

Lec 04

* الدكتور قال ان الفايال هيبقى زي الميديا من حيث طريقة الاستلام ومن حيث انه غير ناس طامه ال concept فبيقى في برودو mcq ورسم للاشارة مع ال time بين هيبقى في حته Programming زياده "جزء Assembly" بيدنا code وبقول انا رسم ال code ده بيدخل اليه او بيدنا رسمه وزكبله ال code بتاعها او تغير فيه حاجه ..

المفروض نعلم الفهرده على :-

- 1- RS-232
- 2- USB
- 3- I²C

start from slide (38) "Serial Buses" (839)

يعني ايه Serial Buses ؟

يعني بيتا او بيت قبل Bit by Bit عن طريق wire واحد بين

① RS-232



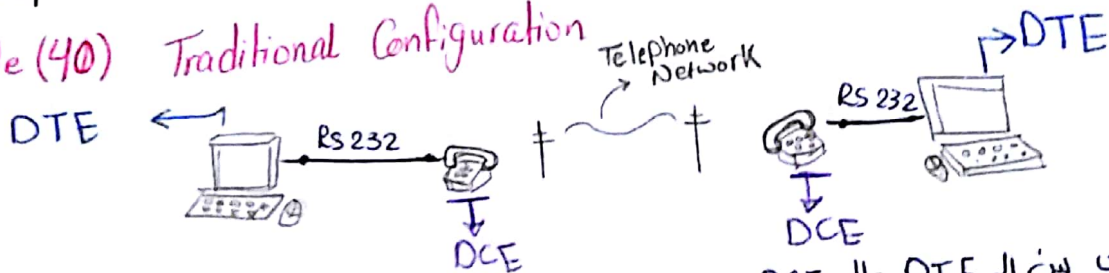
RS 232 بيدقل المعلومات serial والفكره الاساسيه فيه انا بيدقل بين حاجتين :-

- ① DTE = Data terminal equipment (terminal), ex: Computer, Monitor, Keyboard
- ② DCE = Data Communications equipment (modem)

طب يعني ايه modem ؟ الدكتور عقالش بين للفهم يعني :-

As modem is abbreviation of (modulator-demodulator), it is a hardware device that converts data so that it can be transmitted from computer to computer over telephone wires.

@ slide (40) Traditional Configuration



الوصله الي بين ال DTE وال DCE هيردي الي تهمني وهي دي RS-232 .. الي بين ال modem وال modem متهمنين

@ slide (41)

زمان في السنينات على وقت نصينج ال RS-232 كانت السرعات بطيئلسا .. قاول ما تهل كان

Max data rate = 20 KB/s with a max cable length of 15 metre.

بين دلوقت سرعته اتمست ممكن يوصل ل 1Mb/s .. ده تطور بالنسبة لزمان بين مازال بطي

* لو عاين ازود السرعه لازم اتقص طول ال cable .. عشان في حاجه اسمها Propagation time وهو الوقت الي باخده عشان اوصل عن طرف ل طرف .. فلو ال time ده كان كبير اوي .. اكبر من ال time الي بيت في ال actual data على ال Bus .. على ما اول Bit توصل هكون خلعت البت كله وممكن حد باقي يدخل بيت فيحصل interference والداتا بتوطل.

Configuration of RS-232

* Serial

* Point to point

Point to point ← يعني Receiver واحد بيكلم Transmitter واحد .. في حالات تانية هنشوفها بعدين ممكن تبقى point to multipoint أو multipoint to multipoint connections من الـ

@ slide 42

There are 2 modes of operation

① Asynchronous → CLK بين الـ sender والـ Receiver هيفيش بس بيكون في بيدل يعرفني انا بيبدأ وامتى بخلص ← start bit stop bit

② Synchronous → بتتبع مع الداتا عثمان الـ CLK signal لازم يبقى في sender والـ Receiver يتزاهوا مع بعض ويكونوا على نفس الـ CLK

معظم الـ RS-232 ← Asynchronous - عثمان أسهل وأسرع (Most but not all)

@ slide 43

الجزء ده مهم داخل في البروجكت

ازاي الـ data بتتبع serially على RS232 ؟

الـ RS232 بياخد حاجة عامة زي الـ Packets بس الوجهه اللي بيتبع بها بتكون يا إما 7 bits or 8 bits .. في الـ transaction الواحدة يعني

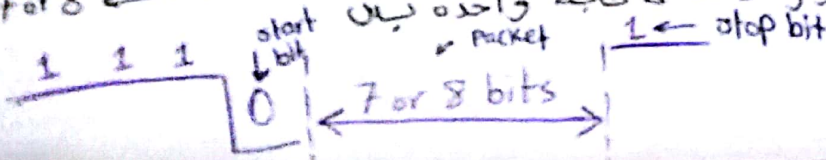
عثمان اتجنب عموماً الـ synch. ومشاكله هنقسم الـ data لأجزاء صغيرة وبعدين يبقى في زي handshaking بين الـ Trans والـ Receiver يعرفوا انا انا انا وامتى بتت وامتى بتت

الـ Asynchro فيه CLKs و 8 bits فيه أكيد .. أي digital فيه بس الفكرة ان الـ CLK هتكون واحدة .. من هتكون بتاعت الـ Receiver والـ Trans ... الـ synch بتكون الـ CLK واحدة

الـ specifications الـ RS-232 حددت ان لو هيفيش حاجة بتتبع الـ Bus يبقى عليه '1'

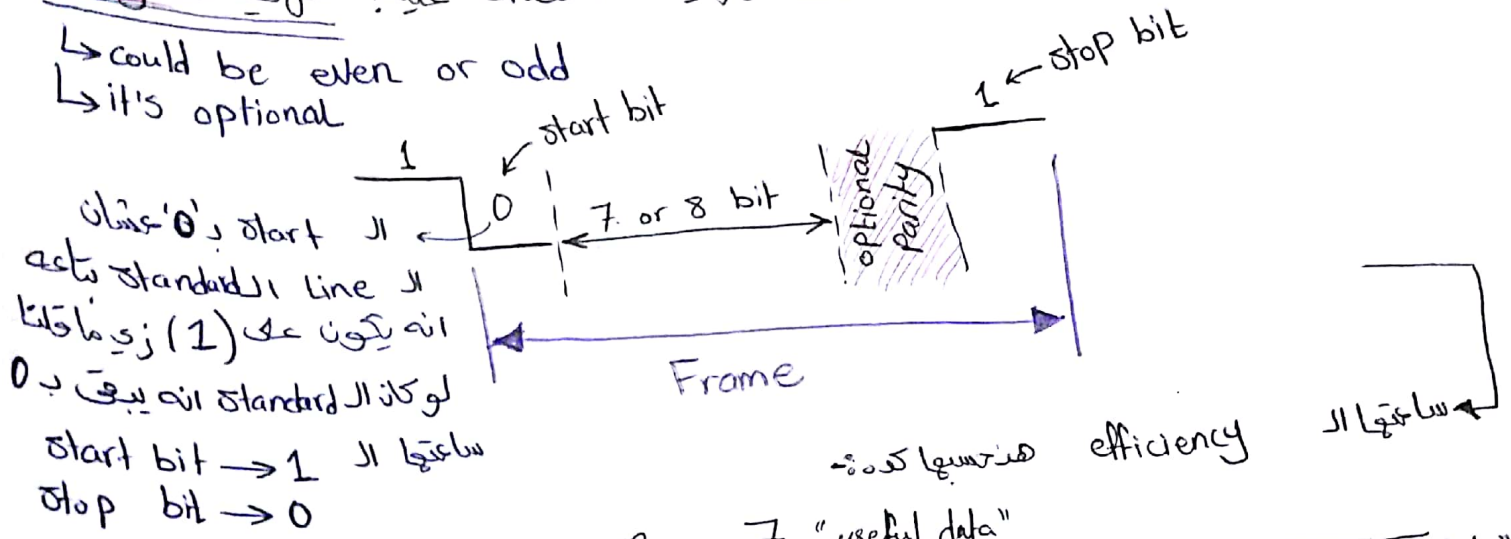
وده صحيح بتضيق للباور بس لها اتفقوا على كده

وقبل ما اتبع الـ packet لازم اعرف الـ Receiver اسبق في حاجة جياك فممكن الـ '1' الـ '0' يعني صيغة '0' ودين هي الـ start bit وبعدين اتبع الـ packet الـ 7 or 8 bits وبعد كده اتبع الـ stop bit باني ارفع '1' ثاني بعد العدد الواحد من الـ packet bits الـ 7 or 8 لازم اتبعه عن الاول فينعتش عن 7 وعنه 8 .. 8 حاجة واحدة بين الـ packet



لأن في احتمال انه خطئ عند Error فإزاي check على اكل في Parity bit

- could be even or odd
- it's optional



efficiency هتجيبها كده

$$\eta = \frac{7 \text{ "useful data"}}{1 \text{ "start bit"} + 1 \text{ "stop bit"} + 1 \text{ "Parity optional"} + 7 \text{ "useful data"}} = 70\%$$

لو هفهم Parity ال η هتبقى اعل $\frac{7}{9} \times 100\%$ لو في أكثر من error ال Parity check هيفشل و ساعتها هتوقف كل ثاني هتقوله

لو عايز اعل even Parity بتشف عدد الواحد لو كانوا 3 ← Parity = 1
لأن بتشف مجموع الواحد شغل ال Parity يكون even لو كانوا 4 ← Parity = 0

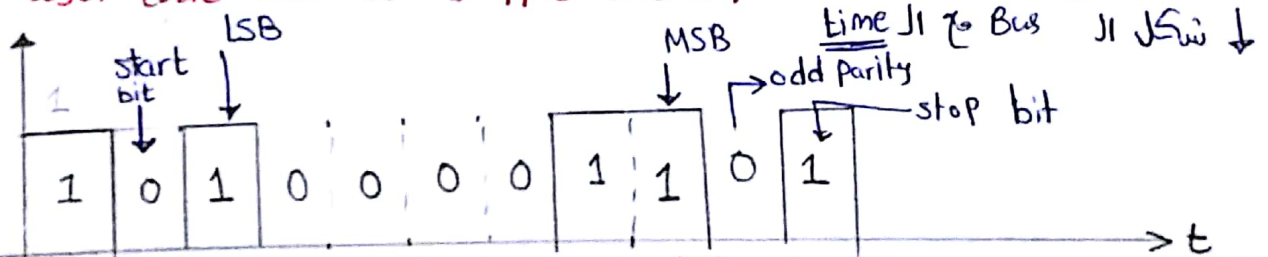
وهكذا لو odd Parity = 0
لو كانوا 3 ← Parity = 1

MSB ← 0 1 0 1 1 0 0 1 ← LSB

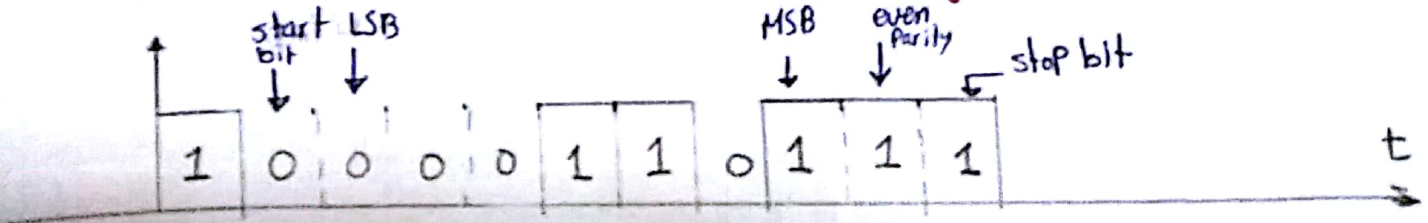
لو عايزه ابعث sequence
بتدبعت الأول كدما اول ال MSB

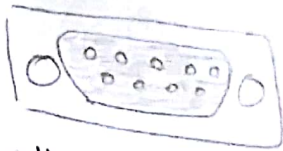
@ slide 45

EX①: ascii code for 'a' = 1100001 with odd Parity



EX②: Plot the transmission of ASCII char 'X' = 1011000 over an asynchr RS - 232 with 7 data bits and even Parity





كان زمان ال connector فيه 25 pin دلوقتى 9-pin هو الي شائع

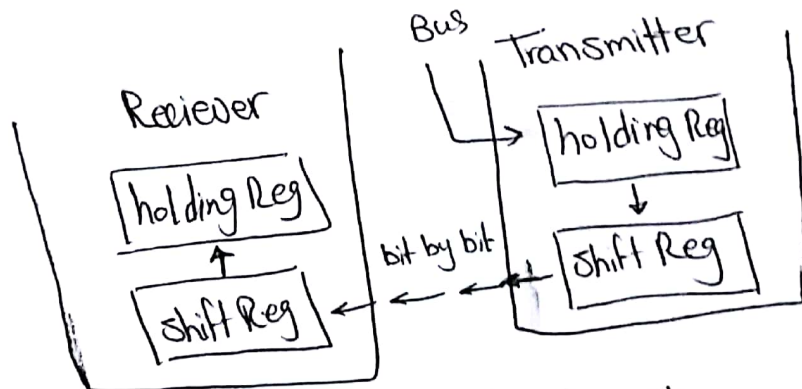
* بالنسبة لـ Transmission Plot فيبديجي في الامتحان والناس بيتدس ال يا اما بيتكلم و ترسم من ال MSB ← LSB و طبعا علم المروض MSB → LSB
 فنظري بالنسبة من الكلام

@ slide (49) Universal Asynchronous Receiver - Transmitter (UART)

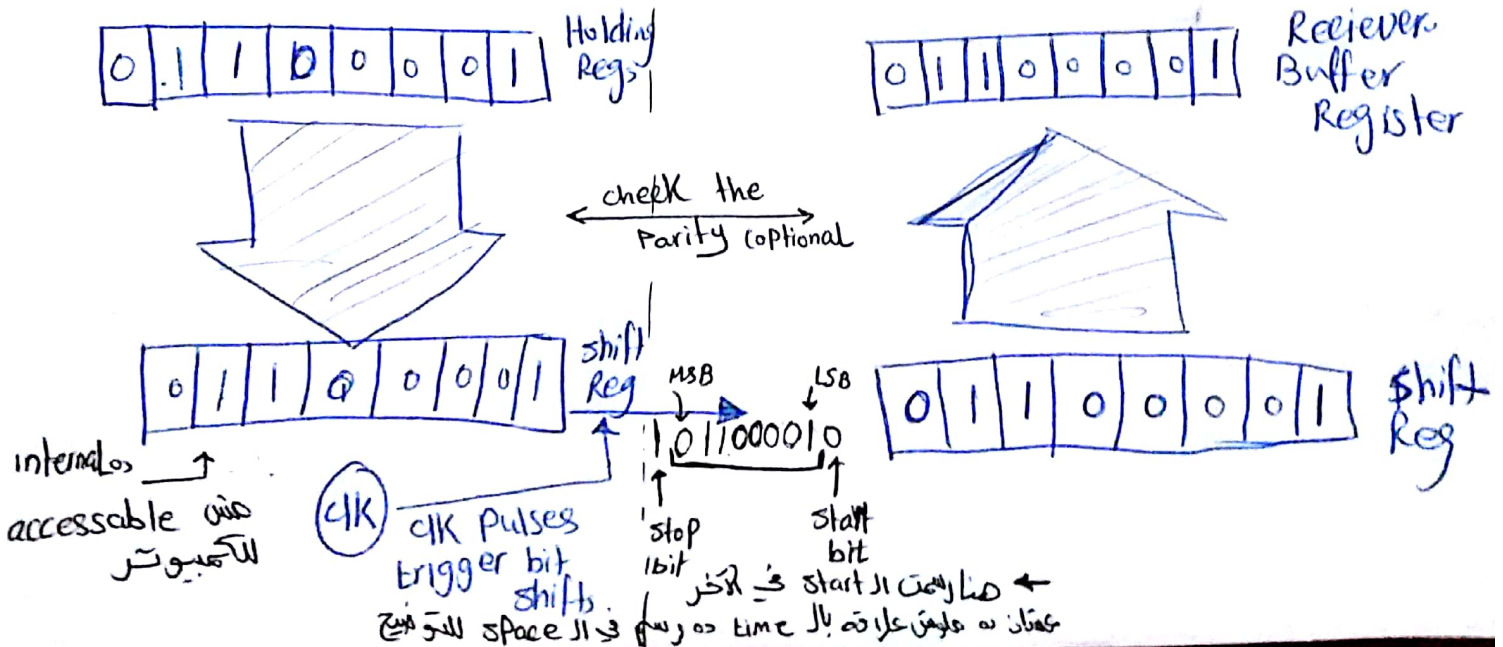
الكمبيوتر ال Buses جواه كلها Parallel فإزاي أوصله بـ System تاني Serial ؟
 عن طريق ال UART -- يتحول من Parallel ← Serial

UART ← جزء هاردوير و جزء سوفتوير .. عنده Transmission engine و Reception engine
 واحد سلكيت واحد من ال Transmitter

محتاج في كل engine هينم Register ثابت باخذ فيه الحاجه الي جياي عنشان ال Bus
 يرجع بكل وبعدين اخذ copy من ال Register التاني أو ال (Holding Register) وأخزنها في ال shift register و كل clk اخذ bit -- ده في ال Transm.
 عند ال Receiver الحاسب .. كل ما تجيي bit ادخلها في shift Reg وتول ما تكمل اعملها copy في holding Reg



Serial Data transmission , Serial Data Reception



Stop
bill

④

28.9

(c)

external
ROM

کھیل

RS 232

* We

1-2

Pattern

Trans

2- M

are
synchron

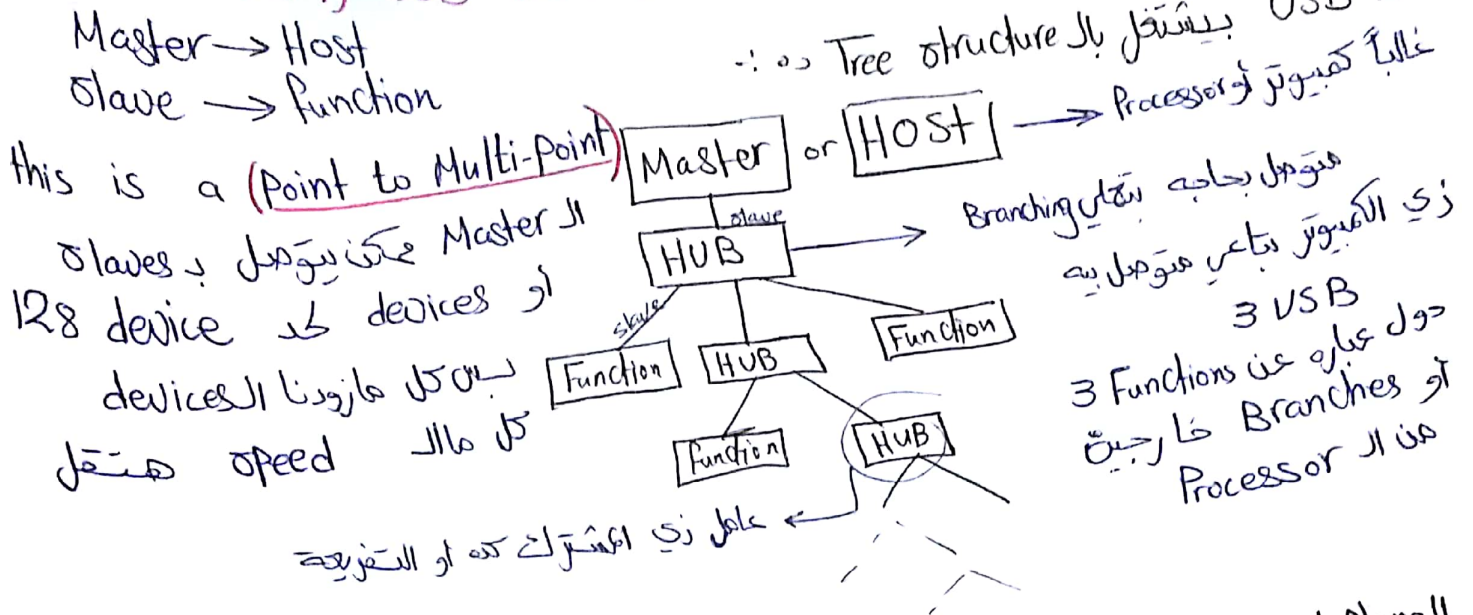
الفرق

فہرست

pgal.

--- 8F

@ Slide (54) USB technical Details



Smart Phone ال بيشغل ك Master لما بوحله بأي USB-device لكن بيشغل ك Slave لما بوحله بالكمبيوتر

* اكتبه لكايه الدكتور قال انه بيحب يرمي فيها كتر بالاختبار ↓↓↓

ال Function او ال slave عرفها مات Interrupt ال Host .. لازم ال Host هو اللي يوج يسأل كل Function عندك حاجة؟ ويرفض يوري عليهم كلام .. فال Receiving Data عن ال Function بيلم عن طريق Polling

* ال USB فيه Power Line ده معناه لما هوصل device على ال Host او الكمبيوتر هيلقي انه اسحب منه power فبيفهم ان في device جديد دخل ويبدأ يبوراه على Driver وهو لوحده هينزله ويتعرف عليه بكمس زمان ممكنش في كده وكان لازم كل حاجة تعرفها ب disk او لو مكتوبه عليه (hot plugging) فيه كده كان جلد جداً عشان فيه لاصيه دي

* مش لازم اجب '0' او '1' ببيتهم المعروفة .. ممكن اجب حاجة على شكل code تبق زي

@ Slide (55) USB physical Interface

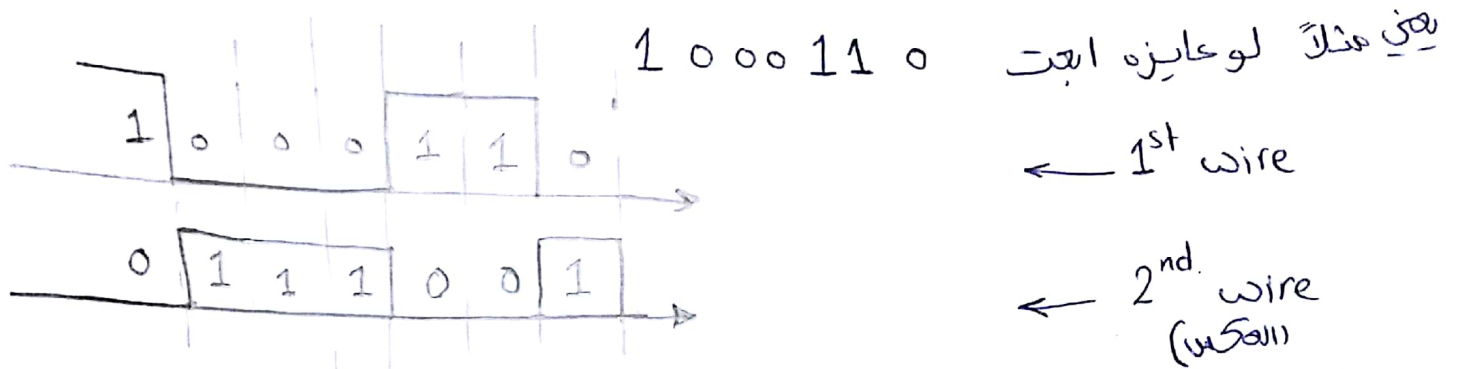
ميم بين ال Trans وال Rec.
وده بس عشان نفهم نوع ال Line المستخدم
في ال USB

↳ Differential signaling

↳ Half Duplex

↩ يا إما ال Host بيتبع
ال Function او العكس .. بس
ميفهمس الاثنين مع بعض

* Differential Signaling → البيانات [2-wires] والعكس يتبعها

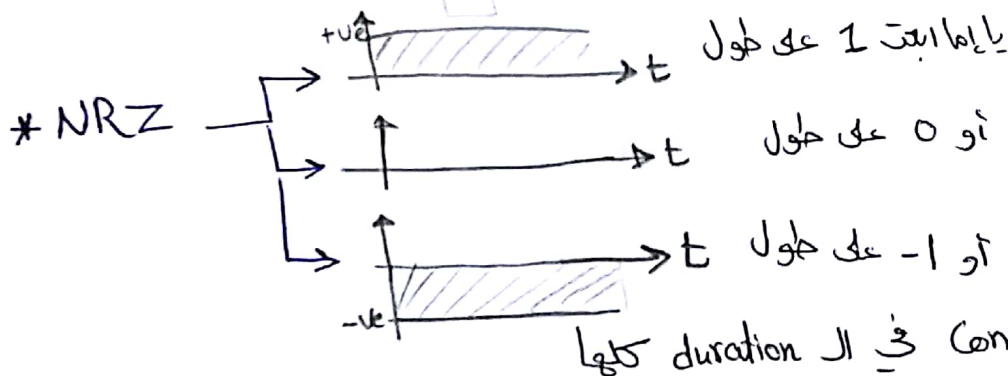


وبالتالي الـ error Margin أقل بكثير
Noise Cancellation*
CMRR *

← لو فحصنا أي USB cable هنلاقي فيه 4 wires واحدة الـ Data والثانية للـ inverse والثالثة Gnd والرابعة Vcc

الـ USB بيأخذ الـ Input Binary من واحد وامضار يعلم coding بحاجة اسمها NRZ
Non - Return to zero

في الـ NRZ الشكل ده المفروض هنشغوش رجعت لا zero ← ده اسمه RZ لأن



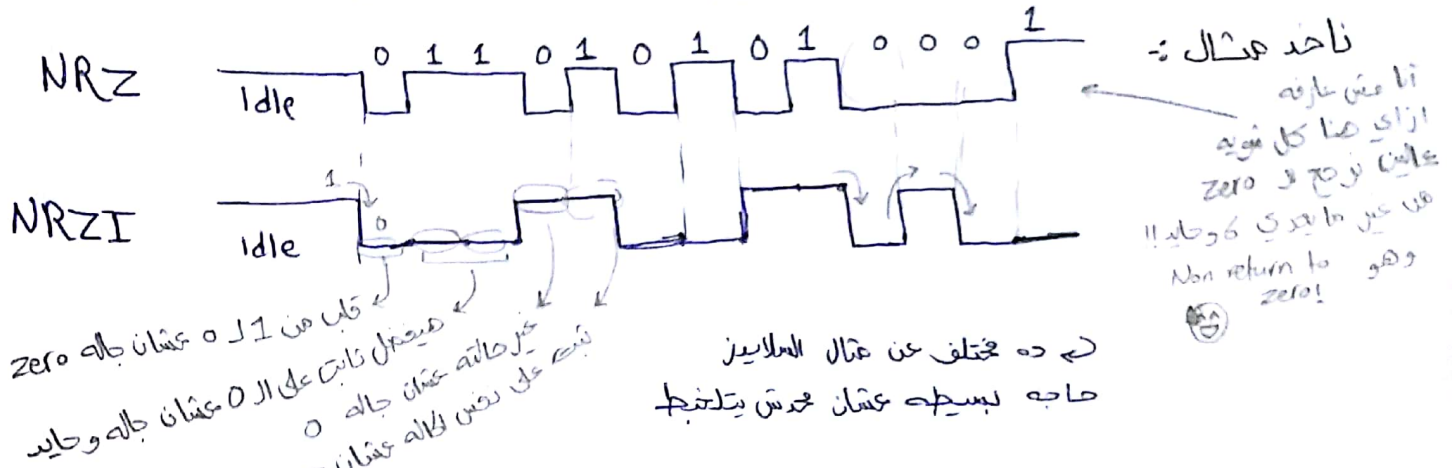
علشان يتجنب أن الـ CLK تبتوط وهو صيوط له؟
قاله لو الـ Bus فخذ مبعوث عليه حاجه ثابتة لفترة كبيرة كده هيقى idle "خامل أو نائم" والـ CLK هتوقد الـ synch. وهتبتوط فلان كل تنويه افكرها إن في Transition من 1 لـ 0 أو من 0 لـ 1
مشان ترجع ثاني
لـ فلو لقيت sequence ← 6 وحيد هخط zero ولو 7 وحيد هخط zero بعد أول ستة بمرور

لـ عند الـ receiver بقى بيكون ظاهم لو لقى 6 وحيد يبقى العايبه ← bit stuffing هيعديها
ويكمل من غيرها وياخد الـ 7 بده يرجع مكانه رقم 7

عشان احنا بنكتب differential signaling
 فممكن عندها NRZ و NRZI
 → Non-Return to zero Inverted

at NRZI → zero is encoded as a transition
 → one is encoded as no change

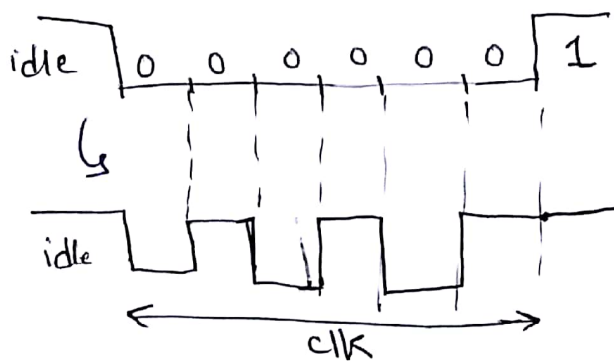
يعني لو جاتلك '0' اقلب من الحاله اللي كنت فيها لو 0 و لو 1
 ولو جاتلك '1' خليه زي مانتة مستقرش لو 1 و لو 0



فكرة لو عاين ابيت CLK بطريقة ال NRZI فانا صبت شوية اصفار
 وهي هتسجم ل CLK لان ال zero يعني من الحاله 0 ← 1 و 1 ← 0

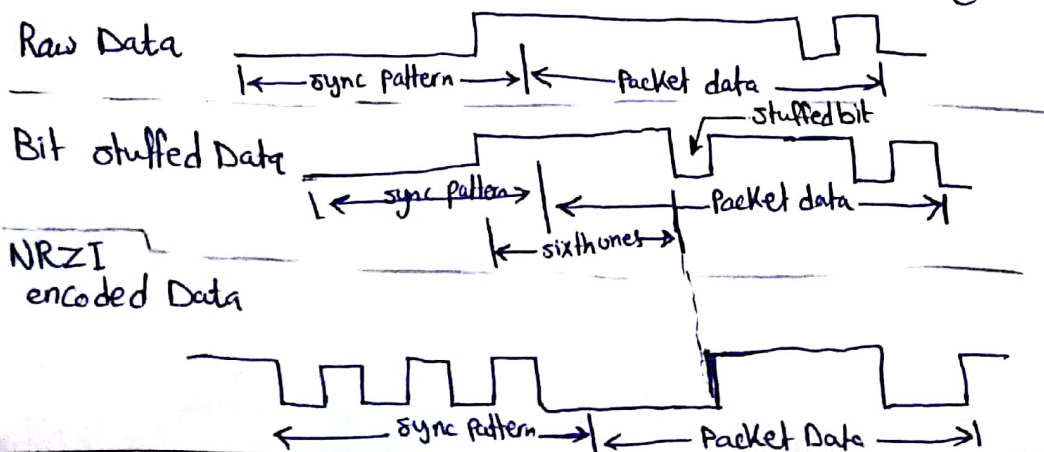
@ slide (56)

قبل ما بعت ال data بيت 'one' 1 و 'zeros' 7 كله :-



ال 7 اصفار دول CLK والواحد
 عامل زي ال start بتاعت زمان بيقول
 bit انا خلاص بيتي ال CLK والي جاي Data
 ولو لقيت ال Data بتاعتي فيها واحد كثير
 هزور zero زياده بعد كل 6 و حايه

الرسمه في العلايين موضحه للشرح ده :-



@ Slide 58 Data Formatting

دول أساسيين

* Start of Packet (8 bits, 7 zeros + 1 one) →

* Address field (11 bit) → 7 → 128 device
→ 4 → internal use
← زي تقسيمه ال AHB ال AMBA Bus

* Start of Packet & Address are mandatory → يعني أماشي الزام

* CRC → Like the Parity check as a Concept
بس ده يستخدمه مع data كيبو .. بيحل function ويطلع رقم ال Receiver بيقارن برقم ال CRC الي اتبعت والي هو هيحسبه عشان يشف ال error قدياه.

في (57) slide الرسمه احزها لو لاحظنا هين هلاقها differential وده بيخبر عن ابي خلاص وحملت لاخر ال Packet

* البروجكت احوال يكون مقارنه بين ال UART و ال USB مع رسم الي بيحل على ال Line وميزات وعيوب كل حاجه .. التفصيل الدكتور هيقولها قريب ..