

Décodeur et générateur de CW

Jean-Paul Gendner F5BU - f5bu@orange.fr

Avec comme signal d'entrée typique la sortie BF d'un récepteur, le système décrit (Fig.1) est principalement un décodeur de signaux CW à base d'un petit circuit de mise en forme du signal BF et d'un module Arduino Nano. Deux utilisations assez différentes sont possibles :

- ▶ Utilisation du montage en « autonome », juste pour le décodage de la télégraphie, avec le texte qui défile sur un afficheur LCD de 4 lignes de 20 caractères ;
- ▶ Utilisation du montage connecté à un ordinateur PC, avec un programme spécifique (Fig. 2), qui permet :
 - de disposer d'un grand écran et d'un affichage amélioré du décodage ;
 - de pouvoir transmettre de la télégraphie en entrant du texte au clavier dans une deuxième fenêtre du programme du PC ;
 - de pouvoir sauvegarder les contacts dans un fichier ADIF, pouvant être importé par la plupart des programmes de Log ;
 - etc.



Fig. 1 - Montage complet avec LCD et connexion PC

Mise en forme du signal

Le circuit de mise en forme du signal est constitué d'un amplificateur opérationnel (U1A sur le schéma de la Fig. 3) monté en inverseur, dont le gain est réglable manuellement de 2 à 1000 à l'aide d'un potentiomètre logarithmique (Fig.4). Pour profiter au maximum de la dynamique de sortie de l'AOP, le point de référence pour l'amplification est choisi proche de la masse, et le gain pour les alternances positives est limité par la diode D1.

Remarques : La loi logarithmique du potentiomètre de gain utilisé n'étant pas proche de la loi théorique, les positions du potentiomètre pour les principales valeurs de gains ont été relevées puis utilisées pour tracer l'échelle. Les dessins des échelles sont réalisés avec mon programme Galva .

Ce circuit est suivi d'un monostable re-déclenchable de 2 ms (U2A). La fréquence du signal BF étant au minimum de 500 Hz, donc d'une période inférieure ou égale à 2 ms, ceci permet d'obtenir en sortie des pulses continus de la durée des pulses CW (en dehors des parasites).

U1B est monté en trigger de Schmitt pour détecter si le signal de sortie de l'amplificateur est proche de la saturation. Son signal de sortie attaque aussi un monostable re-déclenchable de 2 ms afin d'illuminer une LED, ce qui, à l'expérience, facilite grandement le réglage du gain.

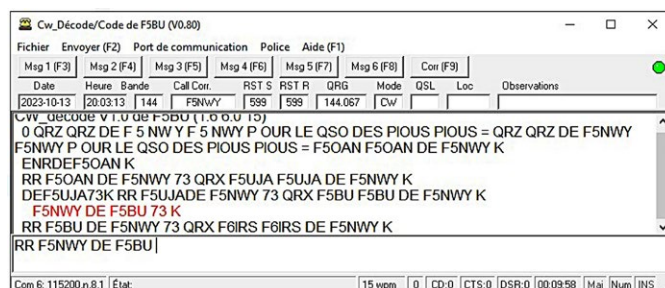


Fig. 2 - Copie d'écran du programme PC (avec une fenêtre de petite taille)

L'Arduino Nano

L'Arduino Nano reçoit sur sa broche A3 le signal de sortie de U2A, c'est à dire des pulses (normalement) conformes aux signaux CW à décoder. Le LCD est de préférence un modèle à 4 lignes, (mais un modèle à deux lignes convient aussi) et à connexion I2C (solution utilisée) ou 4 bits parallèles.

Pour une utilisation avec PC, le signal D11 commande le transistor Q1, monté en collecteur ouvert pour court-circuiter l'entrée « Manip » d'un TX, et une DEL rouge (J9). Si souhaité, la broche A0 peut recevoir le signal d'un potentiomètre pour définir la vitesse de transmission de la CW. Celle-ci peut aussi être modifiée à partir du PC, mais l'expérience montre qu'il est plus agréable et facile de tourner un potentiomètre, y compris durant la transmission d'un texte. La vitesse programmée par défaut dans le microcontrôleur (μC) est de 15 mots par minute.

Décodeur CW F5BU (2023-08)

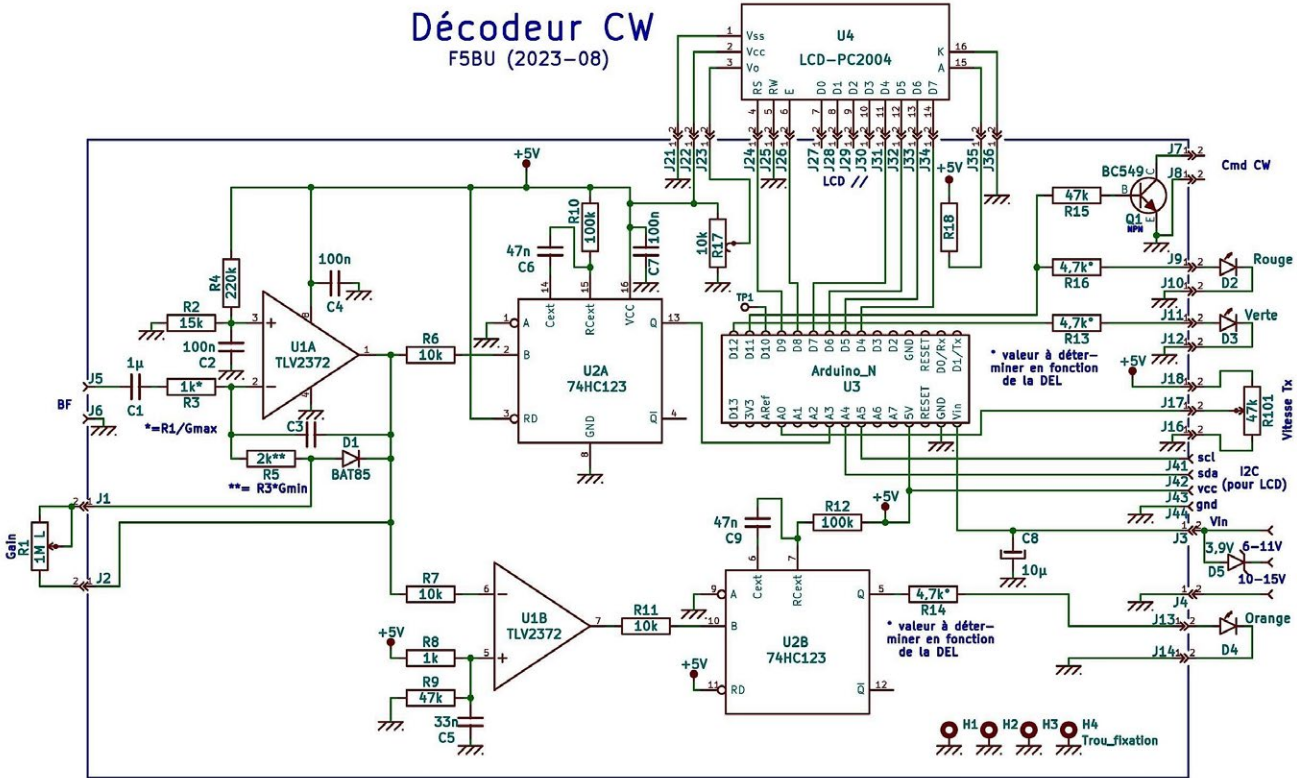


Fig. 3 - Schéma



Fig. 4 - Échelle du gain



Fig. 5 - Échelle de la vitesse de transmission



Fig. 6 - Affichage des messages préenregistrés (#Corr et #RSTS sont remplacés par les valeurs présentes dans la barre de contact)

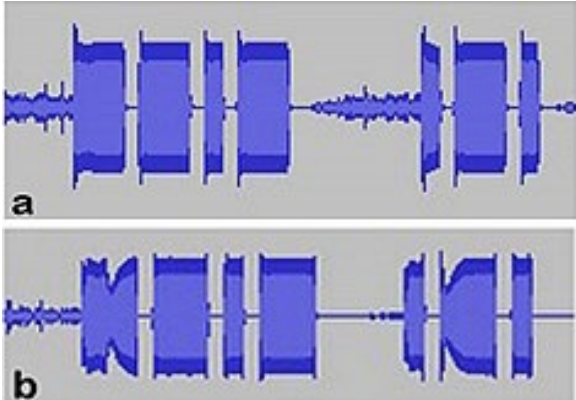


Fig. 7 - Signaux CW courants

La programmation de l'Arduino Nano

Pas facile de décrire le programme qui est assez complexe et comporte plus de 900 lignes, sans compter celles des commentaires.

En résumé simplifié, la bibliothèque PinChangeInterrupt, de NicoHood, qui génère une interruption à chaque changement d'état d'une broche, est utilisée pour détecter, de manière précise, chaque changement d'état de la broche A3. Ceci permet de savoir à tout moment si on est durant un pulse CW ou durant un silence. Ces deux états étant définis, on utilise les interruptions d'un Timer de 2 ms pour obtenir, puis mémoriser, la durée de chacun des états successifs du signal : pulse, silence, etc. Lorsqu'un silence est évalué comme étant suffisamment long, le tableau des durées des pulses et silences est analysé pour déterminer les points et les traits. Pour cela, on commence par déterminer la durée moyenne d'un point. S'il n'y a qu'un pulse, il s'agit soit d'un E soit d'un T, et on est obligé d'utiliser la valeur précédente de la durée du point.

S'il y a au moins un silence, on utilise sa durée, ou la valeur moyenne des silences s'il y en a plusieurs, pour déterminer, pour chaque pulse, s'il s'agit d'un point ou d'un trait. Après cela on recalcule la valeur moyenne du point en tenant compte des silences et des pulses, en divisant par 3 la durée des traits. Le résultat est une suite de 0 (pour les points) et de 1 (pour les traits) qui est recherchée dans le tableau des caractères codés selon Hans Summers GOUPL et Stephen Farthing COXAR (voir mon article « Générateur d'un message CW pour balise » Radio-REF mars 2021 pages 13 à 16) pour afficher le caractère. Les pulses et silences très courts sont traités spécialement pour éliminer le mieux possible les parasites.

Remarques :

- ▶ A propos de l'affichage du décodage sur un LCD. Afficher du texte sur un LCD est enfantin. La difficulté est au niveau du défilement vertical du texte lorsqu'on arrive à la fin de la dernière ligne. Il faut alors décaler les 3 dernières lignes à la place des 3 premières et effacer la dernière ligne. Or, réalisée « normalement », cette opération prend plusieurs centaines de ms et bloquerait le traitement du décodage du signal d'entrée durant tout ce temps. Cette opération est donc réalisée en affichant 1 seul caractère à chaque passage dans la boucle (Loop) du programme. Vu la fréquence de passage dans cette boucle, cela n'est pas perceptible.
- ▶ Au-delà d'un certain seuil, le réglage de la vitesse de transmission se fait en fonction de la fraction de la tension d'alimentation fournie par le potentiomètre R101.

Ce potentiomètre étant un potentiomètre linéaire, c'est une programmation linéaire de la vitesse de 8 à 50 mots par minute (wpm) qui a été utilisée dans un premier temps. Mais ceci ne donnait pas satisfaction, car le réglage en début d'échelle, aux environs de 10 wpm, était trop pointu. C'est alors une échelle avec une fonction à la puissance 4 qui a été visée. Toutefois, pour ne pas avoir à réaliser des calculs de puissance 4 par le μC , cette courbe est approchée par 6 segments linéaires successifs (Fig. 5).

- ▶ Le début des potentiomètres linéaires n'étant souvent pas linéaire, une calibration a été réalisée en utilisant, dans le programme, non pas les valeurs théoriques, mais les valeurs mesurées sur le curseur de R101 pour les 8 graduations charnières : fin de la zone « Défaut », 8, 10, 15, 20, 30, 40, 50.

Le programme du PC

Via la liaison USB, le programme du PC reçoit bien sûr les caractères décodés par l'Arduino et les affiche dans une grande fenêtre.

De plus, vu les possibilités d'affichage sur un écran de PC et les facilités de programmation, bien d'autres fonctionnalités sont ajoutées. Notamment :

- ▶ pour chaque caractère reçu, on demande au μC de renvoyer, sous forme codée sur un octet, la vitesse de réception (calculée à partir de la durée du point), qui est immédiatement affichée dans une ligne de statuts du programme du PC ;
- ▶ une deuxième fenêtre, plus petite, sert à entrer du texte à transmettre, soit en direct, soit à la demande ; 12 messages modifiables, préenregistrés dans un fichier de configuration, peuvent être ajoutés dans cette fenêtre par simple appui sur une touche Fx (Fig.6) ; une barre permet de gérer les informations pour les contacts ; le fichier de configuration contient également le numéro du port série pour la liaison USB (le même que celui dans l'IDE Arduino), l'indicatif, et d'autres paramètres ; ...

Utilisation

Le système doit simplement recevoir le signal BF d'un récepteur. La plupart de ceux-ci ayant maintenant des possibilités étendues de filtrage (bande passante en CW pour mon TS-2000 réglable entre 50 et 2000 Hz), aucun filtre n'a été incorporé dans le montage.

Pour que le décodage se fasse dans les meilleures conditions, il faut régler le gain et/ou le niveau de sortie BF du récepteur pour chaque signal.

Les meilleurs résultats sont généralement obtenus lorsqu'on est en limite de saturation, c'est à dire que la LED (orange, J13) de saturation s'illumine sur les pointes de signal supérieures aux signaux CW (LED verte, J11), qui sont souvent présentes (Fig.7a).

Toutefois, le réglage dépend de la forme des signaux, et il faut trouver celui qui donne les meilleurs résultats. Par exemple, pour un signal présentant des creux (Fig.7b), ce qui n'est pas perceptible à l'oreille, il faut augmenter le gain. Dans certains cas, lorsque le niveau de bruit remonte de manière importante entre deux signes ou mots, il peut être intéressant de supprimer ou de modifier la constante de temps du contrôle automatique de gain (CAG).

Plusieurs paramètres utilisés par le μC pour le décodage dépendent beaucoup des opérateurs. Ils sont définis par défaut dans le programme, mais peuvent être modifiés dans les premières lignes du code source. Ce qui nécessite évidemment de recompiler le programme du μC . Afin de pouvoir modifier des paramètres en cours de décodage, le programme du PC permet d'en modifier facilement les plus importants :

- ▶ `k_e_cars`, qui est le coefficient multiplicateur de la durée du point pour définir la durée à partir de laquelle un espace sera considéré comme un espace entre caractères. Sa valeur par défaut est de 1,6, la valeur théorique étant de 3.
- ▶ `k_e_mots`, qui est le coefficient multiplicateur de la durée du point pour définir la durée à partir de laquelle un espace sera considéré comme un espace entre mots. Sa valeur par défaut est de 6 pour une valeur théorique de 7.
- ▶ `V_TX`, qui est la vitesse souhaitée, en mots par minutes pour la transmission.

Historique

Essayant de me remettre pour la énième fois à la télégraphie, j'ai commencé à participer aussi régulièrement que possible au QSO local des « Pious Pious », sur 144,067 MHz animé par Bruno F5NWY. Mais ayant dû suspendre ma participation durant quelques temps à plusieurs reprises, j'ai vite réalisé qu'il me sera impossible de dépasser les 10 mots / minutes, et cela encore à condition d'en faire régulièrement. Alors, aimant bricoler avec les Arduino, pourquoi ne pas essayer de réaliser un décodeur ?

Le premier montage comportait un ampli suivi d'un trigger de Schmitt, et le μC déterminait la durée de chaque pulse (ici pulse BF de moins de 2 ms) et de chaque silence. Cela marchait assez bien lorsque le rapport signal/bruit était bon.

Suite à des discussions avec des amis, j'ai alors essayé d'utiliser un circuit décodeur de tonalité LM567 après amplification du signal. Mais, cette solution a été abandonnée relativement rapidement, car le temps de réponse du décodeur était incompatible avec de la CW, même lente, dès que l'on cherchait à obtenir une bande passante étroite.

Le montage suivant a été celui décrit.

Côté programmation du μC , il y a aussi eu de nombreuses versions. Au moins 6 ou 7, mais il serait trop long de décrire toutes mes cogitations et essais successifs !

Remarque : Pour mieux extraire le signal du bruit, il serait sans doute intéressant de le gérer directement via la carte son du PC, mais cela, je ne sais pas le faire.

Réalisation

La Fig. 8 montre le circuit imprimé réalisé et la Fig. 9 l'intégration du montage dans son boîtier spécifique. Il fallait trouver un boîtier assez grand, mais pas trop. Ne trouvant pas quelque chose qui me satisfasse, je me suis lancé pour dessiner une image 3D de ce que je souhaitais avec FreeCad (Fig.10) et, ne disposant pas d'une imprimante 3D, je l'ai fait réaliser, pour voir, par le fabricant qui réalise les circuits imprimés.

Pour une utilisation avec PC, l'alimentation se fait via la connexion USB. Autrement, le montage, qui consomme environ 62 mA (dont une quarantaine pour l'éclairage du LCD), peut être alimenté soit par V_{in} avec une tension entre 6 et 12 V (pour ne pas risquer de dépasser la puissance maximum pouvant être dissipée par le régulateur du module Arduino) ; soit par la même entrée, mais avec une diode Zener en série et alors avec une tension supérieure ; soit par du 5 V sur le 5 V du montage.

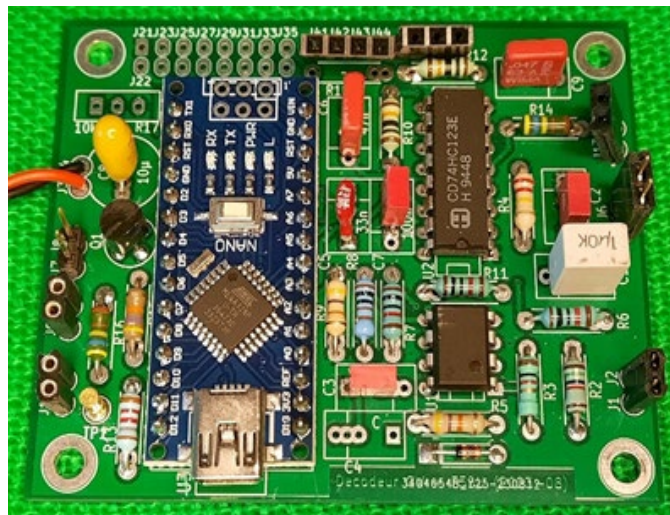


Fig. 8 – Le circuit imprimé

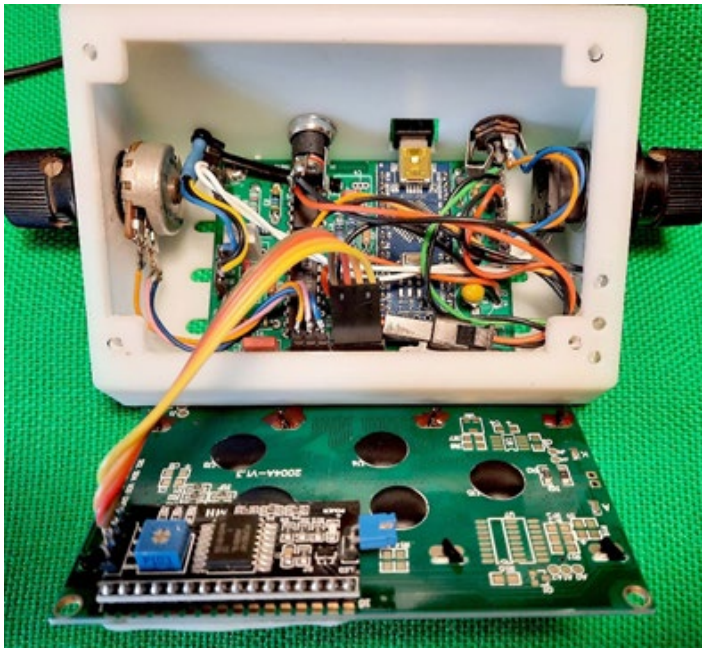


Fig. 9 - Intégration du montage dans son boîtier

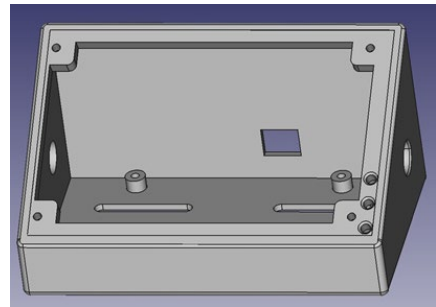


Fig.10 - Image 3D du boîtier

Pour ceux qui seraient intéressés pour réaliser ce montage, il me reste quelques circuits imprimés (et il est possible d'en refaire si besoin).

Un fichier zip contenant le schéma, le fichier .ino pour l'Arduino, le fichier d'installation du programme pour le PC, son fichier d'aide, le fichier .dat pour les échelles, etc., peut être envoyé sur simple demande par courriel.

Le fichier .stl pour la réalisation du boîtier peut également être fourni.

POLO MANCHE LONGUE HOMME/FEMME

POLO HOMME : PEF040 - 100 % coton ringspun - PRIX : 32.00 €

POLO FEMME : PEF041 - 100 % coton piqué peigné ringspun - PRIX : 32.00 €



SAVOIR FAIRE

- courant faible
- matériels électroniques
- assemblages
- intégrations du matériel
- installations sur site
- tirages de câble
- montages véhicule
- vérifier les conformités
- faire des mesures
- contrôler le matériel
- respecter les procédures
- ...

SAVOIR ETRE

- qualifier sont travail
- gérer les priorités
- respecter les objectifs
- travailler en autonomie
- faire preuve de curiosité
- être polyvalent
- implication personnel
- dynamisme, réactivité
- déplacements
- ambition et persévérance
- ...



J'AIME
LES CHALLENGES

J'AI CHOISI UN MÉTIER DE
TERRAIN

JE SUIS
TECHNICIEN RADIO

Etre dans l'action:

Pour moi, c'est s'assurer du bon déroulement d'un chantier. Accompagner et guider mes équipes en trouvant des solutions pour finaliser l'installation et la mise en service d'un réseau radio . Alors oui, j'ai choisi d'être technicien radio, un métier de terrain.

TALCO LR recrute un technicien monteur dans les télécoms sur Montpellier (34).

www.talco-lr.com

