



# télémesure avec détection des pertes de connexion

Grâce aux modules nRF24L01+

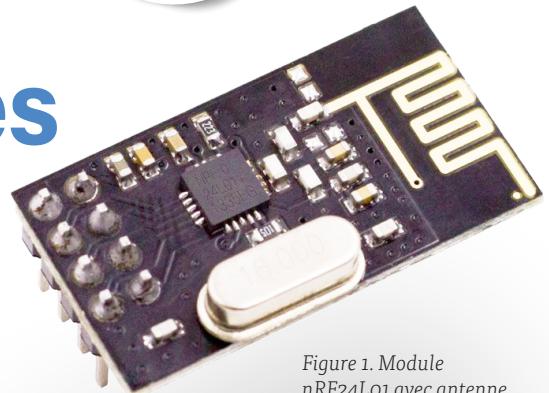


Figure 1. Module nRF24L01 avec antenne imprimée.

**Roel Arits (Pays-Bas)**

La transmission de données sans fil autorise de nombreuses applications (par ex. dans le domaine de la domotique). En revanche, il faut que la communication soit robuste et fiable. Ce projet de détecteur de mouvement à infrarouge passif (PIR) montre comment détecter la perte de connexion avec des modules de communication bidirectionnelle.

Les modules RF nRF24L01+ (fig. 1) sont intéressants et peu coûteux pour la communication bidirectionnelle sans fil [1]. Ils disposent d'un bus d'interface périphérique série (SPI) qui permet de les configurer et de les commander avec un microcontrôleur. On peut trouver sur l'internet de nombreux exemples de projets utilisant des cartes Arduino combinées avec ces modules RF.

Avec le projet décrit ici, je veux ajouter une fonction supplémentaire qui utilise la communication bidirectionnelle pour détecter la perte

de communication entre l'émetteur et le récepteur. C'est un peu étrange de parler d'un émetteur et d'un récepteur lorsqu'on utilise la communication bidirectionnelle, puisque les modules agissent à la fois comme émetteur et récepteur. Mais pour plus de clarté, nous qualifions un des modules d'émetteur car sa tâche principale est de transmettre l'état d'un capteur (PIR) à un autre module, qui reçoit les données pour un traitement ultérieur.

La capacité de détecter la perte de communication est très utile pour les capteurs à distance, car lorsqu'il n'y a pas de communication, on peut perdre des données sans en être averti. Il est également utile, lors de l'installation du capteur, de vérifier si les deux modules RF nRF24L01 se « voient » effectivement et ne sont pas hors de portée. La figure 2 donne une vue d'ensemble du projet et montre la communication entre les deux modules nRF24L01 – ainsi que l'implémentation de la détection de perte de communication.

## Schémas des circuits

Commençons par le schéma de l'émetteur (fig. 3). Le nRF24L01 doit être alimenté en 3,3 V, mais les broches d'E/S du nRF24L01 tolèrent le 5 V. Ainsi, connecter le bus SPI du nRF24L01 directement à un Arduino Pro Mini en 5 V ne pose aucun problème.

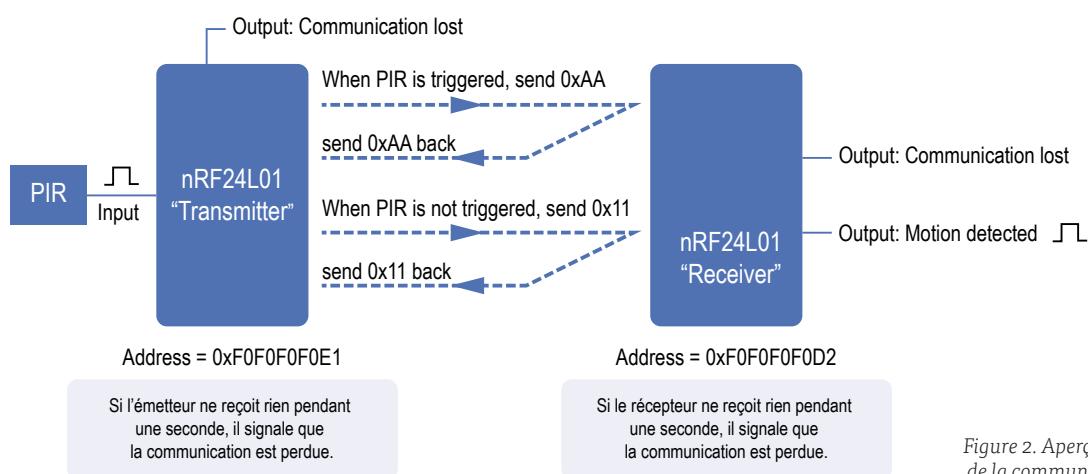


Figure 2. Aperçu des étapes de la communication.

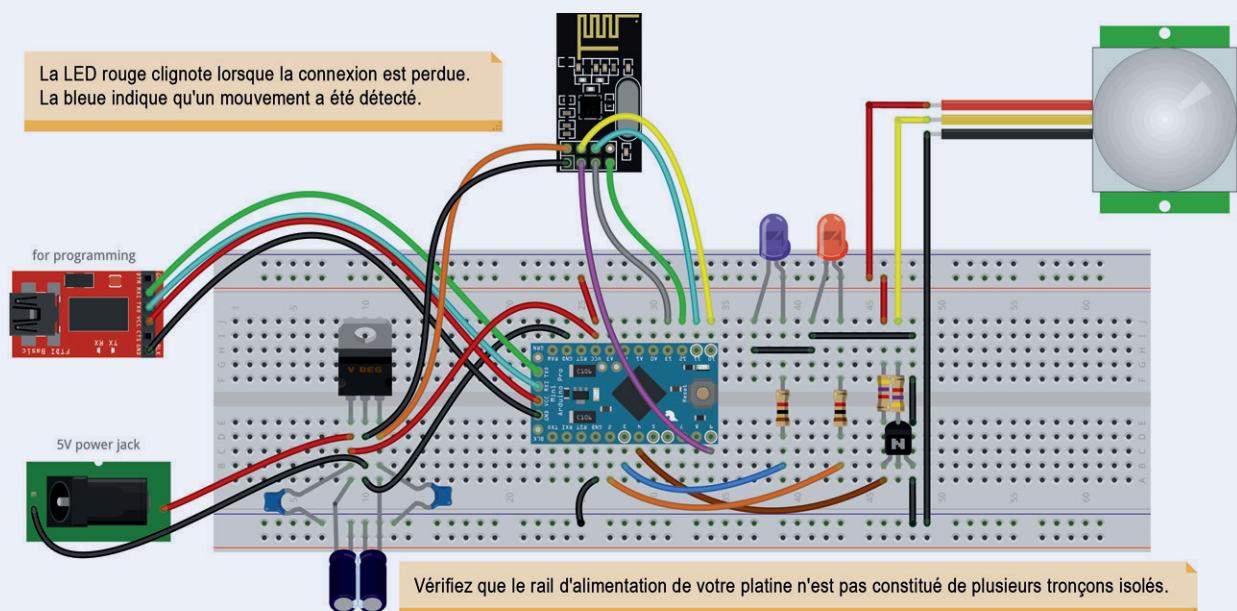
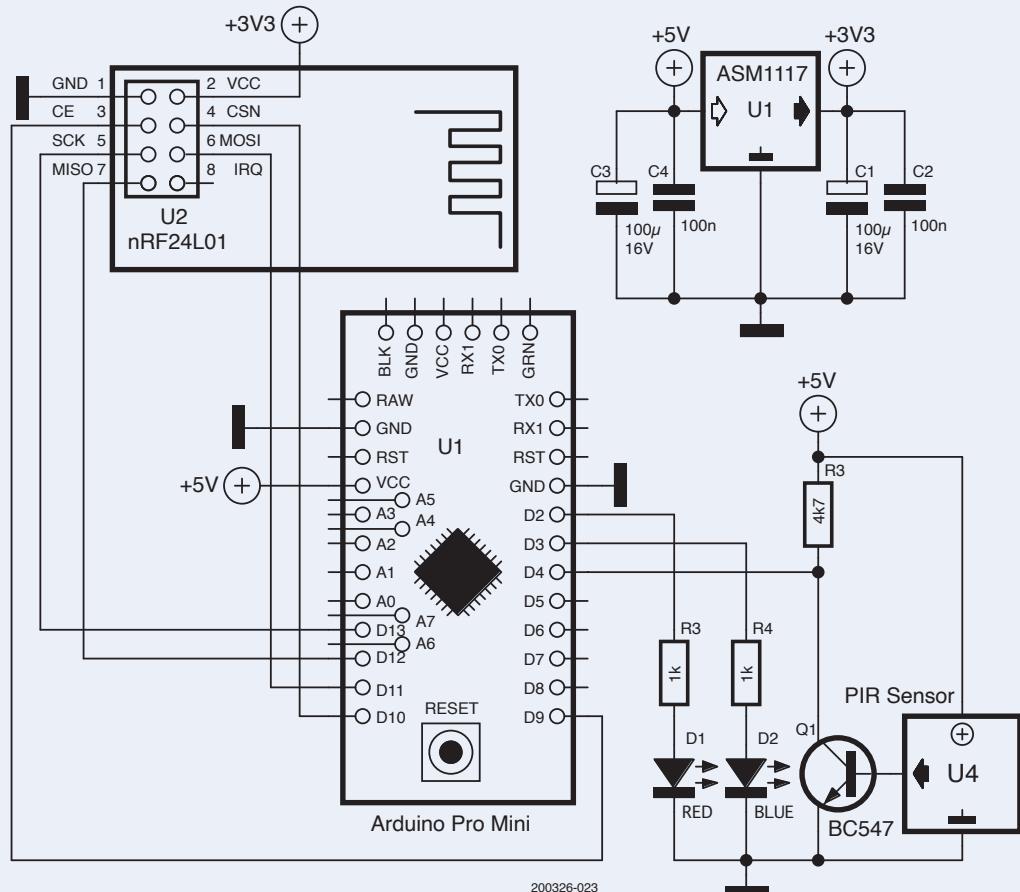


Figure 3. Schéma de la partie émetteur.



Figure 4. Module capteur PIR.



Figure 5. Émetteur avec capteur PIR.



Figure 6. Unité de réception.

Les condensateurs de découplage de l'alimentation doivent être placés aussi près que possible de l'Arduino et du module nRF24L01, respectivement, pour supprimer tout bruit de commutation en provenance de ces cartes. Ceci est souvent négligé dans les projets Arduino et peut causer toutes sortes de problèmes inattendus. Il est également recommandé d'utiliser plusieurs condensateurs de découplage en parallèle avec des valeurs différentes (par exemple,  $C_1 = 100 \mu\text{F}$  et  $C_2 = 100 \text{ nF}$ ). Les condensateurs électrolytiques ne sont pas aussi efficaces à des fréquences élevées que les condensateurs en céramique ou les condensateurs à film de polypropylène. En mettant différents types de condensateurs en parallèle, on crée un filtre plus efficace sur une plus large gamme de fréquences.

Le capteur PIR (**fig. 4**) est connecté en D4 à l'Arduino. Lorsque le capteur PIR nécessite une tension différente de 5 V, il est possible de s'en accommoder. Le transistor Q1 décale (et inverse) le niveau de tension dans le cas où on utilise des capteurs avec, par exemple, une tension de sortie de 3,3 V ou moins. De cette façon, vous pouvez connecter d'autres types de capteurs sans modifier le matériel. Lorsque le capteur PIR a une sortie de niveau 5 V, Q1, R4 et R3 peuvent être omis et la sortie du capteur PIR directement connectée à la broche 4 de l'Arduino Pro Mini. Dans ce cas, il faut adapter le croquis Arduino, afin que l'entrée de déclenchement soit active au niveau haut au lieu de l'être au niveau bas : `#define TRIGGER_ACTIVE_LEVEL 0 /* 0 = actif bas, 1 = actif haut */`

La LED rouge (D1) clignote lorsque la connexion entre l'émetteur et le récepteur est perdue. Lorsque la connexion est rétablie, la LED rouge cesse de clignoter et tout fonctionne à nouveau normalement. La LED bleue (D2) indique que le capteur PIR détecte un mouvement. Cet événement sera transmis au récepteur sous forme d'un code de déclenchement sur un octet. Lorsqu'aucun mouvement n'est détecté par le capteur, un code de présence est transmis au récepteur. De cette façon, le récepteur sait s'il y a détection de mouvement ou non. Le code suivant montre les définitions du code de présence et du code de déclenchement :

```
#define LIVE_BEAT_CODE 0x11 /* code transmis comme
    signal de présence de l'émetteur au récepteur */
#define TRIGGER_CODE 0xAA /* code transmis lorsque
    l'émetteur détecte une activité à l'entrée de
    déclenchement */
```

La **figure 5** montre l'émetteur avec le capteur PIR.

## Récepteur

En guise d'accusé de réception, le récepteur (**fig. 6**) renvoie à

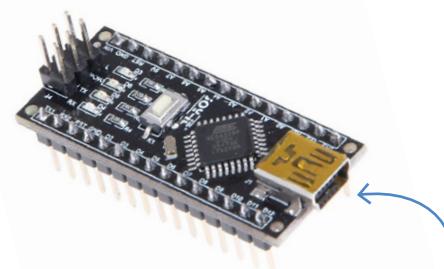
l'émetteur le même code que celui qu'il a reçu. Grâce à cette communication continue entre l'émetteur et le récepteur, tous deux peuvent facilement détecter la perte de connexion.

Le circuit du récepteur (**fig. 7**) est presque identique au circuit de l'émetteur. Dans le récepteur, la LED rouge (D1) se met à clignoter lorsque la connexion est perdue et l'Arduino envoie un signal au haut-parleur en utilisant la commande `tone (x, y)`, avec  $x$  = fréquence et  $y$  = durée. Le haut-parleur est de type  $8 \Omega$  et est connecté à l'Arduino via un condensateur électrolytique. Plus la valeur du condensateur est élevée, plus le volume du haut-parleur sera fort. Pour un signal à 50 Hz pendant 150 ms, on utilise le code suivant :

```
tone(SPEAKER_PIN, 50, 150) ; /* émet un signal
    à 50 Hz pendant 150 ms sur la sortie haut-parleur
    */
```

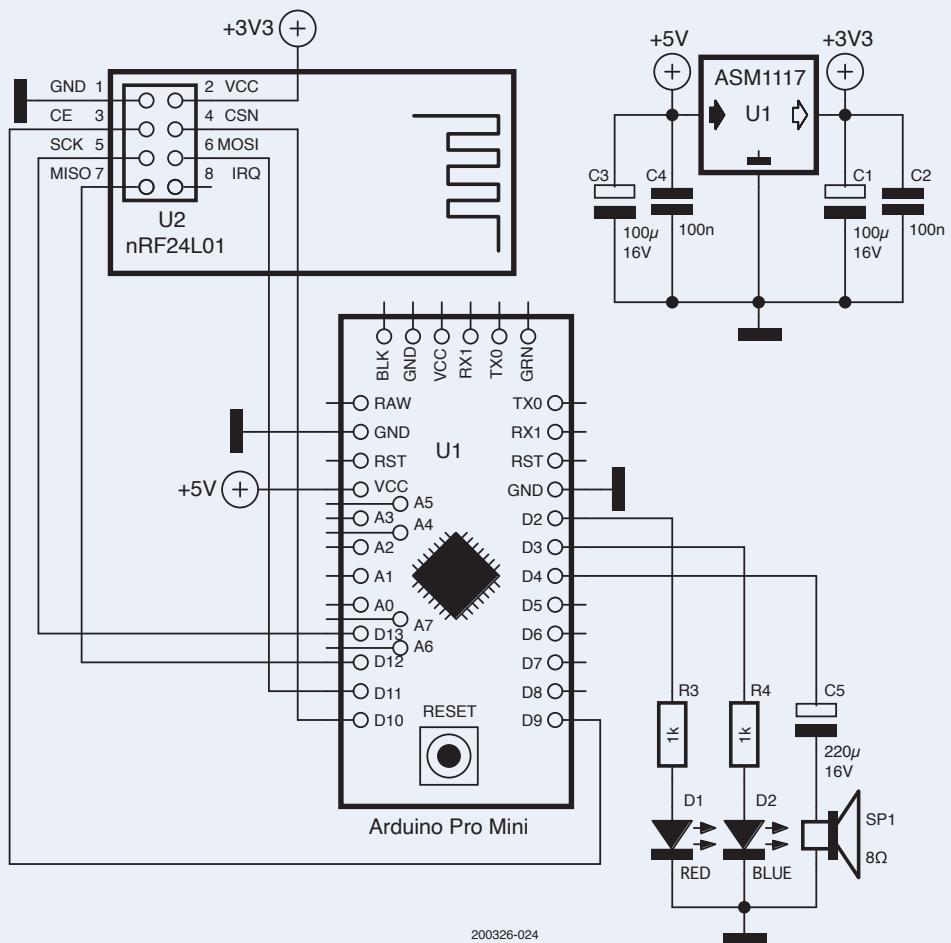
J'ai opté pour un son à basse fréquence (grave) car il est moins gênant/dérangeant que les sons plus aigus. La LED bleue (D2) indique qu'un mouvement est détecté. Vous pouvez connecter un relais à cette sortie via un transistor pour allumer/commander tout autre appareil lorsqu'un mouvement est détecté. ↗

200326-04 - VF : Denis Lafourcade



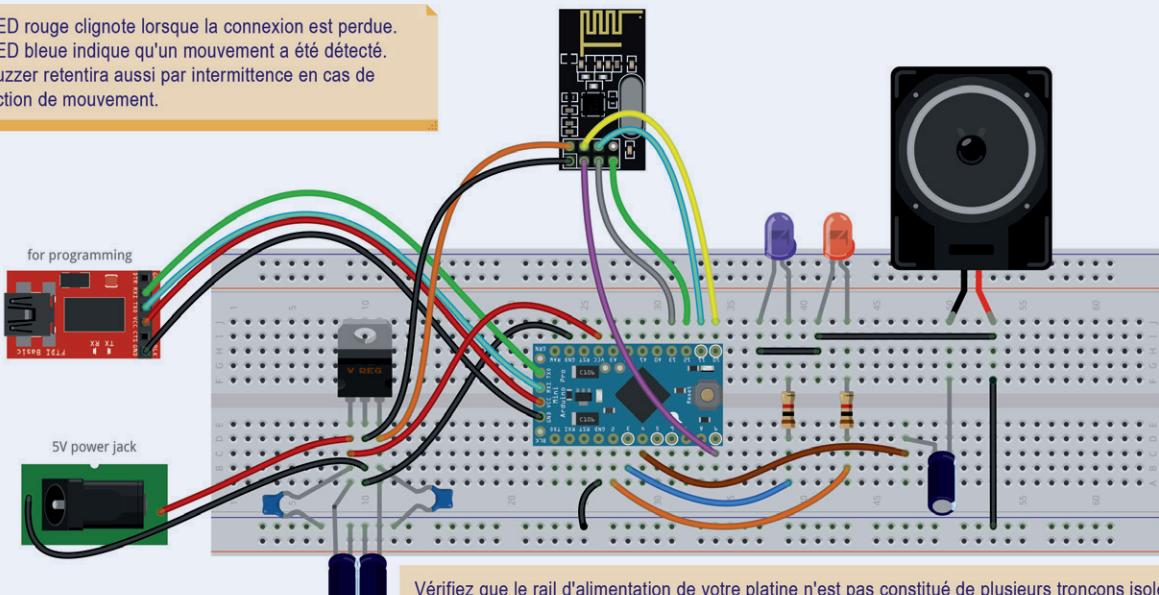
## PRODUITS

- Nano V3 de Joy-IT [www.elektor.fr/18615](http://www.elektor.fr/18615)
- HC-SR501 - Module détecteur de mouvement PIR [www.elektor.fr/18420](http://www.elektor.fr/18420)
- Module émetteur-récepteur sans fil nRF24L01+ (2,4 GHz) [www.elektor.fr/19283](http://www.elektor.fr/19283)



200326-024

La LED rouge clignote lorsque la connexion est perdue.  
La LED bleue indique qu'un mouvement a été détecté.  
Le buzzer retentira aussi par intermittence en cas de détection de mouvement.



fritzing

Figure 7. Schéma du récepteur.

## LIENS

- [1] Fiche technique des modules RF : [www.nordicsemi.com/Products/nRF24-series](http://www.nordicsemi.com/Products/nRF24-series)
- [2] Ce projet sur Elektor Labs : [www.elektormagazine.fr/labs/remote-sensing-with-connection-loss-detection-using-nrf24l01-modules](http://www.elektormagazine.fr/labs/remote-sensing-with-connection-loss-detection-using-nrf24l01-modules)