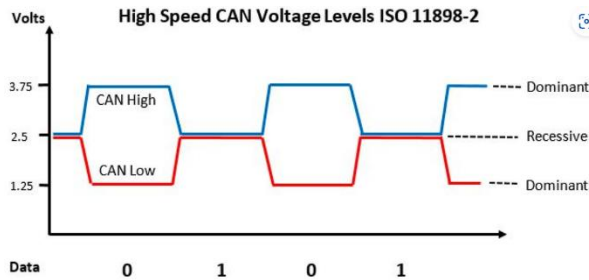


CAN

- ❖ ده عنده طرق كثير لل Check Error
- Reliability ودي اهم ميزه في ال CAN وده كله Hardware
- ده مفهوش Clock احنا بنشتغل Baud Rate ... وشلنا ال Clock عشان مش عايزين Noise
- شغال Half Duplex مينفعش يشتغل Full Duplex عشان احنا شغالين على BUS
- ده بينقل داتا عن طريق Differential Voltage
- Asynchronous عشان يضمنلي امشي مسافات كبيرة

- ❖ ال CAN سلكتين ومع ذلك Half Duplex .. لأني السلكة الثانية مش بتاعت Clock
- ❖ عشان نقلل ان السلكين جنب بعض ممكن ال Magnetic Field الى بيطلع من واحد يآثر على الثاني
- بعملهم Twisting Pairs زي ضفيره يعني



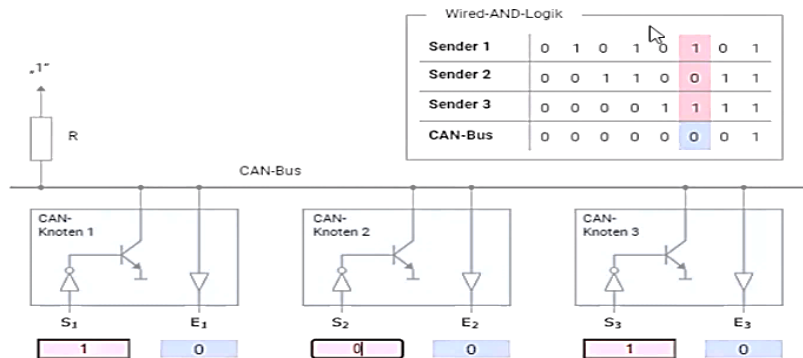
- ❖ ال CAN Transceiver
- ده الى بيعمل التحويل بتاعت Differential Voltage

- ❖ ال Can Controller
- ده الى بيبتعت ويستلم رسايل

- ❖ هنا ال Zero هو الى مسيطر والواحد مش مسيطر على ال BUS وده نفس حوار ال I2C بس فرق Volts
- ودي حاجة Hardware (لأن ال 1 بيكون 2.5V بس ال 0 بيوصل ل 3.75V وده الأكبر وبيسطر)

- ❖ هنا كل ال MC المتوصلين بالBUS بيعتوا داتا ويقروها في نفس الوقت وهي دي ال BUS ANDing

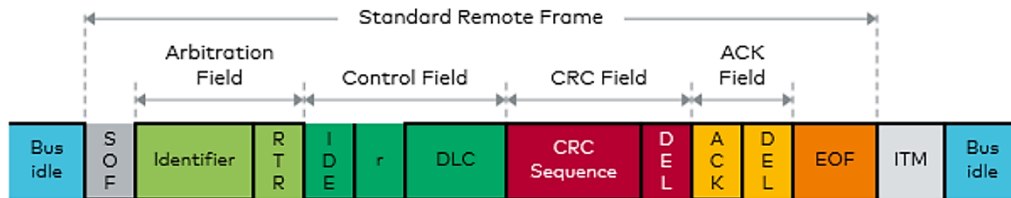
CAN Bus Logic



© 2010-2017, Vector Informatik GmbH. All rights reserved. Any distribution or copying is subject to prior written approval by Vector. V2.0

When two or more nodes try to transmit a message simultaneously on the bus, a bit-wise arbitration process is used to determine which message has priority. During arbitration, the bus is monitored by all nodes, and any node that detects a dominant bit on the bus while transmitting a recessive bit immediately stops transmitting and gives way to the message with the dominant bit.

ال Arbitration ده بالتبسيط كده
لما واحد يكتب حاجة على ال BUS ويلاقيها مش موجودة بي فصل نفسه



زاي هنبعث رسائل في ال CAN
❖ هو قالك هنريح دماغنا ومش هنعمل Slaves في ال CAN

Serial , Asynch , Half Duplex , Multi master no slaves

- اول حاجة ببعثها Start of the Frame ولازم تبقي Zero عشان ال Line بواحد
- بعدين نبعت ال ID (وده هو عنوان الرسائل ، احنا عملنا لكل رسالة عنوان)
- نبعت من MSB الى LSB عشان لو في رسالتين وصلوا مع بعض يضمن ان العنوان الصغير هو اللى يكسب

=====

- ❖ فانا مثلاً ببعث رسالة رقم 2 على ال BUS وال Microcontroller الى عايز الرسالة دي هياخذها والباقي لأ
- وممكن الرسالة يكون ليها كذا Listener بس مينفعش يبق ليها كذا Owner يبعثها
- عشان مييجيش الاتنين يكتبوا ويدخلوا في بعض

والطريقة دي اسمها **Message Oriented Protocol**

=====

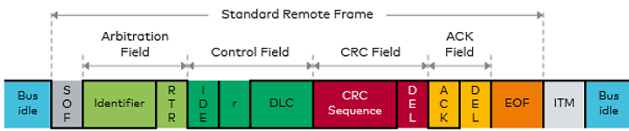
- ❖ مين اللى بيعمل Check على الرسالة ويقول ملهاش لازمة ويرميها
- ده ال MC مش الكود
- فكده انا عملتلك موضوع تنظيم الداتا Hardware بدل ما اعمل software (لأني ممكن اعملها بال I2C وابتعت قبل كل داتا رقمها وبعدين استقبلها واعمل حاجة معينة وكدا)

=====

Frame CAN

Start of Frame : بيمسك هو ال BUS والباقي مش بيبقوا Slave ، لأ ده هما قاعدين يسمعوا ومستنيين داتا

Identifier : ده هو Message ID وده بيكون عنوان الرسالة (11-bit)



RTR : (1-bit) Read/Write بتاعت ال BIT عامل زي ال

- ده انا Remote frame ولا Data Frame
- 0 : ده Data frame وده معناه اني انا صاحب الرسالة وهبعثلك داتا

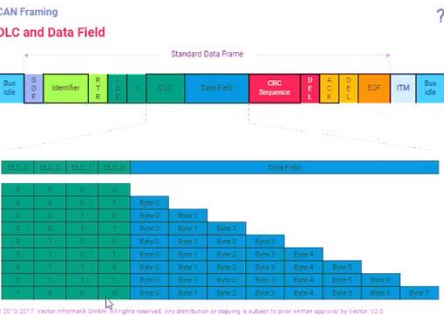
IDE : (1-bit)

- ده بيقولك ده فريم عادي ولا فريم Extended

r : (1-bit)

- Recessive دي بسببها فاضية يمكن استخدمها في المستقبل

DLC : عدد ال Bytes اللى هبعثها (4-bit)

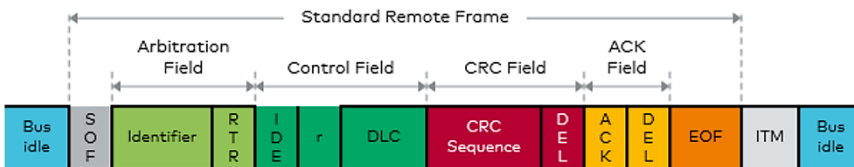


- مش معني اني مسكت ال BUS وان الكل بيسمعني يبقي ابعث عطلول .. لازم يكون في Data Length
- وال DLC ده 4-bit عشان اختر حجم الداتا ما بين (0 - 8) .. طب تحديد حجم الداتا هيغيد في اى ؟
- عشان ميحصلش Starvation ومحدث ياخد ال BUS لفترة طويلة يعني اللى بيتكلم اخره في الرسالة 8 BYTE .. بعدين لازم تقفل عشان لو حد ثاني مستني

CRC Field

- ده بيعمل Check على الداتا زي ال Check Sum .. ولو لقيها بايطة بيرميها ودي بتتعمل على كل حاجة مش الداتا بس

- وده هنا بيتعمل Hardware



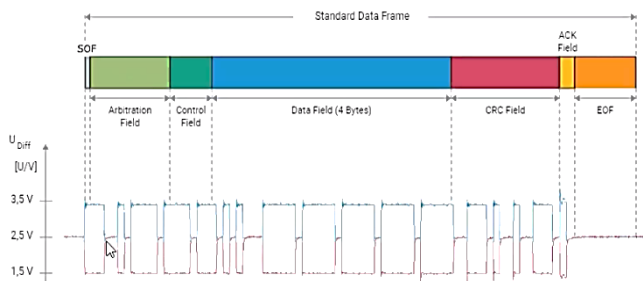
DEL : (1-bit) CRC delimiter

- دي عشان ادي ال MC ياخدر فعل
- تسبب وقت لل Peripheral انه ياخذ وقت عشان يخلص الى معاه (يحسب ال CRC)

ACK

- اللى هيرد عليا ب zero ده ال listener على الأقل لازم واحد يرد عليا-

Physical Transfer of a Data Frames in Standard Format



طب امتي مش هيرد عليا ب ACK ... ؟ ؟

- لو مفيش ولا node على ال BUS عايزة الرسالة
- كل الناس استقبلت ال CRC غلط

DEL : (1-bit) ACK delimiter

- عشان لو محدث رد عليه يبقي في وقت ياخدر فعل

END of frame

- دي stop وبيقفل ال Frame وبيحط 7 BIT كلهم ONES

ITM

- بعد كده 3 BIT بين كل فريم والثاني ... مينفعش حد يتكلم فيهم لازم الكل يسكت

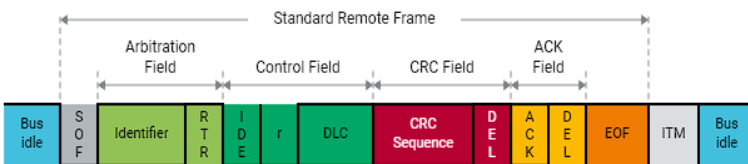
Remote Frame

❖ ده بستخدمه لو عايز اطلب داتا من باقي الMC

❖ ببعت ال Identifier bits : ودي هي ال ID بتاعت الرسالة اللى انا عايزها
كل الناس على الBUS بتشوف انه Remote Frame بيقره عادي (broadcasting)
بس اللى بيستقبله ويرد عليه هو صاحب الرسالة

- هو مش بيبيعلي الداتا في نفس الفريم بتاعي زي ما كنا بنعمل في ال I2C ويرد عليا في نفس الفريم لأنه معندناش Slaves هنا
- إنما هنا انا بطلب وبقوله عايز الداتا دي وهو يرد عليا ب ACK بس في الفريم بتاعي

Standard Remote Frame

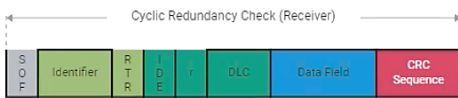


❖ RTR

- دي زي ال Read/Write
- بتكون بواحد عشان عايز داتا ... Read

❖ DLC

- بتبقي ب Zero عشان انا مش عايز ابعت داتا



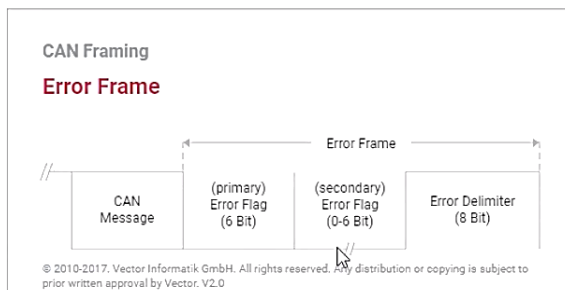
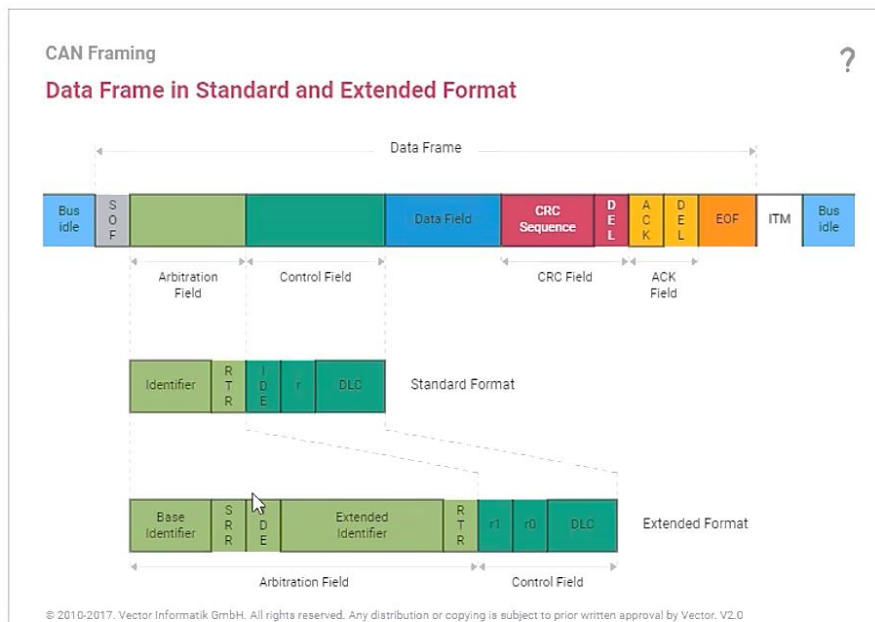
❖ **CRC** : ده بيتعمل عن طريق ال CAN Controller مش بال Software

- **طب ليه** ببعت CRC وانا مش ببعت داتا أساسا !!!
- ده لأنه مش بيتعمل على الداتا بس ده بيتعمل على كل الحاجات اللى في الفريم

🌀 بعد ما ببعت ال Remote Frame مين بيرد عليا ويبعتلي الداتا ؟

- المفروض ال MC اللى صاحب الرسالي دي يبعلي Frame جديد فيه الداتا اللى انا طلبتها منه
- ممكن الرسالة يكون ليها كذا Listener بس مينفعش يبقي ليها كذا Owner يبعها
- يعني كله هيشوف الرسالة بس واحد بس اللى هيرد عليا عشان ميحصلش Conflict

وده بيحصل عن طريق ان ال Can Controller بيستقبل ال Remote Frame بعد كده المفروض هنا يبعلي Interrupt انه استقبل الرسالة اللى عنوانها كذا



Error Frame

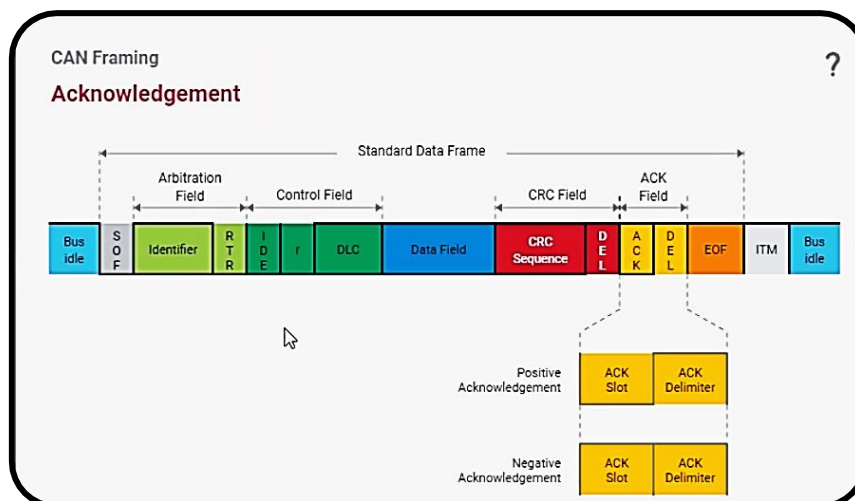
لما بيكون في error ، بيعت لكل الناس ان في error

بيعت Error flag وده بيكون 6 اصفار بعدين 6 اصفار ثاني

بعدين بيعت Error Delimiter وده بيكون 8 bit كلهم 1

عشان اسكت واسيب ال BUS زي ما كان في ال IDLE

الاحتمالات اللى ممكن تحصل



❖ فكرة ال Arbitration

- ان ال MC جواه Unit بتعمل Check لل Data اول ما تلاقي داتا مختلفة عن اللى هي بعثها تفصل نفسها وتبطل تبعت داتا عشان متبوظش ال Frame
- اللى بيخسر بيكون عارف ان الداتا موصلتش فلما ال BUS بيفضي هو بيبعتها ثاني

❖ احتمال ان واحد بعث الرسالة رقم 5 وواحد رقم 6 .. مين هيكسب

- رقم 5 عشان العنوان الأصغر هو اللى بيكسب بسبب عملية ال ANDing اللى على ال BUS

❖ احتمال ان اتنين يبعثوا Data Frame نفس الرسالة في نفس الوقت دي مش هتحصل

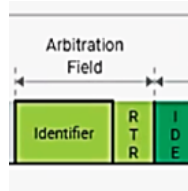
- لأن الرسالة ملهاش غير Owner واحد بس لأن الرسائل ليها عناوين وكده

❖ لو الاتنين طلبوا نفس الرسالة في نفس الوقت Remote Frame ؟

- مش هتحصل مشكلة لأنهم ال MC هتوصله نفس الرسالة انه لازم يطلع رسالة رقم كذا
- والاتنين دول هيكونوا مستنين وقاعدين يستقبلوا

❖ احتمال ان الاتنين يبعثوا في نفس الوقت بس واحد يبعث Data Frame والثاني يبعث Remote Frame

- هيكسب اللى يبعث ال Data Frame لأن ساعتها ال arbitration هتحصل على ال BIT بتاعت RTR ودي بتكون ب Zero ودي بتكون مسئولة عن تحديد نوع ال Frame



❖ فكرة ان ال arbitration مش بيعدي الحجة دي

- عشان انا مش محتاج اعمل arbitration غير على ال address
- وحوار ال Read/Write

❖ مينفعش في ال CAN كل شوية يديك interrupt ان في داتا جت ... طب دي حلوها زاي ؟

- لأن كل الناس بتسمع الرسائل في نفس الوقت (Broadcasting)

❖ لكن هيديك interrupt لما تيجي رسالة من الرسائل بتاعتك

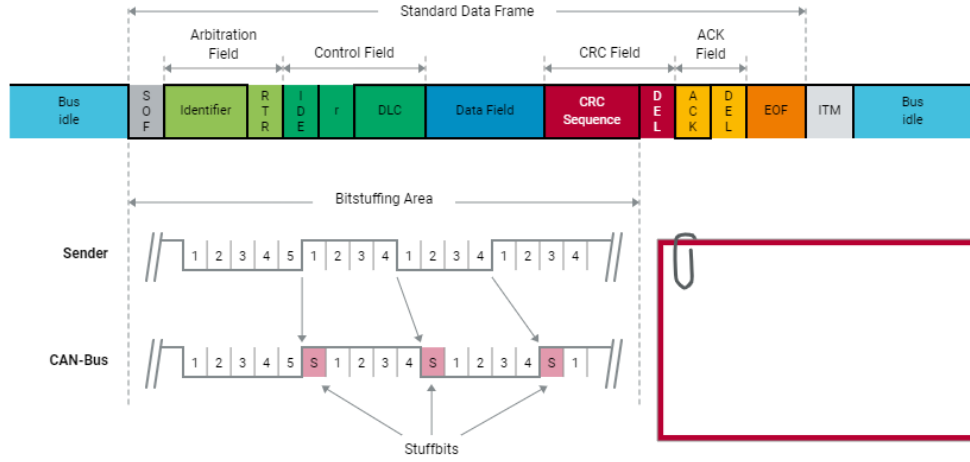
فانا هعرفه في ال Config .. اى هي الرسائل بتاعتي والكلام ده

عشان يضمنوا سلامة وصحة الداتا

❖ عملوا حاجة اسمها Bit Stuffing

- اى Can Controller عايز يبعث داتا .. مينفعش يبعث 6 Bits زي بعض ورا بعض
- طالما جه 5 BITS زي بعض السادسة بتكون عكسهم 000001

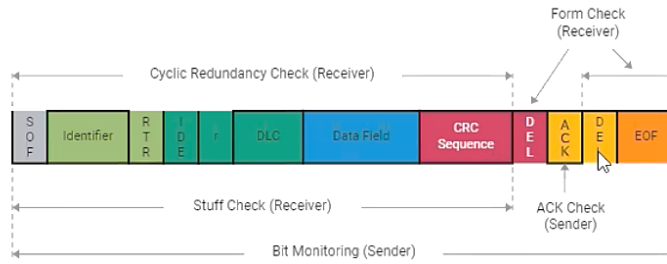
Bitstuffing



طب لو عايز ابعت 5 اصفار بس .. هتعمل مشكلة؟؟

- لأ طبعا لأن اللي هيستقبل عارف حجم الداتا وعارف ان ال BIT الزيادة دي Bit Stuffing
- فلما يستقبل الرقم ده بيكون عارف ان stream of data ماشي صح وان الواحد ده مش تبع الداتا
- فلما الاقي 6 BITS ورا بعض زي بعض ... يبقى كده في مشكلة وده شكله ال Error Frame اصلاً

أنواع ال Errors في ال CAN



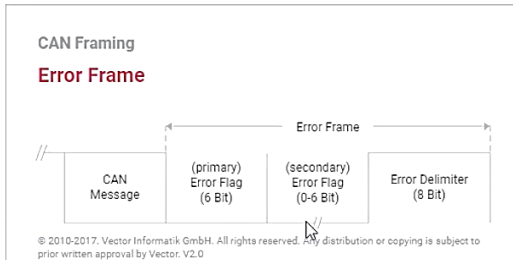
Bit Monitoring

❖ ان انت كـ Sender بتبعت الداتا وتقرأها ولو لقيت Bit مش زي اللى بعتته بيحصل الاتي

- لو انت في ال arbitration Field يبقى هتسكت وتخسر ال arbitration
- لو انت لقيت داتا غير اللى بتبعتها في اى جزء تاني غير ال (arbitration field) يبقى هيديني ان حصل

Error

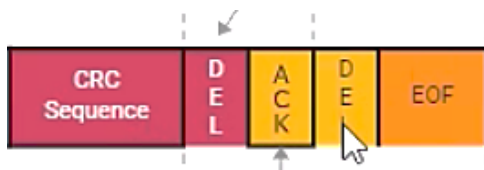
🌀 فاللى بيعرف الخطأ ده هو Sender



Form Check

❖ ده عامل زي ال Frame Error بتاع ال UART بالظبط

انا عندي شوية حاجات في ال Frame لازم تكون بـ 1 ... فانا بستغلهم عشان اتأكد من الداتا



- زي ال CRC delimiter
- وال ACK delimiter
- والد End of Frame

فكده لو ال 3 دول فيهم مشكلة يبقى ال Frame غلط ... عشان كده اسمه Form check عشان شكل ال Frame هيبنى غلط اصلاً

🌀 فاللى بيعرف ال 3 أخطاء دي ويتأكد منها هو ال Receiver

Cyclic Redundancy Check (CRC Check)

- ده هنا بجمع الداتا واشوف مجموعهم ولو فيه اختلاف يبقي الداتا دي بايظة

فالى بيعرف الخطأ ده هو ال Receiver

39. What is CRC error?

- Transmitter node calculates a 15-bit Cyclic Redundancy Checksum (CRC) depending upon the message carried by the data field.
- The receiver node will also generate a 15 bit CRC checksum, then it compares with the received one.
- If both varies the CRC error is flagged.

ACK Check

- لو انا بعت داتا وكل الناس مردتش فمحدث هيبتلي ACK (1)
- لكن لو واحد بس استلم الداتا سليمة هيبتلي ACK (0)

فساعتها لو لقيت مفيش ACK يبقي انا الى بعت الداتا هقول انا غلطان او في مشكلة واطلع ACK error

فالى بيعرف الخطأ ده هو ال Sender

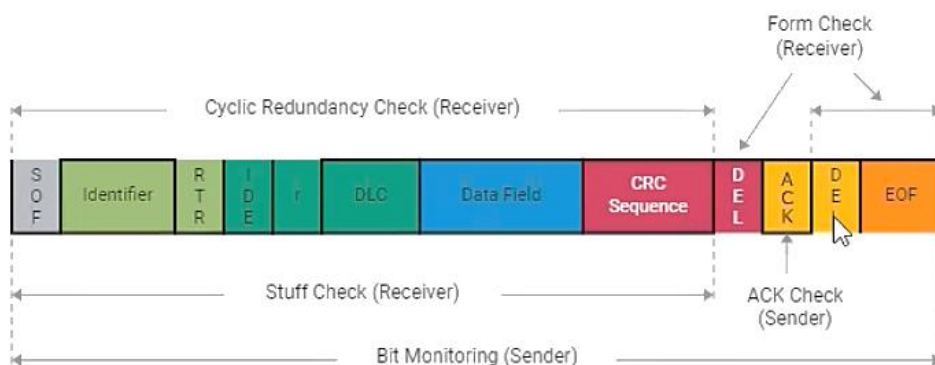
Stuff Check

❖ ده بيحصل لو في 6 bit شبه بعض

فالى بيعرف الخطأ ده هو ال Receiver

Data Protection in the CAN Network

Error Detection



- ❖ أي خطأ من الأخطاء التي فاتت دي بتطلع على هيئة Active or Passive Errors
- لو انت ك MC بتطلع errors كتير يبقى انت فيك مشكلة وهتطلع Passive Errors ومش هتأثر على ال BUS
- لو انت نشيط وشغال تبعت وتستلم صح فلما تطلع هتطلع Active Errors
- طب الكلام ده معناه أي بالظبط ... ؟!

افرض عندي Peripheral متوصله على ال BUS بس فيها مشكلة و أي داتا ببعثتها ليها بتعتبرها غلط وتطلع Error Frame وكده هي بتبوظ الداتا ومعطله ال BUS

❖ فهما قالوا هنعمل نوعين من الأخطاء :

Passive Error Frame - Active Error Frame

- ال Active Error Frame ده اللي هو 6-BITS كلهم اصفار
- ال Passive Error Frame ده اللي هو 6-BITS كلهم 1 - وده كأن واحد عمال يتكلم وتروح عامله Mute

🌀 طب هنحل زاي المشكلة دي وزاي هنطلع Error سواء passive or active ؟

هما عملوا اتنين Counter اسمهم :

each CAN controller has a TEC (Transmit Error Counter) and a REC (Receive Error Counter)

وده هو الحل عشان اتخلص من الأخطاء ...

❖ انا هعمل 2 Counters Units ... واحد لل Receiver وواحد لل Transmitter

❖ ال Units دي كل ما تلاقي error تزود واحد في ال Counter بتاعها حسب نوع ال error جاي منين .. من Receiver or Transmitter

- فلو ال Counter ده عدي ال 128 بيحول ال Error بتاعه ل Passive error Frame وكده مش هياثر على ال BUS هيفضل يقول ان الداتا غلط بس محدش هيسمعه كأني عملته Mute
- ولو ال Counter ده عدي ال 255 بي فصل نفسه تماماً من ال BUS وساعتها يروح صيانة

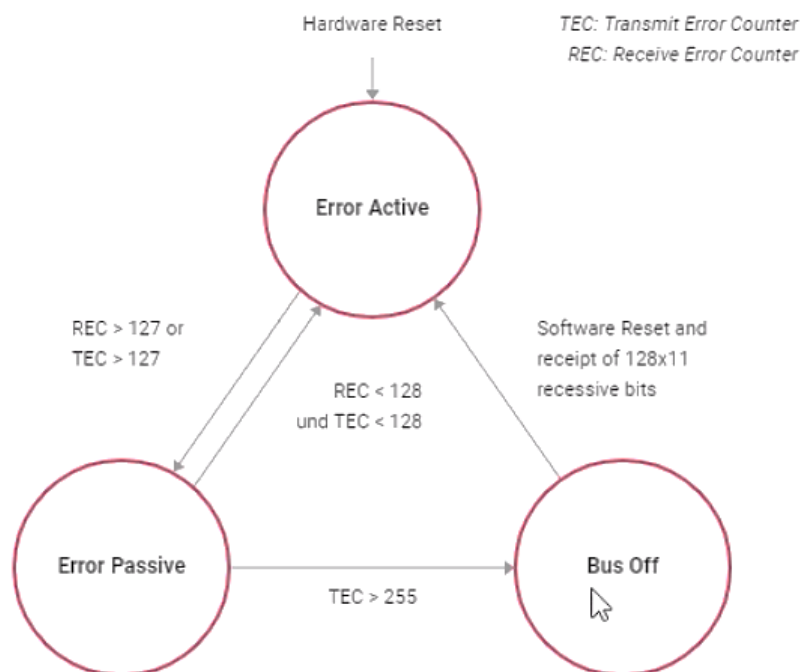
🌀 طب هل اللي عملته passive error وخليته Silent .. هيفضل متعاقب كده علطول !؟؟

- ال Transmitter لو بعت داتا صح هنعمل counter--
- وال Receiver لو استلم داتا صح هنعمله -- counter

فانت لو استلمت داتا مرة غلط ومرة صح هيبقي مرة C++ ومرة C وبكده عمرك ما هتوصل 127 لكن لما بتوصل لل 127 انت كده مش طبيعي ول عديت 255 انت كده اتحرقت ولازم تتغير

Data Protection in the CAN Network

Error Tracking



❖ وال transmitter امتي بيكتشف الخطأ .. في ال ACK check وال BIT Monitoring