

**ADF4351 driven by an Arduino**  
**RF Generator 34.5 to 4400MHz**  
 By Alain Fort F1CJN

March 7, 2016

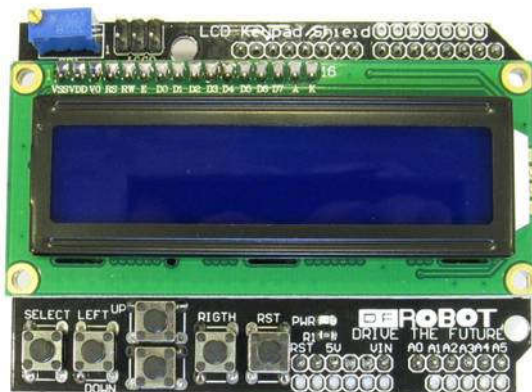
\*\*\*\*\* FRANCAIS \*\*\*\*\*

Ce programme utilise un Arduino Uno muni d'un "LCD keypad shield V1.0" de marque ROBOT, avec boutons permettant de commander une ADF4351 chinoise (40€ sur Ebay) qui génère une fréquence comprise entre 34,5 et 4400MHz au pas minimum de 10KHz.

Une carte équivalente à 34 Euros (mais sans OL Quartz) peut être achetée sur le site de SV1AFN.

Vingt fréquences peuvent être mémorisées dans le mémoire EEPROM de l'Arduino.

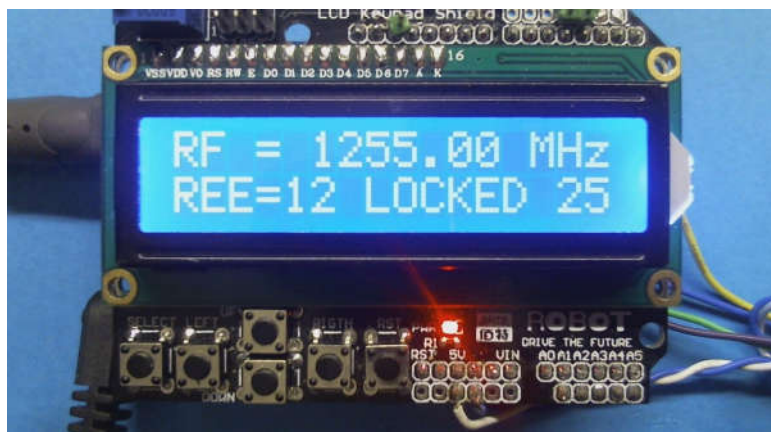
Si une ou plusieurs fréquence sont mémorisées, alors la fréquence en mémoire zéro sera affichée à la mise sous tension



Le curseur se déplace avec les touches LEFT and RIGHT. Les digits placés sous le curseur peuvent être modifiées avec les touches UP and DOWN, ceci pour la fréquence, la mémoire et la fréquence de référence:

- pour la fréquence, il suffit de placer le curseur sur le chiffre à modifier,
- pour la mémoire , il suffit de placer le curseur sur le numéro de mémoire,
- pour la fréquence de référence, il suffit de placer le curseur sur 10 ou 25,
- pour la lecture ou écriture de la fréquence en mémoire, placer le curseur en bas à gauche (passage de REE (lecture EEPROM) à WEE(Ecriture EEPROM)).

Le curseur disparaît après quelques secondes et est ré activé lors de l'appui sur une touche.



#### MEMORISATION

- Pour mémoriser la fréquence en cours, mettre à WEE, puis sélectionner le numéro de mémoire, puis appuyer sur la touche SELECT pendant une seconde. Le mot MEMORISATION apparait alors sur l'écran. Ceci fonctionne quelque soit le placement du curseur excepte sur l'emplacement de la fréquence de référence 10 ou 25.

- Pour mémoriser la fréquence de référence, placer le curseur sur 10 ou 25, puis appuyer pendant 1s sur la touche SELECT.

Note : pour utiliser un 10 MHz externe, la résistance R6 doit être enlevée sur la carte ADF4351.

\*\*\*\*\* HARDWARE IMPORTANT \*\*\*\*\*



Voir le schéma.

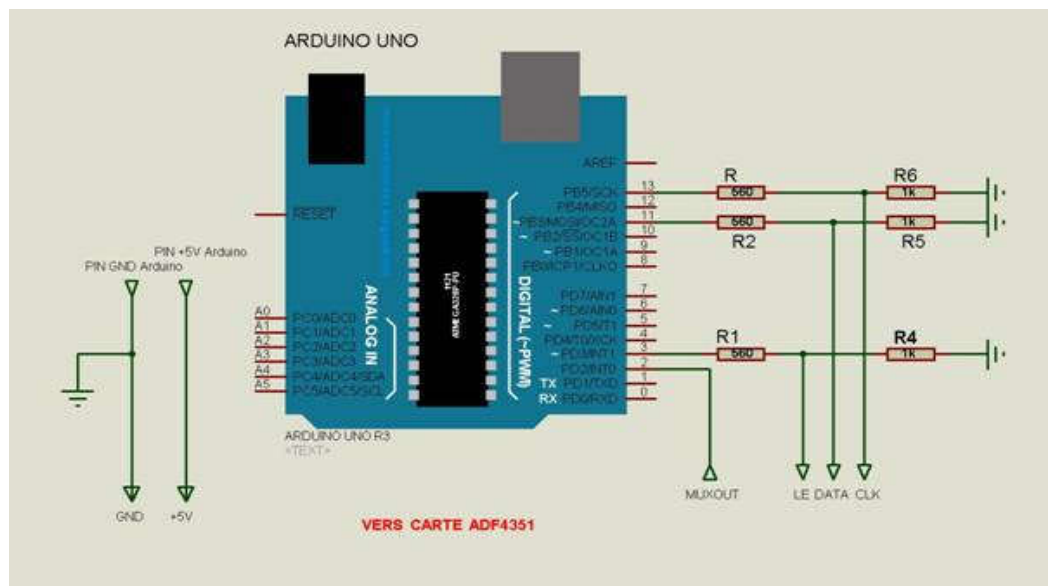
Avec un Arduino UNO : utiliser un pont de résistances pour réduire la tension, MOSI (pin 11) vers ADF DATA, SCK (pin13) vers CLK ADF, Select (PIN 3) vers LE

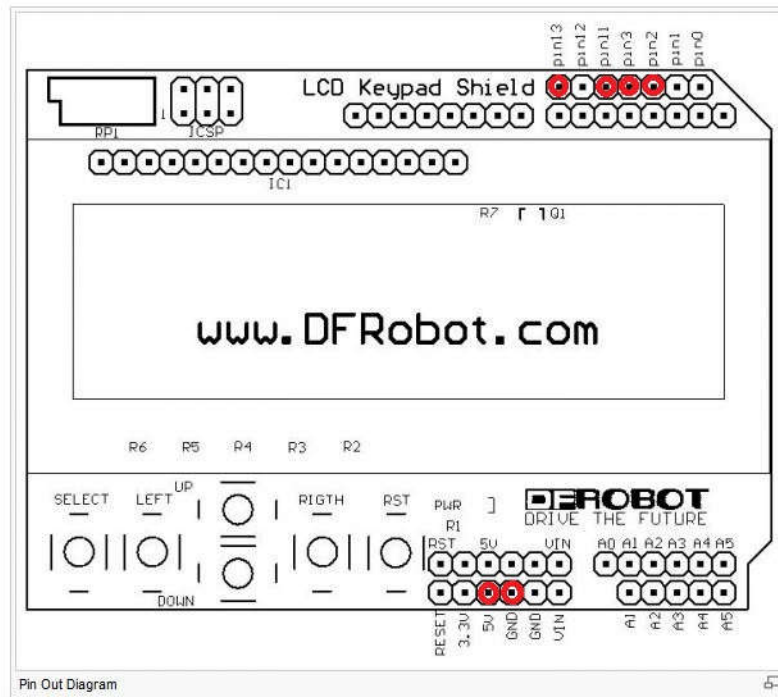
Resistances de 560 Ohm avec 1000 Ohm à la masse sur les pins 11, 13 et 3 de l'Arduino UNO pour que les signaux envoyés DATA, CLK et LE vers l'ADF4351 ne dépassent pas 3,3 Volt.

La Pin 2 de l'Arduino (pour la détection de Lock) est connectée directement à la sortie MUXOUT de la carte ADF4351.

La carte ADF est alimentée en 5V par la carte Arduino (les pins +5V et GND sont proches de la LED Arduino).

**Important : Pour avoir un spectre de sortie "propre", alimenter l'Arduino par la prise alimentation externe et non pas par le cordon USB du PC. L'alimentation externe doit être de 8V à 9V afin de ne pas dépasser la dissipation maximale du régulateur de l'Arduino.**





\*\*\*\*\*SOTFWARE\*\*\*\*\*

Suivre le lien:[ADF4351\\_LCD\\_07032016](#)

Attention: Si vous utiliser un LCD Shield ROBOT version 1.1 au lieu de 1.0, modifier le logiciel (voir le fichier ino).

\*\*\*\*\* ENGLISH \*\*\*\*\*

This sketch uses an Arduino Uno (5€), a standard "LCD buttons shield" from ROBOT (5€), with buttons and an ADF4351 Chinese card found at EBAY (40€). An equivalent card (without Xtal oscillator) can be bought on the SVIAFN site for 37 dollars.

The frequency can be programmed between 34.5 and 4400 MHz with a minimal step of 10KHz.

Twenty frequencies can be memorized into the Arduino EEPROM.

If one or more frequencies are memorized, then at power on, the memory zero is always selected.

The cursor can move with the LEFT and RIGHT buttons. Then the underlined digit can be modified with the UP and DOWN buttons, for the frequency, the memories and the frequency reference (10 or 25 MHz):

- to modify the frequency, move the cursor to the digit to be modified, then use the UP and DOWN buttons,
- to modify the memory number, move the cursor to the number to be modified, then use the UP and DOWN buttons,

- to select the reference frequency, move the cursor on 10 or 25 and select with UP and DOWN.

- to read or write the frequency in memory, place the cursor on the more left/more down position and select REE (for Reading EEprom) or WEE (for Writing EEprom).

The cursor disappears after few seconds and is re-activated if a button is pressed.

#### MEMORIZATION

- For the frequency, select WEE, then select the memory number, then push the SELECT button for a second. The word MEMORISATION appears on the screen. This memorization works then the cursor is anywhere except on the reference 10 or 25 position.

- For the reference frequency, move the cursor to 10 or 25, then press the SELECT for one second.

Note : to use an external 10 MHz signal, R6 must be removed on the ADF4351 PCB.

\*\*\*\*\* HARDWARE IMPORTANT\*\*\*\*\*

See Schematic.

With an Arduino UNO : uses a resistive divider to reduce the voltage, MOSI (pin 11) to

ADF DATA, SCK (pin 13) to ADF CLK, Select (PIN 3) to ADF LE

Resistive divider 560 Ohm with 1000 Ohm to ground on Arduino pins 11, 13 et 3 to adapt from 5V to 3.3V the digital signals DATA, CLK and LE send by the Arduino.

Arduino pin 2 (for lock detection) directly connected to ADF4351 card MUXOUT.

The ADF card is 5V powered by the ARDUINO (PINS +5V and GND are closed to the Arduino LED).

**Important : in order to get a "clean" output spectrum, power the Arduino with an external PSU and not by the USB port. The PSU voltage must be set between 8 or 9V to limit the power dissipation of the Arduino internal regulator.**

\*\*\*\*\* SOFTWARE \*\*\*\*\*

Follow the link : [ADF4351\\_LCD\\_07032016](#)

If you are using a LCD Shield ROBOT version 1.1 instead of 1.0, please read and modify the ino file.

\*\*\*\*\*