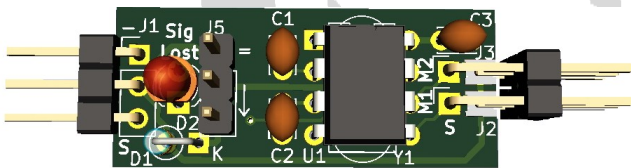


# *Xany2Misc*

**Un cordon  
d'adaptation pour  
interfacer les  
modules Multi-Switch  
à commande par  
impulsion du  
commerce avec  
*OpenAVRc***



## *Manuel Utilisateur Adaptateur Xany2Misc*



Copyright *OpenAVRc* 2018

## Table des matières

1	CE DOCUMENT.....	3
1.1	Versions.....	3
1.2	Copyright.....	3
1.3	Avertissement.....	3
1.4	Contenu.....	3
2	PRESENTATION DE XANY2MISC.....	4
2.1	Vue d'ensemble.....	4
2.2	Facilité de migration vers le 2.4GHz.....	5
2.3	Résistance aux perturbations radio-électriques.....	5
2.4	Spécifications du cordon adaptateur Xany2Misc.....	5
3	SCHEMA DE L'ADAPTATEUR XANY2MSX/XANY2MISC.....	6
4	REALISATION DE L'ADAPTATEUR XANY2MISC.....	7
4.1	Liste de composants.....	7
4.2	Réalisation d'un circuit imprimé.....	8
4.3	Montage de l'ATtiny85 sur support tulipe.....	9
4.4	Chargement du Firmware dans l'ATtiny85.....	9
5	UTILISATION.....	10
5.1	Connexion au récepteur.....	10
5.2	Mode avancé.....	11
5.2.1	Utilisation du port série de Xany2Misc.....	11
5.2.2	Les messages de commande de Xany2Misc.....	12
5.3	Exemple de configurations réelles.....	13
5.3.1	Contrôle d'un Module Conrad 7 fonctions (7 commandes).....	13
5.3.2	Contrôle de 2 Modules Conrad 7 fonctions (14 commandes).....	13
5.3.3	Contrôle d'un Module son Beier USM-RC-2 (16 commandes).....	14
5.3.4	Contrôle d'un Module son Beier USM-RC-2 (30 commandes).....	14
5.3.5	Contrôle du décodeur Multi-Switch NVM (14 commandes).....	15

# 1 CE DOCUMENT

## 1.1 Versions

Version	Date	Raison de l'évolution
0.1	12/06/2021	Création

## 1.2 Copyright

Ce document est Copyright © 2018 - 2021 **OpenAVRc**.

## 1.3 Avertissement

L'équipe **OpenAVRc** n'est aucunement responsable des dommages qui pourraient découler de la mauvaise utilisation ou d'un éventuel dysfonctionnement de l'émetteur **OpenAVRc**, de l'adaptateur **Xany2Msx/Xany2Misc** et/ou des logiciels associés.

Il appartient donc à l'utilisateur final d'en mesurer, d'en assumer les risques et de respecter la législation en vigueur selon le pays d'utilisation.

## 1.4 Contenu

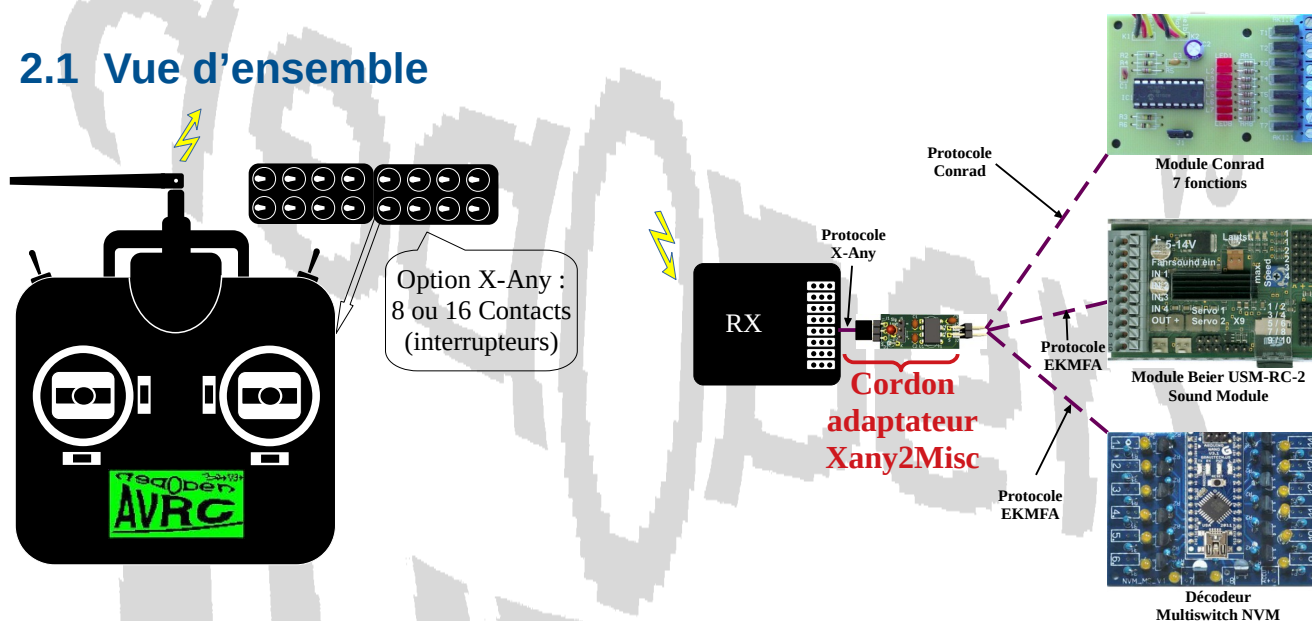
Ce document décrit la réalisation du cordon adaptateur **Xany2Misc** pour l'interfaçage avec les modules Multi-Switches à commande par impulsions supportés du commerce (**Conrad 7 fonctions**, **module son Beier USM-RC-2**, décodeur **Multiswitch NVM**), ainsi que le paramétrage pour son utilisation avec l'émetteur **OpenAVRc**.

Le cordon adaptateur **Xany2Misc** utilise la carte **Xany2Msx**.

La seule différence est le firmware qui se nomme **Xany2Misc**.

## 2 PRESENTATION DE XANY2MISC

### 2.1 Vue d'ensemble



Le cordon adaptateur **Xany2Misc** permet d'utiliser des modules Multi-Switches à commande par impulsions sur une ou deux voies « X-Any » de l'ensemble RC **OpenAVRc**.

Les décodeurs commandés par impulsion du commerce actuellement supportés sont :

- |           |                       |
|-----------|-----------------------|
| 1. Conrad | Module 7 fonctions    |
| 2. Beier  | Module son USM-RC-2   |
| 3. NVM    | Décodeur Multi-Switch |

#### Note :

1. Côté émetteur, il n'y a pas besoin du codeur spécifique correspondant au modèle de décodeur.
2. Le codeur unique et universel est intégré à l'émetteur **OpenAVRc** et utilise le protocole **X-Any**. Il est possible de déclarer jusqu'à 4 instances d' **X-Any** côté émetteur afin d'autoriser jusqu'à 64 commandes Tout-Ou-Rien. Se reporter à la documentation relative à **OpenAVRc** pour configurer **X-Any** côté émetteur.

## 2.2 Facilité de migration vers le 2.4GHz

Bon nombre de modélistes adeptes d'animations (donc de modules Multi-Switches) ne migrent pas vers les radios 2.4 GHz, puisque les anciens codeurs/décodeurs Multi-Switches sont rarement supportés en 2.4 GHz, ou bien, il est nécessaire de réinvestir dans des modules compatibles avec le 2.4 GHz.

Avec le cordon adaptateur **Xany2Misc**, il est très facile de migrer un ensemble RC FM/PPM vers l'émetteur **OpenAVRc** en 2.4 GHz tout en conservant le bon fonctionnement des modules Multi-Switches existants : en effet, **X-Any** est un protocole universel fonctionnant aussi bien en FM/PPM qu'en 2.4 GHz.

## 2.3 Résistance aux perturbations radio-électriques

Du fait de l'utilisation du protocole **X-Any** pour transporter les ordres entre l'émetteur et le récepteur, les corruptions liées aux perturbations radio-électriques sont détectées à la réception à l'aide de la somme de contrôle (checksum 8 bits).

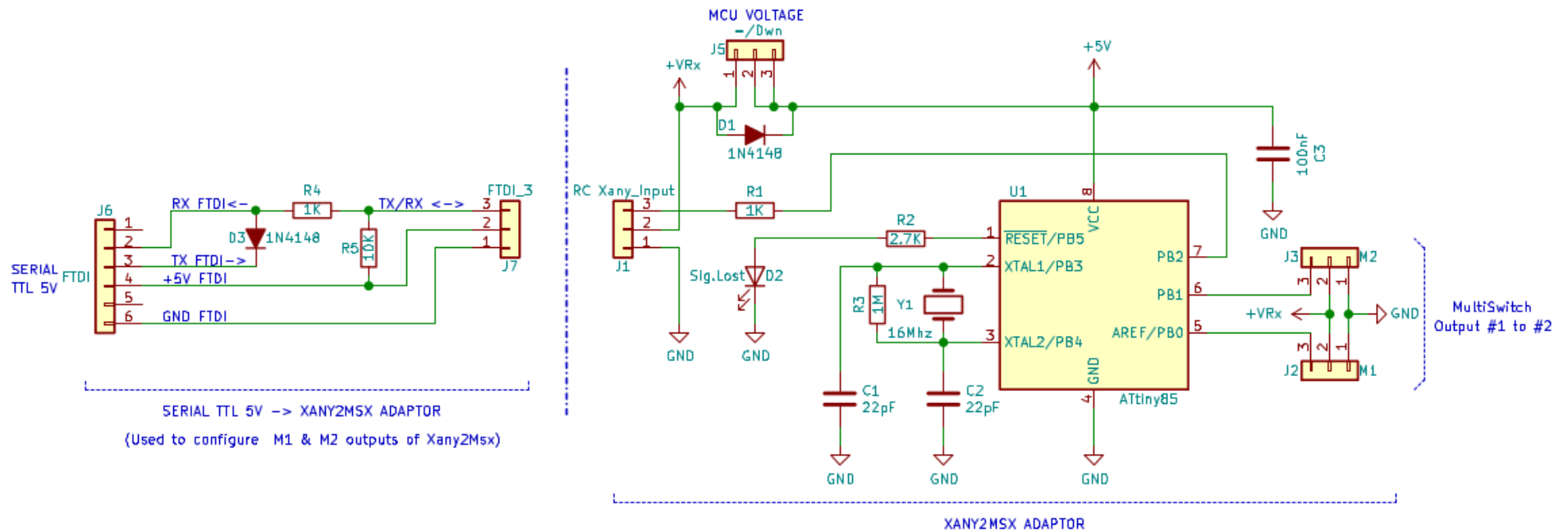
Contrairement au protocole FM/PPM, le cordon adaptateur **Xany2Misc** est en mesure de détecter la corruption lors du transport et peut ainsi passer en mode Failsafe en fournissant des commandes à 0 au(x) décodeur(s) Multi-Switch connecté(s) sur la(les) sortie(s) de **Xany2Misc**.

Donc, finis les soubresauts incontrôlés en sortie de décodeur Multi-Switch !

## 2.4 Spécifications du cordon adaptateur Xany2Misc

Spécification	Valeur / Caractéristique	Note
Alimentation	+3.3V à +6.6V	Mettre le cavalier « tension récepteur » sur « = » ou « ↓ » selon la valeur de la tension fournie par le récepteur
Protocole Entrant	<ul style="list-style-type: none"><li>- Protocole numérique <b>X-Any</b> universel utilisé par <b>OpenAVRc</b> pour les accessoires distants</li><li>- Contrôle d'intégrité par checksum 8 bits</li><li>- Fonctionne avec tous les protocoles, y compris en FM/PPM et en 2.4 GHz :<ul style="list-style-type: none"><li>- Protocole PPM</li><li>- Protocoles SPIRfMod</li><li>- Protocoles MultiMod</li></ul></li></ul>	Contrairement à bon nombre de modules Multi-Switches, l'adaptateur <b>Xany2Misc</b> pour Multi-Switches fonctionne également avec les modules HF en 2.4 GHz
Protocoles Sortants	Les protocoles des modèles de Multi-Switches à commande par impulsions actuellement supportés sont : <ul style="list-style-type: none"><li>- Conrad Module 7 fonctions</li><li>- Beier Module son USM-RC-2</li><li>- NVM Multi Switch</li></ul>	Les 2 sorties Multiswitch <b>M1</b> à <b>M2</b> peut être configurée pour piloter l'un de ces 3 modules.
Failsafe	Désactive toutes les sorties si pas de signal valide pendant 1.5s : La Led rouge « <b>Signal Lost</b> » s'allume.	En présence de signal valide, a Led rouge « <b>Signal Lost</b> » est éteinte.

### 3 SCHEMA DE L'ADAPTATEUR XANY2MSX/XANY2MISC



An adaptor for using many commercial Multiswitch decoders with OpenAVRc transmitter  
It behaves like a protocol converter: X-Any protocol → Manufacturer protocol  
Supported decoders: Futaba MS8 (1513), Robbe MS16 (8369), Robbe MS12+2xProp (8370)  
Multiplex Multinaut MS12+2xEngines (75882), Graupner Nautic MS16 (4159)

**OpenAVRc Team (C) 2018 – 2021**

Sheet: /

File: Xany2Msx.sch

**Title: Xany2Msx Adaptor**

Size: A4

Date: 2021-04-04

**Rev: 1.0**

KiCad E.D.A. eeschema 5.1.9-73d0e3b20d88ubuntu20.04.1

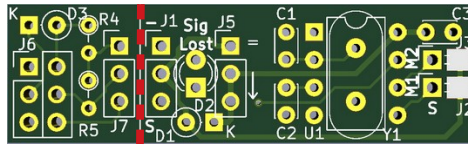
Id: 1/1

## 4 REALISATION DE L'ADAPTATEUR XANY2MISC

### 4.1 Liste de composants

Composants	Quantité	Remarque
Support « tulipe » 8 points	1	Indispensable, car présence Quartz
Quartz 16MHz	1	
Condensateur céramique 22 pF	2	
Condensateur film plastique MKP 100 nF	1	Découplage d'alimentation
Résistance 1K 1/4 W	2	
Résistance 2.7K 1/4 W	1	
Résistance 10K 1/4 W	1	
Led rouge diamètre 3 mm haut rendement	1	
Diode 1N4148	2	
Microcontrôleur ATtiny85-20PU	1	A charger avec le HEX du Firmware
Cordon de servo avec connecteur 3 points femelle	1	
Barrette de connecteur 3 points au pas de 2.54 mm	6	2 pour les sorties <b>M1</b> à <b>M2</b> , 1 pour la sélection du niveau d'alimentation et 3 pour l'adaptateur FTDI
Cavalier pour barrette 3 points	1	Pour sélection du niveau d'alimentation
Gaine thermorétractable (diamètre à froid : 18 mm)	2 cm	Uniquement si câblage « en volant »

## 4.2 Réalisation d'un circuit imprimé

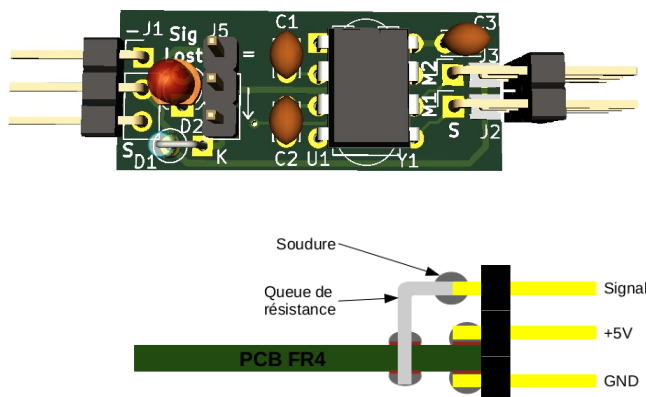


Note :

A réception du circuit imprimé, il est nécessaire de le couper au niveau de la ligne en pointillé **rouge** afin de séparer les 2 parties.

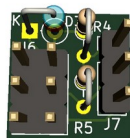
Le circuit imprimé proposé est composé de 2 parties :

1. L'adaptateur **Xany2Misc** proprement dit : (partie droite)



Montage des connecteurs J2 et J3 de M1 et M2

2. Un adaptateur FTDI vers **Xany2Misc**: (partie gauche)



Cet adaptateur FTDI vers **Xany2Misc** est nécessaire pour le paramétrage avancé de **Xany2Misc**.

Une fois paramétré, cet adaptateur FTDI vers **Xany2Misc** est déconnecté et le montage se limite à l'adaptateur **Xany2Misc** (montage plus compact dans le modèle réduit).



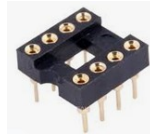
## 4.3 Montage de l'ATtiny85 sur support tulipe

### **ATTENTION :**

L'**ATtiny85** doit être programmé **avant** son positionnement sur le circuit imprimé de [Xany2Misc](#).

De plus, pour plus de compacité, le **quartz de 16MHz** est logé sous l'**ATtiny85**.

C'est pourquoi, il est nécessaire de le monter **l'ATtiny85** sur un support tulipe de 8 points :



Il est également possible d'utiliser 2 portions (de 4 points) de barrette tulipe sécable :



Dans les 2 cas, bien vérifier l'orientation de l'**ATtiny85** avant de mettre sous tension [Xany2Misc](#).

## 4.4 Chargement du Firmware dans l'ATtiny85

TO DO

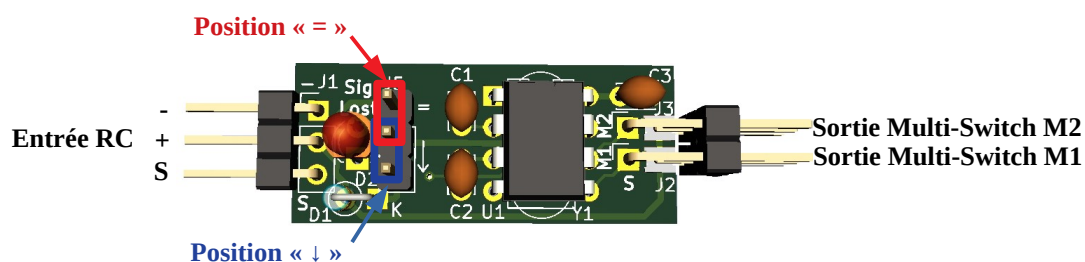
## 5 UTILISATION

### 5.1 Connexion au récepteur

Avant de connecter le cordon adaptateur **Xany2Misc** au récepteur, il est impératif de mesurer la tension fournie par le récepteur.

Si la tension disponible entre les broches – et + du connecteur 3 points de la voie utilisée est :

1. Inférieure à 5.7V, placer le cavalier « tension récepteur » sur « = »
2. Supérieure à 5.7V, placer le cavalier « tension récepteur » sur « ↓ »



## 5.2 Mode avancé

Le cordon adaptateur **Xany2Misc** dispose d'un accès pour les configurations avancées : un port série TTL.

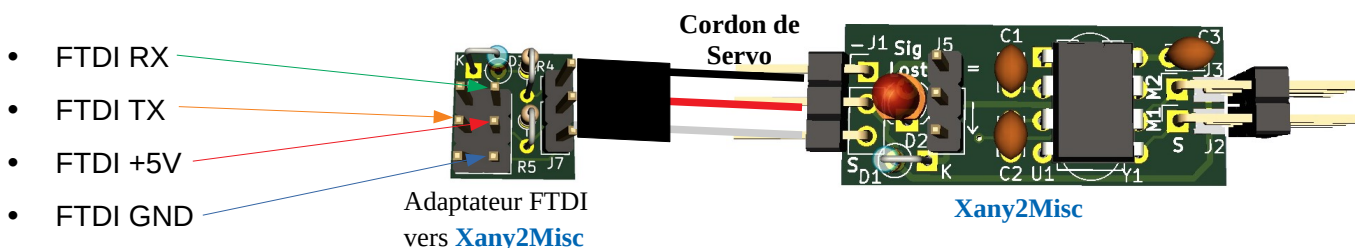
Pour l'utiliser, il est nécessaire de déconnecter l'entrée RC de **Xany2Misc** afin de ne pas perturber le fonctionnement de l'accès série.

C'est cet accès série qui permet de définir le modèle de décodeur Multi-Switch à connecter aux sorties **M1** et **M2**.

### 5.2.1 Utilisation du port série de Xany2Misc

Pour accéder au port série de **Xany2Misc**, il faut un câble USB/Série TTL par exemple de type « FTDI ».

Les points nécessaires sur le câble USB/Série TTL sont :

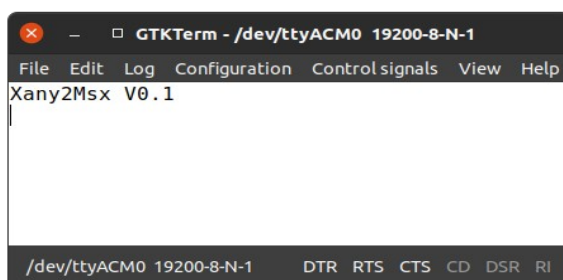


Le câble USB/Série TTL doit être capable de fournir le +5V pour alimenter **Xany2Misc** car pendant la configuration avancée, **Xany2Misc** doit être déconnecté du récepteur.

1. Connecter le câble USB/Série TTL sur le port USB d'un PC pour le côté USB
2. Sur le PC, ouvrir un Terminal série, par exemple, PuTTY, TeraTerm, HyperTerminal, GtKTerm, ou encore CoolTerm avec les paramètres suivants : 19200 bauds, 8 bits de données, 1 bit de stop, pas de parité.

Selon le Terminal série, il peut être nécessaire d'activer les retours à la ligne automatique sur réception de CR/LF pour avoir un bon affichage.

3. Connecter le câble USB/Série TTL sur le connecteur J6 de l'adaptateur FTDI vers **Xany2Misc**
4. Connecter J7 de l'adaptateur FTDI vers **Xany2Misc** à J1 de **Xany2Misc**, ceci va alimenter **Xany2Misc**
5. **Dans les 3 secondes** après le branchement sur **Xany2Misc**, appuyer sur la touche « Entrée » de votre clavier, le message « Xany2Misc VX.Y » doit apparaître sur le Terminal série comme illustré ci-dessous. Si ce n'est pas le cas, déconnecter J7 de J1 de **Xany2Misc** et recommencer à l'étape 4 ci-dessus.



Exemple de connexion avec le Terminal GtKTerm sous Linux

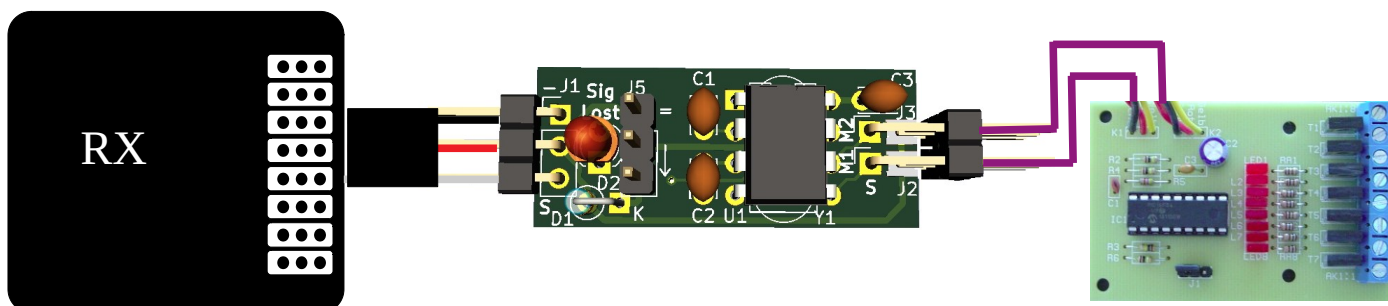
## 5.2.2 Les messages de commande de Xany2Misc

La liste des messages supportés par **Xany2Misc** est donnée dans la table suivante :

← Commande/ → Réponse	Action	Remarque
← Enter → Xany2Misc Vx.y	Si envoyée pendant les 3 secondes après la mise sous tension, passe en mode Terminal	Si échec et 3 secondes écoulées, débrancher puis rebrancher le connecteur J7 3 points du câble USB/Série TTL
← Mx? → Mx=F ou Mx=F1 ou Mx=F2	Si x est compris entre 1 et 2, retourne la configuration de la Sortie Multiswitch N°x : - F étant le Fabricant: - C pour Conrad - B pour Beier - N pour NVM	Renvoie ERR, si : - Valeur x non comprise entre 1 et 2 Exemple de réponses : M1=C0 M1=C1 M1=C2 M2=B1 M2=B2 M2=N0
← Mx=F0 ou Mx=F1 ou Mx=F2 ou Mx=OFF → Mx	Définit la sortie Multiswitch x comme étant du Fabricant F pour le code Produit P ou désactive la Sortie (OFF)	Exemple de commande: M1=C1 M2=OFF
← D? → D=DurMs	D=Durées R, B, I, L pour EKMFA R: Durée de Reset (validation) B: Durée de Burst I: Durée Inter-Burst L : Durée de validation de la dernière commande	Exemple de réponses : ← R? → R=0100 ← B? → B=0050 ← I? → I=0050 ← L? → L=0150
← D=DurMs → D	D=Durées R, B, I, L pour EKMFA R: Durée de Reset (validation) B: Durée de Burst I: Durée Inter-Burst L : Durée de validation de la dernière commande	Renvoie ERR, si : - Valeur DurMs non comprise entre 10 et 2000 ms Exemple de réponses : ← R=200 → R ← B=100 → B ← I=100 → I ← L=300 → L
← W? → W=0xHHHH[HHHH]	Retourne la valeur des sorties (hex)	Sur 8 digits si Mx=B2, sinon 4
← W=0xHHHH[HHHH] → W	Définit la valeur des sorties (hex)	Sur 8 digits si Mx=B2, sinon 4
← Fx=0/1 → Fx	Appelle la fonction x pour Beier/NVM Seule, Fx=1 est supportée.	Désactive/Active la sortie x pour Conrad 7 fonctions

## 5.3 Exemple de configurations réelles

### 5.3.1 Contrôle d'un Module Conrad 7 fonctions (7 commandes)



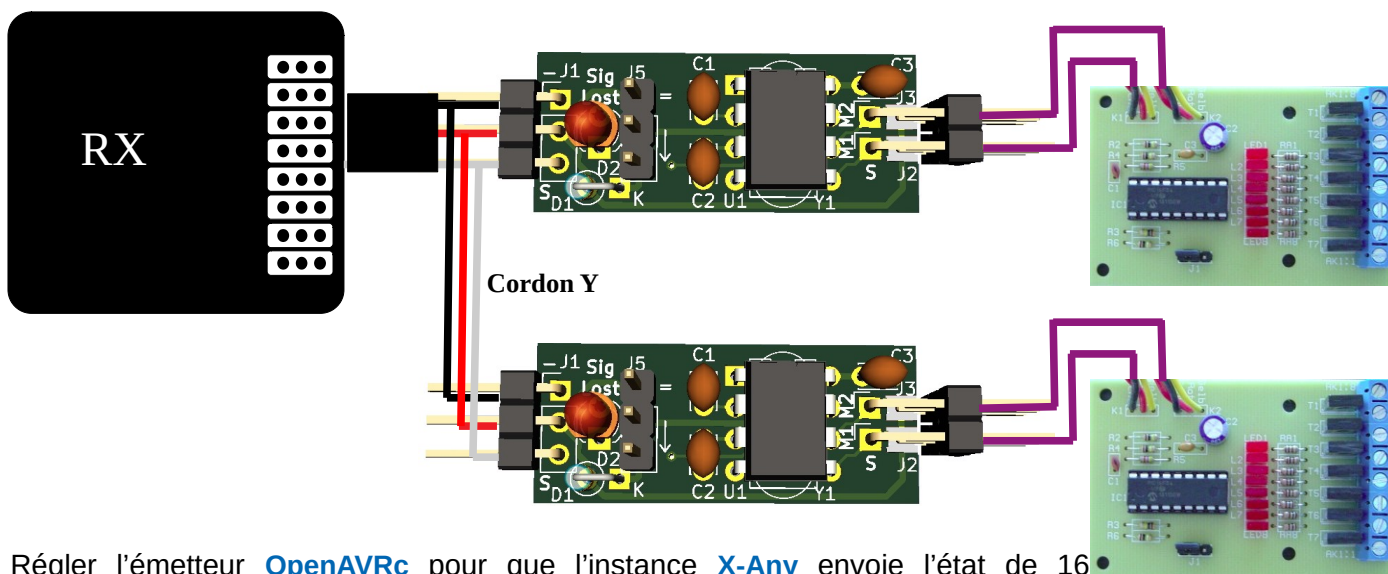
Régler l'émetteur **OpenAVRc** pour que l'instance **X-Any** envoie l'état de 8 contacts (et pas 16).

Pour commander un **Module Conrad 7 fonctions** sur les sorties M1 et M2, il suffit de paramétrer **Xany2Misc** avec les commandes :

M1=C0 ou M2=C0

Fx=0/1 pour désactiver/activer la sortie x (x de 1 à 7)

### 5.3.2 Contrôle de 2 Modules Conrad 7 fonctions (14 commandes)



Régler l'émetteur **OpenAVRc** pour que l'instance **X-Any** envoie l'état de 16 contacts.

Pour commander deux **Module Conrad 7 fonctions** sur les sorties M1 et M2 de deux adaptateurs **Xany2Misc**, il suffit de paramétrer **Xany2Misc** avec les commandes :

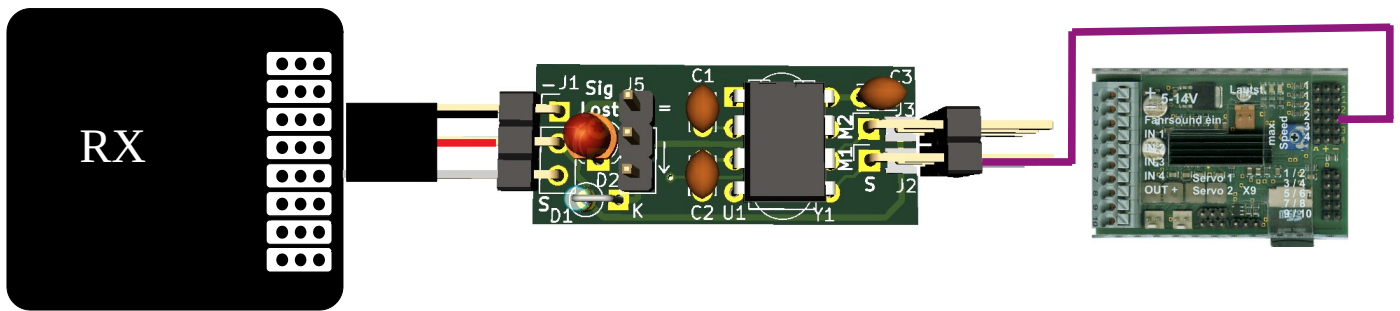
M1=C1 ou M2=C1 sur le premier adaptateur **Xany2Misc**

Fx=0/1 pour désactiver/activer la sortie x (x de 1 à 7) pour le premier adaptateur **Xany2Misc**

M1=C2 ou M2=C2 sur le second adaptateur **Xany2Misc**

Fx=0/1 pour désactiver/activer la sortie x (x de 9 à 15) pour le second adaptateur **Xany2Misc**

### 5.3.3 Contrôle d'un Module son Beier USM-RC-2 (16 commandes)



Régler l'émetteur **OpenAVRc** pour que l'instance **X-Any** envoie l'état de 16 contacts.

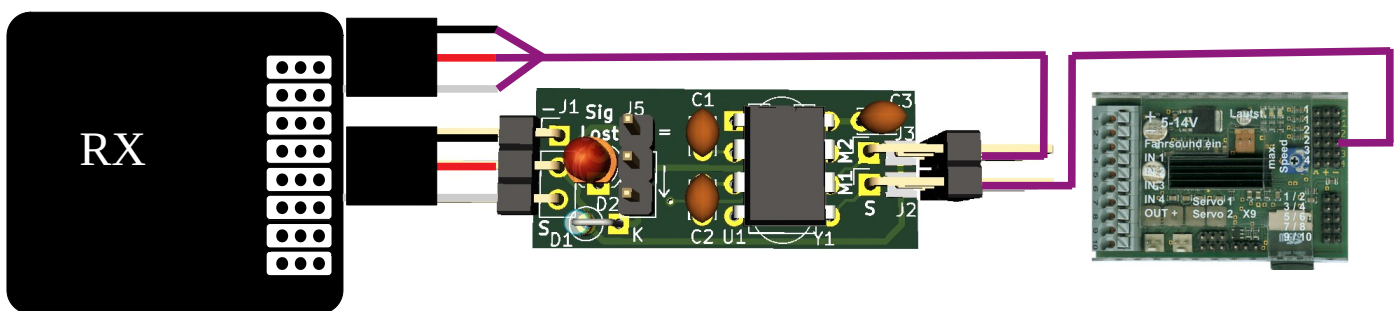
Pour piloter les 16 premières commandes d'un **Module son Beier USM-RC-2** sur la sortie M1, il suffit de paramétrer **Xany2Misc** avec la commande :

M1=B1 ou M2=B1

Cette configuration permet d'envoyer 16 commandes en mode **EKMFA** depuis l'émetteur jusqu'au **Module son Beier USM-RC-2**.

Fx=1 pour activer la fonction x (x de 1 à 16)

### 5.3.4 Contrôle d'un Module son Beier USM-RC-2 (30 commandes)



Régler l'émetteur **OpenAVRc** pour que 2 instances **X-Any** envoient l'état de  $2 \times 16 = 32$  contacts.

- J1 de l'adaptateur **Xany2Misc** sera connecté à la voie correspondant à la première instance **X-Any**
- M2 de l'adaptateur **Xany2Misc** sera connecté à la voie correspondant à la seconde instance **X-Any**

Pour piloter les 30 commandes d'un **Module son Beier USM-RC-2** sur la sortie M1, il suffit de paramétrer **Xany2Misc** avec la commande :

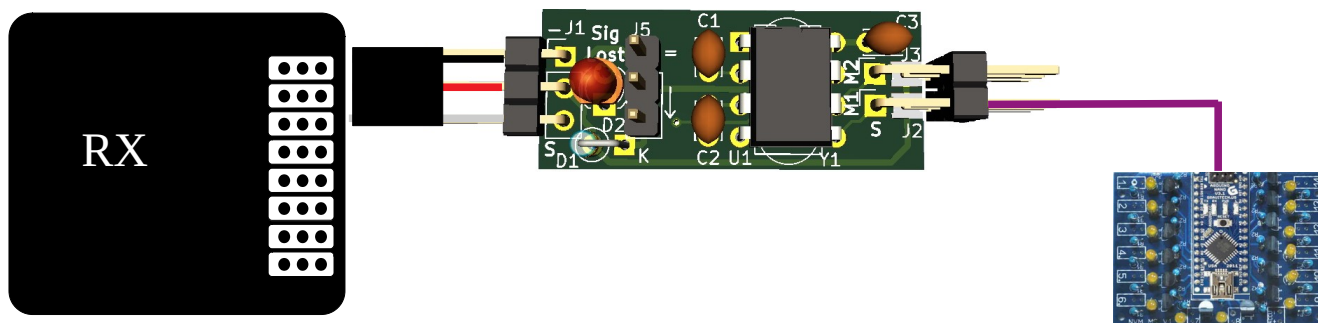
M1=B2 ou M2=B2

Cette configuration permet d'envoyer 32 commandes en mode **EKMFA** depuis l'émetteur jusqu'au **Module son Beier USM-RC-2**.

Seules, les 30 premières commandes sont interprétées et exécutées par le **Module son Beier USM-RC-2**.

Fx=1 pour activer la fonction x (x de 1 à 30)

### 5.3.5 Contrôle du décodeur Multi-Switch NVM (14 commandes)



Régler l'émetteur **OpenAVRc** pour que l'instance **X-Any** envoie l'état de 16 contacts (et pas 8).

Pour commander le décodeur **Multi-Switch NVM** sur les sorties M1, il suffit de paramétrer **Xany2Misc** avec les commandes :

M1=N0 ou M2=N0

Seules, les 14 premières commandes sont interprétées et exécutées par le décodeur **Multi-Switch NVM**.

Fx=1 pour activer la fonction x (x de 1 à 14)