیت داره که.

ما میدونیم  $P=VI$  که ضریب t بشه انرژی کل مصرفی رو میده. ما وقتی میگیه یه باتری داریم 3.7 ولت و 750mAh.

750mAh یعنی اگر یک ساعت روشن باشه تا 750mA جریان رو میتونه فراهم کنه. حالا اگر 2 ساعت اون موقع باید جریان رو نصف کنیم. که اینم با فرض 3.7V هستش.

حالا اگر بخوایم بر حسب وات بگی، 3.7 رو ضربدر 750 میکنیم میشه 2775mWh ظرفیت باتری. که اگر تقسیم بر 1000 کنیم میشه 2.775Wh

109: در این شکل هم میبینیم همینطور که باتری دیسشارژ میشه ولتاژ هم به مرور کم میشه.

113: current consumption اینطوری محاسبه میشه که ما یک سری وسایل جانی داریم هرکدوم در یک زمان Ts ای جریان Is رو میکشه بعد همه رو با هم جمع میکنیم تقسیم بر زمان کل میکنیم.

**بحث Power Consumptions Considerations:**

یکی از مهم ترین دغدغه های IoT مصرف توانه با توجه به این که به برق شهر وصل نیست.

مصرف توان اثر خودش رو در شبکه هم میذاره چون پروتکل شبکه رو اون thing هم باید اجرا کنه. هرچند یک طرف Cloud هستش و به برق شهر وصله ولی اون طرفش وسیله IoT هستش. پس در Design اون شبکه تاثیر خودشو میذاره.

سعی میکنیم تا حدی این مصرف رو بباریم پایین که به یک Performance قابل قبولی برسیم و روی اون تاثیر بدی نذاره.

(شماره گذاری اسلاید ها در این جلسه طبق شماره footer هستش نه شماره PDF)

110: ظرفیت باتری معمولا بر اساس آمپر ساعت (Ah) اندازه گیری میشه. باتری ها از 100 تا 4000 میلی آمپر ساعت تنوع دارند. علاوه بر این واحد، ما از وات ساعت (Wh) هم استفاده میکنیم که این 2 قابلیت تبدیل شدن به هم دارند.

\*111: اگر بخواهیم در شرایط آزمایشگاهی ظرفیت باتری رو اندازه بگیریم. Full charge استفاده میکنیم و x آمپر جریان ثابت ازش میکشیم. همین طوری که جریان میکشیم و باتری دیسشارژ میشه، حالت ایده آل در شکل پایینی معلومه این که تا به جایی میکشیم تا صفر بشه. و زمان T رو ضریدر X میکنیم و آمپر ساعت بدست میاد. وقتی این کار صورت میگیره متوجه میشیم ولتاژ کم کم افت میکنه و از یه جایی بسته به کاربرد یک threshold میگیریم (EODV) و میگیم دیگه فرض ما اینه که دیگه باتری کامل دیسشارژ شده و جریان قابل توجهی رو برامون تولید نمیکنه.

\*112: یک آزمایش رو مشاهده میکنیم از ساعت 15 شروع شده و 19:55 به بعد دیگه به 3 ولت رسیده دیگه اون جریان مد نظر ما نبوده. بعد محاسبه میبینیم ظرفیت واقعی 500 با ظرفیت اسمی 750 متفاوت است.

\*115: باتری که mAh750 ساعت است. اگر من 750mA ازش بکشیم به ساعت کار میکنه. ولی آیا تو این مدت زمان جریان ثابت میکشیم؟ طبیعتا خیر.

ولی به دیوتی سیکل وابستس ما مثلا در شبکه هر 2 دقیقه ارسال که میکنیم کل این 2 دقیقه جریان نمیکشیم و تنها موقع ارسال جریان میکشیم.

اگر N<sub>s</sub> تا منبع مصرفی مثل Peripheral ها و .. داشته باشیم هرکدام یک جریانی در یک مدت زمان T<sub>s</sub> مصرف میکنند. اگر همه این هارا تقسیم بر مدت زمان کلی T<sub>on</sub> بکنیم جریان متوسط I<sub>on</sub> بدست می آید.

\*116: اگر مدت زمان Sleep هم در نظر بگیریم رابطه جریان متوسط کل بدست میاد. چون توی زمان sleep همه چی خاموش نیست. (T<sub>ci</sub> - T<sub>on</sub>) میشه مدت زمان خاموش. و T<sub>ci</sub> میشه مدت زمان کل.

اگر ظرفیت C رو تقسیم بر این جریان متوسط کل بکنیم مدت زمانی که باتری روشنه بدست می آید.

مدت زمانی که باتری روشنه چطور بدست میاد؟ ظرفیت باتری تقسیم بر I<sub>avg</sub> میشه.

\*118: ما وقتی میخوایم Power consumption یک باتری رو حساب کنیم یا به عبارتی Battery life یک باتری رو میخوایم حساب کنیم، این به ولتاژ کاری و Operating Current وابستس (I<sub>on</sub>). بعد بر اساس کارکردی که اون سخت افزار برای ما داره. ما میتونیم Power مورد نیاز دستگاه رو به چندین بلاک تقسیم کنیم:

- Data sensing and Acquisition: یک دیتایی رو بگیریم sense کنیم که به ذات سنسور وابستس
- Data handling or processing: پردازش هایی که روی دیتا داریم
- ارسال این داده های پردازش شده به یک جا
- System management: کاری

همه این عناصر با هم Power consumption دستگاه رو تعریف میکنن.

119: در شکل میبینیم وقتی که پردازشی میخوایم انجام بدیم توان زیاد میشه و انرژی ناشی از Networking خیلی زیاد میکنه. از شکل میبینیم مصرف توان متغیره.

میزان مصرف در Networking از همه بیشتره. چون هم مودم ها باید روشن باشه هم یک سیگنالی 2 سر آنتن ها قرار میگیره خودش یک توانی بسته به میزان مسافتی که ارسال میشه داره. بنابراین باید تا میتونیم مدت زمان ارسال رو کم کنیم. این که مدت زمان رو کم میکنیم به اصطلاح میگیم Low duty cycle داریم. مثلا املاح خاک در کشاورزی دیگه هر یک دقیقه تغییر نمیکنه که هی ارسال کنیم و باید تا میتونیم این زمان ارسال رو کم کنیم که توی روز در حداقل زمان ممکن ارسال بشه.

\*120: ما دنبال ماکسیمیم T MSG هستیم به طوری که روی Performance تاثیر منفی نذاره. چون هرچی این فاصله رو بیشتر کنیم توان مصرفی کمتری خواهیم داشت. علتشم واضحه من هر 60 دقیقه یک بار یک مسیج رو ارسال کنم مسلما توان کمتری رو دارم مصرف میکنم تا این که هر 10 دقیقه یکبار.

\*128: یکی از سوالاتی که بایستی پاسخ به آن داده بشه اینه که چرا فرکانس روی مصرف توان تاثیر داره که در لکچر بعدی میبینیم چرا.

\*1: در این جلسه در مورد شبکه و فناوری های ارتباطی صحبت داشته باشیم.

در ابتدا مقدمه ای از network داریم که لایه های network یک دوره میشه. سپس وارد لایه های پایین physical و link می شیم و بررسی میکنیم در ارتباط های با سیم و بی سیم چه پروتکل هایی داریم و چرا برای IoT مناسب نیستند.

در لکچر بعدی وارد لایه network میشیم و در آخر در لکچر های بعدی وارد transport و application میشیم.

\*3: Protocol Stack: فرایند ارتباط از A به B فرایند پیچیده ای است. باید task های زیادی اتفاق بیافته تا یک پیام از نقطه A به B ارسال شود.

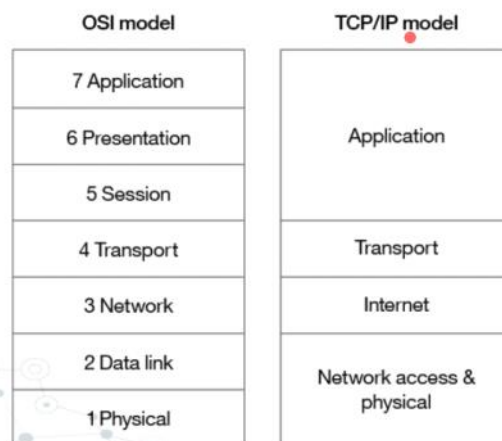
در ارتباط با پدیده های پیچیده، اگر ما یک structure کاملاً روشن داشته باشیم که سیستم پیچیده رو به یک سری قطعات تقسیم میکنه. و به صورت ماژول بندی نگاه کنیم کارمون راحت تره. ما مجموعه ای از task هارو کنار هم نگر میداریم و ماژول ماژول تقسیم بندی میکنیم.

این ماژولاریز کردن (لایه بندی) مزایای داره:

- نگهداری سیستم رو برا ما راحت میکنه.
- آپدیت کردن سیستم هم راحت میکنه: فقط ماژول مربوطه تغییر میکنه و بقیه تغییر نمیکنن.

این تجربه در درس برنامه نویسی هم هست. وقتی function تقسیم بندی میکنیم هرکدام وظیفه مشخصی داره که با team working میتونیم راحت تر توسعه بدیم.

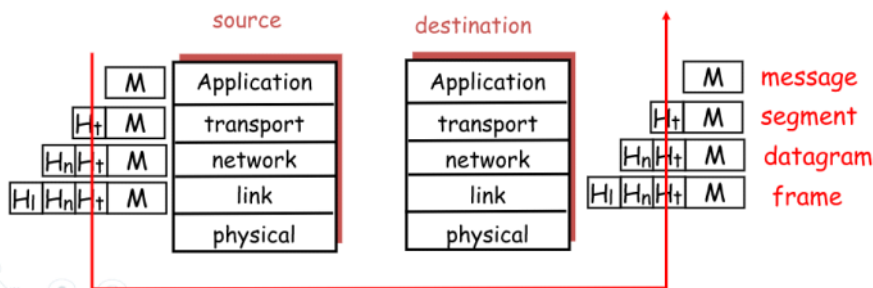
\*4: یک سری مدل برای لایه بندی داریم. مدل تئوری 7 لایه ای OSI و TCP/IP:



در تصویر مشاهده میکنیم هر کدام از لایه ها در TCP/IP معادل کدام لایه ها در OSI هستند. در TCP/IP سطح Abstraction بالاتری داره.

اگر جایی بگیریم TCP/IP 5 تا لایه داره منظورمون لایه پایینه اس که به 2 تا تقسیم کردیم.

\*5: هر لایه ای با لایه بالاتر و پایین تر در ارتباطه. در Source لایه بالایی داده رو به پایینی انتقال میده.



به طور خلاصه:

- Application: پیام اصلی رو تولید میکنه. داده ای که قراره رد و بدل بشه. مثل سرچی که توی گوگل داریم. اینا بایتری میشه و مقصد هم معنی مسیج رو متوجه است.
- Transport: وظیفه انتقال اطلاعات و برقراری connection با مقصد رو برعهده داره. وظیفه این که مقصد وقتی داده رو دریافت کرد Acknowledge بده یا نده. این که دوباره ارسال بکنه یا خیر و.. برای انجام کردن این کار ها یک سری header به مسیج اضافه میکنه.
- Network: لایه Network مسیج + Ht رو مشاهده میکنه. پس از مشخص کردن مقصد IP Routing میکنه. چون مقصد معمولاً Node بغل دست ما نیست. پس وظیفه اصلی مسیریابی است. هدر های Hn مثل Source IP و Destination IP رو اضافه میکنه.
- Link: وظیفه انتقال packet فقط به node بغلی رو داره.
- Physical: لایه ای هست که کل Message اپلیکیشن و هدر لایه های بالا رو دریافت میکند و سیگنال فیزیکی رو تولید میکند که بین Node ها انتقال باید تا به مقصد Message لایه Application برسد.

در مقصد:

- Physical: داده های بایتری رو دریافت میکنه.
- Link: کنترل خطا میکنه اگه اوکی بود به لایه بالایی میده
- Network: به Destination IP نگاه میکنه اگر خودش بود که میده لایه بالا وگرنه انتقال رو انجام میده
- Transport: هدر های خودش رو چک میکنه و اخر سر به Application میده
- Application: داده هارو دریافت میکنه

\*6: در این نگاه لایه ای از Application به Physical کدوم لایه بیشتری تاثیر از IoT رو میگیره و دستخوش تغییر میده؟

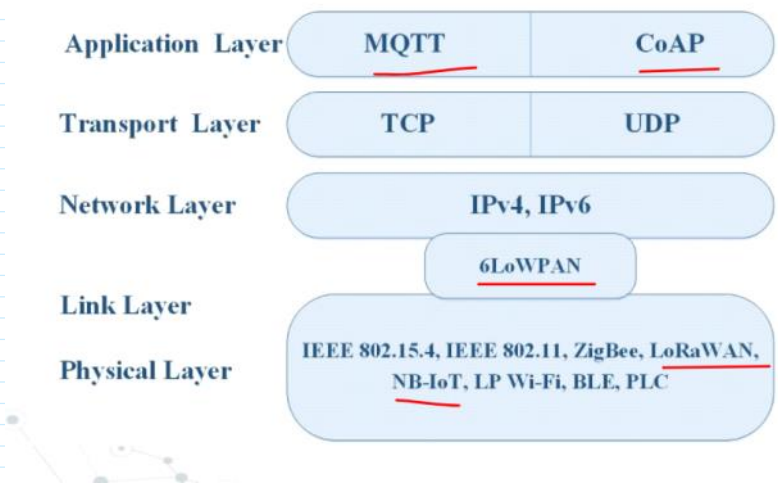
ما در اپلیکیشن IoT خیلی Abstract نگاه کنیم یک سری Things داریم که داده ها رو ارسال و دریافت میکنیم و یک سری سرور هم داریم در بخش Backend داریم که الگوریتم ها رو انجام میدهند. در دیتاسنتر ها مشکل انرژی نداریم چون جاهای ثابتی اند. Things مشکل انرژی رو داره که حتما Physical و Link رو داره. این 2 دستخوش تغییرند (در ذات IoT) مثلا Physical وقتی میخواد یه سیگنال رو تولید کنه مهمه با 1mW تولید کنه یا چی. لایه های Transport و Network هم دستخوش تغییر است. در Transport اگر بخواهیم Ack بگیریم. ممکنه این Ack رو حذف کنیم چون اگر داده رو ارسال کنه و منتظر باشه Ack بیاد مصرف انرژی بیاد بالا. اگر Ack رو هم حذف کنیم Reliability بد میشه. برای اونا یه سری کارهای دیگه میکنیم. مثلا مسیج رو چندبار ارسال کنیم. اما یه اپلیکیشن هم در سمت Things خواهیم داشت که با سمت سرور میخواد صحبت کنه. که اون هم دستخوش تغییره.

\*7: CDMA یعنی 3G.  
RPL و 6LoWPAN خاص IoT اند که جلوتر میبینیم.

\*8: در این درس ابتدا از لایه های پایین شروع میکنیم چون بیشترین تغییرات در اینجا هست. IEEE 802.15.4 برای فاصله متوسط (Medium Range) خوبه. از قبل IoT بوده و Customized شده.  
IEEE 802.11 میشه وای فای  
NB-IoT، LoRaWAN مخصوص IoT دیزاین شده.

با توجه به این که پروتکل IP برای IoT سنگینه از 6LoWPAN برای سبک کردن استفاده میشه.

موارد خاص IoT زیرشون خط کشیده شده:



-- پایان مقدمه

حالا میخواهیم درمورد 2 لایه پایین صحبت کنیم.

وقتی میگی IoT Access منظورمون Physical + Link هستش و از things تا اولین زیرساخت قرار داره.



ما یه سری Node داریم با x نشون دادیم. اینا وصل میشن به یه Base که در بلندی قرار داره و به Core شبکه وصله. در Core ما Datacenter های با پردازش بالایی داریم. در edge computing در همان base station پردازش انجام میشه که تاخیر کمتر بشه.

- \*12: از لحاظ Transmission Medium ما باید یک وسیله فیزیکی بین فرستنده و گیرنده داشته باشیم.
- Guided Media: یعنی Wire. Copper twisted pair در سیم های تلفن موجوده. Coaxial cable در کابل های تلویزیون مشاهده میشه.
  - Unguided Media: به کمک سیگنال الکترومغناطیس می باشد و محصور به یک محیط محدودی نیست. معمولاً Wireless transmission خطاب میشه.

- \*13:
- فایده استفاده از Wireless:
- نیاز به سیمکشی نداریم
  - اجازه Mobility داریم.
- ولی اگه جایی قابلیت Wire داریم بدون شک باید بریم سمتش.

چه لایه هایی بیشترین تاثیر رو از ارتباط بی سیم دارند؟ لایه فیزیکی. زیرا کلمه بی سیم از همان لایه فیزیکی آمده. یعنی بین 2 نقطه فیزیکی سیم نداریم. آیا فقط لایه فیزیکیه؟ وظیفه Data Link یکیش LLC بود که وظیفه تشخیص و تصحیح خطا رو داشت. یکی هم MAC که اگه یک مدیا مشتری داریم بتونیم کنترل کنیم Access به اون رو. یعنی مشخص شه



که کدوم node وظیفه ارسال اطلاعات رو الان داره.

ولی سوال الان اینجاس که آیا **Data link برای wireless swire** پیکه؟ خیر. **MAC** تفاوت داره. در سیم اگر 2 نفر همزمان سیگنال ارسال کنند هر 2 متوجه میشوند. زیرا سیم از یک متر تا حداکثر 1000 متر خواهد بود و توانایی تشخیص موجود است. ولی در wireless ممکنه 2 نفر متوجه collision نشن و گیرنده نتواند تشخیص بدهد که دیگری چه سیگنالی ارسال شده. برخورد در هوا اتفاق میافته ولی مثل این میمونه که یک موج سینوسی شخص A ارسال کرده که دامنه و فرکانس و فاز متفاوتی دارد و یک دیتا رو نمایندگی میکند. یک موج سینوسی هم شخص B ارسال میکند که آن هم دامنه و فرکانس و فاز متفاوتی دارد. این 2 موج سینوسی در هوا جمع میشوند. یعنی  $x_A(t) + x_B(t)$  دریافت شود و دامنه یک سیگنال دامنه دیگری را خراب کند. بر فرض اگر A یک دامنه دارد و دامنه B هم پایین است. دامنه B میاد دامنه A هم میکشه پایین و گیرنده فکر میکنه سیگنال صفر ارسال شده. در دریا هم اگر 2 موج مکانیکی به هم برسند. اگر هر 2 در پیک باشند همدیگر رو تقویت میکنند ولی اگر خلاف هم باشند جمع آن ها یک خط صاف میشه. در شبکه بی سیم هم رویداد و تشخیص متفاوتی دارد. در شبکه سیمی به راحتی از طریق اختلاف ولتاژ میتوان تشخیص داد. مثلا اگر یک اختلاف ولتاژی توسط A روی سیم وجود باشد و B هم یک اختلاف ولتاژ دیگری قرار دهد شخص A میتواند احساس کند و متوجه شود موج دیگری موجود است. ولی در شبکه بی سیم:



الان یک base station اون وسط داریم و 2 نفر در دو طرف مستقل هستند و از وجود سیگنال هم اطلاع ندارند. اما در Access point موجود در base station میتواند collision صورت بگیرد. به همین دلیل ما MAC سیمی و بی سیم متفاوتی داریم.

هرچه میریم در لایه بالاتر تاثیر بی سیم کمتر میشه. در این جا در مورد تاثیر بی سیم بر لایه های شبکه حرف زدیم و دیدیم در لایه های پایینتر تاثیر بیشتری میگذارد. قبل تر در مورد اثر IoT در دیزاین لایه ها گفته شد. در آن جا IoT یک دستگاه محدودی از نظر:

- مصرف انرژی
- قدرت پردازش
- حافظه

می باشد. بنابراین وظایفی که دارد بایستی سبک باشد.

**Digital Modulation (سوال محصل):** ما میتونیم توی سیم مثلا فقط 2 سیگنال ارسال کنیم که یکیش مثلا 10 کیلوهرتز باشه یکیش 11 کیلوهرتز. بعد گیرنده اگه 10 کیلوهرتز دریافت کرد 0 منطقی و 11 کیلوهرتز دریافت کرد 1 منطقی تفسیر کند. این که پارامتر های سینوس و کسینوس چگونه 0 و 1 رو نمایندگی کنه ← FSK (Frequency-Shift Keying)

14: در ارتباطات بیسیم یک مفهوم مهم فرکانس سیگناله. فرکانس نقش طلای ارتباطات بی سیم را دارد. در یک سیگنال متناوب فرکانس معنای عکس دوره تناوب را دارد. در صورت غیرمتناوب بودن ما آن رو به صورت مجموع سیگنال های پریودیک مینویسیم. که معروف ترینش سینوس و کسینوس بود (تبدیل فوریه). فوریه به ما کمک میکرد در مورد محتوای فرکانسی سیگنال ارتباط داشته باشیم حتی وقتی سیگنال متناوب نباشد.

سیگنال ها در فضای بی سیم در طیف الکترومغناطیس منتقل می شوند. 0 و 1 در میکروکنترلر ها از طریق اختلاف ولتاژ مثل 0 تا 5 ولت در نظر میگیریم. ولی اگر این پالس هارو در 2 سر آنتن قرار دهیم به صورت بی سیم انتقال نمایانند. ارتباط بی سیم یعنی اگر یک آنتن فرستنده داشته باشیم یک اختلاف ولتاژ در فرستنده قرار دهیم و آنتن گیرنده امواج الکترومغناطیس رو دریافت کند. برای انتشار این موضوع **فرکانس سیگنال** بسیار مهم است.

سیگنال های پالسی رو در تبدیل فوریه دیدیم که پس از تبدیل فوریه Low frequency بود. یعنی فرکانس های پایین نقش بیشتری داشتند. چون تبدیل فوریه یک تابع sinc رو تشکیل میداد.



\* ممکنه بگیم چون یهو از 0 به 1 میره باید فرکانسش زیاد باشه ولی چون این تغییرات ناگهانی و بیشتر اوقات بی تفاوته فرکانس پایین محسوب میشه (مدت تناوب هم وابستس در فرکانس).



برای این که سیگنال از یک آنتن به دیگری برسد فرکانس باید از یک حدی بیشتر باشد. اثرش رو در ادامه میبینیم:

15: در سیگنال های متناوب یک فرکانس دارد. اما سیگنال غیرمتناوب مجموعه ای از فرکانس ها را دارد. چون مجموع یک سری سیگنال سینوسی هستش که هرکدوم یک سری اندازه دارند. اون سینوس هایی که اندازشون بیشترین نقش در تشکیل سیگنال رو دارند معمولا فرکانس سیگنال غیرمتناوب رو همین سیگنال سینوسی های اصلی در نظر میگیریم. اصلی هم اونایی هستن که اندازه فوریه بیشتری دارند. در جدول این اسلاید دسته بندی رو میبینیم. در آخر هرچه به چند تراهرز نزدیک تر میشیم به visible light که نور هستش نزدیک تر میشیم. طول موج هم که داشتیم  $c/f$ . مسلما هرچه فرکانس بیشتر میشه طول موج کمتر میشه.

16: دو قانون حاکم انتشارات امواج در ارتباطات بی سیم استوار است:

- وقتی میخوایم یک سیگنال ارسال بشه ساینز آنتن مهمه. امروزه بدنه گوشی هم آنتن 4G، 3G، Wi-fi و ... رو داراست. ساینز در ارسال و دریافت اطلاعات مهم است. **قانون:** ساینز آنتن مورد نیاز برای good reception (propagate شدن داده) معکوسا متناسب با فرکانس است. یعنی اگر ساینز آنتن رو L بگیریم این L متناسبه با  $1/f$ . که f فرکانس سیگنالی است که میخواد در آنتن propagate بشه. به عنوان یک قانون سر انگشتی میگن ساینز آنتن برابر است با  $L = \lambda/4$ . با توجه به جدول اسلاید 15، مشاهده میکنیم از HF به بعد معقول است که آنتن داشته باشیم. آنتن های گوشی اواخر UHF هستش الان. اگر این قانون تنها قانون مورد نظر بود میرفتیم میچسبیدیم به تراهرز ها. ولی این تنها قانون نیست. بلکه یک قانون بازدارنده هم داریم
  - ما 2 نوع آنتن داریم. آنتن های directional یا جهتی مثل دیش ماهواره و nondirectional مثل غیر جهتی مثل گوشی موبایل که در هر جهتی قرار بگیریم ارتباط برقراره. طبق این قانون توان سیگنال دریافتی با یک آنتن بدون جهت متناسب با معکوس توان 2 فرکانس است. یعنی متناسب با  $1/f^2$  می باشد. یعنی هرچی فرکانس رو ببریم بالا تضعیف بیشتری و سیگنال ضعیف تر دریافت میشه.
- پس فرکانس ارتباطی یک جا بینابین مناسب است. طبق جدول 15 باید از VHF بایم بالا ولی خیلی هم نباید بایم بالا چون تضعیف زیاد میشه و مثلا 1 سانت 10 سانت میشه انتقال داد. به دلیل این 2

قانون ما یک رنج فرکانسی رادیویی داریم.

17: رنج فرکانس رادیویی قابلیت اینو دارن که یک سایز مناسب دارند و تضعیف زیادی هم ندارند. از بالای 30 گیگاهرتز تضعیف زیاد میشه از کمتر از 30 مگاهرتز هم طول آنتن زیاد میشه پس بین این بازه قرار میگیره.  
گل سرسبد این رنج فرکانس رادیویی نیز بین 300 مگاهرتز تا 4-3 گیگاهرتز است. به همین دلیل محدودیت به آن طلای ارتباط پی سیم میگویند.  
این فرکانس بسیار گران هم هست بنابراین در هر دولتی یک سازمان تنظیم مقررات موجود است که وظیفه تخصیص این فرکانس هارو دارند.  
این فرکانس ها یک رنج unlicensed دارند که باید پول بدیم بخریم. یک هزینه سنگین که اپراتور همراه اول و ایرانسل داره میده بابت خرید این لایسنس ها هستش. مثلا محدوده 2 مگاهرتز رو حول 2 گیگاهرتز میخرند. یعنی  $2\text{GHz} \pm 2\text{MHz}$

از 30 گیگاهرتز به بالا رو هم میگویند Milimeter wave که دیگه به درد ارتباط همه جهته نمیخوره.  
ارتباطات نظیر کنترلر تلویزیون عموما Infrared میباشد که از 300 گیگاهرتز به بالا میباشد. به همین دلیل از موانع عبور نمیکنند (با معادلات ماکسول میتوان اثبات کرد) و کنترلر باید حتما در جهت تلویزیون باشد.

18: فرکانس های رادیویی این خوبی رو دارند که انحنای زمین تأثیری روی آن ها نمیگذارد.



یعنی اگر فرض کنیم یک انحنای زمین داشته باشیم، این 2 node در دیدرس هم نیستند اما امکان برقراری ارتباط رو دارند. یعنی مثلا سیگنال یکیش بره بخوره به جو برگرده برسه به اون یکی.

\*در آخر این نکته حائز اهمیت است که با توجه به این که مصرف توان در IoT خیلی اهمیت دارد و قانون دوم در سیگنال های پی سیم، سعی بر این است که فرکانس های sub-GHz برای IoT انتخاب شود.



16: تضعیف (فانون دوم) به سه عامل بستگی دارد:

- محیط: وقتی در طبقه دوم هستیم و یک سیگنال خارج از ساختمان رد میشه، وقتی از چندتا دیوار عبور میکنه (بسته به متریال دیوار هم داره که چه فلز و آجر هابی استفاده شده) در موج تاثیر میگذاره.
- فاصله: هرچی فاصله بیشتر تضعیف بیشتر
- فرکانس: هرچه فرکانس بیشتر تضعیف بیشتر. درسته فرکانس های بالاتر به ما اجازه compact antenna داشته باشیم اما تضعیف زیاده.

توان سیگنال دریافتی ما متناسب است با معکوس مربع فرکانس.

17: نمونه تضعیف بسیار زیاد رو در نور میتونیم مشاهده کنیم. چون فرکانس فوق بالایی داره وقتی یک مانع موجود باشه دیگه عبور نمیکنه. در اینترنت های نسل آینده به کمک نور تلاش میشه که داده منتقل بشه ولی به دلیل فرکانس زیاد برای short range استفاده میشه.

18: ما وقتی میخوایم یک موج الکترومغناطیس در یک فضا انتشار کنیم باید از سازمان تنظیم مقررات لایسنس اشو بگیریم که پول سنگینی داره. مثلاً وقتی ایرانسل 40 مگاهرتز حول 1.2 گیگاهرتز بگیره. ماکسیمیم منهای مینییم فرکانس میشه پنهان باند. در شکل هم ببینیم Bandwidth از تفاضل frequency range حاصل شده.

یک سری فرکانس Unlicensed هم داریم (ISM). مثلاً وای فای حول فرکانس 2.45 گیگاهرتز و 5.8 گیگاهرتز هستش و آزاده. یک سری علت برای این که وای فای ضعیف میشه اینه که همسایه ها همگی حول یک فرکانس خاص کار میکنن و رو هم تداخل میندازن.

19: سه بحث مهم داریم:

- Coverage: سیگنال ما تا چند متر اونورتر امکان Communication داره. (تقسیم بندی LAN و WAN)
- Capacity: مستقل از coverage و فاصله میخوایم ببینیم ظرفیت چند بیت در ثانیه است.
- Latency: فرض کنیم 10Mb/s داریم فاصله 10 متره. برامون مهم نیست. میخوایم بگیم از نقطه A تا B در کل چقدر طول میکشه.

از نظر فرکانسی در قانده اول هرچی فرکانس رو ببریم بالا Coverage کمتر میشه. چون در فرکانس های بالا تضعیف بیشتر میشه (جلسه قبل). ساب گیگاهرتز برای IoT خوبه چون هم توان کمتره هم فاصله بیشتری ارسال میتونه بکنه. از 30 مگاهرتز تا 1 گیگاهرتز اون 30 مگاهرتز هم صرف نظر کنیم کلاً 1 گیگاهرتز داریم که خیلی محدوده.

ولی Capacity با افزایش فرکانس بیشتر میشه. چون پهنای باند بیشتری داریم.

Latency هم مثل Coverage هستش هرچی فرکانس کمتر تاخیر کمتر. یکی از دلایلی اینه که مثل دوجرخه های اسپرت هرچی چرخ دهنده شعاع کمتری داریم سریعتر انتقال میدیم. باید امواج الکترومغناطیس و معادلات ماکسول بررسی بشه تا دقیق علتش مشخص شه.

20: وقتی میگیم 700 مگاهرتز باید بدونیم زیر 1 گیگه پس چون بالای 30 مگه هست رادیویی و فاصله زیادی میشه ارسال کرد. ولی data rate کمه. تاخیر هم کمه ولی Ultra low نیست و برای تاخیر های نانوفانیه باید بریم فرکانس های بالاتر.

حالا میخوایم سیگنال رو ارسال کنیم. یک تعریف سیگنال نمایش فیزیکی داده است. دیتا یک مفهوم مجازی است که ما در مبنای 2 قرار داد میزنیم. حالا میخوایم در 2 سر آنتن ارسال کنیم. بایستی در یک هویت فیزیکی این دیتا منتقل شود. که امواج الکترومغناطیس می باشد. یکی از مهم ترین سیگنال ها سینوسی است که دامنه و فرکانس و فاز سه فاکتور مهم در آن می باشد. ما هرکدام رو میتونیم به نحوی کد کنیم و تحت نمایندگان 0 و 1 معرفی کنیم.

21: وقتی که سیگنال منتشر می شود. 3 موضوع در انتشار امواج اثر میگذاره:

- فاصله: وقتی فاصله بیشتر باشه تضعیف بیشتره
- محیط: یکیش Shadowing هستش. مثلاً اگر در سایه یک ساختمان قرار بگیریم (مشابه شکل) بخشیش بازتاب میکنه و بخشیش وارد میشه. با توجه به این که 2 بخش میشه هرکدام تضعیف شده رو میگیره. Scattering هم همینطور که مثلاً به تابلو راهنمایی میخوره به چندتا تقسیم میشه. Diffraction هم در کوه مشاهده میکنیم.
- فرکانس. هرچه فرکانس بالاتر تضعیف بیشتر.

یک سری مدل ها برای این مدل سازی تضعیف موجوده. یکیش

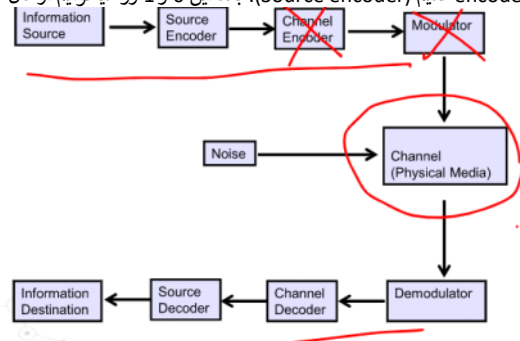
$$\text{Receiving power proportional to } 1 / (d \times f)^2 \text{ (ideal)}$$

هستش که نسبت توان دریافتی به فرستنده. یعنی اگر توان فرستنده 1 وات باشد، اون که در گیرنده دریافت میشه در رابطه فوق میبینیم که d فاصلس ولی در حالت real که محیط در نظر گرفته میشه بعضاً به جای توان 2 به توان آلفا میرسه. چون محیط هم اثر میگذاره. در ایده آل محیط خارج شهریه که موانع نداریم (مثل بیابون)

22: ممکنه یه سیگنال مستقیم به گیرنده برسه. ممکنه هم به چندتا جا برخورد کنه و بازتابش به گیرنده برسه. پس یه سیگنال از مسیر های مختلف ممکنه به گیرنده برسه.

23: ساختار Transceiver (ترکیب transmitter و receiver):

وقتی یک Information source مثل کیبورد داریم و میخوایم بهصورت بی سیم منتقل کنیم (کیبورد بیسیم) یا مثلاً درجه سد داریم که 4 حالت بسته و باز دارد. اولین کاری که میکنیم باید این داده هارو به 1 و 0 encode کنیم (Source encoder). بعد این 0 و 1 رو میخوایم ارسال کنیم. فرض کنیم 2 بلوک بعدی نباشه



آیا اینا مستقیم به سر آنتن میتوانن برن؟ نه. چون موج مستطیلی در تبدیل فوریه یک sinc میشه که فرکانسش پایینه و نمیتونه منتقل بشه.

برای راه حل یک ماژولی تحت عنوان ماژولاتور نیاز داریم که 0 و 1 رو دریافت میکنه و یک سری امواج سینوسی تولید میکند که این امواج در فرکانسی هستن که قابلیت radiate رو دارند (فرکانس رادیویی).

در گیرنده اولین بلوک میشه demodulator که سینوسی دریافت میشه و باید بتوانه از علایم داخل سینوس که تضعیف شده است متوجه بشه که چه 0 و 1 هابی ارسال بشه. پس بین ماژولاتور و دیمازولاتور باید یک قرارداد موجود باشه. ماژولاتر سینوس هابی که ارسال میکنه در محدوده رادیویی قرار داره و در channel که بی سیم است منتقل میشه.

ترکیب ماژولاتر و دیمازولاتر که در هر communication task در هر فاصله ای استفاده میشه رو Modem میگویند. چون ما فرستنده خاص نیستیم و گیرنده هم نیاز داریم هم ماژولاتور نیاز داریم هم

24: ماژولاتور یک بخشی از لایه فیزیکی است.

25: با توجه به این که دامنه و فاز و فرکانس در سینوس اهمیت دارد. میتونیم با ASK بیایم بگیریم که گیرنده بر اساس دامنه سیگنال سینوسی مشخص کند که 0 ارسال شده یا 1. یک نمونه دیگه اینه که به فرکانس نگاه میشه و فرکانس بیشتر به 1 داده میشه و فرکانس کمتر به 0 (FSK). یا میشه از PSK استفاده کرد که با تغییر فاز باشه. مثلا فاز 180 درجه برای 1 باشه و فاز 0 درجه برای 0.

\* در FSK توجه کنیم هر دو فرکانس در محدوده رادیویی هستش.

\* برخی فکر میکنند در Digital Modulator سیگنال دیجیتال ارسال میشه در هوا. اما الان میدانیم سیگنال پیوسته میباشد و تنها تعداد سیگنال هاش گسسته است و تنها نمایندگی یک سیگنال دیجیتال رو بسته به ASK,FSK,PSK دارد.

23: حالا Channel encoder برا چیه؟ وقتی ASK و FSK و PSK داریم ممکنه دچار خطا بشه. باید بتونیم این خطا رو تشخیص بدیم و Channel encoder میاد یک سری 0 و 1 هارو عامدانه به دیتا اضافه کنیم و بتونیم تشخیص بدیم Demodulator دچار خطا شده یا نه. ساده ترینش Parity Check هستش.

25: FSK خطای کمتری نسبت به بقیه دارد ولی چون فرکانس زیادی میخواد گرونه. PSK هزینه بالایی نداره و تا حدی در برابر خطا مقاومه.

میشه یه کاری هم کرد مثلا 4 تا دامنه در نظر بگیریم که اگر مثلا دامنه اولی ارسال شد 00 در نظر بگیریم دامنه دومی رو 01، دامنه سومی رو 10 و دامنه چهارمی رو 11 ارسال میکنیم. یا میتونیم 8 تا فرکانس در نظر بگیریم که بتونیم 3 بیت رو انتقال بدیم. پس به طول کلی m تا سیگنال در نظر بگیریم  $\lg(m)$  تا میتونیم بیت ارسال کنیم. با این کار بیت ریت از symbol(baud) rate بیشتر میشه.

مثلا اگر بخوایم 1024 تا داده رو در یک سیگنال ارسال کنیم باید 10 تا دامنه (یا فرکانس یا فاز) داشته باشیم و با این کار بیت ریت 10 برابر baud rate میشه.

**حالا چرا همینطوری زیاد نمیکنیم؟** چون خطا هم بیشتر میشه. چون بازه ها به هم نزدیک میشه و دقتمون کمتر میشه.

یک سوال مهم اینه که ماژولاتور پروتکل چه می باشد. این که بلوتوث، وای فای و 3G و 4G هرکدام از چه ماژولاتور استفاده میکنن که هرکدوم روی دقت، خطا و هزینه تاثیر میگذارند.

پس از physical layer issue به Communication Layer می‌رسیم.

2: Communication criteria ویژگی‌هایی که باید هنگام select کردن و یا برخورد با smart object در نظر بگیریم رو داره. برای انتخاب بایستی معیار داشته باشیم.

3: هنگام مقایسه انتخاب‌های متعددی داریم. تعداد زیادی فناوری communication موجود است. هم برای IoT هم برای غیر آن. بسیار گسترده هستند و حتی یک جاهایی ممکنه انتظار ادغام داشته باشیم. یعنی در یک جا 2 تا پروتکل استفاده بشه. قبل وارد شد جزئیات مهم است که در مورد معیار یک جمع بندی داشته باشیم. جدا از تنوع تکنولوژی‌های ارتباطاتی IoT از چه معیارهایی متفاوت هستند و میتونیم تمیزشون بدیم.

4: این 6 تا معیار برای تأثیر گذاری انتخاب رو داریم.

- Range: این که چه range رو پوشش میده. بسته به کاربرد فاصله سنسور تا gateway چقدر باشه.
- Frequency bands: باند فرکانس‌های Unlicense باشه (هیچ هزینه ندیم) که از یک جهت هزینه کم میشه ولی ممکنه تداخل بشه چون انحصار ما نیست
- Power Consumption: محدودیت مصرف توانی. آیا خیلی Critical عه که که Ultra low باشه یا معمولی.
- Topology: توپولوژی شبکه که اون پروتکل از چه توپولوژی‌هایی Support میکنه و کدام مناسب تره.
- Constrained Devices: دیوایس‌ها از نظر مصرف توان، از نظر حافظه و از نظر قدرت پردازش اهمیت دارن.
- Constrained-node networks: مستقل نیست. شبکه‌هایی که خاص Constrained-node دیزاین شده. یک سری پروتکل داریم که از ابتدا مخصوص IoT نوشته شده. یک سری هم داریم که customize شده مثل bluetooth و ...

5: Short Range: ما زیر 1 متر تا چند ده متر فاصله دیوایس داریم.

- کابل سریال و USB
- Parallel cable
- در قسمت wireless که مواردش اومده.

Medium Range: چند ده متر تا چند صد متر (تشکیل LAN).

\*PLC(Power Line Communication): خطوط برق که سراسر جغرافیا کشور موجود هست. دیتا ریت بسیار پایین و نویز بسیار بالاست. این اختلاف ولتاژ به صورت برق AC و در فرکانس کاری 50 هرتز می باشد. به همین دلیل Radiate نمیتونه بکنه. چون سیم هاشون هدفش برای رسیدن Power بوده نه Communication نمیشه سینوس‌های با فرکانس بالا رو ارسال کرد.

6: Long Range: بیشتر از یک مایل (1.6 کیلومتر) که به آن WAN هم میگن.

LPWA: شبکه‌های مخصوص IoT در range زیاد.

8: معیار دوم frequency bound هستش که مفصل در موردش بحث شده.

Licensed: هزینه بالا مثل شبکه‌های سلولی که اپراتور‌ها اون فرکانس رو خریده. اپراتور‌ها 2 منبع هزینه‌ای بالا دارن یکیش خرید لایسنس فرکانس و دومی ایجاد زیرساخت (ساخت دکل و نیروهای انسانی و ...)

Unlicensed: در کارهای پزشکی، علمی و صنعتی استفاده میشه که در کشور‌های مختلف فرکانس‌های مختلفی جهت ISM قرار داده شده.

9: در Licensed‌ها یوزرها باید Register بشه مثل سیم کارت که میخریم. در IoT هم همین. وقتی فرکانس‌های لایسنس دار میریم باید دیواس رجیستر بشه و هزینه پرداخت بشه.

حسن فرکانس‌های لایسنس دار تضمین میکنه که انحصار داریم و کس دیگه‌ای اگر استفاده کرد میشه شکایت کرد و جریمه بشه. پس **کیفیت بیشتر** رو میتونیم تضمین کنیم.

کیفیت سرویس:

- data rate: مثلاً یک camera داریم باید کاملاً انتقال پیدا کنه.
- delay: دیتا باید در 2 میلی ثانیه بره برگرده.

\* وقتی میگیسم ISM این صرفاً یک اسمه منظور این نیست که تمام صنعتی‌ها میاد روی ISM. مثلاً پزشکی از راه دور قطعاً فرکانس لایسنس داره چون حساسیت داریم.

ولی یک کاربرد صنعتی میشه مثلاً سوله رو مثال زد در یک جاده مخصوص و مثلاً میخواد از این ور خط انتقال (تولید) تا اون ور خط تولید یک communication داشته باشه. مثلاً اگر در خودرو در تولید قرار گرفت یک ربانی که مثلاً میاد یک قسمتی رو پیچ و مهره میکنه بهش بگه 2 تا داره میاد تا پیچ اونارو محکم کنه. در محیط محدود بیشتر به درد LAN میخوره.

10: در Ulicensed به این معنی نیست که Regulation نداریم. پول نمیدیم ولی باید قواعد رو رعایت کنیم. مثلاً اگر بخوایم یک محصولی رو وارد آنالایسنس وارد کنیم اولین چیز که میگن اینه که سازمان تنظیم مقررات کشور بره certificate بده که قوانین رو رعایت میکنه بعد وارد کشور بشه. **حالا این Regulation‌ها شامل چه چیزایی است؟**

- Transmit power: مثلاً میگن در این باند فرکانسی نهایتاً 100 میلی وات میشه سیگنال ارسال کرد
- Duty Cycle: مدام نمیتوان داده ارسال کرد. یعنی یک واحد زمانی ارسال کن ولی 99 تای دیگر داده ارسال نکن.
- Channel Bandwidth: خارج باند اجازه نداریم
- Channel Hopping: وقتی از یک کانال میریم کانال دیگه اونم محدوده

پس آنالایسنس معنیش نیست حالا که لایسنس نداره با هر پهنای باندی بتوانیم استفاده کنیم.

11: فرکانس روی propagation و ماکسیمم رنج تأثیر میگذاره. هرچی فرکانس بالا بریم تضعیف بیشتری و در نتیجه رنج کمتری رو میشه با یک توان ثابت ارسال کرد. مثلاً آگه 1 وات در اختیار داشته باشیم با فرکانس پایین رنج بیشتری داریم تا فرکانس بالا.

پس فرکانس بالا برای کاربرد‌های رنج پایین هستش و فرکانس‌های آنالایسنس Sub-GHz با یک توان ثابت فاصله بیشتری رو میشه پوشش داد.

12: ولی مشکل Sub-GHz نرخ داده پایینه و نمیشه داده زیادی ارسال کرد.

13: بسیاری کاربرد‌ها در این فرکانس‌های سبز رنگ قرار دارن.

14: (دقیقه 30).

مثلاً در یک خانه هوشمند به برق شهر وصل داشته باشیم و گره هامون توان داشته باشن. اما یک سری گره‌ها Battery Power هستن و اون موقع انتخاب Communication Technology بسیار مهم میشه. چون Communication Interface نقش زیادی روی توان میذاره.

15: معیار 4م توپولوژی شبکه است. توپولوژی معنیش این است که ارتباط گره‌ها با هم به چه صورت است و انتقال داده به چه نحوی انجام میشه.

در توپولوژی star یک base station (Access Point) مرکزی داریم که همه گره‌ها به اون وصل میشن.

در Mesh یک سری نقاط تحت عنوان Mesh Router داریم که گره‌ها حول این Mesh Router یا ارتباط مستقیم برقرار میکنن یا از طریق این مش‌ها به هم وصل میشن.

در Peer To Peer گره‌ها با گره‌هایی که در رنج خود قرار دارد ارتباط مستقیم برقرار میکنند.

برای Long Range و Short Range کلا Star Topology رایج ترینشه چون یک access point در یک نقطه جغرافیایی قرار میگیره و یک پوشش ارتباطی به وجود می آید. اگر چه بعضی جاها مش هم داریم.

16: دو بحث دیگه داریم که در اون 4 تا معیار دیده میشن ولی تحت معیار های مستقل هم به خاطر تاکید نگاه میکنیم.

یکیش دیواس های کانسترنیت هستش که توی کاربرد آیا دیواس کانسترنیت داریم؟ یعنی چی؟

در RFC 7228 برای این که یک تعریف واحدی از Constrained Node بده. منابع محدودی داره و روی ویژگی های شبکه تاثیر میذاره.

3 تا کلاس براشون در این RFC 7228 تعریف شده:

- کلاس 0: مثل گره Push button که battery-powered هستن و میخوان 1 بایت داده رو ارسال کنه. مموری کمتر از 10 کیلوبایت داره و حافظه مربوط به اون هم کمتر از 100 کیلوبایت
- کلاس 1: مموری حدود 10 کیلوبایت و استوریج حدود 100 کیلوبایت داره
- کلاس 2: رم و مموری بالای 50 کیلوبایت و پردازش و استوریج بیشتر از 250 کیلوبایت

17: در این جدول جزئیات بیشتر رو میگه.

\* در کلاس 0 نمیشه IP Stack رو آورد یعنی پروتکل TCP/IP رو روش نمیشه بالا آورد و مکانیزم امنیتی نمیشه روش آورد.

\* در کلاس 1 یک نسخه سبک از IP Stack رو میشه پیاده سازی کرد که در فصلای بعدی میگیریم.

\* در کلاس 2 قشنگ میشه IP stack رو پیاده سازی کرد.

18: شبکه های Constrained-node: شبکه هایی که برای Constrained-nodes دیزاین شده و فرضش این بوده ک برای گره های تعریف بشه و انتقال داده اتفاق بیافته که گره ها محدود هستن.

\* نسخه 802.11ah نسخه کاستومایز شده برای IoT هستش.

19: یک اسم دیگه هم برای Constrained-node استفاده میشه ک Low-power and lossy network میگن. (LLNs)

20: آخرین بحث در معیار این است که این شبکه ها وقتی کاستومایز IoT بشن چندتا پارامتر رو باید بررسی کرد:

- Data rate و Throughput چقدره

- Latency چقدره

- Payload و Overhead چقدره

(دقیقه 45)

21: در IoT از 100 bps داریم. که خیلی پایینه ولی به درد خیلی کاربرد میخوره. و تا چند مگابیت در ثانیه هم داریم که به درد LTE میخوره. اما معمولا Actual Throughput کمتر از data rate است زیرا بعضی وقتا داده ها برای هدر برخی برای channel coding جهت تصحیح خطا اضافه میشود. بنابراین از 100 مگابیت مثلا میگیریم داریم ممکنه 20 مگابیت اش داده باشه. معمولا جهت ایجاد تفکیک به اون بیت داده میگن throughput و به bps میگن data rate.

22: مثلا فرض شود یک پروتکل داریم که اگر در گیرنده اگر دچار خطا شویم تصحیح کنیم. آنگاه از 150 kbps دیتا ریت واقعی برسه به 75 kbps.

23: در خیلی کاربرد ها که میخوایم acceess technology و communication technology رو میخوایم تعیین کنیم latency مهمه. اگر packet دچار loss بشه باید retransmission انجام بدیم ولی همین کار delay داره و باید مشخص بشه برای اون کاربرد تاخیر تا چه حدی جایزه و همین باعث بشه آنالایسنس ها در انتخابمون قرار نگیره.

24: در مورد Overload و Payload مثال زده شد. مثلا به عنوان یک requirement عه IPv6، Maximum Transmission Unit ما 1280 بایت است. حالا اگر لایه فیزیکی اجازه نده 1280 بایت ارسال کنیم باید fragmentation انجام بشه که در لکچر بعدی در موردش صحبت میشه. اون موقع اگر بخوایم به 10 تا 128 بایتی تقسیم کنیم مشکلاتی نظیر تاخیر ایجاد میکنه چون باید بشکنه و از اون ور سر هم کنه. هنوز استفاده از IP مخصوصا در کلاس های 0 و 1 یک Open topic هستش و کلی مسائل حل نشده داره.

چرا اصرار داریم روی IP؟ چون همه گیر هستش و اگه بشه سازگار با IoT کنیم بهتره تا یک پروتکل جدید.

26: نکته مهم اینه که در خیلی از کاربرد های IoT معمولا small payload size داریم یعنی سائز داده های ارسالی کمه. در حد 1 تا 10 بایت. اطلاعات کمی داریم. هرچند camera تعداد بایت زیادی داره ولی عموم دستگاه های IoT به صورت small payload size هستن. در خیلی از کاربرد ها داده های املاح و رطوبت در 3 تا 5 بایت میشه کد کرد.

در بسیاری از تکنولوژی های IoT در لایه فیزیکی payload size کوچیکی رو Offer میدن. میگن کلا 127 بایت رو در MTU ساپورت میکنن مثلا. یا در LoRaWAN تا 250 بایته. چرا سائز لایه کم گرفته شده؟ یکی از مهم ترین دلیل هاش Loss زیاد است. در شبکه های معمول Loss خیلی کمه چون در فرکانس لایسنس دار هستیم و از لحاظ پردازش بسیار خوب هستیم و تا 1280 بایت هم میشه فریم رو ارسال کرد. ولی در شبکه ای که Lossy می باشد فرض شود در بیت آخر از 1000 بایت (8000 امین بیت) خطا ایجاد شود. در این صورت نمیدونیم کدوم بیتنه و باید retransmission انجام بشه. پس حجم بسیاری داده باید دوباره ارسال و تولید بشه و هم شبکه شلوغ میشه هم باتری مصرف میشه. از طرفی چون در آنالایسنس هستن data rate کمی دارن و زمان زیادی طول میکشه 1000 بایت ارسال بشه.

بر فرض اگر فرض کنیم 100 bps شبکه داریم. انتخاب 100 بایت 100 بایت بهتره یا 1000 بایت 1000 بایت؟ اگه 1000 بایت انتخاب کنیم یعنی فریم هامون 1000 بایتی هستش و گیرنده باید کل 1000 بایت رو یک جا unpack کنه و بده لایه بالاتر. با توجه به 100 bps زمان زیادی طول میکشه و اگر loss بشه کلی زمان دوباره طول میکشه ولی اگر small payload size داشته باشیم راحت تر میتونیم دوباره ارسال کنیم.

پس Small payload size به 2 دلیل ساخته شده:

- دیتا ریت پایین

- قراره load time over the air داره. یعنی وقتی 1000 بایت بسته بندی بشه فریم ها زمان زیادی طول میکشه و در هوا زمان زیادی هستش. در صورتی که ما duty cycle داریم و در شبکه آنالایسنس نمیشه رگ باری ارسال کرد و باید یک فاصله رو ارسال نکنیم

در صورتی که مثلا ما در ethernet همیشه گفت payload size تا 1500 بایت رو ساپورت میکنه. البته خب در لایه فیزیکی بیت بیت میره ولی گیرنده وقتی 1500 بایت تشکیل شد میده لایه بالاتر.

7: در این شکل میبینیم که Low Power WAN (LPWAN) از شبکه های سلولی هم range بیشتری دارد که باید سوال بشه که چرا انقدر به خودمون زحمت میدیم که رنج بیشتری رو پوشش بدیم. یکی از دلایل اینه که در بحث IoT علاوه بر Low Power برای ما Low Cost اهمیت زیادی دارد. چون مثلا در یک کشاورزی هوشمند وقتی میخوایم یک سری سنسور قرار بدیم. شبکه آن نباید شبکه گرانی باشد. شبکه ای باشد که بابت دیتا به اپلیکیشن هوشمند بره و در آن یک تصمیم گیری بشه و برگرده به Actuator اگر این هزینه زیاد بشه کسی به سمت پیاده سازی IoT نمیره علارغم مزیت هایی که دارد.

یکی از این هزینه ها اینطوری پایین میاد که در فرکانس آنالیزنس باشه. روش دیگه که Operator ها نیاز به زیرساخت کمتری داشته باشند. حالا جزئیات رو میبینیم جلوتر ولی یکی از مهم ترین دلایل این که برای range بالاتر میریم سبب کاهش هزینه Operator ها و End User میشه.

14: حالا وارد جزئیات میشیم و در مورد 802.15.4 صحبت میکنیم که در دسته short range ها می باشد. حدود 300 کیلو بیت در ثانیه data rate دارد و مصرف توان متوسط و low power ای دارد.

15: هدف گذاری 802.15.4 Low Cost هستش. پس متوجه میشیم در بحث فرکانس ارتباطی باید سراغ فرکانس های آنالیزنس رفته باشه. اگر short range باشه که infrastructure خاصی نمیخواد ولی اگر wide area بود هزینه ها علاوه بر فرکانس آنالیزنس بایستی infrastructure ارزونی رو هم داشته باشه.

نسخه های 802 دو لایه پایین رو در IEEE تعریف میکنن. مثلا Ethernet میشه 802.3. که کلا در مورد Physical و data link layer صحبت میکنن.

حال ما یک Task group در 802.15 داریم که در مورد personal area network (short range) ها رو توضیح میدن. در 802.15.4 علاوه بر فرکانس آنالیزنس و رنج پایین بایستی بسیار ساده باشند و پیچیدگی نداشته باشند. با کاهش پیچیدگی سبب میشه با هزینه کمتری بشه تولیدش کرد.

16: یکی از کاربرداش این است که در toys and games کنسول های بازی ازش استفاده میکنن. در Home Automation میشه HVAC که رطوبت محیط رو میخواد تنظیم کنه از این پروتکل استفاده میشه البته رقیب هم داره.

17: هرکدام از پروتکل ها و استاندارد ها یک چرخه عمر (Life cycle) ای دارند. البته لزوما به این معنی نیست که در انتها از بین میره. همونطور که در اسلاید میبینیم بعد هرکدام یک سال وجود داره که backward compatibility داره. یعنی 2006 میتونه 2003 رو بفهمه ولی لزوما 2003 نمیتونه 2006 رو بفهمه.

18: نکته مهمی که نیاز است در این مرحله بدونیم که این 802.15.4 فقط physical و mac layer است و نمیتواند یک شبکه و protocol stack کامل باشه. بقیه پروتکل ها مثل ZigBee که یک پروتکل استک هستند از 802.15.4 در لایه های پایین استفاده میکنن. مثل TCP/IP که در لایه پایینی از 802.3 که ethernet است استفاده کند. معروف ترین Zigbee هستش که از 802.15.4 استفاده میکنه.

19: اولین بار میبینیم ZigBee در 2004 اومده که یک سال بعد از زمانی که 802.15.4 اومده.

20: فهمیدیم Protocol Stack میشه تمام لایه های شبکه رو داره. منظور از vendor specific یعنی یک Vendor ای بر اساس کاربردش یک اپلیکیشن خاصی درست کرده.

21: Alliance منظور یک گروهی متشکل از شرکت خصوصین که برای Interoperability یک Alliance تشکیل میدهند که Protocol stack رو نهایی و استاندارد سازی میکنند.

23: ZigBee یک نسخه دیگری تحت عنوان ZigBee IP دارد. در ZigBee عادی لایه IP رو در بالاش نداره. ZigBee IP یکی از کاربرداش Smart Energy است که برای مدیریت انرژی و اندازه گیری هوشمند می باشد. وقتی میگی Smart Energy یعنی در لایه اپلیکیشن یک سری جزئیات تعریف شده.

24: به دلیل overhead ای که IPv6 زیادی داره یک Adaption layer گذاشتیم که کمش کنه که جلوتر مشاهده میکنیم.

26: در لایه فیزیکی میدانیم با چه سوالاتی مواجه هستیم. یکیش اینه که wireless است یا با سیم. فرکانس کاری چه هست. چه range ای دارد. با دریافت رنج فرکانسی مصرف توان و محدوده پوشش رو میشه یک تخمینی زد. یک سوال دیگه اینه ماژولیشن اش به چه صورته و چطوری 0 و 1 هارو در هوا منتشر میکنه.

28: در ماژولیشن بر اساس یک نوع یک نوع ماژولیشن خاص که به آن DSSS میگن کار میکنه. جزئیاتش رو نمیشه ورود کرد. کلیتش اینه که یک روش CDMA است که یوزر ها نه در زمان و نه در زمان فرکانس از هم جدا بلکه در حوزه کد از هم جدا.

29: همانطور که میبینیم از 2006 به بعد بیشتر بر اساس PSK و ASK ساخته شده.

- OQPSK: از 4 تا فاز مختلف استفاده میشه. یعنی مثلا 0 و 90 و 180 و 270 درجه. حسنش اینه که من ماژولاتور 4 تا سیگنال در ورودی داریم و 2 بیت 2 بیت ارسال میکنیم. در اینجا data rate 2 برابر Symbol rate هستش.
- BPSK: به صورت باینری هستش یعنی 2 تا سیگنال داریم و 1 بیت 1 بیت ارسال میکنه.
- ASK: اینم از نوع باینری هستش.

31: یکی از نکات در لایه فیزیکی فرمت packet و فریم در لایه فیزیکی است. PSDU در واقع که 0 تا 127 بایت تعریف شده است. هدر های لایه بالایی است که MAC layer به لایه فیزیکی تحویل میده و حدود 1000 بیت دریافتی رو میده به ماژولاتور. لایه فیزیکی در گیرنده باید بداند شروع این فریم از کجاست.

Preamble: به داده هایی گفته میشه که فرمتش از قبل معلومه. پیش درامدی است که میخوایم به طرف اعلام کنیم. مثلا بین ارتباط بین انسان ها وقتی دست رو میریم جلو یک حالت preamble ایست که میخوایم بگیم سلام میکنیم و اونم دستش رو میاره. پس یک فرمتیه که بین فرستنده و گیرنده توافق میشه و وقتی ارسال میکنیم گیرنده میفهمه قراره داده ارسال بشه. همونطور که از Frame length از اسمش معلومه تعیین میکنه از بین 0 تا 127 بایت چه حجمی دارن و چون از 0 تا 127 هستش یک بایت داره.

33: یکی از مهم ترین چیزایی که باید در لایه mac layer ببینیم اینه که توپولوژی های آن چیا هستن. در 802.15.4 ما سه نوع توپولوژی star, mesh, peer-to-peer رو داریم. در توپولوژی mesh 2 تا node میتونن به هم وصل بشن هم میتونن به یک mesh router وصل شن و ارتباط با اینترنت براشون فراهم بشه.

34: در mac layer عه 802.15.4 دو نوع Node داریم. چرا 2 دسته؟ یکی از دلایلیش اینه که ما بتوانیم node هایی داشته باشیم که هم low cost باشن هم انرژی کمتری مصرف کنن. در مواقعی که ما نیاز نداریم full function باشیم میتوانیم هزینه و انرژی رو کمتر کنیم.

- FFD: تمام جزئیات 802.15.4 رو دارن.
- RFD: فانکشن های کمتری رو ساپورت میکنن

35: برای این که Personal Area Network داشته باشیم، باید یک FFD داشته باشیم که بهش PAN Coordinator میگن. FFD\* ها با هر دستگاه دیگری میتوانن ارتباط برقرار کنن ولی RFD ها با خودشون نمیتونن و فقط میتونن با FFD ها ارتباط برقرار کنن. اگر داده ای برای ارسال به هم داشته باشن باید از طریق FFD یا PAN Coordinator با هم صحبت کنن.

36: (دقیقه 15)

37: یک شبکه 802.15.4 در آخر مشابه این شکل هستن. به FFD ها ZigBee Router هم میگن چون packet هارو میتونن بین گر ها برقرار کنن. به کل این شبکه که یک Coordinator دارن میگن PAN که یک ID هم داره. حالا میشه یه PAN دیگه در کنارش باشه که بخواد ارتباط برقرار کنه.

- 38: در حالت کلی 4 تا تسک رو انجام میده MAC در 802.15.4:
- یک beacon که مشخص میکنه Coordinator چگونه Select شود.
  - انتساب دستگاه ها به PAN
  - امنیت
  - اگر خطایی در ارتباط وجود داشته باشد 1 و 0 جاشون عوض شد بتونیم تشخیص بدیم.

- 39: ما 4 نوع frame داریم :
- Data frame: انتقال داده رو مدیریت میکنه.
  - Beacon frame: همیشه PAN Coordinator منتشر میکنه که یک سیگنال مشخص است. مثلا Coordinator میگه Hello من فلانیم Coordinator فلان شبکه و با این کار اعلام هویت میکنه

یکی دیگه از کاربرد های beacon هم اینه که Coordinator مدام سیگنال beacon رو میفرسته و علتش CSMA هستش. یعنی یوزر ها اول گوش بدن اگر چیزی سنس کردن اون موقع حق ارسال دارن و اینطوری collision رخ نمیده ← در اینجا slot بندی هم میشه و علاوه بر sense کردن وظیفه slot بندی هم میکنه که یوزر ها فقط اول slot ها حق ارسال داشته باشند ← Slotted Aloha برای سنکرون شدن یوزر ها و تشخیص slot ها فاصله بین Hello کردن Coordinator رو به 10 تا slot مثلا تقسیم میشه.

- Acknowledgement: اگر یک داده رسید به گیرنده بگیم که ACK بفرسته اون فریم ( دقیقه 24)

40: کل 127 بایت در اختیار MAC است. MPDU همیشه کل دیتایی که در MAC تولید میشه و میدن به Physical layer. بسته ای که در مک درست میشه سه بخش داره:

- MHR
- MSDU: سرویسی است که MAC به لایه بالا میده. و لایه های بالا تر انقدر داده میتونن در اختیار MAC بدن که بده به لایه فیزیکی.
- MFC: فوتر نقش Logical link control دارد که یک سری داده 0 و 1 اضافه میکنه که گیرنده قابلیت تشخیص خطا رو داشته باشه.

- 41: این جا جزئیات بیشتری از فریم مک میبینیم:
- MHR: 4 تا 20 بایت برای Addressing هستش که مشخص میکنه مقصد و مبدا کجان. حالا این که 1 بایت یا 2 بایت اختصاص بدیم بستگی به کاربران داخل PAN می باشد. در حالت 4 بایتی 2 باید سورس 2 بایت مقصد و برای 20 بایتی هم باز نصف نصفه.
  - Sequence number: مثلا ما گیرنده ایم و هر بسته دریافتی رو میخوایم ACK بدیم. باید اول بگیم فلانی من بسته رو دریافت کردم. کدام بسته؟! باید یک شماره گذاری داشته باشیم که بعد به آخر رسیدیم دوباره reset میکنیم.

43:

- Frame control: یکی از کاراش اینه که نوع frame رو مشخص میکنه. ( از اون 4تا). و Addressing mode هم مشخص میکنه که 4 تا 20 بایت کدومشه

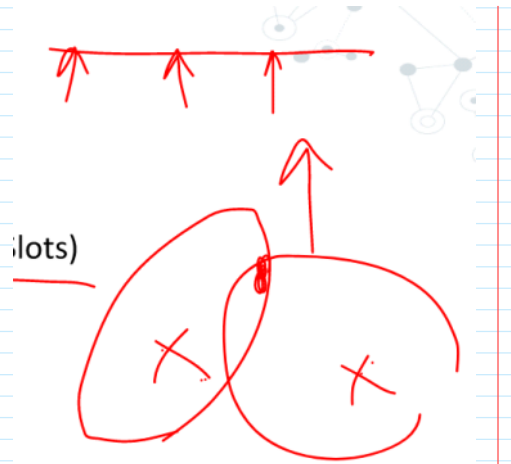
44: در Frame control 3 بیت اولش نشون میده نوع Frame چیه. با وجود این که 4 تا داریم 3 تا بیت میگیریم چون در آینده ممکنه کاربرد جدید بیاد و یک فریم جدید اضافه بشه. در این مواقع نباید پروتکل از بین بره بلکه بیت هایی رزرو نگه میداریم. در بخش ACK اش که 0 باشه فرستنده میگه ACK نمیخوام ولی 1 باشه میگه میخوام وقتی ACK داشته باشیم از طرف گیرنده دیگه دیتا مهم نیست

47: چرا برای beacon frame ها هم sequence number داریم؟ چون مطمئن بشیم 2 تا beacon با هم فرق دارن و اگر یکی اون وسط lost بشه و ما beacon سوئی رو دریافت کردیم فکر نکنیم اینا 2 تا متوالی ان و اشتباه slot هارو تقسیم کنیم. Superframe specification هم برا وقتی که یک بسته ای خیلی حیاتی هستش و نمیخواد منتظر بقیه بمونه برا اونا یک slot رزرو میشه.

50: ما سعی میکنیم تا حد امکان هدر هارو کوچیک کنیم. از این رو در 802.15.4 ما یک short address هم داریم که 16 بیتی و در 2 بایت آدرس رو به جای 20 بایت اختصاص میدیم که محدودیتش اینه که مختص به یک PAN هستش.

51: CA: ما وقتی ارتباط سیمی داریم وقتی فرستنده ها داده ارسال میکنن میتونن متوجه بشن که collision رخ داده ولی در پی سیم نمیشه:





از این رو در پی سیم سعی شده Collision Avoidance رخ بده که یکیش همین slot بندی است.

53: همونطور که گفته شد در Slotted CSMA بین 2 تا beacon رو مثلا 16 تا اسلات بندی میکنیم:



بعد از این ما بین یوزرا رندم میتونن به اسلات بردارن ولی به سرپاش GTS عه که رزرو شده برای کارای حیاتی.

54: در این اسلاید هم توضیح داده میشه که 16 تا اسلات بین superframe تخصیص داده میشه که اولین slot برای beacon هستش.

56: وقتی که یک داده میخواد به coordinator بفرسته از CSMA استفاده میکنه و ارسال میکنه.

اما وقتی که coordinator میخواد به دستگاه بفرسته این اول announce میکنه در beacon بعد دستگاه مورد نظر از طریق یکی از اسلات ها میگه من آمادم دریافت کنم و اون موقع coordinator از CSMA-CA استفاده میکنه برای ارسال

57: وقتی هک که Non-beacon هستیم slotted نیست و هر موقع sense کردیم ارسال میکنیم که تضمینی نیست.



- 1: به طور کلی، 802.15.4 نسخه هاش با سال اون نسخه ای که release میشه معرفی میشه. یک موقع هابی هم یک سری حروف میگذارن. اینطوری میگن که IEEE یک Amendment (الحاقی) به Specification های پروتکل ها اضافه میکنه. مثلا میبینیم 802.15.4e-2012 اون e ایش میشه الحاقی که معمولا در موضوع IoT هم هستش. بعد ها این الحاقی ها میشن یک اصل برای نسخه بعدی
- 2: الحاقی g برای این استفاده شد که از این پروتکل در smart grid استفاده بشه که تبادل انرژی بین خانه ها و انرژی خورشیدی و .. رو اشاره داره. فاصله node ها short تا medium هستش و طولانی نیست.
- شیکه های SCADA شبکه های کنترل صنعتی هستن.
- Public lighting اینه که چراغ های یک کوچه همیشه روشن نباشه بلکه مثلا یک فردی عبور کرد روشن بشه.
- 3: در الحاقی e مربوط به mac layer هستش که خواسته بهبود بده
- 4: مهم ترین تفاوت بین 802.15.4 و 802.15.4g در حجم payload size می باشد.
- 5: لایه فیزیکی g و e: وقتی payload رو زیاد میکنیم دیگه در IPv6 استفاده میکنیم دیگه به fragmentation نیاز نداریم. از بدی حجم payload بالا هزینه زیاد از لحاظ انرژی است چون وقتی به خطا داشته باشیم کلهش باید از اول ارسال بشه. از جهت اشغال کانال هم هزینه داره.
- بنابراین از CRC استفاده بشه که یک سری بیت به صورت هوشمندانه اضافه میشه که قدرت بیشتری برای تشخیص و تصحیح خطا رو در اختیار میگذاره.
- 6: تفاوت دیگه بحث Security عه. در الحاقیه g یک Auxiliary Security header رو داریم.
- 7: موقعی که داشتیم structure format رو در لایه mac برای 802.15.4 میگفتیم گفتیم سه تا بخش داره که تو شکل میبینیم.
- در frame control سه بیت اول نوع frame رو مشخص میکنه که beacon بود یا ack یا ...
- یکی از این فلگ ها security بود که آیا فعال هست یا نیست. اگه  $flag = 1$  باشه یعنی داده encrypt شده. اگر این فلگ امنیت 1 شده باشه یک هدر اینجا اضافه میشه تحت عنوان Auxiliary Security Header. که از سهم frame payload اون وسط کم میشه و به هدر اضافه میشه.
- 9: دومین پروتکل از دسته short range ها که از 802.15.4 هم short range تر است BLE می باشد. نسخه Bluetooth low energy اول توسط نوکیا در 2006 اومد البته اسمش Wibree بود. بعد merge شد و در قالب Bluetooth core specification version 4 اومد. iPhone 4s هم اولین گوشی ای بود که این رو دشات.
- 10: بسته به انتخاب در لایه فیزیکی سرعتش از 1 Mb/s.2 هم بیشتر میتونه بشه و به 1 برسه.
- برای cost پایین:
- استفاده از فرکانس آنالیزنس
- معماری ساده
- 11: در مقایسه با پروتکل هاش short range، BLE از اسمش معلومه انرژی کمتری داره و استک اش شبیه بلوتوث کلاسیکه.
- Controller: لایه فیزیکی و لینک. معمولا هم یک SoC هستش
- Host
- Application یا Profile
- 12: معماری از سه بخش تقسیم شده:
- Link Controller: کنترل کننده لینک که لایه فیزیکی و لینک
- Host: مجموعه ای برای یک کاربرد خاصی مورد استفاده قرار بگیره یک سری Functionality هابی بالای لایه فیزیکی و لینک هست که برای انواع profile ها مورد استفادس
- Profile
- 14: BLE برای data streaming خوب نیست و زمانی استفاده میشه که یک small packet ای سریع ارسال بشه. هیچ موقع یک دورین رو نمیبینیم که BLE استفاده کنه.
- دو حالت:
- Master: تحت عنوان Central device میتونه به تعدادی Slave وصل بشه
- Slave: در mode save energy هستن و عموما در sleep mode هستن و وقتی wake up میشن که بخوان packet ای از master دریافت کنن.
- 16: در مقایسه بین data rate و throughput، میبینیم که در کلاسیک 1 Mbit/s اش شده 0.7 ولی در BLE همون شده 0.27. یکی از دلایل میتونه هدر های بیشتر در نظر گرفت. یا مثلا این که مستر برای بیدار کردن باید اسلیو رو بیدار کنه (البته باید دقیق تر بررسی کنیم)
- 17: در لایه فیزیکی BLE 40 تا کانال داره ( نصف شد) که 3 تاش برای کارای کنترلیه که مشخص بشه مستر و اسلیو کیه.
- 19: این ماژولیشن مورد بررسی قرار نمیگیره تو این درس. فقط در این حد بدونیم که بین فرکانس ها پرش میشه به صورت شبه تصادفی. وقتی میگیم شبه تصادفی چون از نظر ناظر بیرون تصادفیه ولی بین فرستنده و گیرنده توافق شده.
- 20: سه فرایند توسط L2CAP که در Host هست انجام میشه.
- Advertisment: تعیین مستر و اسلیو
- Scanning: کسی که تازه وارد شده میخوان ببینه مستر کیه اسلیو کیه
- Connection establishment
- 21: یک روش ساده ای که device ها میتونن حضورشون رو به بقیه بفهمونن و اجازه میده ک یک connection ای برقرار بشه. وقتی یکی master شد این Advertisment تحت عنوان broadcast data میتونه لیست Supported services، device name و TX Power Level رو بفرسته.
- 40 تا کانال داشتیم که 3 تاش برای Advertisment بود و توسط اون 3 کانال ارسال میشه.
- سوال: چرا یکی از Advertisment ها فرستنده Power level سیگنال ارسالی رو هم میفرسته؟ در shareit یکی از کارایی که میکنه اینه که device های اطراف رو شناسایی میکنه که عموما توسط Advertisment هستش. بعد میاد یک سری دایره هابی میگذاره. این یک مفهوم داره و نشون میده که فلان دستگاه نزدیک تره از بقیه. این کار اینطوره که مثلا با 2 mW یکی ارسال میکنه ولی ما درگیرنه 1.5 mW دریافت میکنیم. بعد یکی دیگه 1mW میگیره. اینطوری میفهمیم کدوم نزدیک تره.
- 23: در Active Scanning ما بعد شنیدن Advertisment ما Scan request میفرستیم تا جواب بده. چون ممکنه ما یک سری Advertisment باشه که علاقه بهش نداشته باشیم. مثلا در shareit وقتی

کلیک نکنیم یعنی علاقه نداریم و نیازی هم به تبادل داده نداریم.

25: مثال shareit رو اینجا بهتر میشه مثال زد. ما وقتی کلیک میکنیم initiator میشیم و تقاضای ارتباط میدیم. اون فردی که درخواست ارتباط رو میده میشه Master و اون یکی میشه slave.

26: Hybrid: برای یک عده master برای یک عده slave

:27

2: این که میگیم نوی سال 1997 دیتا ریت 2 مگ بود بعد 1999 شد 6 تا 54 مگ اینا دستوری نیست وابسته به لایه های پایینیه.

3: در سال 2008 میبینیم برای Fast roaming بود که برای یک محیط بزرگی استفاده میشه. مثلا بخوایم در دانشگاه امیرکبیر یک 802.11 واحدی ارائه بدیم. یعنی مثلا از دانشکده کامپیوتر در میام بریم تو محوطه راه بریم بسته به این که نزدیک کدوم Access Point میشیم اتوماتیک به سرعت خودش وصل بشه نه این که دستی وصل کنیم.

نسخه 802.11ah رو ما علاقه داریم که در ساب گیگاهرتز داره کار میکنه. که دو هدف داره:

- Low Power
  - Range بالایی رو در مقابل بقیه نسخه ها ایجاد بکنه.
- که از بدی هاش اینه که data rate کم میشه که قبلا مرور کردیم.

4: در 802.11.ad چون بالای 1 گیگاهرتز هستش رنجش کم هستش و بالعکس 802.11.ah میتونیم در یک کیلومتری اش باشیم کانکشن داشته باشیم ولی دیتاریت کمه

5: معماری 802.11 یک LLC داره ک مربوط به اضافه کردن اضافه کردن بیت برای تصحیح و تشخیص خطاست که بین همه نسخه ها مشترکه. اما mac layer مهمترین فانکشن آن DCF عه که یک مکانیزم برای دسترسی به کانال میتی بر رقابت است که بر اساس یک مکانیزمی کاربرا به مدیا دسترسی داشته باشن که جلوتر میبینیم.

6: WiFi قبل 802.11.ah روی ساب گیگاهرتز کار نمیکرد که یک سری چیزارو سابورت نمیکرد ک در اسلاید میبینیم.

7: WiFi HaLow اون Low اش یعنی Low Power و Ha اش برعکس ah هستش و تلاش شده سعی Hello درش بیارن. که با هدف مصرف توان کمتر نسبت به وای فای قبلی و رنج بیشتر طراحی شده. رنجش تا 1.5 کیلومتر میتونه پوشش بده که نسبت به قبلی قبلی که تا 500 تا 600 متر رو پوشش بده و همین سبب شده به ah بگیم دیگه LAN نیست و به گونه ای WAN عه ولی حالا جز LAN حسابشون میاریم.

8: جاهایی که میخوایم مانیتور محیط بکنیم یا زمین کشاورزی داریم یا سنسور در محیط صنعتی داریم یا خانه هوشمند، یک سری سنسور در کاربرد IoT داریم که حداکثر در 1 کیلومتری gateway قرار میگیرن.

همچنین میتونه نقش Backhaul aggregation داشته باشه: بهش وصل بشن که به اینترنت ارتباط برقرار کنن. وقتی اهمیت داره که سنسور یا پخوان با cloud ارتباط داشته باشن.

کاربرد سومشم اینه که وای فای رنج رو بالاتر میره و فاصله بیشتری رو پوشش میده. حتی گفته شده که 802.11ah برای cellular traffic offloading هم خوبه. یعنی شبکه cellular مثل شبکه های داخل گوشی که داره ترافیک اینترنتی رو دریافت و آپلود میشه offload بشه به 802.11ah. یعنی در یک سری محیط به جای این که به شبکه cellular وصل بشم و به base station وصل شیم به access point های وای فای وصل بشیم. به اصطلاح میگن offload میشیم و load ما از شبکه cellular به 802.11ah آف میشود. مزیت این کار برای Operator و مشتری چیه؟ برای Operator در جاهای شلوغ و فقی پیاده سازی کنه پهنای باند بیشتری براش میمونه. یعنی چی؟ هرچند که برای Access Point باید هزینه بدیم، این هزینه کمتر از اون فرکانس لایسنس دار هستش. و این یوزر ها در یک شعاع یک کیلومتری Mobility کمی دارن ← مثل بازار که مشتری ها یک ساعت تا دو ساعت در یک مکان هستن. و با این کار ما فرکانس آنالینس رو برای اونا میذاریم و فرکانس های لایسنس هارو میذاریم برای بقیه که Mobility زیادی در جاده ها دارن و...

9: روی سابگیگاهرتز آنالینس کار میکنه. که فرکانس های ISM رو داریم مشاهده میکنیم

10: براساس OFDM که پهنای باند رو بر اساس یک سری فرکانس تقسیم میکنه و هر فرکانس رو به یه سری کاربر میده. در مقایسه بین ah و ac، میبینیم ac تا 60 گیگاهرتز داره و ah تا یک دهم اون کانالاش هستش. انتظار داریم دیتا ریت هم یک دهم ac باشه.

برای ah میبینیم که دیتاریت 100 کیلوبیت در ثانیه هست که برای برخی کاربرد های IoT میشه گفت more than enough هم هست! مثلا سنسور های کشاورزی دیگه دیتاریت بالایی نمیخواد.

11: از نظر Coverage میبینم حتی یک af هم داریم که پوشش اش از ah بیشترم هست.

12: در لایه فیزیکی گفتیم که مالتیپلکسینگ اشون از نوع OFDM هستش. فرکانس کاری هم که گفتیم ساب گیگاهرتز هستش. در لایه 802.11 MAC یک بحث کلی داریم که برای مقابله با error موثر ترین جا در MAC هستش. میدونیم که یکی از وظایف مهم MAC چیه؟ Logical Link Control که تصحیح و تشخیص خطا هستش. البته لایه های بالا مثل TCP هم برای خطا کنترل میکنن که میگن ارسال مجدد بشه اما بهترین جا MAC Layer هستش.

802.11h از یک پروتکل frame exchange استفاده میکنه که به این صورت که اون Source station که میخواد داده ارسال کنه داده اش رو ارسال میکنه و مقصد ACK میفرسته و اگه مبدا ACK رو دریافت نکرد از اول میفرسته. (کلیت Frame exchange protocol).

برای انجام (قبل از ارسال داده) این کار بایستی 4 تا بسته رد و بدل بشه. اول RTS فرستاده بشه که بگه من فلانی هستم و میخوام برای بهمانی داده ارسال کنم (خود داده رو نمیفرسته). مقصد که این رو دریافت میکنه clear to send میفرسته و میگه که OK من آمادم دریافت کنم. بعدش مبدا داده رو ارسال میکنه و مقصد هم ACK به فرستنده میفرسته.

چرا از Aloha و عادی استفاده نکردن؟ میومدن و اگه Collision رخ داد از مکانیزم های Aloha مثل صبر به زمان رندم و ارسال مجدد استفاده میکردن. این مکانیزم پیشنهاد دادی چه مزیتی داره؟ با ارسال RTS و CTS اصلش بر این اساسه که اگر قراره Collision ای رخ بده برای چی حجم زیادی داده ارسال بشه؟ میان رقابت رو برای تقاضا میدارن و اگر Collision هم داد روی RTS رخ میده و بالاخره یکی RTS اش به مقصد میرسه و CTS وقتی برگردونده شد بقیه متوجه میشن و ساکت میشن. ولی لزومی نداره بقیه داده بدون collision باشه اگر این روش efficient تره.

چه زمانی ممکنه collision داده رخ بده؟ وقتی مثلا 2 تا node در رنج هم باشن RTS وقتی بشنون جفتشون متوجه هم میشن و وقتی CTS رو میشنون اون یکی میفهمه که برا اون نیست و ساکت میشه:



ولی اگه یک گره ای خارج از دید اینا باشه دیگه RTS اونارو نمیشنوه و وقتی CTS میشنوه میفرسته.

13: تمرکز Mac layer در 802.11ah در Power consumption بوده و اجازه داده Wifi station هم sleep mode داشته باشند. قابلیت اینو داره که user های زیادی بهش وصل بشن و short low-bit transmission rate یعنی مقدار بیت کمه و زمان کوتاهی برای ارسال نیاز دارن برای کاربر های کشاورزی و ... مناسب است.

14: برای توپولوژی هم star هم mesh ساپورت میشه. در 802.11ah از sectorized beam transmission یا آنتن های جهت دار استفاده شده: باعث میشه در یک فرکانس در یک جهت خاص به یک یوزر ارسال کنیم. و همین فرکانس رو میشه برای کاربری که در نقطه جغرافیایی دیگری هم هست استفاده کنیم. ولی اگر از omni-directional استفاده میکردیم دیگر نمیتوانستیم یک فرکانس رو به 2 کاربر ارسال کنیم (تداخل به وجود میومد).

15: دیدیم که 802.11ah برای پاسخ به نیاز IoT یعنی هزینه کم و برآورده کردن نیاز دیوایس های زیاد و مصرف کم توان customized شده و هر 3 نیاز مورد نظر رو برآورده میکنه.

16: تا الان Short Range ها که PAN ها و LAN ها بودن رو بررسی کردیم. حالا وارد Long range میشیم که به اسم WAN میشناسیم.

17: در مقایسه بین Cellular ها، WiFi روی آنالایسنس هستش و رنج کمتری نسبت داره. ولی بدی شبکه های سلولی اینه که به شدت انرژی مصرف میکنه. بنابراین شبکه های Long range مثل LPWAN اومدن که خاص IoT طراحی شدن. برای برآورده نیاز هایی اعم از مصرف توان کم و رنج بالا

18: وقتی Long Range باشه زیرساخت هزینش میاد پایین. مثلاً در شهر تهران اگه 750 کیلومتر مربع در نظر بگیریم، اگر Access Point قرار بدین که تا 10 یا 15 کیلومتر رو پوشش بده، اگر 10 در نظر بگیریم میشه یک دایره با شعاع 10 کیلومتر که میشه مساحت 314 کیلومتر مربع. و اگر 750 رو تقسیم بر 314 کنیم میبینیم در حالت ایده آل با 2 تا Access point میشه پوشش داد. البته خب اینا ایده آله و وقتی که موانع نداریم.

در شرایط واقعی با 10 تا 20 Access Point میشه تهران رو پوشش داد. که هرکدام 400 500 دلار قیمتشه. با 3 تا پراید میشه اپراتور IoT راه انداخت که هزینه پایینی است. باتری هاش تا 10 سال کار میکنه. هزینه ها برای اپراتور از 2 جهت در نظر گرفته میشه:

- زیرساخت: که برای IoT ارزونه
  - فرکانس: که برای IoT آنالایسنس است
- پس هزینه هایی که برای ارسال پکت ها در نظر میگیریم باید در حد تک تومن باشه!

22: چرا LAN ها و WAN های فعلی برای IoT مناسب نیستند؟ علتش رو نوشته

24: برای 5 تا هدف که تعیین کردیم باید چیکار کنیم؟

برای Long Range:

- فرکانس رو میاریم روی Sub-1GHz
- تکنیک های مژولایسن خاص هم داریم که تو اسلاید میبینم اینا در برابر نویز خیلی مقاوم دو انتخاب فوق در لایه فیزیکی اتفاق میافته.

25: برای Low Power:

مصرف توان زیاد از کجا میاد؟

- تضعیف زیاد داریم و باید با توان بیشتری ارسال کنیم. همین که میایم روی Sub Ghz تضعیف کم میشه و با توان کمتری میشه ارسال کرد.
- Computation: هرچی برای Medium Access و فیلد های که توی پروتکل دیزاین داریم بیشتر محاسبات کنیم و پروتکل پیچیده تر باشد باید میکروکنترلر بیشتر روشن باشد و این یعنی مصرف توان بیشتر. بنابراین سعی کردن Complexity رو از end device بردارن offload کنن روی gateway. از طرفی چون gateway به برق شهر وصله سمت یوزر پیچیدگی کم میشه

26: چطوری به Low Cost میرسیم. خود Hardware حدود 5 دلار و connectivity subscription هم قراره به ازای هر unit 1 دلار باشه (حالا ماهیانه یا روزانه بستگی به این که چه میزان داده داریم).

برای هزینه کم باشه باید هزینه OPEX و CAPEX اپراتور بیاد پایین. CAPEX میشه هزینه هایی که برای راه اندازی خرج میکنیم مثلاً به جایی رو اجاره میکنیم، سرور میخریم، gateway رو در یک بلندی قرار میدیم، کلا و ...

OPEX میشه هزینه Operation، مثلاً کولر گذاشتن برای Cooling، maintenance، اگه باگی داشت رفع بشه، پاسخ به سوالات مشتری و ....

برای کاهش هزینه CAPEX خب همین که Long range هستن تعداد station ها میاد پایین.

برای OPEX هم وقتی تعداد سایت ها کم باشه نیرو کمتری برای نگهداری میخواد.