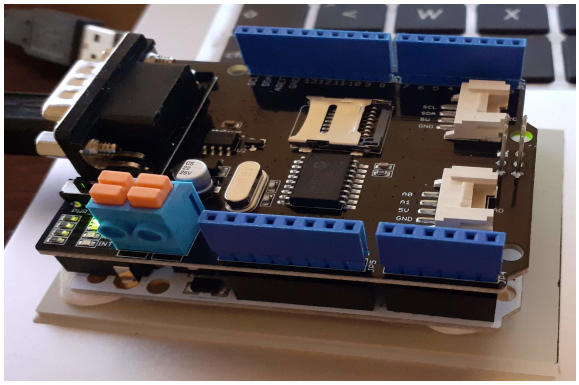
RAPPORT D'EXPERIMENTATION DU 27 FEVRIER 2025

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

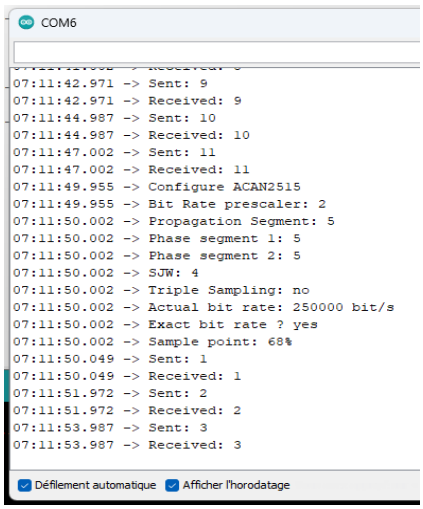


Utilisation du programme de Pierre Molinaro :
https://github.com/pierremolinaro/acan2515/blob/master/examples/LoopBackDemo/LoopBackDemo.ino

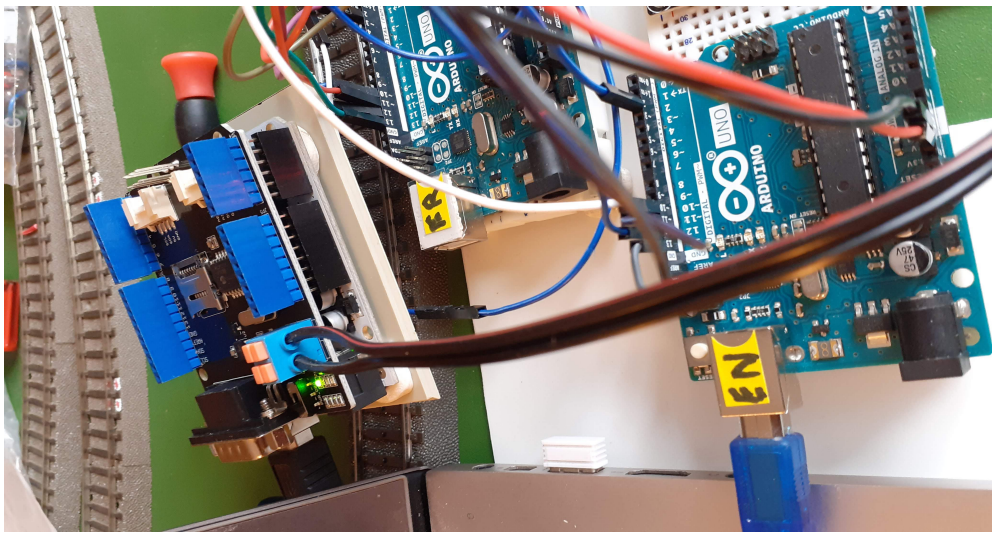
Lignes de programme modifiées :

```
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 :



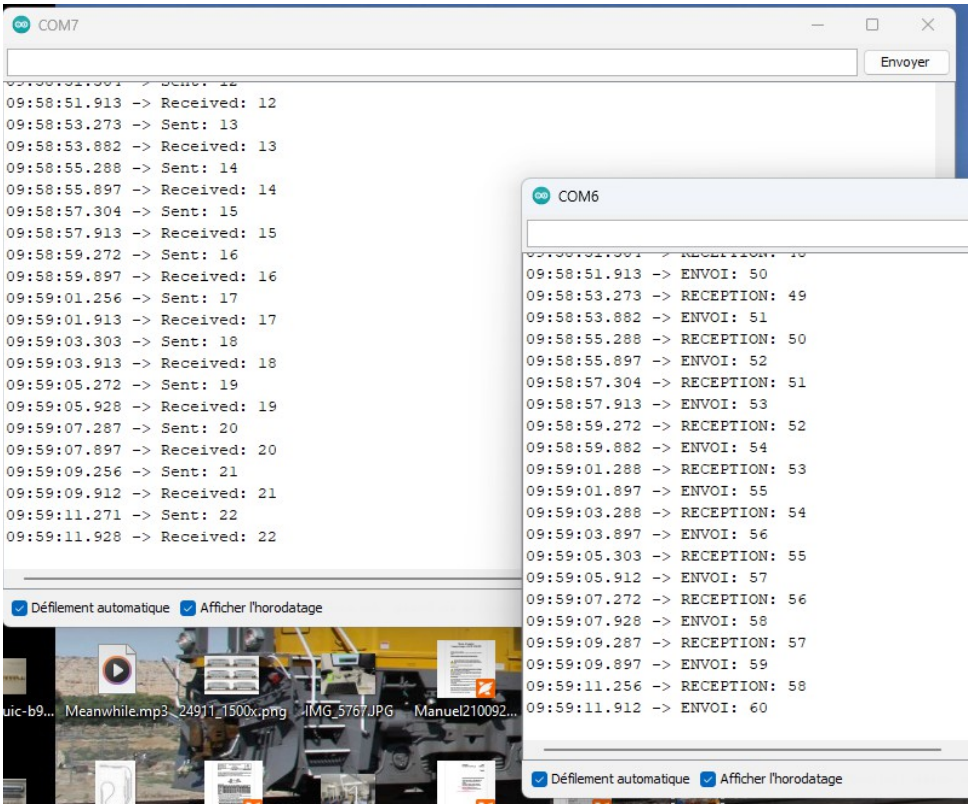
Puis, utilisation de deux CAN-BUS pour émission et réception simultanées (Half-Duplex?)



A partie 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 :

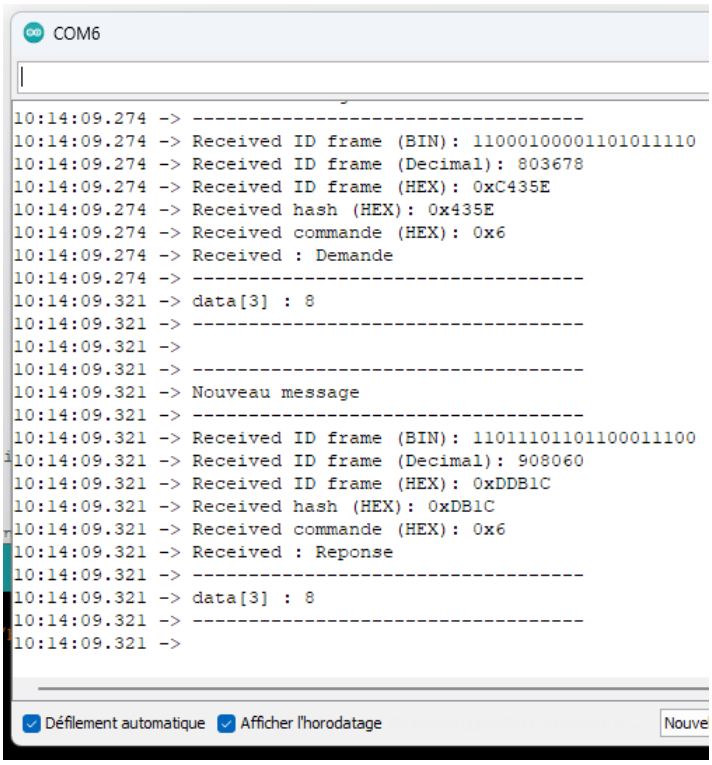


Puis, Utilisation de la carte pour surveiller le Can-Märklin :

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 :



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