**Projet création ODB pour Audi A3**

Le projet va s’articuler autour d’un arduino MkrZero qui est un microcontroleur programmable en C++ et d’un affichage TFT de 1.8’’ couleur

Le but est de pouvoir afficher dans le porte instrument les données suivantes :

* Vitesse instantanée
* Rapport de vitesse engagée
* Distance restante à parcourir (fonction du niveau du réservoir et de la conso moyenne)
* Consommation moyenne depuis le dernier plein d’essence
* Distance parcourue depuis le dernier plein d’essence
* Consommation instantanée
* Conso moyenne et distance parcourue pour un parcours 1, 2 ré-initialisables

**Récupération des données**

Pour millésime > 2000. Tout se passe sur la prise bleue du porte instrument

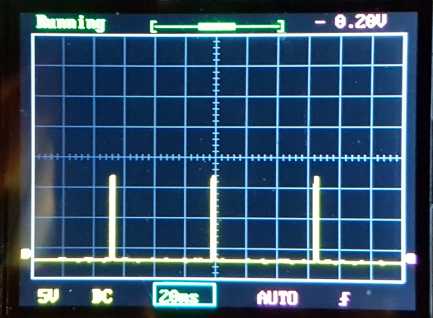
Borne 1 : +12 V après contact

Borne 9 : masse

Borne 30 12V permanent

Borne 25 : signale conso

On récupère des impulsions 12V proportionnelles à la consommation du moteur par unité de temps



Borne 3 : Sortie 1 tachymètre

Idem à la borne 25 : on récupère des impulsions 12V proportionnelles à la vitesse de sortie de boîte de vitesse

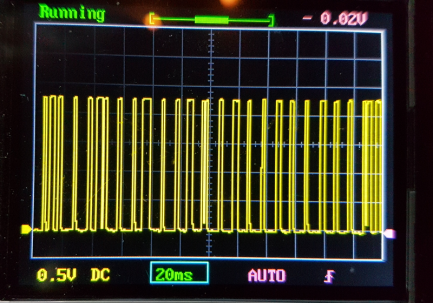
Borne 11 : régime moteur :

On récupère un signal 12V de forme carrée dont la fréquence est proportionnelle au régime moteur



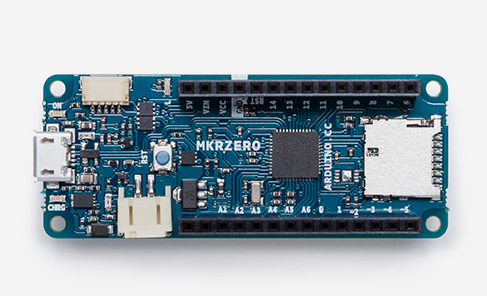
Borne 5 : Transmetteur de niveau de carburant

On récupère ici un signal numérique de type K-Line. Mais c’est en fait la partie analogique (amplitude du signal) qui transmet le niveau de carburant. Elle varie d’environ 1.5V (vide) à 3V (plein)



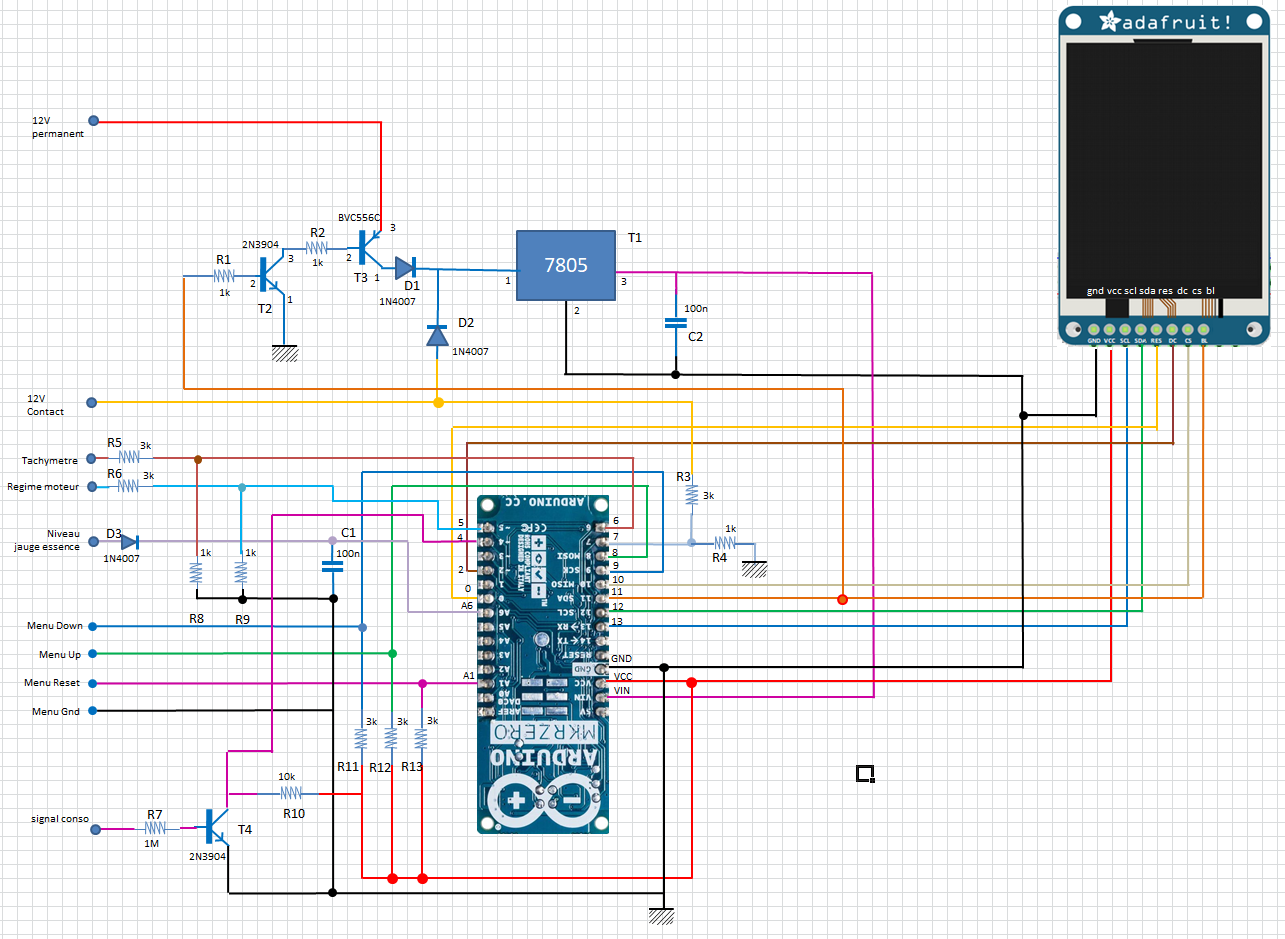
**Traitement des données**

Le montage s’articule autour d’un arduino mkrzero qui possède plusieurs entrées / sorties numériques analogiques ainsi qu’un lecteur de carte µSD. Ce dernier va donc enregistrer les données des différents capteurs, les traités , les enregistrés sur une carte SD (lors de la coupure du contact) et gérer l’affichage sur un petit écran TFT couleur



<https://www.arduino.cc/en/Main/ArduinoBoardMKRZero&>

Ce qu’il faut retenir est qu’un arduino fonctionne sous 5V et que ses entrées /sorties ne tolère que du 3,3V. Il va donc falloir créer un petit montage (appelé shield ) qui va permettre de connecter l’arduino à la voiture et de les rendre compatibles



Alimentation du montage

Les arduino possède une mémoire volatile qui s’efface lors de la coupure de leur alimentation. (ce n’est pas tout à fait vrai, il existe une eeprom mais le nombre de changements de valeurs est limité avant la destruction du composant) .

On a 3 possibilités

* Laisser le montage constamment alimenté grâce au 12V permanent. (non retenu)
* Utiliser le 12V contact et sauvegarder sur la carte SD en permanence toutes les modifications (non retenu car l’écriture sur la carte SD ralentie fortement le programme)
* Laisser le temps à l’arduino de sauvegarder sur la carte SD les données avant de s’éteindre (solution retenue)

Pour cela l’arduino sera alimenté par un régulateur de tension 7805 T1 de 5V et stabilisé par un petit condensateur C2

Lorsque le contact est mis le régulateur est directement alimenté par le +12V contact (avec une diode D2 pour éviter les retours de courant) et délivre 5V sur la borne d’alimentation Vin de l’arduino

Un pont diviseur de tension formé avec les résistances R3 R4 indique à l’arduino (borne 7 input) la présence du +12V contact

Le programme de l’arduino impose alors à sa borne 11 (output) une sortie haute de 3,3V

Les transistors T1 et T2 agissent comme un interrupteur et laissent à leur tour passé le courant (12V permanent) via la diode D1

Lorsque l’on coupe le contact le courant ne passe plus au travers de la diode D2. Mais le régulateur est toujours alimenté par D1 tant que l’arduino imposera une tension de 3,3V à sa borne 11

L’arduino est programmé pour détecter la coupure de courant sur sa borne 7 (via une interruption) . Il exécute alors la procédure de sauvegarde sur la carte SD et bascule ensuite la sortie 11 à 0V ce qui a pour effet de couper l’alimentation via les transistors T1et T2. L’arduino s’éteint jusqu’au prochain allumage

Transformation des signaux d’entrées

Les signaux 12V de l’entrée tachymètre et régime moteur sont simplement rabaissées à 3V par deux ponts diviseur de tension (R5 & R8 pour le tachy et R6 & R9 pour le régime moteur) et connectés respectivement aux bornes 6 et 5 de l’arduino

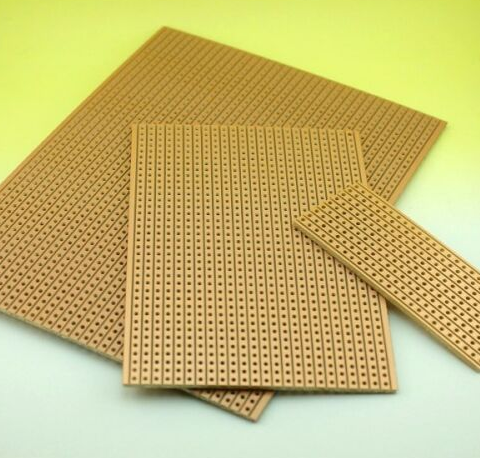
Le signal conso étant très sensible à l’impédance connecté en parallèle, on utilise un transistor 2N3904 (T4) couplé à une résistance (R7) d’un mega ohms. Le signal récupéré au collecteur de T4 est connecté à la borne 4 de l’arduino

Pour le signal de niveau de jauge de carburant seule l’amplitude nous intéresse. On isole donc le signal par la diode D3 et l’on vient charger le condensateur de faible capacité C1 ce qui permet de filtrer le signal et d’obtenir une tension stable proportionnelle au niveau de la jauge. Le signale est ensuite enregistré via la borne analogique A6 de l’arduino

Les boutons des commodo (Menu +, Menu – et reset) seront enregistrés via les bornes 8,9 et A1 qui sont mises au potentiel haut de 3.3V par défaut via les résistances R11, R12, R13

Implantation des composant

J’utilise une platine de prototypage pour réaliser le support de l’arduino ce qui évite les typons et les bains acides



Cotés cuivres les pistes sont interrompues grâce à un petit coup de perceuses (cf ronds rouge sur la vue côté cuivre)

