



## Laser multiple-motif

Université François Rabelais  
Institut Universitaire de Technologie de Tours  
Département Génie Électrique et informatique Industrielle



# **Laser multiple-motif**

Projet E&R ISI

Mathieu DUSSOT  
Axel GERME  
Groupe P1

Sofi RODIER  
Thierry LEQUEU  
2011/2013



# Sommaire

Introduction.....	5
1.Cahier des charges.....	6
1.1.Introduction au problème posé.....	6
1.2.Expression fonctionnel du besoin.....	6
1.3.Les solutions proposées pour répondre à ces besoins.....	6
2.Planning prévisionnel et réel.....	7
3.Laser .....	8
3.1.Descriptif.....	8
3.2.Fonctionnalités.....	9
4.Moteur pas à pas .....	10
4.1.Descriptif.....	10
4.2.Fonctionnalités.....	12
5.Programmation .....	13
5.1.Logiciel.....	13
5.2.Ordinogramme.....	14
5.3.Explication .....	15
6.Réalisation .....	16
6.1.Schéma fonctionnel .....	16
6.2.Carte.....	17
6.3.Disque.....	18
6.4.Composants.....	19
Suivi de projet.....	21
Conclusion.....	22
Résumé.....	23
Index des illustrations.....	24
Bibliographie.....	25
Annexe .....	26

## **Introduction**

Nous avons pour ce dernier projet d'étude et réalisation au semestre 4 à choisir un sujet qui porte essentiellement sur de la programmation. Nous avons donc choisis un sujet qui alliait un projet que nous avions vu d'une approche un peu différente précédemment. Nous avons donc pris comme support un laser firefly que nous possédions et on a voulu en faire ressortir quelque chose c'est là que nous est venue l'idée de faire un disque rotatif sur lequel plusieurs embouts seraient disposés afin d'avoir différents motifs.

# 1. Cahier des charges

## 1.1. Introduction au problème posé

Nous avons pour projet de réaliser un disque rotatif pour faire passer différents motifs avec un laser qui sera placé en dessous.

Le moteur devra donc effectuer un huitième de tours pour changer de motif à l'appuie du bouton avant ou arrière.

## 1.2. Expression fonctionnel du besoin

- Quel moteur utiliser ?
- Quel sera le mode de transmission d'information le plus approprié ?
- Quels seront les dimensions adéquates ?
- Comment donner l'ordre d'avancer ou de reculer ?

## 1.3. Les solutions proposées pour répondre à ces besoins

- On va utiliser un moteur pas à pas pour nous permettre de choisir quand arrêter la rotation sans avoir les contraintes de l'entraînement.
- On utilisera donc un ATmega8535 pour transmettre les informations au moteur.
- On aura un cercle de 10 cm de diamètre en plaque pour circuit imprimé.
- On utilisera des boutons poussoirs pour envoyer les informations au moteur.

## 2. Planning prévisionnel et réel

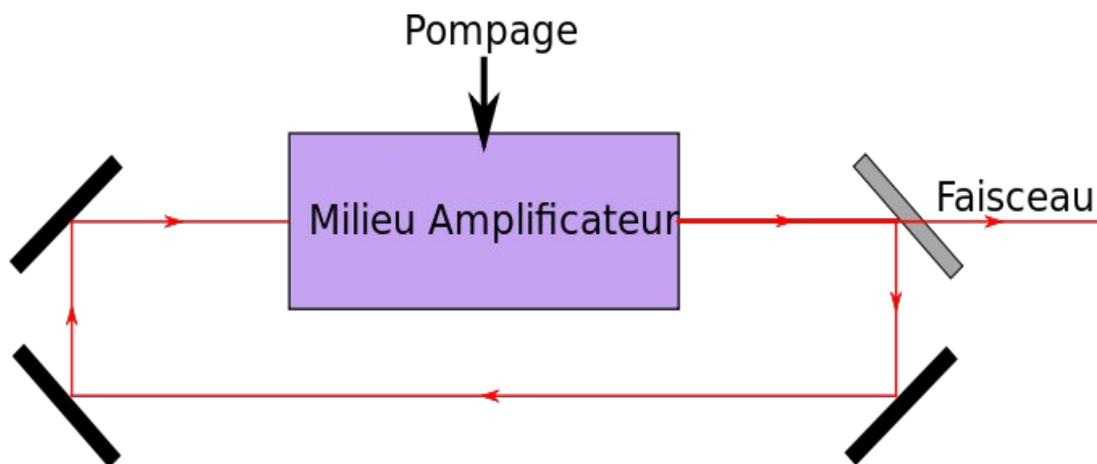
E&R S4	Laser à motif rotatif							Dussot, Germé	
semaines/Taches	6	7	8	9	10	11	12	13	14
Choix du sujet	■								
Étude du moteur pas à pas		■	■		■	■			
Construction mécanique		■	■	■	■				
Programmation					■	■	■	■	■
Test						■	■	■	■
Rédaction du rapport			■	■			■	■	■
Remise du rapport								■	■
Présentation oral									■
Planning prévisionnel	■								
Planning réel	■								

## 3. Laser

### 3.1. Descriptif

Le laser acronyme de « light amplification by simulated emission of radiation » qui en français signifie « amplificateur de lumière par émission simultanée de rayonnement » est un appareil qui permet de produire de la lumière spatialement et temporellement cohérente basé sur l'effet laser.

Un laser est donc, fondamentalement, un amplificateur de lumière dont la sortie est réinjectée à l'entrée. Son l'alimentation en énergie est la source du pompage, la sortie est le rayonnement laser qui est réinjecté à l'entrée par les miroirs de la cavité résonnante, le mécanisme de l'amplification étant l'émission stimulée.



*Illustration 1: Principe de fonctionnement du laser*

On peut classer ces lasers en plusieurs catégories distinctes. Il y a six familles pour les différencier qui sont en fonction de la nature du milieu excité ou bien le fait qu'ils fonctionnent en mode continu ou en mode impulsionnel. [1]

### 3.2. Fonctionnalités

Pour la réalisation de ce projet nous utilisons donc un laser rouge. Le laser choisi s'est imposé nous avons récupéré un jeu de lumière firefly de chez IbizaLight. Nous avons pris le laser rouge et non le laser vert essentiellement pour un choix pratique. La puissance du pointeur laser vert est de 30mW alors que la puissance du pointeur laser rouge est de 100mW.



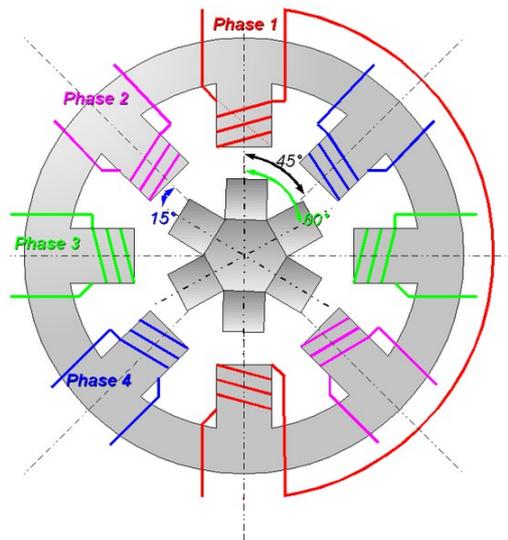
*Illustration 2: Laser firefly Ibiza Light*

Le laser réfléchit la lumière et l'amplifie pour que l'on puisse se servir du faisceau qui passe dans nos motifs. La lumière produite traverse les capsules et peut ensuite être projetée sur un mur ou une surface quelconque. Effet garanti.

## 4. Moteur pas à pas

### 4.1. Descriptif

Le moteur pas à pas est un très simple d'utilisation. Il est basé sur l'électromagnétisme. Il possède plusieurs bobine en opposition et un barreau de fer au milieu du moteur. Quand on alimente l'une des bobine elle devient un électroaimant. Naturellement le barreau de fer va s'aligner avec la bobine suivant son sens de polarisation. Ainsi de suite, on alimente la bobine à coté de la première pour faire tourner le barreau de fer. Pour changer le sens du moteur il suffit de changer l'ordre d'alimentation des bobines. Voici une illustration qui explique le fonctionnement [2]:



*Illustration 3: Fonctionnement d'un moteur pas à pas*

Dans la pratique, le barreau de ferrite a plusieurs dents (ici 6). Dès qu'on alimente la phase 2, il y a une rotation de  $15^\circ$  (i.e.  $60^\circ - 45^\circ = 15^\circ$ ), puis la phase 3, etc. Donc le moteur tourne de  $15^\circ$  dès qu'on alimente une phase. Il faut 24 impulsions pour faire un tour complet. C'est un moteur 24 pas.

Ce type de moteur possède quelques inconvénients et avantages :

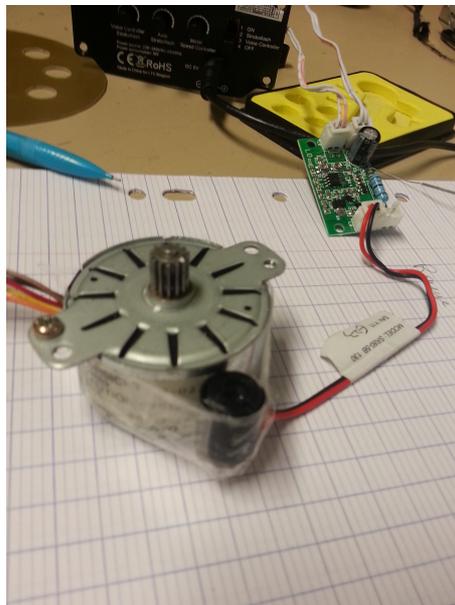
Inconvénient:

- nécessite au moins 3 bobinages pour avoir un cycle complet
- hors tension, le rotor est libre donc cela peut être problématique
- fabrication assez délicate et entrefers doivent être faible

Avantages:

- peu coûteux
- bonne précision
- avec 4 enroulements on obtient 24 pas (modèle utilisé)
- le sens du courant n'a aucune importance

Dans quelques applications, un moteur pas à pas est un moteur est surtout utilisé dans les systèmes de boucle ouverte pour un contrôle de vitesse et de positionnement.

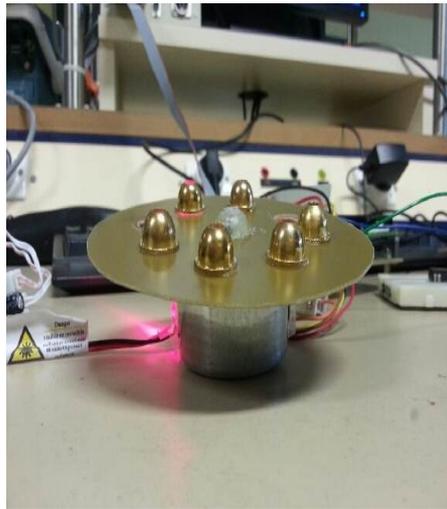


*Illustration 4: Moteur pas à pas  
Sanyo*

## 4.2. Fonctionnalités

Dans notre application, le moteur pas à pas est utilisé pour faire tourner un plateau composé de huit lentilles. Ces huit lentilles serviront à créer un motif avec le laser. Les lentilles sont disposées de façon à ce que le laser tombe bien en face en fonction du nombre de pas.

Le disque comporte donc 8 embouts de laser, on possède un moteur de 24 pas donc tous le 3 pas on trouve un nouveau motif.

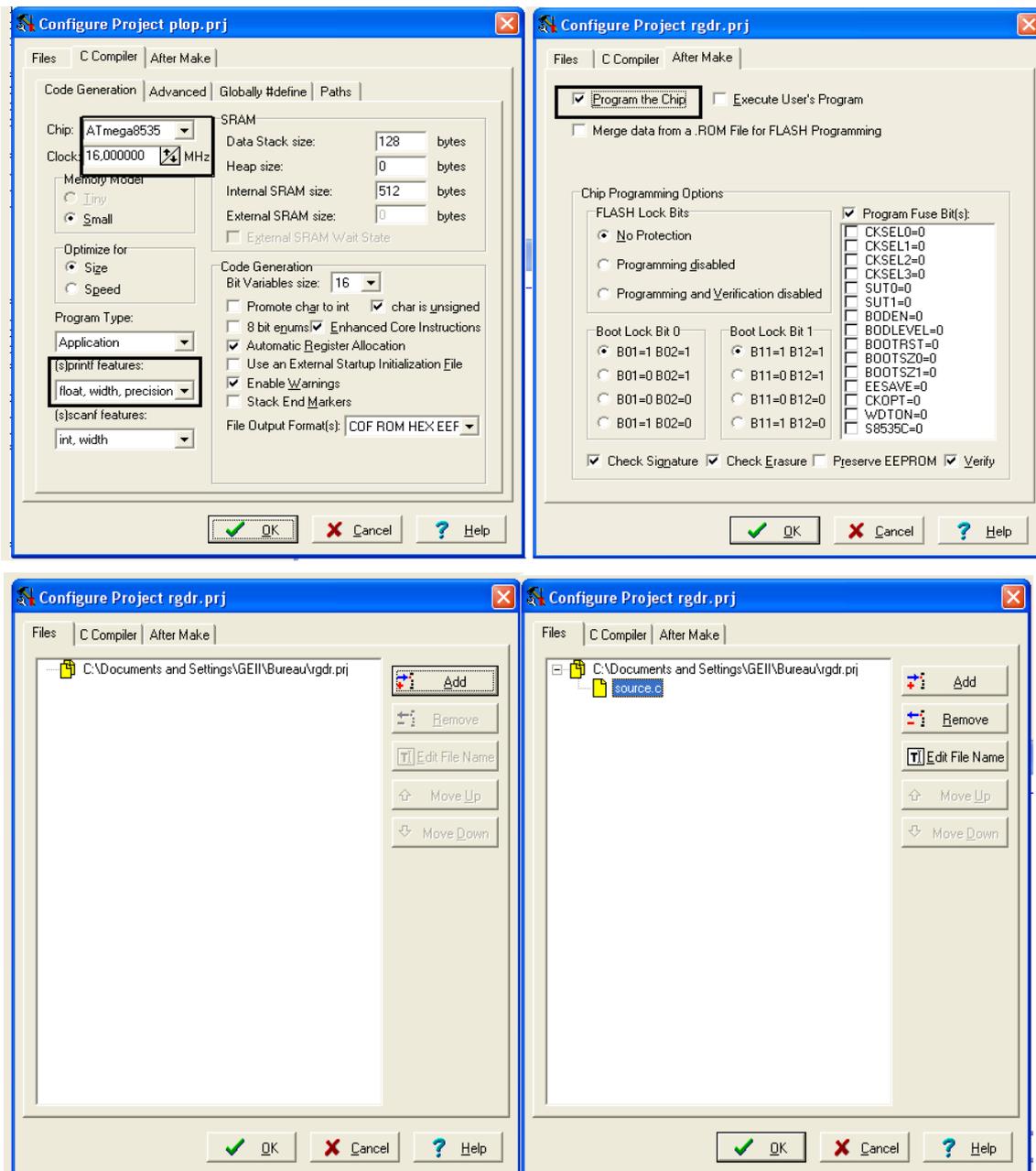


*Illustration 5: moteur pas à pas avec disque*

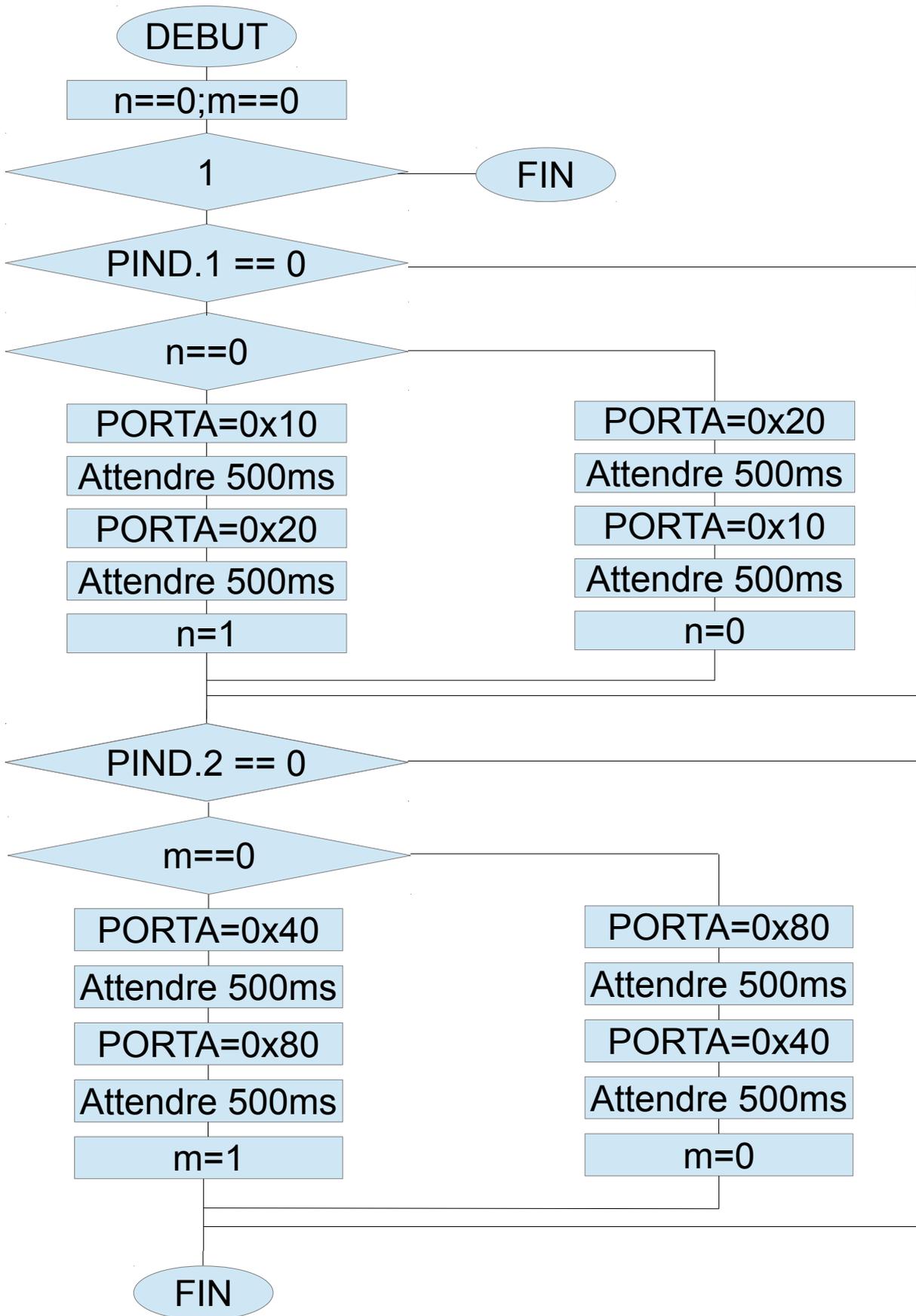
## 5. Programmation

### 5.1. Logiciel

Pour programmer notre ATmega8535, nous avons utilisé le logiciel CodeVisionAVR. Ce logiciel utilise la programmation en C. Tout d'abord, il faut configurer le logiciel pour qu'il soit compatible avec notre ATmega8535. Voici les étapes qui permet cela :



## 5.2. Ordinogramme



### **5.3. Explication**

On peut voir dans l'ordinogramme la configuration du PORT A. La valeur hexadécimale correspond à la valeur de l'octet sur le port.

Exemple :

PORTA=0x20 correspond à la valeur binaire 0010 0000.

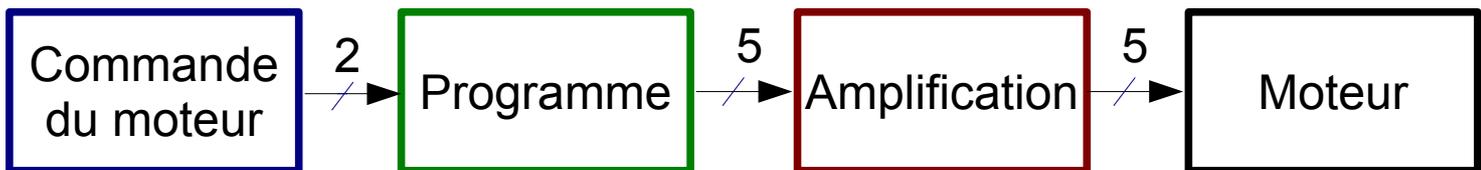
Chaque bit correspond à une des sortie de l'ATMega8535.

L'étape PIND.1==0 correspond à l'appui sur le bouton. Elle est égale à zéro parce que le montage est constitué d'une résistance de tirage qui amène le 5V quand on n'appuie pas sur le bouton.

Pour les variables m et n, on les initialise à zéro. Ensuite on change la valeur une fois que l'on a fait un décalage. Car le moteur est de vingt-quatre pas donc il y a trois pas d'écart entre chaque embout. Comme c'est un nombre impair, quand on ré-appuiera sur le bouton l'ancien fil sera mémorisé puisqu'il faut une alternance pour pouvoir tourner le moteur.

## 6. Réalisation

### 6.1. Schéma fonctionnel



#### **Commande du moteur**

On pilotera le moteur à l'aide des boutons poussoirs présent sur la carte de programmation de M.LEQUEU.

#### **Programme**

Le programme sera géré par un ATmega8535. Il commandera le moteur à l'aide des sorties présente sur l'un des ports que l'on activera suivant le sens du moteur que l'on veut lui appliqué.

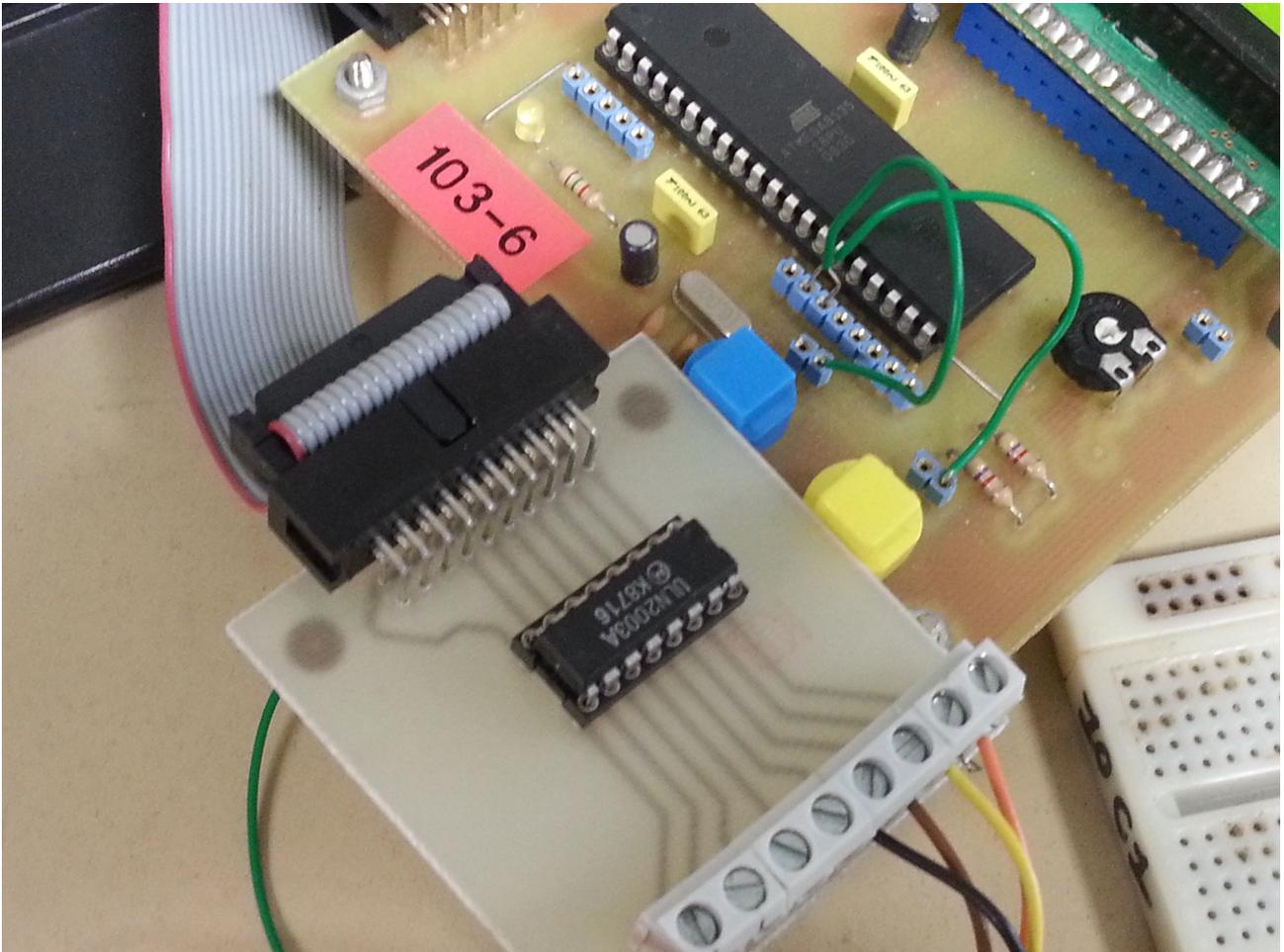
#### **Amplification**

L'amplification est très importante dans notre cas puisque l'ATmega8535 ne délivre pas assez de puissance pour permettre au moteur de tourner.

#### **Moteur**

Le moteur est donc un moteur pas à pas qui permettra d'avoir une grande précision pour aligner le laser aux lentilles. Et il est très facile à commander avec un ATmega8535.

## 6.2. Carte

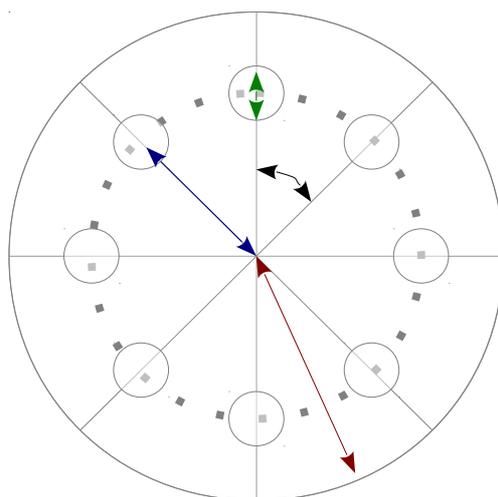


*Illustration 6: Carte d'amplification pour commande moteur pas à pas*

Nous avons réalisé une carte sur laquelle nous avons disposé un ULN2003A ainsi qu'un connecteur pour permettre la connexion avec la carte test comprenant l'AtMega8535 qui gère notre application. Il y a aussi des borniers qui permettent l'alimentation du moteur pour le faire tourner dans un sens ou dans l'autre. Cette carte est très simple du fait que l'on avait besoin de très peu de composants pour la réalisation de cette connexion entre la carte test et le moteur pas à pas.

### 6.3. Disque

Nous avons donc réalisé un disque pour le support de nos embouts. Celui-ci est accroché au moteur et a été spécialement conçu pour un moteur 24 pas. Nous avons donc opté pour un disque que nous avons découpé dans une plaque d'époxy cuivré. On a décidé de ne faire que 8 trous permettant de placer des embouts, car sinon cela aurait été trop serré. En vue du moteur 24 pas on a dû faire en sorte que lorsque l'on appuie sur un bouton le disque effectue  $45^\circ$  afin de passer d'un motif à l'autre.



Dimensions:

↔ : 50 mm.

↔ : 30 mm.

↔ : 10 mm.

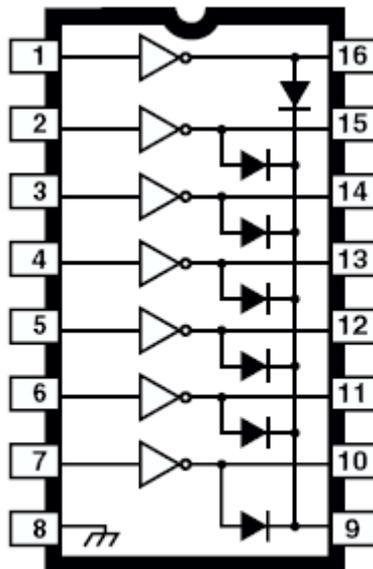
↔ :  $45^\circ$



*Illustration 7: disque en époxy cuivré*

## 6.4. Composants

### 6.4.1. ULN2003A



Le composant ULN2003A est un composant qui permet d'amplifier le signal en sortie de l'ATMega8535 pour la commande du moteur. En effet, la sortie a un faible courant et une tension de 500 mV. C'est pourquoi on utilise ce composant pour augmenter la puissance du signal ce qui permettra une bonne alimentation des bobinages.

Ce composé est fait de 8 transistors qui sont disposé de manière à faire une amplification le signal reçu.

## 6.4.2. AT MEGA 8535

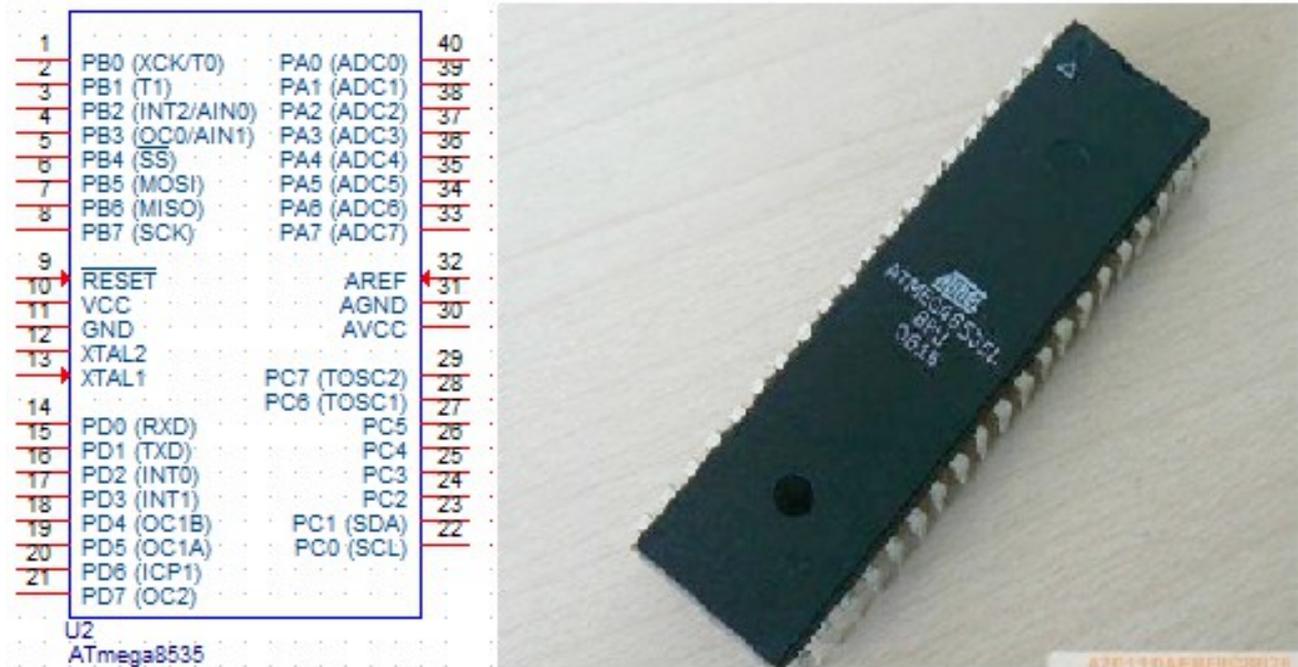


Illustration 8: AtMega8535

Nous nous servons de l'ATMega8535 pour commander le moteur. Nous utiliserons le PORTA pour les différents fils qui piloteront le moteur donc le port sera configuré en sortie. Alors que le PORTD pour l'utilisation des boutons poussoirs sera configuré en entrée. L'ATMega8535 sera présent sur la carte de programmation de M.LEQUEU. Il sera programmé avec le logiciel CodeVisionAVR.[3]

# Suivi de projet

## ***1<sup>er</sup> partie: Élaboration de la partie mécanique***

Nous avons commencé par étudié la bonne dimension du disque pour que celui-ci puisse accueillir les embouts et tenir sur le moteur pas à pas. Le nombre d'embouts devait aussi correspondre à un dénominateur commun de 24 pour que lorsque le moteur fasse un pas il se retrouve en face d'un embout et non à coté. On voulait néanmoins que le dispositif reste d'une taille raisonnable pour être transportable. Nous avons aussi posé le laser sur le moteur pour la réalisation. Nous avons eu des difficultés pour déterminer le commun du moteur pas à pas n'ayant aucune documentation sur celui-ci.

## ***2e partie: Élaboration de la carte***

Nous avons réalisé une carte qui nous permet de faire une connexion entre la carte test et le moteur pas à pas. Sur cette carte nous avons disposés un bornier pour l'alimentation du moteur pas à pas, un connecteur pour la liaison avec la carte test et un ULN2003A qui gère la transmission d'information. Nous avons utilisé la carte test pour ne pas perdre de temps à réaliser notre propre carte avec l'ATmega8535. La réalisation de la carte a pris peu de temps et n'a suscité aucune difficulté.

## ***3e partie: Programmation***

Pour la programmation nous avons modifiés les programmes de test qui étaient fourni avec l'AtMega8535. Nous avons eu quelques difficultés du fait que pour que le moteur avance de plusieurs pas il faut alimenté deux fils ceci est dû au fait que le moteur est un ensemble de 4 bobinages qui permettent de faire bouger la barre ferrite , soit le noir et le marrons pour un sens soit l'orange et le jaune pour un autre sens , une fois sur deux. Comme nous avons utilisé 8 capsules ,entre chacune d'elles il y a 3 pas et donc nous devons faire une alternance et mémorisé le dernier fil sur lequel une information a été envoyé.

## **Conclusion**

Nous avons donc pu terminer sur un projet aboutit qui esthétiquement n'est pas encore à son aspect finale mais qui marche correctement et comme nous le voulons au niveau de la programmation. Nous avons passé un peu de temps sur la partie mécanique mais nous avons su garder le temps nécessaire pour programmer, tester et résoudre les quelques soucis que nous avons pu avoir. Nous pouvons donc retirer de cette dernière expérience d'étude et réalisation que le choix du sujet pour un temps de réalisation court est très important, que l'organisation est primordiale en prenant en compte la possibilité de ne pas réussir du premier coup. Le travail d'équipe a très bien fonctionné et nous a permis de nous découvrir l'un et l'autre.

## Résumé

Lors de ce projet nous avons conçus une carte électronique pour faire la liaison entre la carte test et le moteur pas à pas afin d'avoir une amplification du signal. Nous avons aussi réalisé un disque qui est une partie beaucoup mécanique et qui était moins inintéressante au niveau de ce projet d'étude et réalisation accès sur la programmation. Enfin nous avons réalisé un programme qui gère le fait que le moteur avance et recule en fonction de l'appuie sur un bouton poussoir qui est sur la carte test.

## Index des illustrations

Illustration 1: Principe de fonctionnement du laser.....	7
Illustration 2: Laser firefly Ibiza Light.....	8
Illustration 3: Fonctionnement d'un moteur pas à pas .....	9
Illustration 4: Moteur pas à pas Sanyo.....	10
Illustration 5: moteur pas à pas avec disque .....	11
Illustration 6: Carte d'amplification pour commande moteur pas à pas .....	16
Illustration 7: disque en époxy cuivré.....	17
Illustration 8: AtMega8535.....	19

## Bibliographie

- [1] **wikipedia** . *laser* , 2013, [En ligne]. (Page consultée le ) <<http://fr.wikipedia.org/wiki/Laser>>
- [2] **wikipedia**. *moteur pas a pas*, 2013, [En ligne]. (Page consultée le ) <[https://fr.wikipedia.org/wiki/Moteur\\_pas\\_%C3%A0\\_pas](https://fr.wikipedia.org/wiki/Moteur_pas_%C3%A0_pas)>
- [3] **thierry Lequeu** . , 2013, [En ligne]. (Page consultée le ) <<http://www.thierry-lequeu.fr/>>

## Annexe

```
#include <mega8535.h>

#include <stdio.h>

#include <delay.h>

// Alphanumeric LCD Module functions

#asm

    .equ __lcd_port=0x15 ;PORTC

#endasm

#include <lcd.h>

// Declare your global variables here

unsigned char a=0x55;

unsigned char tampon[20];

void main(void)

{

// Declare your local variables here

    int n;

    int m;

    n=0;

    m=0;

// Input/Output Ports initialization

// Port A initialization

// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out Func0=Out

// State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0

PORTA=0x00;

DDRA=0xFF;

// Port B initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
```

```

// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTB=0x00;
DDRB=0x00;
// Port C initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTC=0x00;
DDRC=0x00;
// Port D initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTD=0x00;
DDRD=0x00;
// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
// Mode: Normal top=0xFF
// OC0 output: Disconnected
TCCR0=0x00;
TCNT0=0x00;
OCR0=0x00;
// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer1 Stopped
// Mode: Normal top=0xFFFF
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off

```

```
// Input Capture on Falling Edge
// Timer1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x00;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;
// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: Timer2 Stopped
// Mode: Normal top=0xFF
// OC2 output: Disconnected
ASSR=0x00;
TCCR2=0x00;
TCNT2=0x00;
OCR2=0x00;
// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
// INT2: Off
```

```
MCUCR=0x00;
MCUCSR=0x00;
// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x00;
// USART initialization
// USART disabled
UCSRB=0x00;
// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;
// ADC initialization
// ADC disabled
ADCSRA=0x00;
// SPI initialization
// SPI disabled
SPCR=0x00;
// TWI initialization
// TWI disabled
TWCR=0x00;
// LCD module initialization
lcd_init(16);
lcd_init(16);
lcd_gotoxy(0,0);
lcd_putsf("laser multimotif");
lcd_gotoxy(0,1);
lcd_putsf("Dussot Germe");
```

```
while (1)
{
    if(PIND.1 == 0)
    {
        if(n==0)
        {
            PORTA=0x10;
            delay_ms(500);
            PORTA=0x20;
            delay_ms(500);
            PORTA=0x10;
            delay_ms(500);
            PORTA=0x20;
            delay_ms(500);
            n=1;
        }
    }
    else
    {
        PORTA=0x20;
        delay_ms(500);
        PORTA=0x10;
        delay_ms(500);
        PORTA=0x20;
        delay_ms(500);
        PORTA=0x10;
        delay_ms(500);
        n=0;
    }
}
```

```
}  
if(PIND.2 == 0)  
{  
  if(m==0)  
    {  
      PORTA=0x40;  
      delay_ms(500);  
      PORTA=0x80;  
      delay_ms(500);  
      PORTA=0x40;  
      delay_ms(500);  
      PORTA=0x80;  
      delay_ms(500);  
      m=1;  
    }  
  else  
    {  
      PORTA=0x80;  
      delay_ms(500);  
      PORTA=0x40;  
      delay_ms(500);  
      PORTA=0x80;  
      delay_ms(500);  
      PORTA=0x40;  
      delay_ms(500);  
      m=0;  
    }  
}
```

```

//PORTA=0x00;
if(PIND.1 == 0)
{
sprintf(tampon,"sens horaire",a);    //pour lire le port A PORTA=%3d et PORTA=0x%x
lcd_gotoxy(0,2);
lcd_puts(tampon);                    //on affiche la valeur de a.
}
if(PIND.2 == 0)
{
sprintf(tampon,"sens antihoraire",a);
lcd_gotoxy(0,3);
lcd_puts(tampon);                    //on affiche la valeur de a.
}
delay_ms(100);
}
}

```