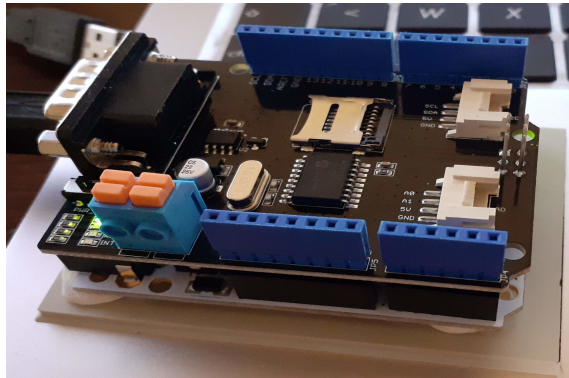


RAPPORT D'EXPERIMENTATION DU 27 FEVRIER 2025

Utilisation du Shield Sseed V2.0 directement « Out of the Box » sans toucher à rien.
Utilisation du Clone Arduino de Gotronic.



Utilisation du programme de Pierre Molinaro :

<https://github.com/pierremolinaro/acan2515/blob/master/examples/LoopBackDemo/LoopBackDemo.ino>

Lignes de programme modifiées :

```
32 static const byte MCP2515_CS = 9 ; // CS input of MCP2515 By Default
33 static const byte MCP2515_INT = 2 ; // INT output of MCP2515 By Default

45 static const uint32_t QUARTZ_FREQUENCY = 16UL * 1000UL * 1000UL ; // 16 MHz

56 Serial.begin (38400) ;

66 ACAN2515Settings settings (QUARTZ_FREQUENCY, 250UL * 1000UL) ; // CAN bit
rate 125 kb/s // Modified Duperche 250
67 settings.mRequestedMode = ACAN2515Settings::LoopBackMode ; // Select
loopback mode
```

Résultat sur la console :

```
07:11:42.971 -> Sent: 9
07:11:42.971 -> Received: 9
07:11:44.987 -> Sent: 10
07:11:44.987 -> Received: 10
07:11:47.002 -> Sent: 11
07:11:47.002 -> Received: 11
07:11:49.955 -> Configure ACAN2515
07:11:49.955 -> Bit Rate prescaler: 2
07:11:50.002 -> Propagation Segment: 5
07:11:50.002 -> Phase segment 1: 5
07:11:50.002 -> Phase segment 2: 5
07:11:50.002 -> SJW: 4
07:11:50.002 -> Triple Sampling: no
07:11:50.002 -> Actual bit rate: 250000 bit/s
07:11:50.002 -> Exact bit rate ? yes
07:11:50.002 -> Sample point: 68%
07:11:50.049 -> Sent: 1
07:11:50.049 -> Received: 1
07:11:51.972 -> Sent: 2
07:11:51.972 -> Received: 2
07:11:53.987 -> Sent: 3
07:11:53.987 -> Received: 3
```

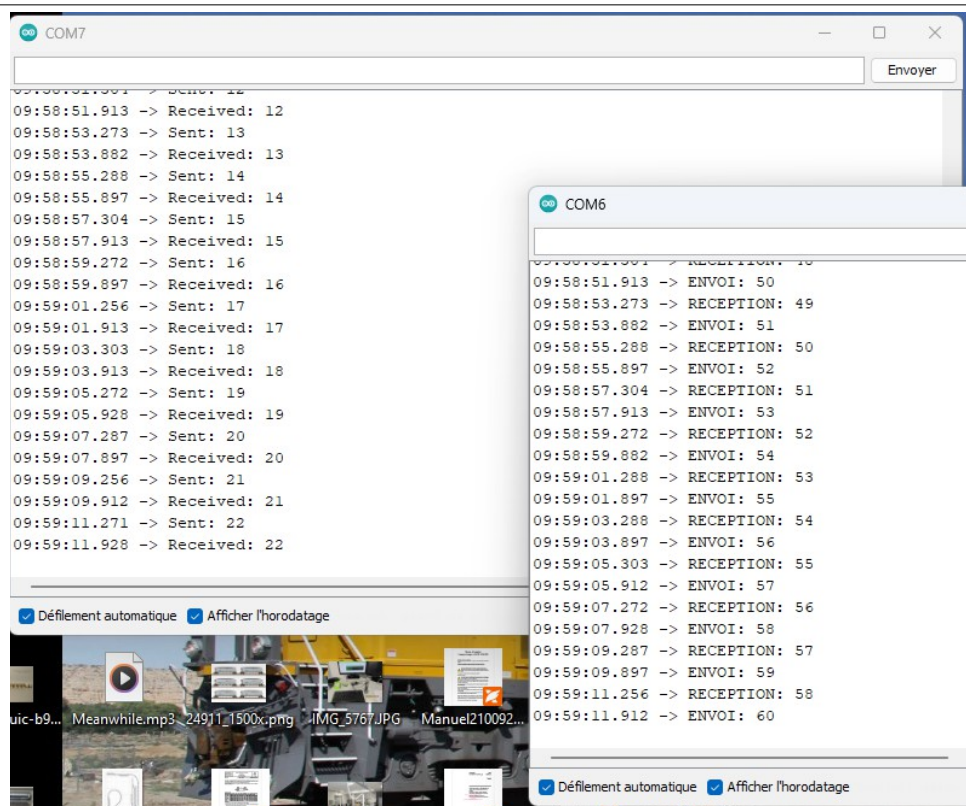
Puis, utilisation de deux CAN-BUS pour émission et réception simultanées (Half-Duplex?)



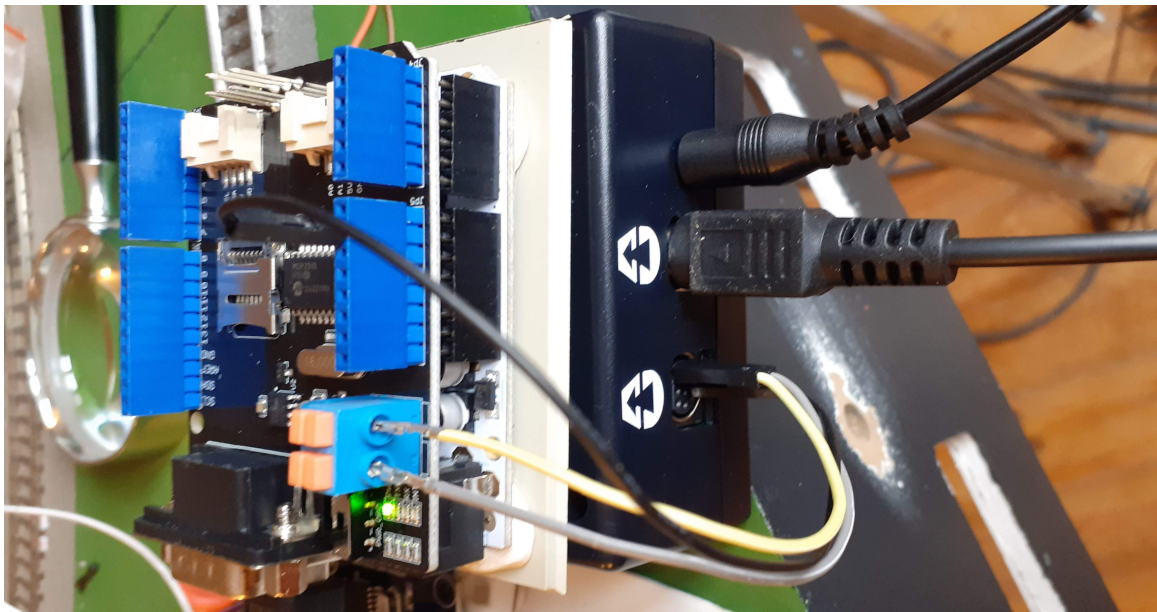
A partir des programmes EN et FR :

Mode_No_LoopBack_FR.ino version retouchée pour les paramètres CS et INT
Mode_No_LoopBack_EN.ino version non modifiée.

Résultats :



Puis, Utilisation de la carte pour surveiller le Can-Märklin :



Avec le Programme de Bobille Christophe, légèrement adapté :

```
// Bobille Cristophe du 27-02-2025
#include <ACAN2515.h>

static const byte MCP2515_CS = 9; // CS input of MCP2515
static const byte MCP2515_INT = 2; // INT output of MCP2515

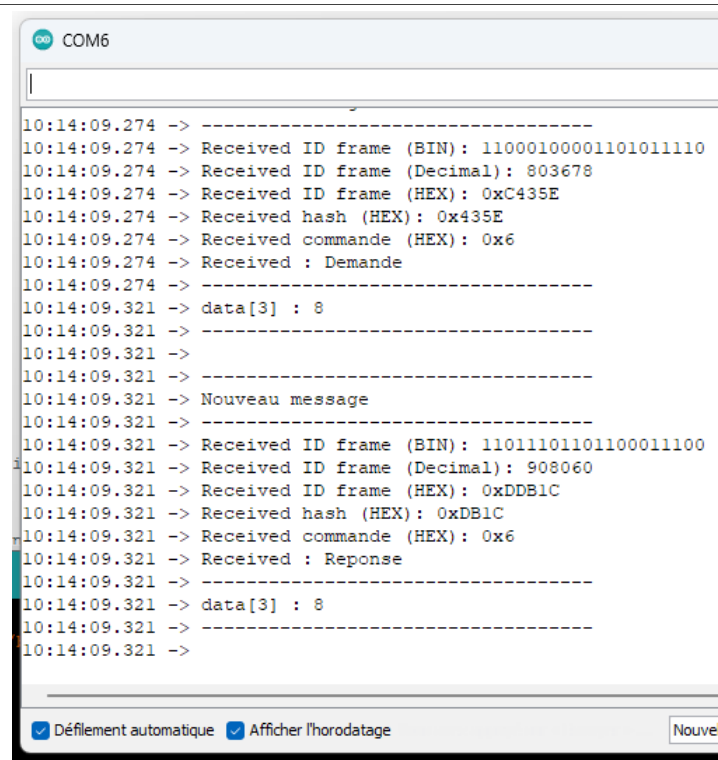
//-----
//  MCP2515 Driver object Sur SEEED V2.0
//-----

ACAN2515 can(MCP2515_CS, SPI, MCP2515_INT);

//-----
//  MCP2515 Quartz: adapt to your design
//-----

static const uint32_t QUARTZ_FREQUENCY = 16UL * 1000UL * 1000UL; // 16 MHz
```

Avec comme résultats :



```
COM6
10:14:09.274 -> -----
10:14:09.274 -> Received ID frame (BIN): 110001000001101011110
10:14:09.274 -> Received ID frame (Decimal): 803678
10:14:09.274 -> Received ID frame (HEX): 0xC435E
10:14:09.274 -> Received hash (HEX): 0x435E
10:14:09.274 -> Received commande (HEX): 0x6
10:14:09.274 -> Received : Demande
10:14:09.274 -> -----
10:14:09.321 -> data[3] : 8
10:14:09.321 -> -----
10:14:09.321 -> -----
10:14:09.321 -> Nouveau message
10:14:09.321 -> -----
10:14:09.321 -> Received ID frame (BIN): 11011101101100011100
10:14:09.321 -> Received ID frame (Decimal): 908060
10:14:09.321 -> Received ID frame (HEX): 0xDDB1C
10:14:09.321 -> Received hash (HEX): 0xDB1C
10:14:09.321 -> Received commande (HEX): 0x6
10:14:09.321 -> Received : Reponse
10:14:09.321 -> -----
10:14:09.321 -> data[3] : 8
10:14:09.321 -> -----
10:14:09.321 -> -----
☒ Défilement automatique ☒ Afficher l'horodatage Nouve
```

Data[3] = 8 correspondant à la locomotive T3.