arduino, lecteur-son, em

Téléphone BMR

/!\ en cours de rédaction 22 octobre 2019

C'est un superbe téléphone sensuel en forme de bouche qui a servi à diffuser une suite d'entretiens sur le sujet des règles. Il a été utilisé comme borne d'écoute pour la journée/soirée “Bois mes règles” organisée par Gast! à Quimperia / La Baleine en mars 2018.

Le clavier du téléphone permet de déclencher la lecture d'un enregistrement que l'on peut écouter dans le combiné, comme un téléphone quoi!

Le circuit du téléphone est remplacé par un arduino nano qui déclenche la lecture d'un fichier mp3 par un circuit DFPlayer, le haut-parleur utilisé est le haut-parleur original du combiné. Le clavier original du téléphone est aussi réutilisé et l'ensemble est alimenté par un transfo qui récupère le courant du secteur pour en faire un courant 9V continu.

Réutiliser le haut parleur du téléphone

Le combiné est relié au poste par une prise RJ11 (seuls quatre fils sont utilisés) :

- Vert : micro
- Rouge : micro
- Jaune : HP
- Noir : HP

On se servira des fils jaunes et noirs pour relier le haut-parleur au circuit

Réutiliser le clavier du téléphone

Chaque touche active une liaison entre deux broches.
Méthode pour retrouver la matrice :

- numéroter les broches ou les nommer
- sur du papier, dessiner une grille avec autant de colonnes et de rangées que sur le clavier et dessiner le clavier
- en utilisant un multimètre pour tester la continuité, tester toutes les combinaisons de broches possibles, en appuyant sur chaque touche, l'une après l'autre
- noter pour chaque touche les 2 broches utilisées

Connexions du clavier sur nappe : R4, R3, R2, R1, VSS, C1, C2, C3, C4, PT

	C1	C2	C3	C4
R1	1	2	3	
R2	4	5	6	FLASH
R3	7	8	9	
R4	*	0	#	REDIAL

telephone_BMR_test_clavier_002.ino (cliquer pour afficher le code)

telephone_BMR_test_clavier_002.ino

```
/* deuxième essai avec une autre library pour l'afficheur 7 segment
 *  https://github.com/bremme/arduino-tm1637
 *  SevenSegmentTM1637 de Bram Harmsen
 */

// ************************************************
// Définitions pour l'afficheur U7 segments à 4 chiffres
#include "SevenSegmentTM1637.h"
```

```
const byte PIN_CLK = 2;   // define CLK pin (any digital pin)
const byte PIN_DIO = 3;   // define DIO pin (any digital pin)
SevenSegmentTM1637    display(PIN_CLK, PIN_DIO);

// ****************************************************
// Définitions pour le clavier matrice
#include <Keypad.h>

const byte ROWS = 4; //quatre rangées
const byte COLS = 4; //quatre colonnes
char keys[ROWS][COLS] = {
  {'1','2','3','Z'},
  {'4','5','6','F'},
  {'7','8','9','Z'},
  {'*','0','#','R'}
};

/* Correspondances entres les broches identifiées sur le téléphone et les pins de l'arduino
        R1 : 7       C1 :  8
        R2 : 6       C2 :  9
        R3 : 5       C3 : 10
        R4 : 4       C4 : 11           */
byte rowPins[ROWS] = { 7,  6,  5,  4}; // connecter les rangées à ces pins
byte colPins[COLS] = { 8,  9, 10, 11}; // connecter les colonnes à ces pins


Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );

void setup() {

  Serial.begin(9600);
  // Initialiser l'afficheur à 4 chiffres
  display.begin();            // initializes the display
  display.setBacklight(100);  // set the brightness to 100 %
  display.print("INIT");      // display INIT on the display
  delay(1000);                // wait 1000 ms

}

void loop() {

  char key = keypad.getKey();

  if (key){
    display.print(key);
    Serial.println(key);
    delay(300);
  } else {
    display.clear();
  }
}
```

Sources et ressources

Bibliothèque arduino keypad : http://playground.arduino.cc/Code/Keypad
Bibliothèque pour afficheur 4 digits 7 segments de Bram Harmsen : https://github.com/bremme/arduino-tm1637
Utiliser un clavier matriciel : https://playground.arduino.cc/Main/KeypadTutorial

Article extrait de : http://www.lesporteslogiques.net/wiki/ - **WIKI Les Portes Logiques**
Adresse : **http://www.lesporteslogiques.net/wiki/openatelier/projet/telephone_bmr?rev=1583234911**
Article mis à jour: **2020/03/03 12:28**