

12 Télécommande comodo

ouvrir la porte du garage sans lever les mains du guidon

Bruno Clerc (France)

En arrivant chez moi au guidon de ma moto, en pleine côte montante, les mains pleines de gants, chercher la télécommande du portail puis celle de garage, c'est galère ! C'est pourquoi je me suis mis à la recherche d'une solution pour ouvrir à distance le portail et la porte de garage sans avoir à appuyer sur les boutons de leurs télécommandes respectives.



Bien sûr, on peut cacher une photorésistance pour détecter un appel de phare, mais ce n'est pas très sécurisé. Mieux vaut garder les télécommandes radio à code tournant. Mais, dans ce cas, il faudra un dispositif pour appuyer sur leurs boutons sans lever les mains du guidon, par exemple en faisant un appel de phare.

Voici le cahier des charges du dispositif :

- Alimenté par l'alimentation du véhicule (12 V après contact).
- Comme on doit connecter le dispositif aux boutons-poussoirs des télécommandes, on en profite pour les alimenter à travers le dispositif. On élimine ainsi les piles.
- Détecter la demande d'émission via un signal venant du phare (+12 V phare allumé).
- Suivant la détection, simuler un appui sur le bon bouton de télécommande.

Un petit microcontrôleur ATtiny85 est chargé de détecter l'appel de

phare. Si l'appel est trop court ou trop long, rien ne se passe. Si, en revanche, il dépasse une première durée (mais pas de trop), c'est le bouton de la première télécommande qui est activé. Si la durée de l'appel de phare dépasse un deuxième seuil (mais pas de trop), le bouton de la deuxième télécommande est activé.

Sur ce principe j'ai réalisé différents prototypes qui actionnent les boutons-poussoirs soit avec des relais soit avec des optocoupleurs. Cela fonctionne bien, mais la solution n'était pas tout à fait satisfaisante. Un premier inconvénient de cette approche est le sacrifice des télécommandes qui font désormais partie du véhicule. L'encombrement de l'ensemble est un autre point négatif.

Une relation m'a fait découvrir deux marques de télécommandes programmables (JMA et Silca). Alors, l'idée a été d'associer le dispositif à une télécommande programmable dédiée au véhicule. En remplaçant la pile de la télécommande programmable et son support par une petite platine avec le ATtiny85, on obtient un ensemble propre et compact.

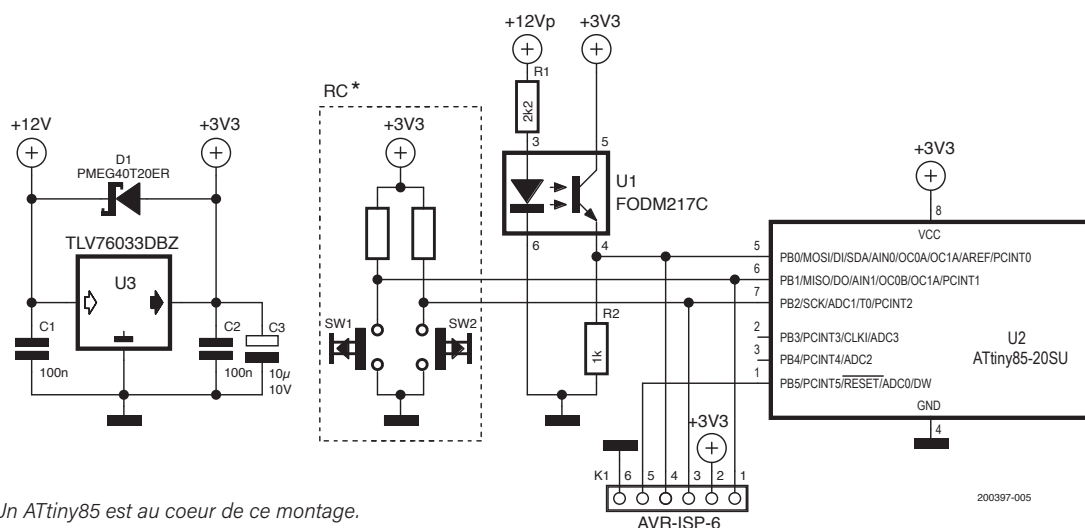


Figure 1. Un ATtiny85 est au coeur de ce montage.

Tableau 1. Connexions entre K1 et l'Arduino UNO en mode programmeur.

K1	UNO	Signal
1	D12	MISO
2	+5V	
3	D13	SCK
4	D11	MOSI
5	D10	RST
6	GND	

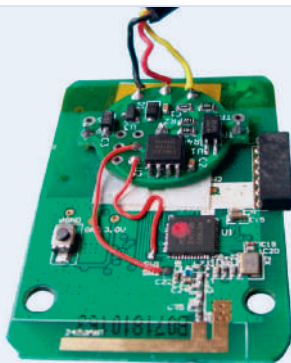


Figure 2. La platine remplace la pile de la télécommande.

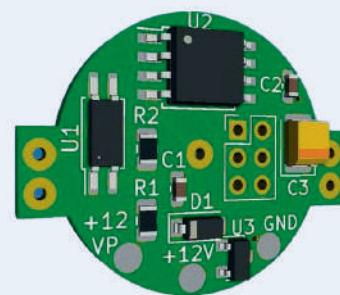


Figure 3. Aperçu en 3D de la platine dessiné par l'auteur.

Quelques mots sur le schéma électrique

Figure 1 montre le schéma du montage. Ce circuit se connecte au véhicule avec trois fils :

- > +12 V après contact
- > Masse véhicule
- > Phare (+12 V_P phare allumé)

Régulateur U3 assure l'alimentation en 3.3 V du micro et de la télécommande. La diode D1 protège le montage contre des inversions éventuelles de l'alimentation.

L'optocoupleur U1 est le détecteur d'appel de phare. Il fournit une protection contre les variations de la tension issue du véhicule (de 12 V à 14 V environ). La résistance R1 limite le courant à travers sa LED à environ 5 mA. La sortie d'U1 est connectée au port PB0 de l'ATtiny85. Lorsque le phare est activé (V_P = 12 V), PB0 voit un niveau haut (3,3 V), sinon c'est 0 V grâce à R2.

Dès que l'allumage du phare est détecté, le micro commence à compter (avec une résolution de 300 ms). Quand le phare s'éteint, le compteur s'arrête et sa valeur est comparée à des seuils prédéfinis. Ainsi, si l'appel de phare a duré entre 600 ms et 1,8 s, le port PB1 est activé durant 4 s pour simuler un appui sur un bouton-poussoir. Si la longueur de l'appel de phare est entre 2,1 s et 4,5 s, c'est le port PB2 qui est activé. Les ports PB1 et PB2 sont connectés en parallèle aux boutons-poussoirs de la télécommande (le rectangle RC dans le schéma) sans avoir besoin de relais ou d'optocoupleurs.

Programmation du micro

Le programme du micro a été écrit à l'aide d'Arduino [1]. Pour pouvoir le compiler, il faut d'abord installer le *Boards Package* pour ATtiny mis à disposition sur GitHub par David A. Mellis [2]. Vous y trouverez également les explications comment l'installer.

Pour charger le programme dans le micro, j'ai utilisé une carte Arduino UNO en mode programmeur (*Arduino as ISP*). La procédure est détaillée sur [2] (cliquez *this guide*). Le **tableau 1** indique comment connecter l'Arduino UNO au connecteur K1 de notre montage.

↓ Téléchargez le projet

www.elektormagazine.fr/summer-circuits-22



Mise en place

Pour ce montage, j'ai réalisé une petite platine circulaire de la taille du support de pile de la télécommande que j'ai montée à la place du support de pile (**Figure 2 et 3**). Les sorties PB1 et PB2 du µC ont été raccordées aux boutons SW1 et SW2 de la télécommande avec des fils fins. Une petite entaille sur la partie arrière du boîtier de télécommande laisse passer un câble à trois conducteurs pour connecter l'alimentation et la détection phare. Pour protéger la platine, j'ai appliqué une fine couche de vernis acrylique sans solvants aromatiques.

Bonne route ! ◀

200397-04

L'auteur

Bruno a découvert l'électronique vers 12 ans grâce à son frère aîné. Curieux de tout, ayant soif d'apprendre, ne connaissant rien, démontant tout sans trop souvent bien remonter, il décide de faire des études d'électronicien à Bordeaux. Ensuite, Bruno a exercé dans les systèmes basse tension tertiaires puis dans l'aéronautique et divers autres postes. À l'arrivée des microcontrôleurs, ne pratiquant pas la programmation, il se consacre à l'entretien des équipements hi-fi vintage. Tout bascule quand, il y a environ 5 ans, son frère lui donne un Arduino UNO. Bruno se découvre une nouvelle passion et devient *Arduino47*. Parti de zéro, aujourd'hui, il remercie l'ensemble de la communauté Arduino qui l'a aidé à avancer dans son apprentissage. Pour cette raison, sans aucune prétention de connaissance ou de reconnaissance, il essaie de prendre le temps de publier quelques montages de son cru. Son crédo est « inventer, ce n'est ni imiter ni copier. »



Produits

- > **W. A. Smith**, *Explore ATtiny Microcontrollers using C and Assembly Language (SKU 20007)*
www.elektor.fr/20007

LIENS

- [1] Code source code sur Elektor Labs: <https://www.elektormagazine.com/labs/comodo-remote>
- [2] ATtiny Boards Package pour Arduino: <https://github.com/damellis/attiny>