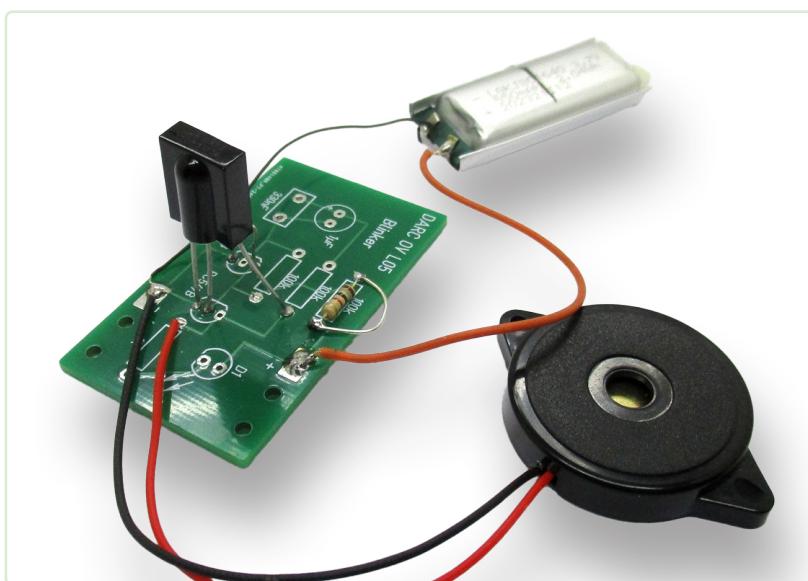


30 télégraphie infrarouge

Burkhard Kainka (Allemagne)

L'auteur a construit cet émetteur/récepteur IR simple pour l'utiliser lors des séances de pratique du code Morse de son club de radio amateur. Le récepteur s'avère également utile pour le test d'un vaste éventail de télécommandes IR.



▲

Figure 1. Récepteur de lumière infrarouge pour les séances de pratique du code Morse (construit ici sur un circuit imprimé provenant d'un autre projet).

Le récepteur (**figure 1**) consiste en un récepteur/démodulateur IR intégré TSOP31236 pour des signaux de 36 kHz. Un transducteur piézo est directement connecté à sa sortie. En plus de détecter les signaux IR en Morse, ce récepteur vous permet de vérifier si la télécommande de votre téléviseur ou de votre système audio fonctionne. De plus, vous pouvez vérifier le niveau de bruit ambiant en relation avec les signaux IR de 36 kHz. Si des tubes fluorescents sont allumés à proximité, vous entendrez des grésillements intermittents provenant du récepteur.

L'émetteur Morse (**figure 2**) active et désactive un signal porteur IR de 36 kHz à l'aide d'un bouton-poussoir. Le signal porteur est modulé par un signal de 800 Hz qui est récupéré après démodulation et transmis à un émetteur piézoélectrique. Des tests effectués avec un générateur de signaux et une LED IR ont montré qu'il est préférable de limiter la durée des impulsions du signal IR de

36 kHz à environ 10 cycles pour assurer une réception optimale. Cela permet d'éviter une atténuation inutile due au contrôle automatique de gain du récepteur. Ces impulsions de 10 cycles, d'une durée de 280 µs, ont un taux de répétition de 800 Hz et se poursuivent pendant que l'émetteur est activé. La période du signal de modulation de 800 Hz est de 1,25 ms.

Circuit

Le schéma du circuit est présenté à la **figure 3**. Il est assez difficile de trouver ou même de construire un interrupteur adapté qui puisse être utilisé comme clé Morse. C'est pourquoi nous avons intégré un capteur tactile capacitif sur le circuit imprimé. Le principe est simple : la broche du port qui lit le capteur est initialement configurée en sortie et mise à 0, de sorte que toute charge sur le capteur est dissipée à la masse. Pour effectuer une mesure, la broche du port est ensuite configurée en une entrée avec une résistance *pull-up* interne. Le contrôleur mesure alors le temps nécessaire pour que la tension sur cette broche atteigne un niveau suffisamment élevé pour passer à l'état haut (1). Cela prend normalement moins d'une microseconde. Lorsqu'un doigt touche la pastille de détection, il a pour effet d'augmenter la valeur de la capacité à cette entrée, cela augmente la capacité à l'entrée, ralentissant ainsi le temps de montée du signal et indiquant que la touche Morse est pressée.

Presque tous les petits microcontrôleurs sont capables d'effectuer ces tâches de base. J'ai choisi l'ATtiny202 parce qu'il peut être programmé via sa broche UPDI et qu'il ne nécessite qu'un programmeur très simple constitué d'un convertisseur USB-série et d'une simple résistance.

Logiciel

J'ai développé le logiciel (**listage 1**) en utilisant Visual Studio Code, PlatformIO, et l'EDI Arduino. Vous pouvez le télécharger depuis [1]. Dans la boucle main, le capteur tactile est évalué en mesurant le temps de charge *t*. Si ce dernier dépasse 1, une série de dix impulsions de



Listage 1. Croquis Arduino.

```
//IRCW202, Touch PA1, IR-LED PA3
#include <Arduino.h>
char t=0;

void setup() {
  PORTA.DIRSET = 0x08;      //Output PA3
  PORTA.DIRSET = 0x02;      //Output PA1
  PORTA.PIN1CTRL = 0x08;    //Pullup PA1
}

void loop() {
  t=0;
  PORTA.DIRCLR = 0x02;

  while(!(VPORTA.IN & 0x02)){
    t++;
  }

  PORTA.DIRSET = 0x02;

  if (t>1){
    for(char n=0; n<10; n++){
      PORTA.OUTSET = 0x08;
      delayMicroseconds (3);
      PORTA.OUTCLR = 0x08;
      delayMicroseconds (21);
    }

    delay(1);
  }
}
```

36 kHz est générée. Chaque impulsion reste à l'état haut pendant environ 3 µs et à l'état bas pendant environ 21 µs. De plus, il est essentiel de tenir compte du temps d'exécution du code estimé entre 3 à 4 µs pour chaque période d'impulsion pour maintenir précisément la fréquence du signal porteur à 36 kHz. Un délai supplémentaire de 1 ms, combiné à la durée de la série d'impulsions de 280 µs, établit la fréquence de modulation de 800 Hz pour la porteuse. Ce signal modulé est généré tant que le doigt reste sur les pastilles de contact.

Circuit Special 2024

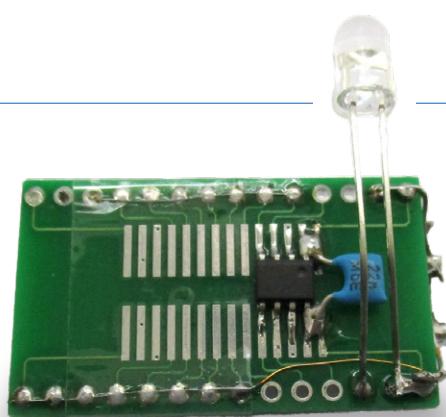


Figure 2. L'émetteur IR est construit sur une carte de prototypage CMS.

Vous pouvez charger le programme compilé, *firmware.hex*, sur l'ATtiny402 avec l'outil *pymcuprog.exe*, sans nécessité de modification préalable. Il est possible de déterminer expérimentalement la taille et la forme des pastilles des capteurs tactiles. Dans notre prototype, les rangées de pavés tactiles sont protégées par un ruban adhésif transparent pour prévenir tout contact direct avec le doigt. ↳

240123-04

À propos de l'auteur

Burkhard Kainka (*b-kainka.de*) est un radioamateur qui a écrit de nombreux livres et articles publiés par Elektor, couvrant un large éventail de sujets. Après avoir enseigné la physique pendant de nombreuses années, il se consacre désormais à sa carrière de développeur de systèmes et d'auteur indépendant.

Questions ou commentaires ?

Envoyez un courriel à l'auteur (b.kainka@t-online.de), ou contactez Elektor (redaction@elektor.fr).

LIEN

- [1] Page web de cet article:
<https://elektormagazine.fr/240123-04>

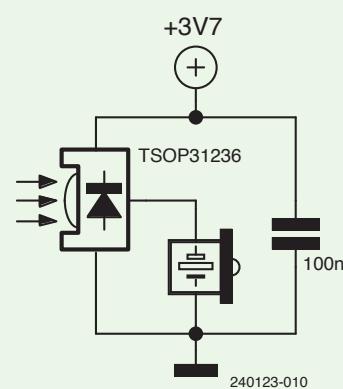
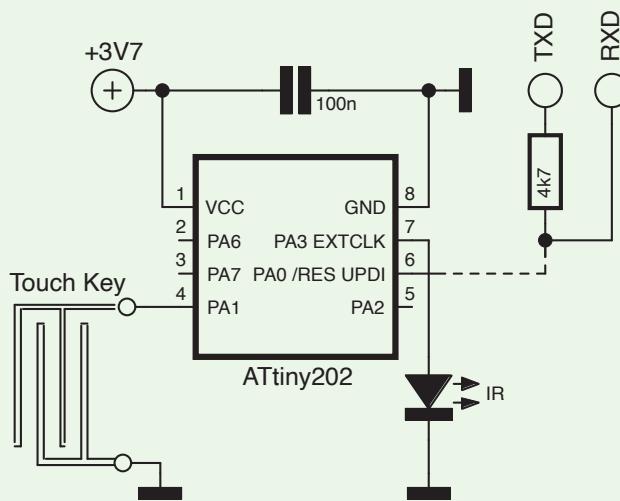


Figure 3. Le schéma de circuit du récepteur est basé sur la puce réceptrice IR TSOP31236 pour les signaux de 36 kHz.