1. CAN設備:

* CAN-bus（Controller Area Network bus）是一種串行通信協議，主要用於車輛中的電子控制單元（ECU）之間進行數據交換。
* CAN協議:CAN-Bus傳輸是需要按照一定協議執行的，CAN協議提供5種偵格式進行傳輸(如下表一)。

(表一:CAN**幀**格式及其用途)

|  |  |
| --- | --- |
| **幀類型** | **幀用途** |
| 數據幀 | 用於CAN節點單元之間進行數據傳輸的幀 |
| 遙控幀 | 用於接收單元向具有相同ID的發送單元請求數據的幀 |
| 錯誤幀 | 用於當檢測出錯誤時向其他單元通知錯誤的幀 |
| 過載幀 | 用於接收單元通知其尚未做好接收準備的幀 |
| 間隔幀 | 用於將數據幀及遙控幀與前面的幀分離開來的幀 |

1. CAN-FD: CAN-FD是在傳統CAN協議基礎上的一次重大改進，通過提高數據速率和增大數據負載，滿足了現代汽車和工業應用中對高效數據通信的不斷增長的需求(如下表二)。

(表二:CAN-FD對CAN-Bus的優化項目)

|  |  |
| --- | --- |
| **項目** | **說明** |
| **控制位** | **EDL**：指示CAN幀或CAN FD幀，標準CAN中為顯性，CAN FD中為隱性。 **BRS**：速率轉換開關，隱性時轉換為可變速率，此時速率可變，BRS到CRC之間使用可變速率，顯性時速率不變。 **ESI**：錯誤狀態指示位，隱性時表示節點處於被動錯誤狀態，顯性時表示節點處於主動錯誤狀態 |
| **優化CRC算法** | 傳統CAN使用位填充的方式來保持同步，這種方法會對CRC造成干擾，導致錯幀漏檢。 CAN FD將填充位加入到差錯校驗碼中進行計算，也就是以填充位的位流進行計算 |
| **主要目的** | 應對車載電子對通信帶。 |

1. CAN驅動實作:

* 首先需要配置Kernel，Enable CAN驅動。接著打開buildroot圖形化介面啟用iproute2和can-utils工具。
* iproute2是一個強大的網路它提供了一套豐富的命令管理工具，用於配置和控制Linux內核的網路子系統。
* can-utils是一組用於Linux環境下的CAN總線工具，提供了配置、診斷和調試CAN總線的實用程序。

1. Linux Kernel CAN編譯:

* 由於STM廠商已經將CAN驅動程式把編寫完成存放在Linux Kernel Source Code中，所以只要透過圖形化配置啟用編譯CAN驅動程式碼即可使用。
* 這個動作相當於在Linux Kernel的KBuild中加入CAN驅動程式編譯。