



Fiche Réalisation  
Mécanisme d'alimentation programmable

<b>Titre : Mécanisme d'alimentation programmable</b>	
<b>Type de réalisation :</b> <i>ensemble électronique</i>	<b>Concepteur :</b> Brice Detailleur <b>Courriel :</b> brice.detailleur@univ-amu.fr
<b>Durée estimée :</b> 1 Jour	<b>Date de la réalisation :</b> 22/02/2017
<b>Fichiers associés :</b> Neant	

### Objectif

Dans cette fiche, nous créons un dispositif de distribution de nourriture, sous forme de poudre pour des cages à souris. Le dispositif est piloté par une interface arduino équipé d'une horloge (RTC) programmable et d'un servo moteur pour permettre la rotation du réservoir contenant la nourriture. Ce dispositif est place au-dessus de la cage à souris.

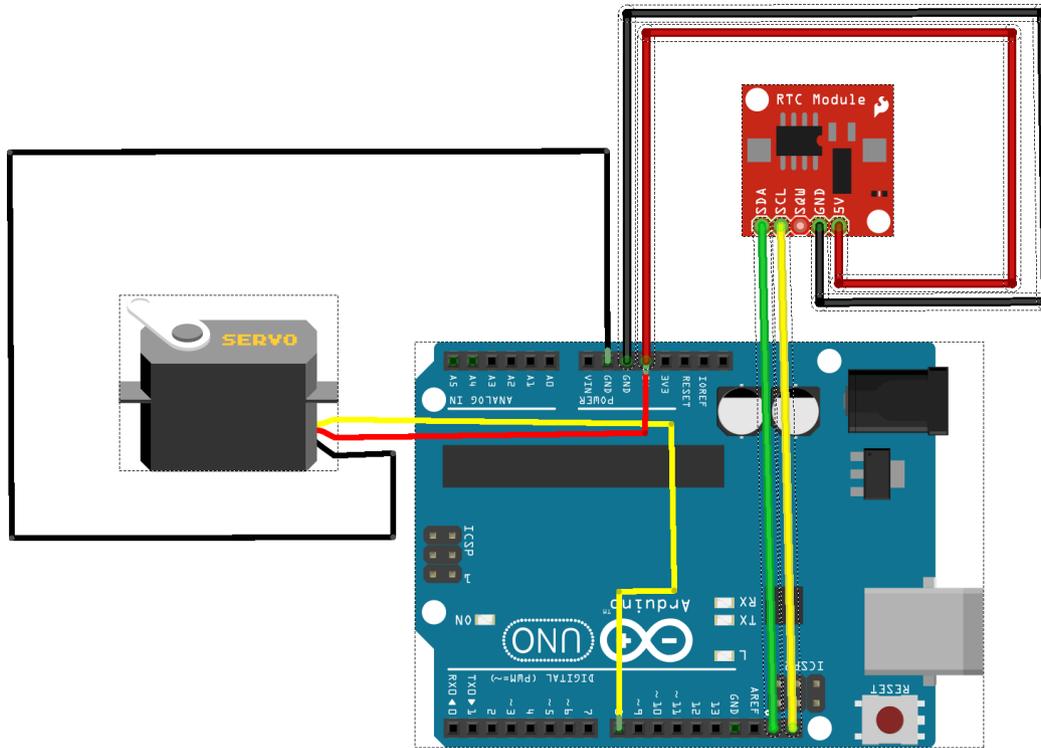
**Matériel** (*Liste/Références/Fournisseurs/Prix unitaires et coût global du matériel nécessaire*)

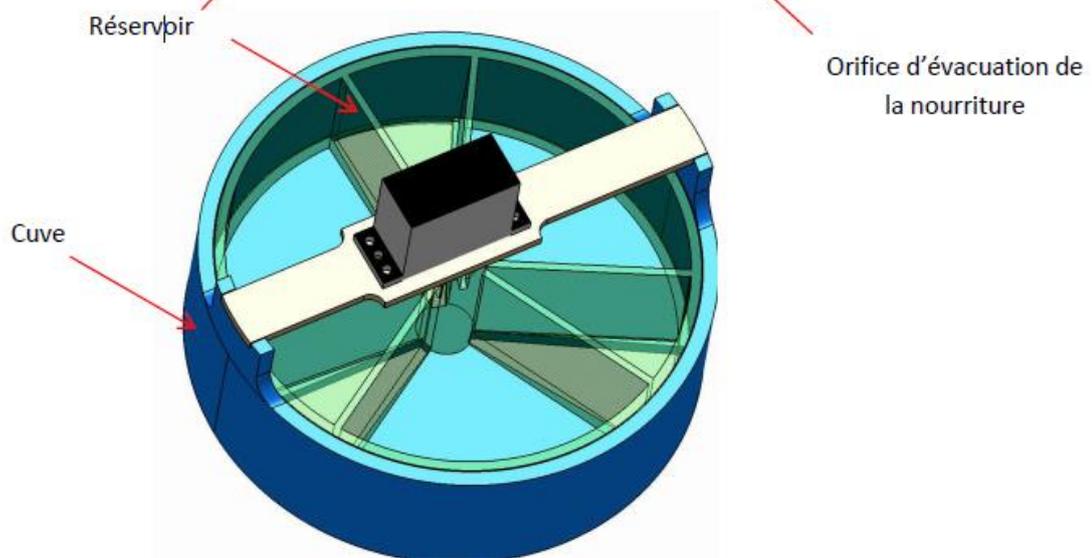
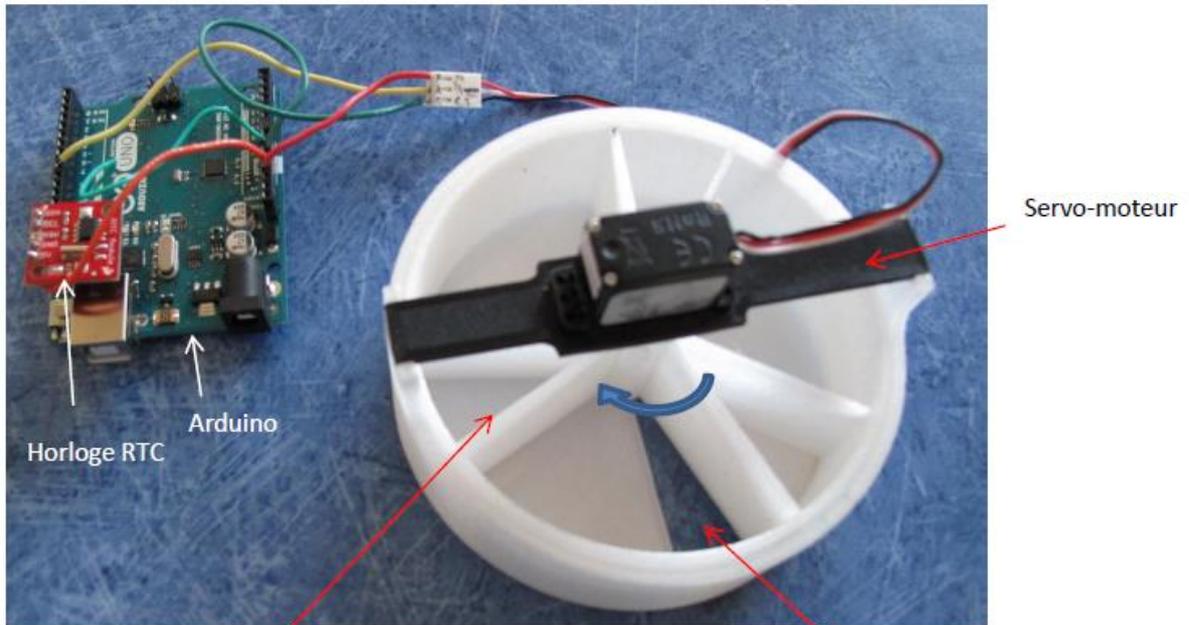
Description	Fournisseur	Référence	Nombre	PU HT
Un microcontroller de type Arduino	RS radioSpare	Ex : UNO REV.3 715-4081	1	20€
Module RTC	Robot Maker		1	1,80€
Servo moteur	Go Tronic	31760	1	4,95 €



## Réalisation

### 1 Schéma fonctionnel





Pour toutes informations, fichiers (CAO et impression 3D),

Contactez : [brice.detailleur@univ-amu.fr](mailto:brice.detailleur@univ-amu.fr) du GT Prototypes, pilotage et périphériques du RTMFM.



## 4 Firmware Arduino

Deux firmaware sont à appliquer. Le premier est à téléverser dans l'arduino pour mettre à l'heure la RTC. Une fois fait il faut installer la pile dans le RTC.

Téléviser ensuite le second firmware pour piloter le dispositif

### Firmware de mise à l'heure du RTC

```
//Mise à l'heure RTC

#include <Wire.h>
#include "RTClib.h"

RTC_DS1307 RTC; //Classe RTC_DS1307

void setup () {
  Serial.begin(57600); //Démarrage de la communication
  Wire.begin(); //Démarrage de la librairie wire.h
  RTC.begin(); //Démarrage de la librairie RTClib.h

  //Si RTC ne fonctionne pas
  if (! RTC.isrunning()) {
    Serial.println("RTC ne fonctionne pas !");

    //Met à l'heure à date à laquelle le sketch est compilé
    RTC.adjust(DateTime(__DATE__, __TIME__));
    //Cela fonctionne également :
    //RTC.adjust(DateTime("Dec 5 2012","12:00:00"));
    //RTC.adjust(DateTime(2014, 1, 21, 3, 0, 0));
  }
}

void loop() {
  //Affichage de l'heure
  DateTime now = RTC.now();
  Serial.print(now.day(), DEC);
  Serial.print('/');
  Serial.print(now.month(), DEC);
  Serial.print('/');
  Serial.print(now.year(), DEC);
  Serial.print(' ');
  Serial.print(now.hour(), DEC);
  Serial.print(':');
  Serial.print(now.minute(), DEC);
```



```
Serial.print(':');  
Serial.print(now.second(), DEC);  
Serial.println();  
delay(3000);  
}
```

### Firmware de pilotage

```
#include <Wire.h>  
#include "RTCLib.h"  
  
int periode=20000; //Temps entre chaque début d'impulsion en microsecondes  
int leServo=8; //leServo est connecté à la pin 8  
  
RTC_DS1307 RTC;  
  
void setup ()  
{  
  Serial.begin(57600); //Démarrage de la communication  
  Wire.begin(); //Démarrage de la librairie wire.h  
  RTC.begin(); //Démarrage de la librairie RTCLib.h  
  
  if (! RTC.isrunning()) //Si RTC ne fonctionne pas  
  {  
    Serial.println("RTC ne fonctionne pas !");  
  }  
  pinMode(leServo,OUTPUT); //On prépare le pin 8 en mode OUTPUT  
  digitalWrite(leServo,LOW); //On l'initialise à l'état bas  
}  
  
void setAngle(int a) //Fonction setAngle pour envoyer les impulsions  
{  
  int duree=map(a,0,179,500,2500); //On transforme l'angle en microsecondes  
  //et on stocke dans la variable duree  
  digitalWrite(leServo,LOW); //On met le pin à l'état bas  
  
  for (int t=0;t<300;t++) //Cette boucle est nécessaire pour laisser  
  //le temps au servo d'aller à sa position  
  {  
    digitalWrite(leServo,HIGH); //On envoie l'impulsion  
    delayMicroseconds(duree); //Pendant la bonne durée  
    digitalWrite(leServo,LOW); //On stoppe l'impulsion  
    delayMicroseconds(periode-duree); //On attend le temps restant  
  }  
}
```



```
        //pour atteindre la période
    }
}

void loop()
{
    //Affichage de l'heure
    DateTime now = RTC.now();
    Serial.print(now.day(), DEC);
    Serial.print('/');
    Serial.print(now.month(), DEC);
    Serial.print('/');
    Serial.print(now.year(), DEC);
    Serial.print(' ');
    Serial.print(now.hour(), DEC);
    Serial.print(':');
    Serial.print(now.minute(), DEC);
    Serial.print(':');
    Serial.print(now.second(), DEC);
    Serial.println();
    delay(3000);

    if (now.dayOfTheWeek() == 4 && now.hour() == 11 && now.minute() == 08)
        //Jour de la semaine : 1 pour lundi, 2 pour mardi, ...
        {
            for (int angle=0;angle<=90;angle+=30)
                //On fait varier l'angle de 0 à 90° par pas de 30°
                {
                    setAngle(angle); //On appelle la fonction setAngle
                }
            setAngle(0); //On fait revenir leServo à le position 0

            delay(60000); //On attend 60 seconds pour éviter un deuxième cycle
        }
}
```