



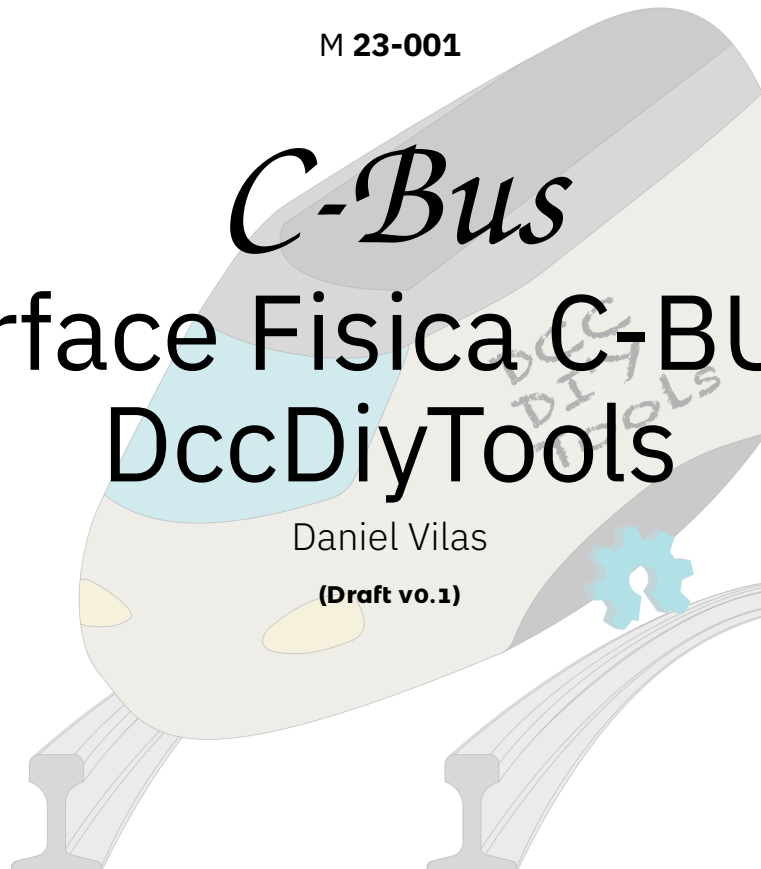
M  
**23-001**  
Draft v0.1

M 23-001

# *C-Bus* Interface Fisica C-BUS en DccDiyTools

Daniel Vilas

(Draft v0.1)



Esta obra está bajo una licencia Creative Commons “Reconocimiento-CompartirIgual 4.0 Internacional”.



# 1 Introduccion

C-Bus es un standard LCB usado y promocionado por MERG®. A bajo nivel utiliza un Bus CAN como transporte fisico de datos entre modulos (electronicos).

La idea de despliegue es usar una topologia de bus:

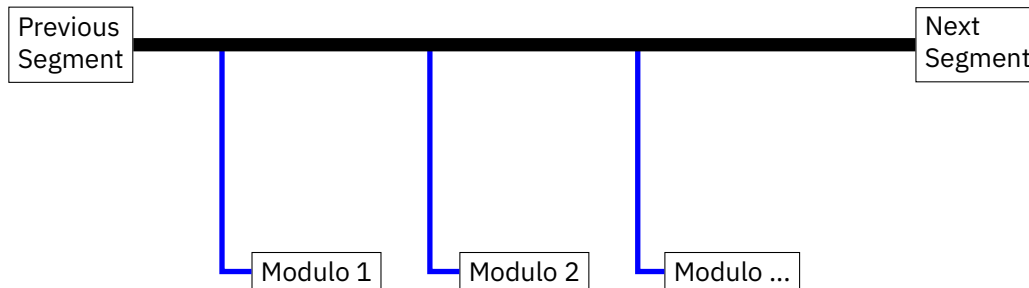


Figura 1: C-Bus Segmento

Al final de un segmento puede haber otro segmento, un repetidor, un convertidor a Ethernet,... Desde el bus a los dispositivos es necesario tener un latiguillo.

## 2 Especificacion MERG

## 3 Topologia

CBus<sup>1</sup> ha sido pensado para seguir lo más fielmente posible una topologia de Bus y en los extremos una resistencia de  $120\ \Omega$ . <sup>1</sup>Usando CAN

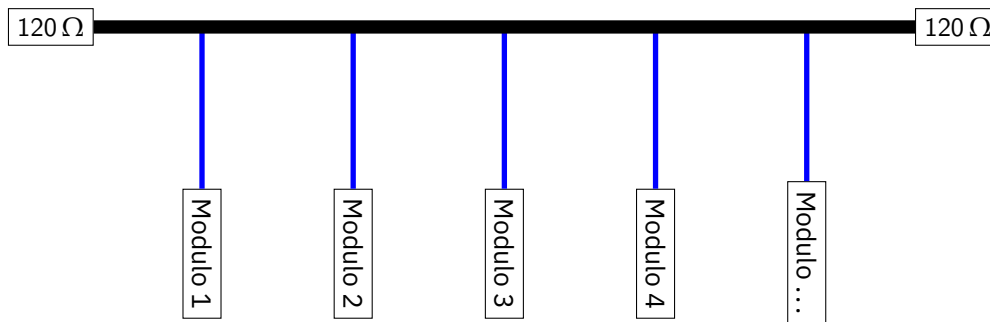


Figura 2: Topologia Bus

Pero a su vez permite algunas ramificaciones siempre y cuando la resistencia entre los conductores diferenciales sea aproximadamente  $60\ \Omega$ . Ya que todos los modulos estaran conectados en paralelo a los 4 conductores.

Ademas CBUS, en general, permite utilizar otros sistemas de transporte de la informacion como puede ser TCP, UDP, MQTT,... Pudiendo haber multiples Buses y segmentos unidos formando una red mayor.

### 3.1 Red

Recordemos que el objetivo de CBUS es poder controlar una maqueta de tren de forma digital. Es decir desde un panel de mandos enviar señales para que suceda un cambio en la maqueta, o al revés, o ante un evento en la maqueta <sup>2</sup> se actualice nuestro panel de control.

El panel de control puede ser físico, (con lucecitas y botones) o bien puede ser una pantalla de un programa como JMRI <sup>3</sup>

El caso mínimo de uso CBus tendra un bus CAN con unos pocos modulos, partiendo el cable donde sea necesario y como mucho un CanUSB o CBusServer para conectar con JMRI. Vease 2. El caso más complejo requería de uno o varios CBusServer cada uno conectado a uno o varios Buses CAN. El/los CBusServer/es se comunicarian mediante TCP<sup>4</sup> entre si y distribuyendo así los eventos entre todos los buses CAN.

Este último caso lo podemos representar con dos buses CAN donde "Modulo 2" sería el CBus-Server y JMRI se conecta a través de Wifi o un CanUSB.

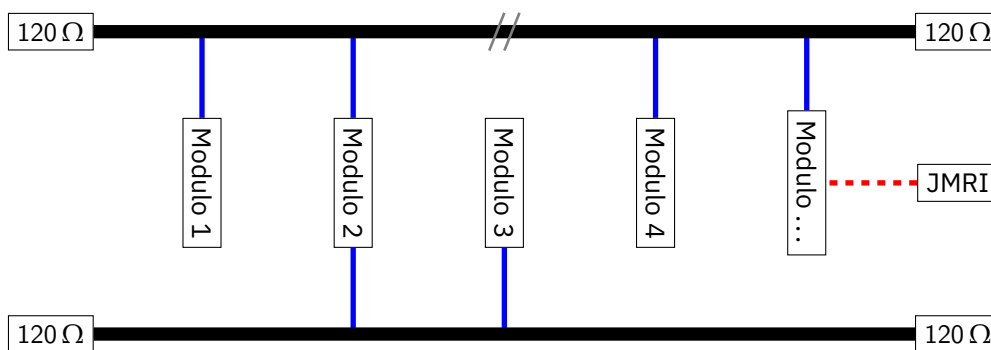


Figura 3: Topologia Bus Ejemplo Red

La red CBUS es pues el conjunto de todos los dispositivos CBUS que se pueden comunicar entre si usando el protocolo CBUS

### 3.2 Segmento vs Bus vs Red vs CBusServer

Sirva este apartado para aclarar y definir estos terminos. Puesto que esto son muy parecidos a una instalacion TCP/IP.

#### 3.2.1 Bus

Como ya se tiene una idea de lo que es la Red, empezaremos con el Bus. Siendo este el canal que permite a todos los dispositivos conectados físicamente al mismo comunicarse con una misma tecnología. Hoy por Hoy solo hay dos disponibles <sup>5</sup>

- **CAN Bus:** Utilizar un par diferencial CAN como comunicacion.
- **CBusServer:** Los dispositivos se conectan mediante una conexion TCP.

De cada bus puede haber varias instancias, que no tienen que comunicarse de por si.

En resumen un Bus es un canal por lo el cual varios dispositivos se comunican entre si, sin tener que cambiar de medio.

Los buses pueden unirse o partirse, lo que hablaremos de redes o de segmentos. Si para unirlos es necesario una logica de transformacion de datos o de medio físico, estaremos ante una red.

<sup>2</sup> Cortos, ocupacion por un tren,...

<sup>3</sup> JMRI o similar, usaremos JMRI por preferencias del autor

<sup>4</sup> Wifi, Ethernet,...

<sup>5</sup> según este punto de vista

### 3.2.2 Segmento

Cuando un bus lo podamos cortar en varios trozos fisicos hablamos de segmentos. En el caso de un Bus Can, al final son dispositivos conectados entre si mediante un par de cables (H y L)<sup>6</sup>. Por lo que estos cables se pueden cortar en segmentos. Por otra parte CBusServer puede ejecutarse en varias maquinas (cada una una instancia) y configurase para que se hablen entre si, pero al final los clientes se conectaran a una sola maquina.

<sup>6</sup>y como mucho Vcc y GND

En resumen un segmento es cada una de las partes en las que un bus se puede partir sin perder la caracteristica de bus.

### 3.2.3 Red

Cuando nos encontremos con varios buses y tengamos que comunicarse entre ambos, nos encontramos con el concepto de Red. Este concepto aparece cuando al unir dos segmentos aparece alguna de estas casuisticas:

- **Cambio de Medio:** Por ejemplo al pasar de CAN a TCP. Esta es una razon un poco debil, por que por ejemplo un repetidor CAN"segun como este echo puede cambiar de medio momentaneamente, o lo mismo un firewall (que no deje pasar tramas no CBUS, como OpenLCB). Y segun lo estrictos que seamos puede ser dos buses o no.
- **Transformacion de datos:** Si los eventos deben ser transformados (cambiar ids, agrupar,...)
- **Diferentes Funciones:** Para cada bus hemos definido una funcion diferente como puede ser eventos locales a un modulo o globales a toda la maqueta. O un bus para traccion y otro para accesorios.

En la implementacion Merg, no existe este concepto, puesto que no lo han necesitado. Tener una Red, implica tener varios buses, con la consecuencia de tener que gestionar la red. Saber para que es cada uno de los buses, asignar identificadores,... Pero nos abre la puerta a diseños más complejos y optimos.

### 3.2.4 CBusServer

CBusServer se corresponde con un protocolo de aplicacion sobre TCP/IP para enviar tramas CBus entre aplicaciones informaticas y como esta vision forma una tecnologia de Bus.

Pero a su vez es un Software de Servidor donde las diferentes aplicaciones clientes se conectan con TCP y se encarga de enviar los paquetes CBus que recibe de un cliente al resto. Este software puede usar el Modulo CanUSB4 de Merg y/o el Modulo CanPiHat para conectarse a una Red CBus directamente.

## 3.3 Ejemplo Practico 1 - Dos Habitaciones

Imaginemosnos una maqueta dividida en dos grandes zonas o habitaciones y con un ordenador central para mostrar el estado general de la maqueta.

Como todo en la vida hay multiples soluciones, que implican desde un bus hasta una red de 4 o más buses.

### 3.3.1 Solucion de un solo bus

Una primera solucion podria ser tener todos los accesorios conectados a un solo bus, tanto de una zona como de la otra.

En esta solución depende que en conjunto no se supere la longitud máxima de cableado y de dispositivos conectados<sup>7</sup>. Pero aun en el muy probable caso de que no, existe otra limitación, que es el número de eventos y la seguridad que los eventos de una zona son ignorados por los dispositivos de la otra.<sup>8</sup>

<sup>7</sup> Muchos metros y decenas de dispositivos

<sup>8</sup> Como mucho interesa algún error para enviar trenes de una zona a la otra

## 4 Cables

## 5 Conectores