

Régulation de vitesse adaptative



Remerciements

Je tiens à remercier toutes les personnes qui ont contribué au succès de mon projet et qui m'ont aidé lors de la rédaction de ce rapport.

Tout d'abord, j'adresse mes remerciements à mon professeur Mr LOMBARD Christophe de l'Université de Franche-Comté qui m'a beaucoup aidé dans ma recherche de solutions pour mon projet. Son écoute et ses conseils m'ont permis de cibler les attentes correspondantes au bon déroulement de mon projet.

SOMMAIRE

I°) Régulateur de vitesse adaptative:.....	P4-5/22
1) Principe:.....	P4/22
2) Fonctionnement:.....	P4/22
3) Certains points à garder à l'esprit:.....	P5/22
II°) Présentation du projet:.....	P6-9/22
1) Travail effectué:.....	P7-8/22
a) Substitution BSI:.....	P8/22
b) Mise en place du projet:.....	P9/22
III°) Problèmes rencontrés:.....	P10-13/22
1) Carte:.....	P10/22
IV°) Solutions retenues:.....	P14-17/22
1) Fabrication Carte et test:.....	P14/22
V°) Réalisation du projet:.....	P18-21/22
1) Prise en main liaison SPI:.....	P18/22
2) Affichage vitesse véhicule et régime moteur:.....	P19/22
VI°) Conclusion:.....	P22/22

I°) Régulateur de vitesse adaptative :

1) Principe :

Le régulateur de vitesse adaptatif (ou ACC pour *Adaptative Cruise Control*) est un dispositif évolué de régulation de vitesse qui permet de conserver une distance préétablie avec le véhicule qui le précède. Un capteur installé à l'avant mesure la distance et la vitesse relative du véhicule qui précède. Le système ACC ajuste la vitesse du véhicule afin de maintenir la vitesse de croisière choisie par le conducteur et une distance minimale prédéterminée. Si le système ACC détecte que le véhicule qui précède ralentit rapidement ou qu'un autre véhicule en provenance d'une autre voie vous coupe, le système applique automatiquement un freinage limité et peut vous avertir au moyen d'un voyant lumineux ou d'un signal sonore. Même si la plupart des systèmes ont recours à un radar, certains ont recours à la détection au laser (Lidar). Lorsque la voie est libre, le système ACC accélère automatiquement le véhicule pour rouler à la vitesse préétablie.

Plusieurs systèmes évolués ralentiront le véhicule jusqu'à l'amener à un arrêt complet en trafic discontinu.

2) Fonctionnement :

Les conducteurs établissent la vitesse désirée et la distance nécessaire pour respecter l'intervalle voulu. La portée de fonctionnement des systèmes ACC standard varie selon les modèles. La portée peut varier de 100 à 200 m selon le système et selon que l'objet est en mouvement ou à l'arrêt. Les systèmes munis d'une caméra vidéo peuvent étendre la portée à 250 m. Les systèmes ACC standards ne peuvent pas être utilisés à des vitesses inférieures à 30 km/h.

Les systèmes couplés à la fonction d'arrêt et de marche fonctionnent quant à eux des vitesses inférieures à 30 km/h jusqu'à l'arrêt complet du véhicule. Avec la fonction d'arrêt et de marche, si le véhicule qui précède s'immobilise et repart dans les 3 secondes qui suivent, le véhicule muni du système ACC suit automatiquement. Si la période d'immobilisation dure plus longtemps, le conducteur doit réactiver le système en appuyant sur l'accélérateur.

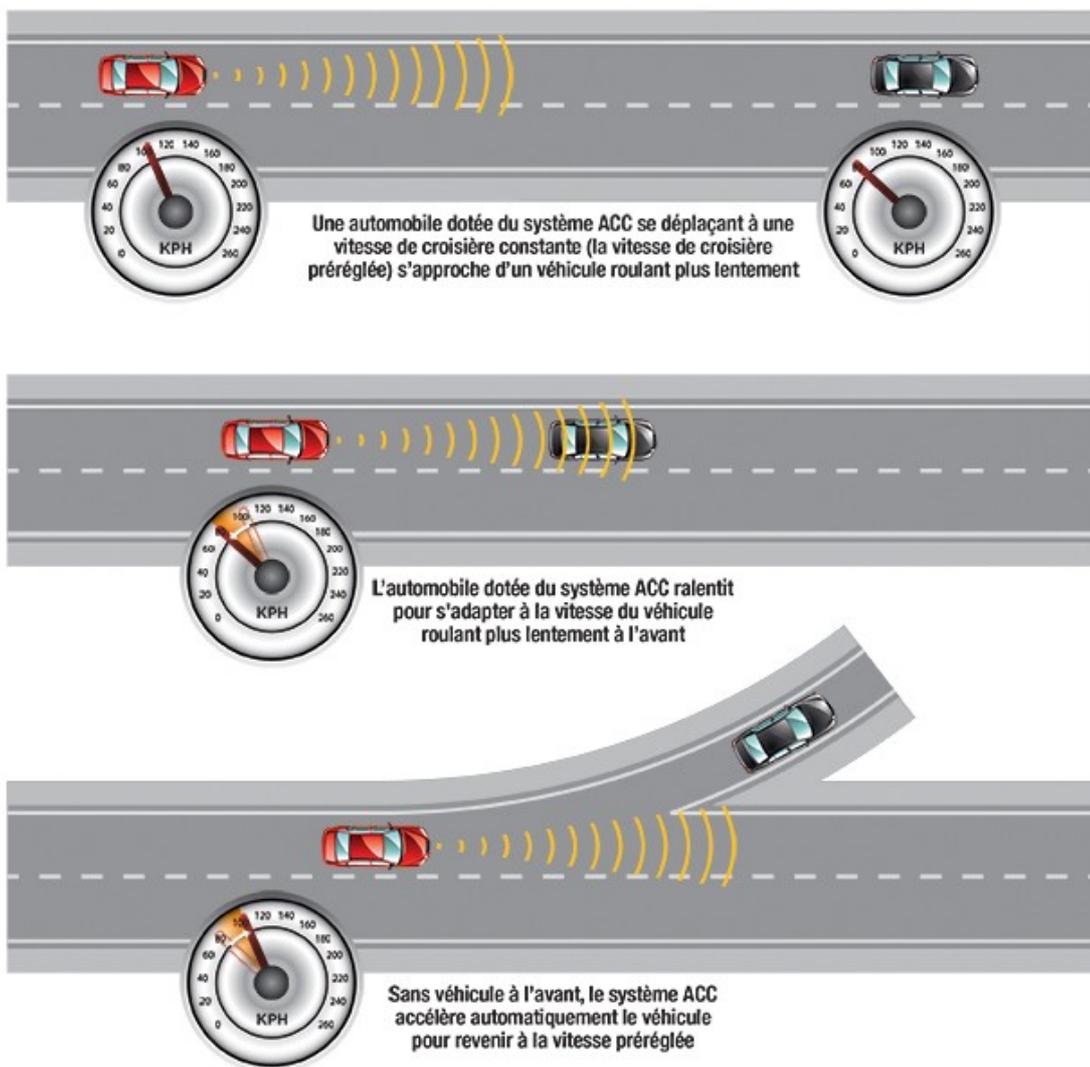
3) Certains points à garder à l'esprit :

Le conducteur doit toujours conserver la maîtrise du véhicule et porter son attention sur la conduite. Le conducteur peut accélérer ou freiner à tout moment comme avec un régulateur de vitesse standard. Le conducteur doit faire preuve de prudence lors d'un dépassement. Le conducteur peut devoir freiner pour éviter une collision si le véhicule qui précède freine brusquement, par exemple.

Les systèmes au laser ne permettent pas de détecter et de suivre des véhicules dans de mauvaises conditions météorologiques ni de détecter des véhicules très sales qui ne réfléchissent pas suffisamment de lumière.

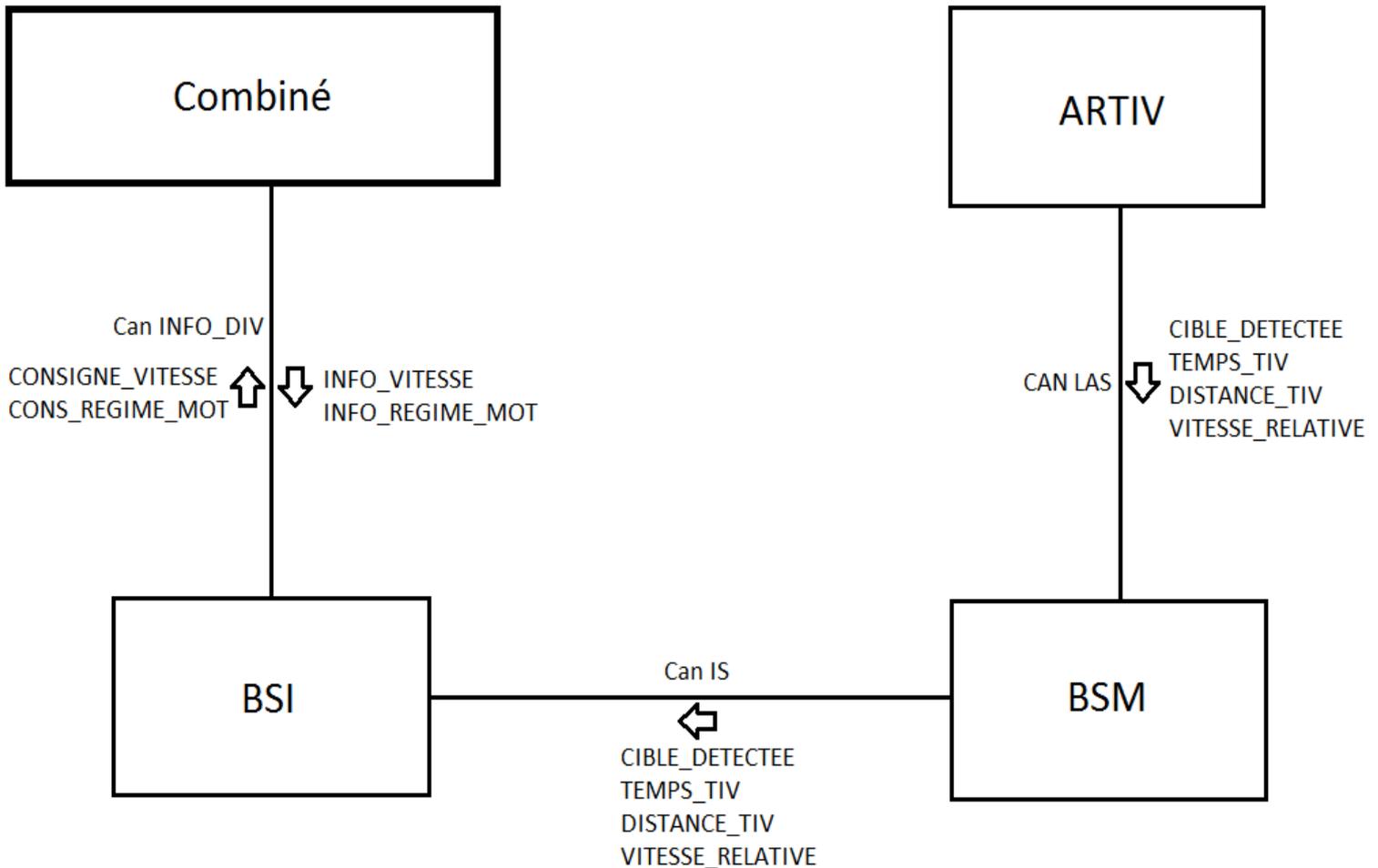
Il est possible que le système ne réponde pas dans le cas de véhicules arrêtés.

Il est donc préférable de lire le manuel du propriétaire fourni avec le véhicule pour se familiariser avec le système et les conditions dans lesquelles le fonctionnement du système est différent ou n'est pas garanti.



II°)Présentation du projet :

Le but de notre projet est de simuler la régulation de vitesse adaptative de la nouvelle Peugeot 308. Afin de le mener à son terme, il faut tout d'abord comprendre les réseaux utiles avec leurs données qui y transitent.



BSI : Boîtier de Servitude Intelligent

BSM : Boîtier de Servitude Moteur

ARTIV : Aide au Respect du Temps Inter-Vitesses

LAS : Liaison Au Sol

IS : Inter-Système

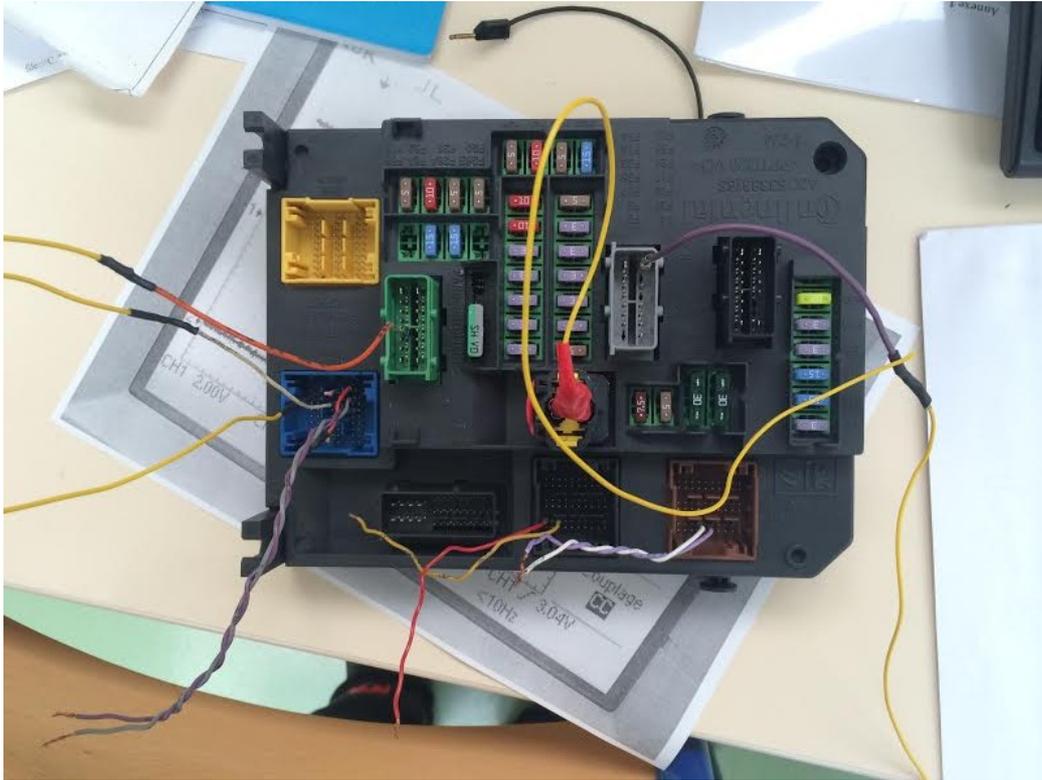
Dans notre exemple, nous simulerons le radar à partir du logiciel CANalyzer pour agir sur le combiné par l'intermédiaire de notre carte qui se substitue au BSI.

1) Travail effectué :a) Substitution BSI

Pour commencer, lorsque l'on a branché le combiné à la BSI grâce aux connecteurs (blanc et marron) ci-dessous avec une alimentation stabilisée 30V/3A en lui envoyant 12V de tension, il ne se passait rien, le combiné affichait « Mode Eco activé » mais consommait 400mA, ce problème était dû à la BSI qui n'envoyait pas de trames. Nous avons refait tous les câblages liés à la BSI afin de lui envoyer des trames pour la réveiller mais sans résultat. Nous avons donc opté pour une création d'une carte qui se substituerait à la BSI.



BSI :



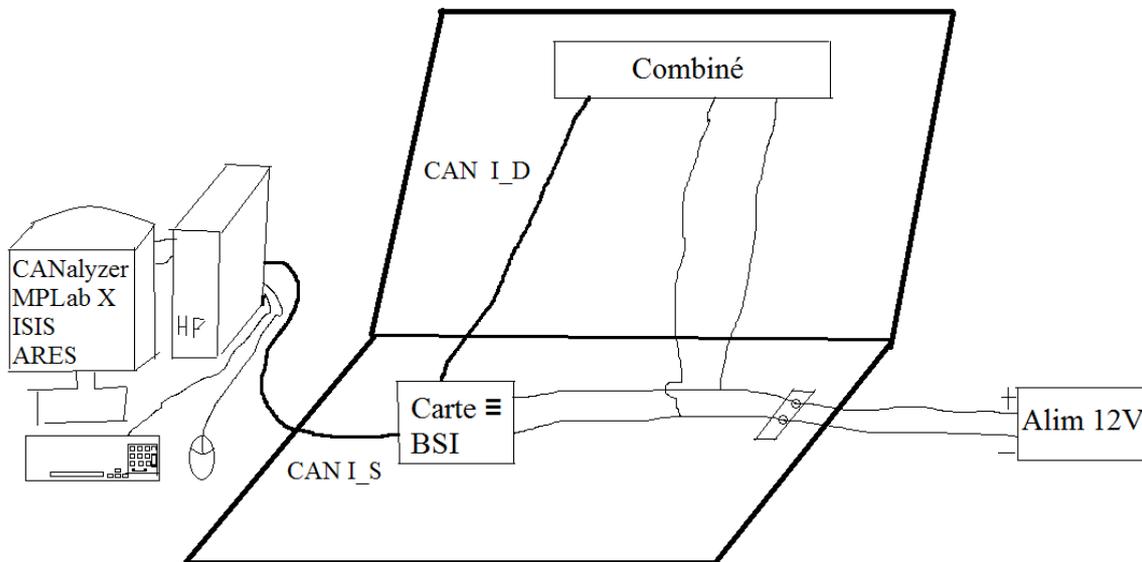
Combiné :



b) Mise en place du projet

Nous avons gardé la même présentation type « Tableau de Bord » des étudiants de l'année dernière afin de disposer d'une maquette ergonomique et facile d'utilisation. Elle est constituée de notre carte se substituant à la BSI pour pouvoir connecter un câble DB9, du combiné et de borniers d'adaptation pour l'alimentation +12V et la masse.

Schéma du montage :

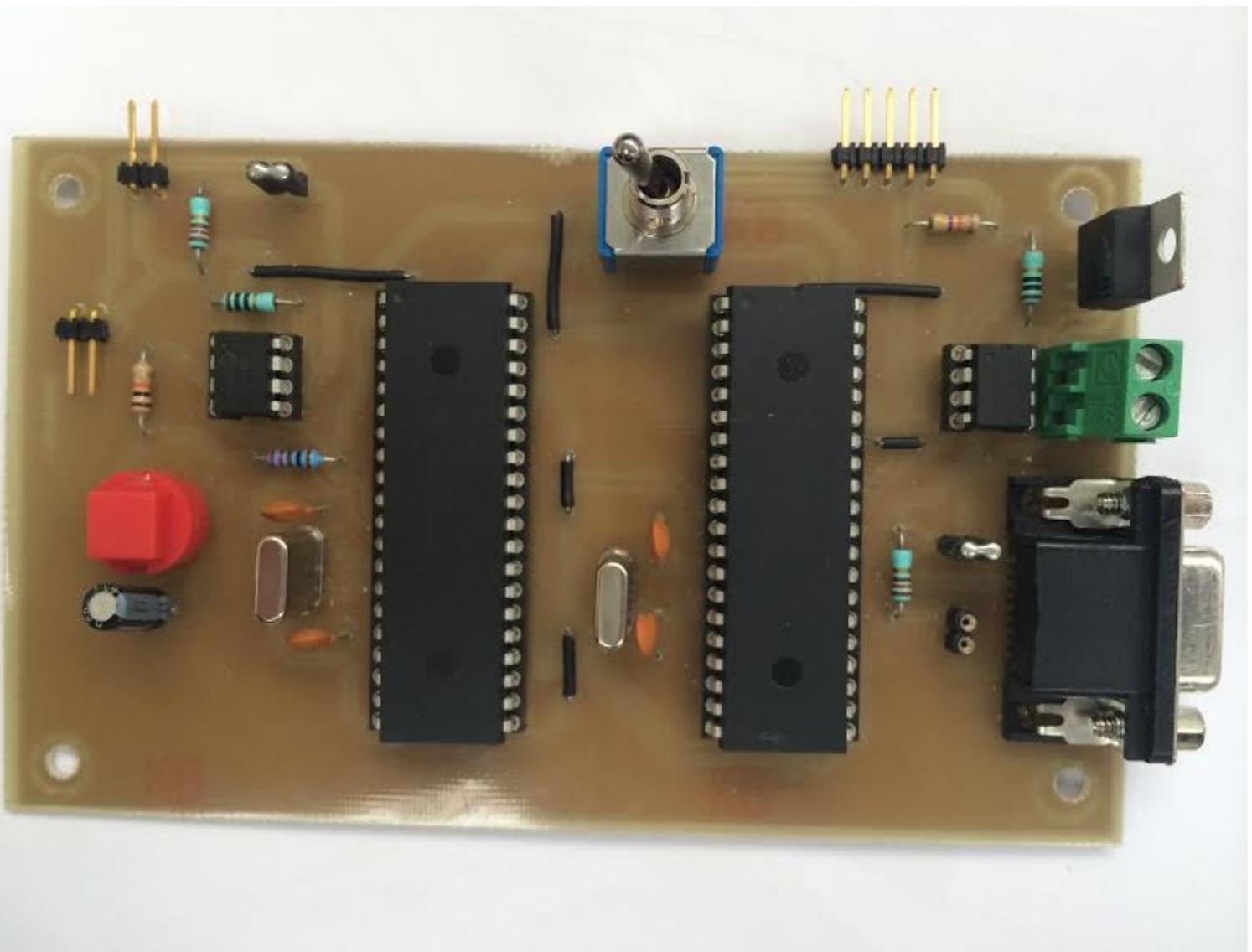


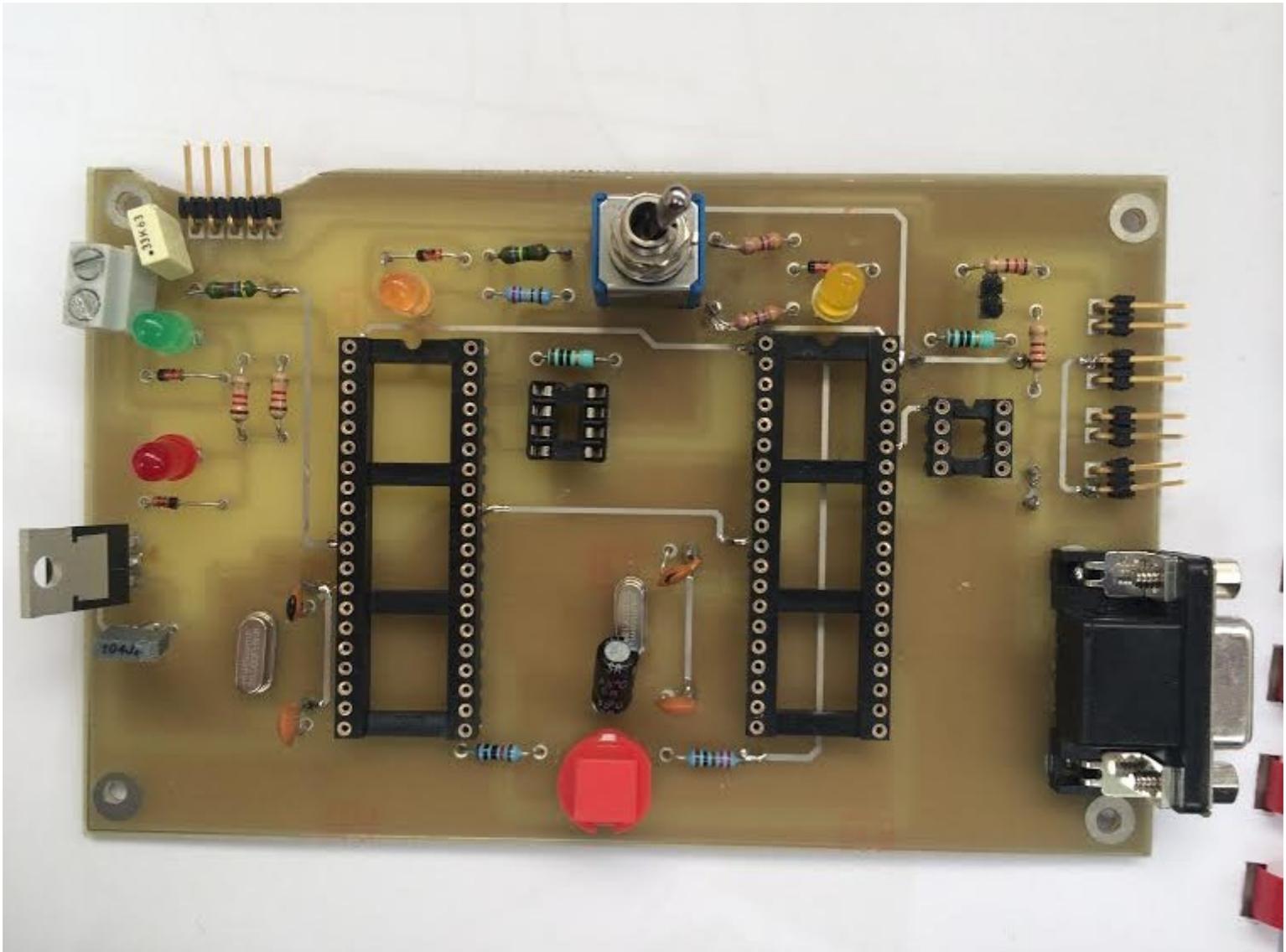
III°) Problèmes rencontrés :

1) Carte :

Dans un premier temps, nous avons branché la carte de l'année précédente afin de constater l'avancement pratique des étudiants mais il y avait un problème de compilation, la cible était trouvée mais ne pouvait pas finir l'exécution. Nous avons donc refait, dans un second temps, une carte similaire à celle déjà existante, nous nous sommes rendu compte qu'il y avait des problèmes de connexions. En effet, certaines broches étaient mal câblées et d'autres manquaient sans parler du problème survenu à cause du switch 6 broches qui était défectueux et causait des problèmes de compilations. Dans un dernier temps, nous refîmes une carte en arrangeant les détails qui nous avait échappé sur notre précédente carte.

Carte existante :



Première Carte :

Les différences avec la carte existante sont la disposition de nos réseaux CAN placés tout à droite afin de faciliter les branchements pour l'utilisateur. L'ajout de LEDs afin de savoir si nous avons bien câblé l'alimentation et sur quel pic nous travaillons en fonction de la position du switch 6 broches placé tout en haut de la photo ci dessus. Nous avons rajouté des condensateurs de découplage afin d'évacuer les harmoniques de hautes fréquences vers la masse et ainsi augmenter l'immunité électromagnétique du circuit en entrée sortie comme le précisait la doc du régulateur LM7805 (78 pour une tension positive et 05 pour avoir 5 V en sortie qui alimente nos pics). Nous avons aussi diminué le nombre de staps, nous n'en avons que 2 sur notre carte au lieu de 14 sur celle existante.

Doc technique LM7805 :

Typical Applications

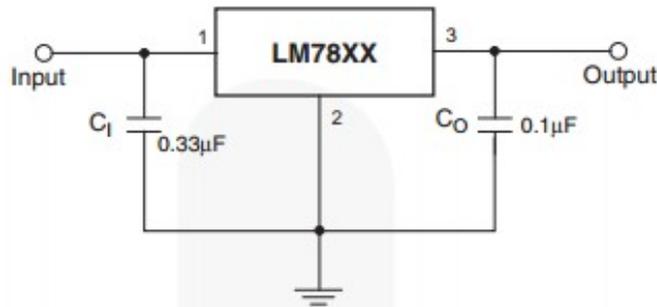


Figure 6. DC Parameters

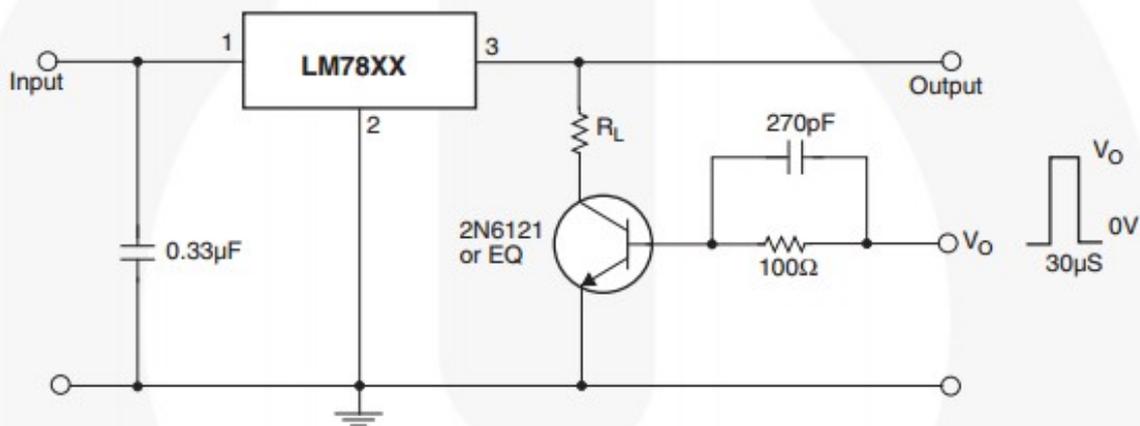
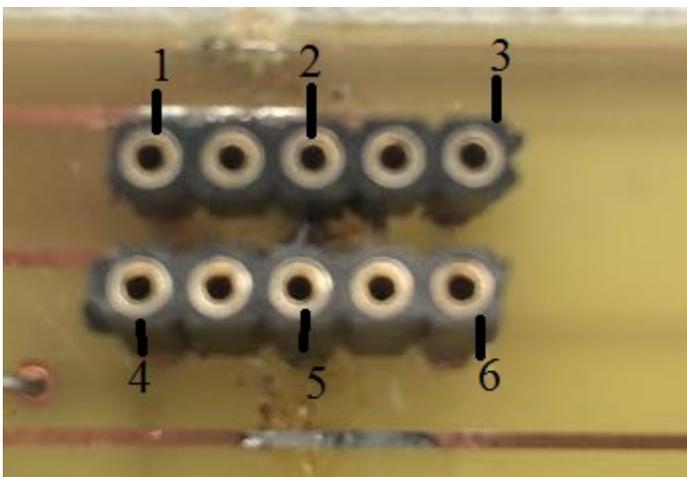
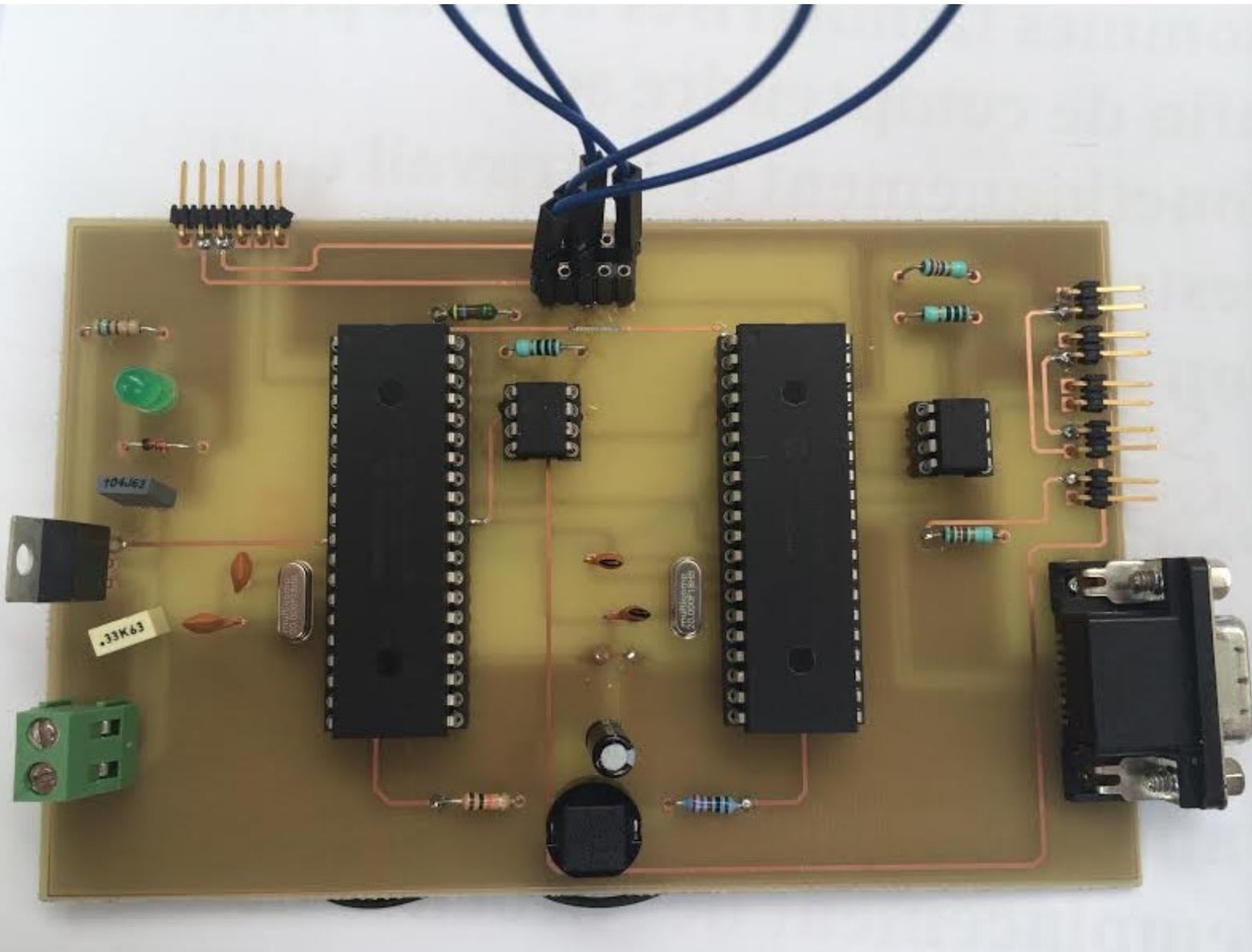


Figure 7. Load Regulation

LM78XX / LM78XXA — 3-Terminal 1 A Positive Voltage Regulator

Dernière Carte :



Nous avons enlevé le switch à 6 broches car il avait un défaut de fabrication. En effet, en réalisant le test de continuité avec un multimètre, nous nous sommes rendus compte que certaines broches étaient reliées entre elles alors qu'il ne le fallait pas. Nous avons opté pour la solution de « cavalier » afin que cela soit plus facile pour programmer nos pics. Sur le schéma ci dessus, sur les entrées femelles tout en haut à gauche est la broche 1, il faut aller de 2 en 2 (cf schéma de côté).

IV°) Solutions retenues :

Étant donné que notre BSI (Boîtier de Servitude Intelligent) était hors service, nous avons trouvé une solution qui consiste à la réalisation d'une carte qui se substituerait à la BSI comme dit précédemment. Elle consiste à gérer les données provenant de l'ordinateur qui sont envoyées via CANalyzer afin de simuler notre réseau IS (Inter Système), le traitement, la mise en forme et l'envoi de ces dernières vers le combiné en simulant notre réseau ID (Info & Divertissement). En effet, les trames qui circulent sur les différents réseaux de communication du projet ne sont pas les mêmes. Il faudra donc effectuer une correspondance entre les différentes données des différentes trames afin d'en simplifier le traitement.

Pour se faire, nous nous sommes aidés des trames créées précédemment pour les différents TP utilisés sur CANalyzer tout au cours de l'année. Il nous manquait quelques trames, c'est Mr Lombard qui nous les a fournies afin que nous puissions avancer.

Nous avons décidé de réaliser une carte électronique principalement composée de deux nœuds qui sont nos microprocesseurs pic18F4580 pour réaliser l'échange des données entre les deux bus de communication :

- le premier (celui de droite) sera chargé de récupérer les données que nous envoyons via CANalyzer. Ces données seront alors traitées et envoyées au second pic à l'aide de la liaison SPI (Serial Peripheral Interface).

- le second effectuera la transition entre les deux bus, IS et ID. Les données séries reçues précédemment seront mises en forme afin de correspondre aux exigences spécifiées dans la définition des trames et ensuite envoyées directement au combiné.

La carte électronique sera donc composée de deux pics ainsi que leur structure associée (quartz, condensateurs de découplages), d'une fonction de remise à zéro (Main Clear), d'une structure commutable permettant la programmation de ces deux pics à l'aide du boîtier pickit 3. Les deux bus CAN seront amenés à la carte par l'intermédiaire de deux connecteurs. Leur résistance de terminaison sera également présente et les niveaux de tension des bus CAN (1,5V et 3,5V) seront adaptés à l'aide des transmetteurs MCP2551.

Il faut faire attention à relier les broches 7 (SS=Slave Select) des deux micros ensemble, les broches 18 (SCL=Clock) et à inverser les branchements des broches d'un pic à un autre pour la liaison SPI comme sur le schéma ci-dessous.

Les connecteurs J1 et J4 ne sont là que pour tester les trames qui circulent sur nos réseaux CAN ID et IS.

Nous avons décidé d'alimenter tout le projet à l'aide d'une seule alimentation 12V ; il a donc fallu ajouter un régulateur de tension de 12V vers 5V afin d'alimenter nos pics.

Le connecteur DB9F (en bas à droite du schéma ISIS carte) assure la récupération des données envoyées via le logiciel CANalyzer. Après avoir vérifié l'exactitude du schéma et la présence de toutes les structures et éléments souhaités, nous avons réalisé le schéma d'implantation et le routage de la carte sous ARES.

Certains composants tels que le switch (SW1 sur le schéma ISIS carte), les LEDs ou encore le Bouton Poussoire (SW2) ne disposaient pas d'empreinte pour l'implantation (tailles des pattes et du boîtier). Il a donc fallu dans un premier créer ces composants afin d'avoir les empreintes souhaitées.

Schéma ISIS carte :

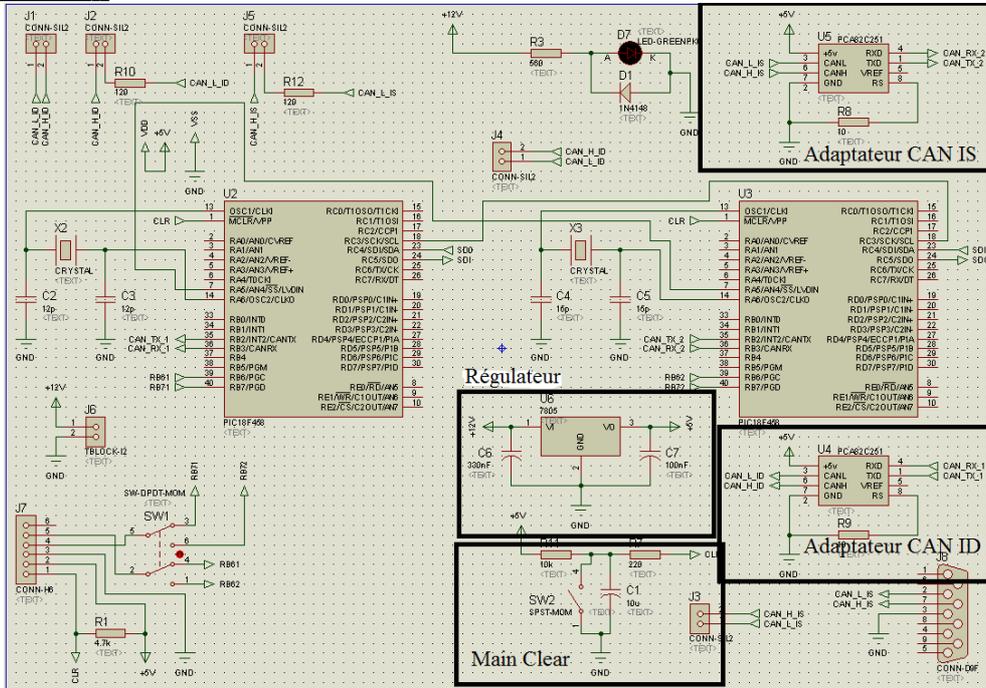
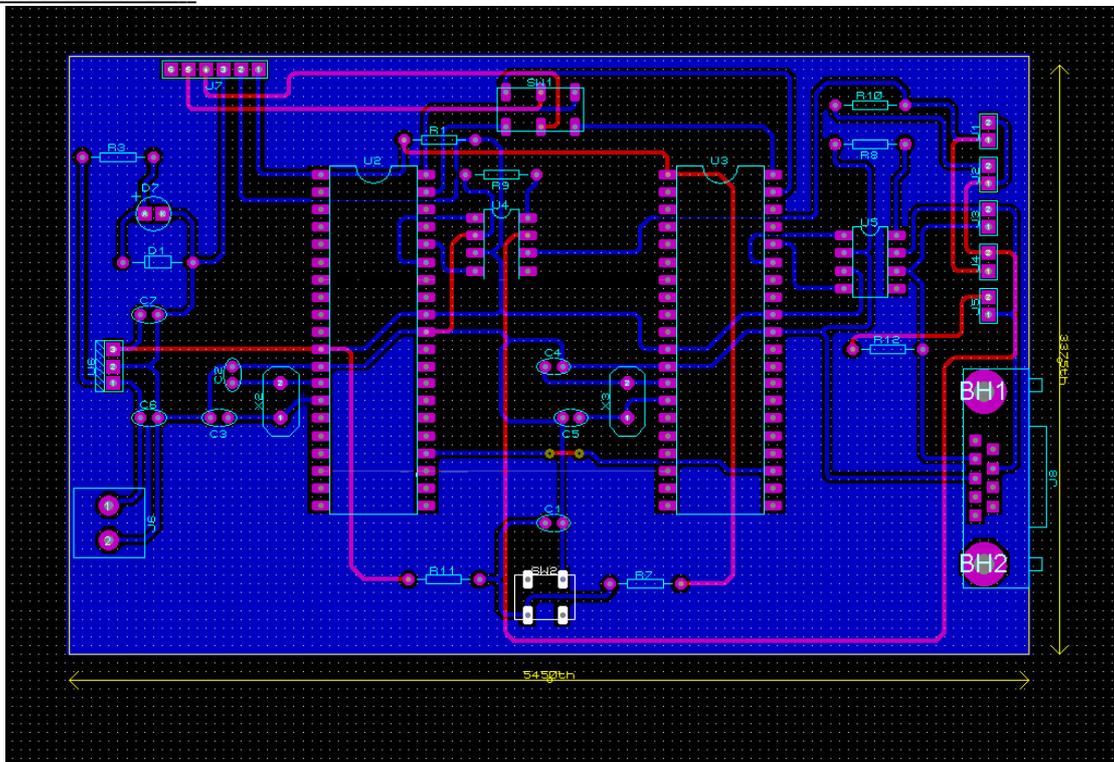


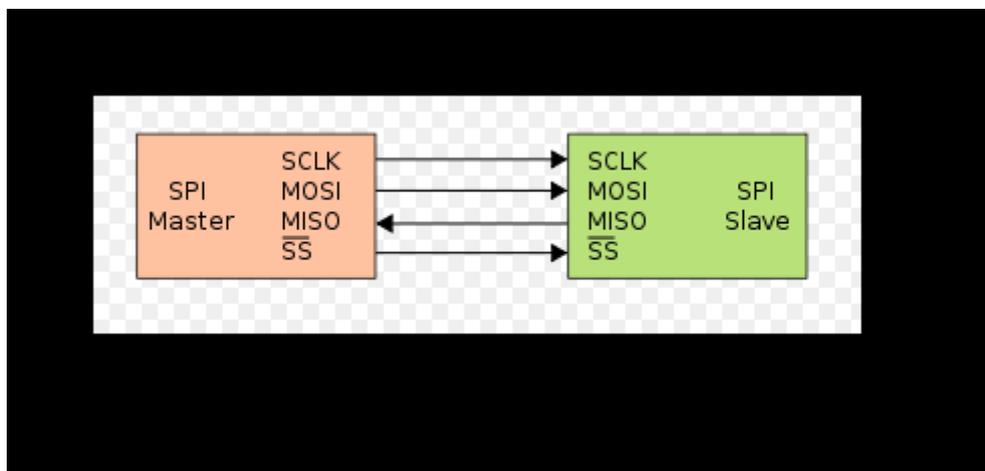
Schéma ARES carte :



1) Fabrication Carte et test :

Après que le typon soit transféré sur la carte puis étamé, nous avons réalisé, une première fois, un test de continuité de l'ensemble des pistes et des connections de notre carte. En constatant qu'aucun court-circuit n'avait été fait, nous avons percé les pastilles aux différents diamètres en fonction des composants à souder. Ensuite nous avons, dans un premier temps, branché notre carte avec une alimentation à 12V et regardé si tout nos supports de composants étaient alimentés correctement. Après validation des tensions d'alimentation, nous avons placé tous les composants sur leur support afin de commencer la programmation de nos pics. Lors du branchement de la carte complète, nous nous sommes rendus compte qu'elle consommait environ 200mA. Cela semble normal étant donné qu'en branchant la BSI, nous avons un courant consommé de 400mA.

Nous nous sommes ensuite attaqué à la liaison SPI, que nous détaillerons plus tard, afin de mieux comprendre son fonctionnement. Pour que la liaison SPI soit fonctionnelle, il faut tout d'abord comprendre ses 4 liaisons utiles entre les deux pics :



Le bus SPI utilise 4 signaux logiques :

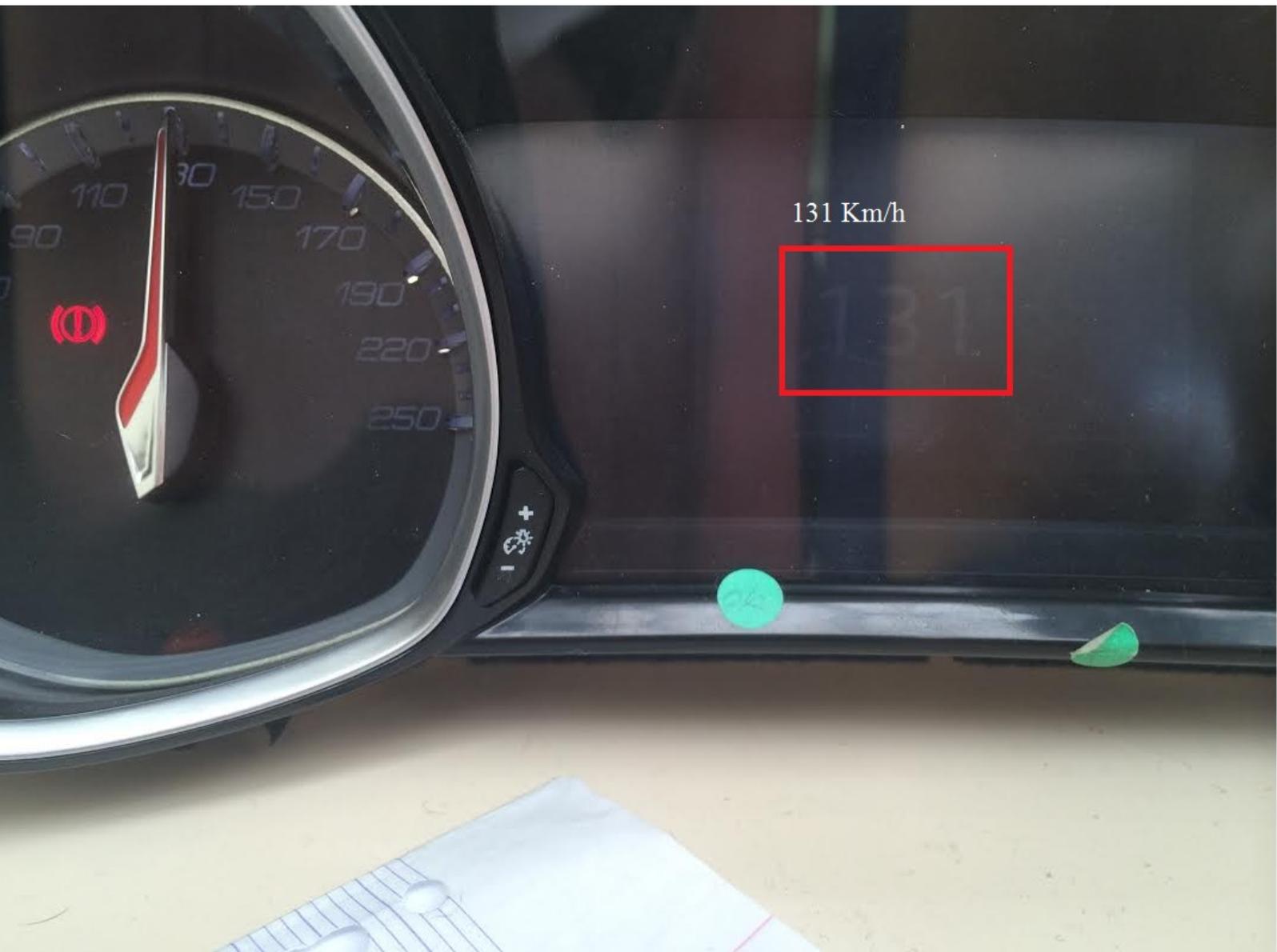
- **SCLK** — Serial Clock, Horloge (généré par le maître)
- **MOSI** — Master Output, Slave Input (généré par le maître)
- **MISO** — Master Input, Slave Output (généré par l'esclave)
- **SS** — Slave Select, Actif à l'état bas (généré par le maître)

Nous avons donc ensuite vérifié la réception des données provenant de CANalyzer. Après avoir créé un programme simple en langage C permettant de décoder les trames CAN réceptionnées, ce qui nous a également permis de valider le fonctionnement de la structure de programmation, nous avons généré des trames connues à partir du logiciel (telles que la trame contenant l'information de la vitesse véhicule et celle du régime moteur) ([Annexe 1 et 2](#)).

Le fonctionnement de l'adaptateur de niveaux MCP2551 a été validé à l'oscilloscope. Nous avons en effet observé en entrée du circuit intégré des niveaux de tension de 1,5V (niveau bas) et 3,5V (niveau haut), correspondant aux niveaux de tension du bus CAN, et en sortie du composant, des niveaux de tension de 0V et 5V. Ainsi, la trame CAN transmise est adaptée pour que le microcontrôleur puisse l'utiliser.

En effectuant l'analogie entre les octets émis par CANalyser et les valeurs des variables affectées dans le programme, nous avons constaté le bon fonctionnement de la réception.

Nous avons effectué la même démarche de vérification pour l'émission des trames CAN vers le combiné (Annexe 3). nous avons constaté la bonne adaptation des niveaux de tension (0-5V vers 1.5-3.5V) et une bonne réaction du combiné. En effet, en envoyant un octet correspondant à la vitesse véhicule souhaité (130 km/h), nous visualisons ce résultat :



A ce stade là, nous avons vérifié le fonctionnement indépendant des structures de notre carte. Nous pouvons alors réaliser l'applicatif de notre projet étape par étape.

V°)Réalisation du projet :

1)Prise en main liaison SPI :

La liaison SPI (Serial Peripheral Interface) entre nos deux pics permet le fonctionnement global de la carte. En effet, sans cette liaison, nous ne pouvons agir sur le combiné directement à partir de CANalyzer.

Le bus SPI est une bus de données série où les circuits concernés fonctionnent en Maître/Esclave. Le maître se charge de toute la communication; c'est à dire qu'il effectue l'envoi des données vers l'esclave et, lorsque l'esclave souhaite retourner une information , ce dernier émet une requête.

Nous avons réalisé après étude des différentes procédures disponibles avec le compilateur, les instructions permettant l'envoi des données sur la liaison SPI:

```

22 }
23
24 void envoi_38D(1Vitesse_roue1,1Vitesse_roue2)//fonction secondaire pour envoyer notre trame 38D
25 {
26     output_high(PIN_D2);//Mise à 1 de la broche 2 du port D
27     output_low(SPI_SS);//Mise à 0 de la broche 5 du port A
28     spi_write(1Vitesse_roue1);//envoi variable
29     output_high(SPI_SS);//Mise à 1 de la broche 5 du port A
30     delay_us(10);//temps d'attente de 10 µs
31
32     output_low(SPI_SS);//Mise à 0 de la broche 5 du port A
33     spi_write(1Vitesse_roue2);//envoi variable
34     output_high(SPI_SS);//Mise à 1 de la broche 5 du port A
35     delay_us(10);//temps d'attente de 10 µs
36
37     output_low(SPI_SS);//Mise à 0 de la broche 5 du port A
38     spi_write(1Reg_Mot);//envoi variable
39     output_high(SPI_SS);//Mise à 1 de la broche 5 du port A
40     delay_us(10);//temps d'attente de 10 µs
41
42 }
43
44

```

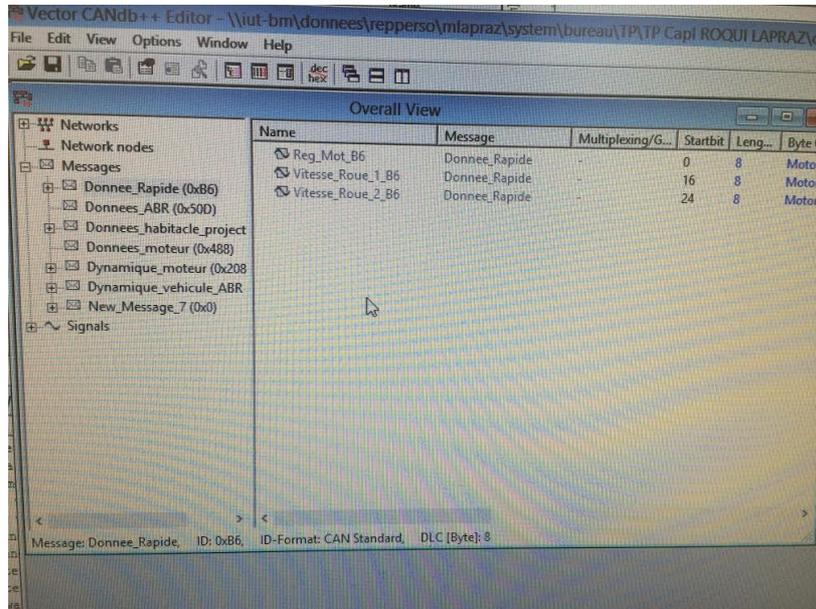
Cette procédure effectue l'envoi de la vitesse du véhicule en deux temps et celle du régime moteur. Comme cette dernière est contenue sur 2 octets et que la liaison SPI ne permet l'envoi que d'un octet à la fois, il a fallu ouvrir trois dialogues l'un à la suite de l'autre.

Afin d'effectuer ce dialogue, nous mettons la ligne 'SS' au NL0. La fonction 'spi_write' permet l'envoi des données du maître vers l'esclave.

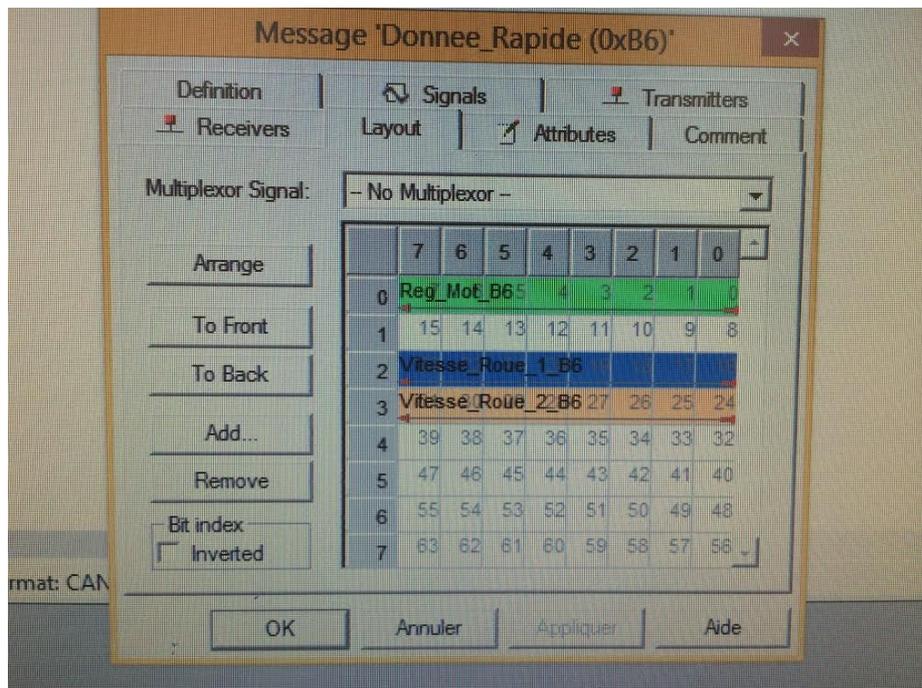
2)Affichage vitesse véhicule et régime moteur :

Afin d'éteindre nos voyants et d'afficher la vitesse et le régime moteur, nous avons du créer une trame avec l'identifiant B6 :

Données trame B6 :



Détails trame B6 :



Nous nous sommes aidés du schéma ci dessous qui était utilisé par les étudiants de l'année dernière :

Trame	Nom		Nbr Octets	Bus
B6 / P	Données Rapides		8	INFO CAR/CONF
Octet	Bit	Description		Codage
8	0-7	??		
7	0-7	??		
6	0-7	??		
5	0-7	??		
4	0-7	Affichage Vitesse (partie faible) ✓		Delta de 1 = 0,01 km/h
3	0-7	Affichage Vitesse (partie forte) ✓		Delta de 1 = 2,6 km/h
2	0-7	??		
1	0-7	Tr/Min ✓		Delta de 1 = 35 tr/min

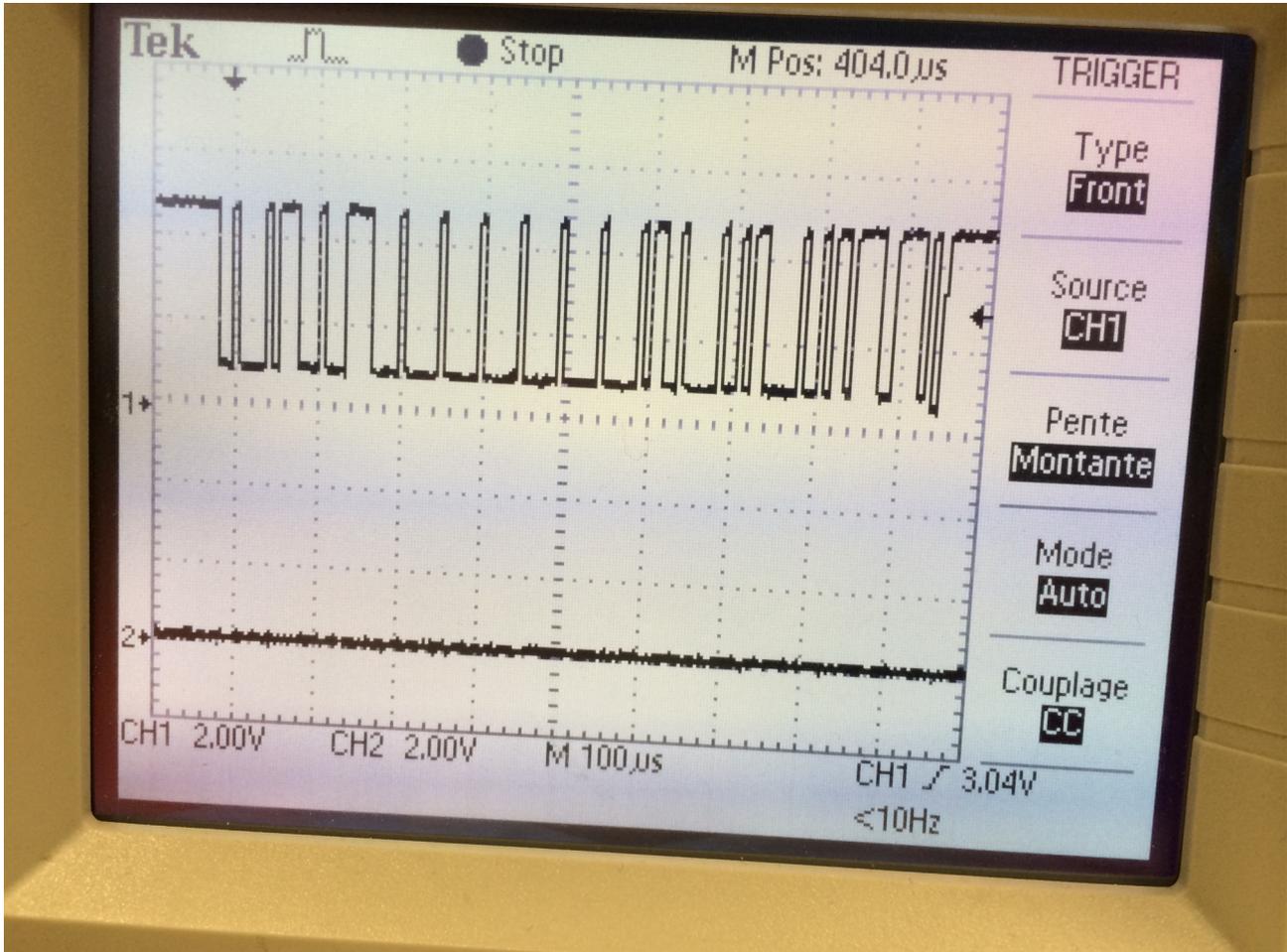
Nous visualisons ensuite dans CANalyzer les valeurs que l'on veut transmettre au combiné :

Trace						
Time	Chn	ID	Name	Dir	DLC	Data
15.99...	1	b6	Donnee_Rapide	Tx	8	6d 00 00 00 00 00 00 00
~			Reg_Mot_B6 2997.5 tr/min	[6d]
~			Vitesse_Roue_2_B6 0 km/h	[0]
~			Vitesse_Roue_1_B6 0 km/h	[0]
15.99...	1	38d	Dynamique_vehicule_ABR	Tx	5	32 c8 00 00 00
~			Vitesse_vehicule_roues 130 km/h	[32c8]
15.90...	1	50d	Donnees_ABR	Tx	6	00 00 00 00 00 00 00

Nous avons un problème avec l'envoi du régime moteur. En effet, lorsque l'on envoie une consigne pour un régime moteur, nous n'avons pas du tout le résultat espéré. Dans notre exemple ci dessus, nous envoyons une consigne à 3000tr/min or sur le combiné l'aiguille affiche 7000tr/min. De même si nous envoyons une autre consigne, le combiné interprète tout seul sans aucun rapport ni coefficient nos données envoyées pour le régime moteur.

Nous avons alors visualisé à l'oscilloscope les trames émises par CANalyzer comme le montre le schéma ci dessous :

Trame CAN ID :



Nous nous sommes rendus compte que nous envoyions deux fois une consigne de vitesse (38D et B6) mais seule la trame de la 38D était prise en compte car lorsque l'on a mit une consigne de vitesse à 130Km/h sur la 38D et 0Km/h sur la B6, notre combiné affichait celle de la 38D.

Sur l'Annexe 1, on remarque bien que notre consigne de régime moteur à 3000 tr/min est envoyée grâce à la fenêtre graphique mais sur le combiné, elle n'a pas la valeur souhaitée.

VI°) Conclusion :

Ce projet nous a dans un premier temps de comprendre plus en détail le fonctionnement d'un système électronique récent, qui est au sein de l'industrie du transport (automobile, transport routier). L'étude et la simulation d'un fonction d'actualité, sujette à de nombreuses critiques nous a permis d'en voir les avantages, mais aussi les points sensibles tels que le fonctionnement en situation très spécifique.

Les problèmes rencontrés n'ont pas enlevé d'intérêt au projet; au contraire, ils nous ont obligé à définir des solutions différentes avec des moyens différents et une manière de voir le projet différemment. L'applicatif réalisé a du coup été hors de nos attentes et la réalisation de la carte est toujours une plus-value pour le projet ainsi que notre expérience dans ce domaine.

Même si nous n'avons pas mené le projet à son terme, nous pensons qu'il pourra servir de base pour les années à venir et pourra être facilement exploitable étant donné le bon fonctionnement de la carte.

