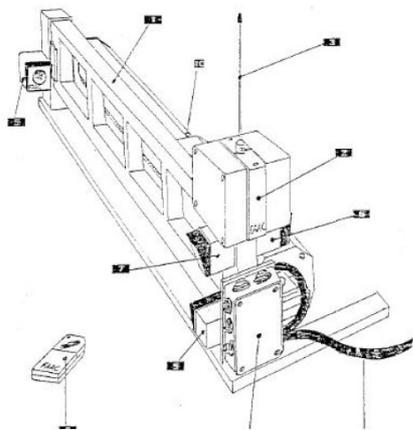


# ALGORIGRAMMES ET ALGORITHMES

## 1- Ecriture d'algorithmes à partir d'un cahier des charges



Ouvre portail FAAC

1-1 : **Réaliser** tout d'abord l'algorithmique du fonctionnement du portail simplifié : Pour cela on ignorera le faisceau lumineux chargé de détecter les passages ou les obstacles.

En conséquence le fonctionnement se réduit alors aux éléments suivants :

- Au départ le portail est fermé, en attente d'une commande (clé ou télécommande ou code).
- Lors d'une commande, le portail s'ouvre pendant 9s, puis reste ouvert (« pause ») pendant 5s, puis se ferme pendant 9s. Au cours de cette séquence, la commande est inopérante.
- Il faudra commander deux voyants pour visualiser les phases d'ouverture et de fermeture.

1-2 : **Réaliser** maintenant l'algorithmique le plus proche possible du fonctionnement complet (avec prise en compte du faisceau). Au fonctionnement précédemment décrit, s'ajoute les fonctionnalités suivantes :

- La coupure du faisceau pendant la pause, prolonge celle-ci : toute coupure du faisceau relance la temporisation de 5s.
- La coupure du faisceau ou une commande, au cours de la fermeture, provoque une réouverture immédiate d'une durée équivalente au temps de fermeture écoulé.

## 2- Ecritures d'algorithmes à partir d'algorithmes

### 2-1 : **Faire clignoter** les Leds du PORTB

```
#include <90s8535.h>
unsigned int i;

void main(void)
{
    DDRB=0xff;
    PORTB=0xff;

    while(1)
    {
        PORTB=0x00;
        for (i=0;i<=60000;i++)
        {
            ;
        }
        PORTB=0xff;
        for (i=0;i<=60000;i++)
        {
            ;
        }
    }
}
```

2-2 : Même exercice que précédemment mais en utilisant un **sous-programme tempo**.

```
#include <90s8535.h>
unsigned int i;

void tempo(void)
{
    for (i=0;i<=60000;i++)
    {
        ;
    }
}

void main(void)
{
    DDRB=0xff;
    PORTB=0xff;
    while(1)
    {
        PORTB=0x00;
        tempo();
        PORTB=0xff;
        tempo();
    }
}
```

2-3 : Réalisation d'un **chenillard** en utilisant un tableau

```
#include <90S8535.h>
unsigned char
table[8]={0xfe,0xfd,0xfb,0xf7,0xef,0xdf,0xbf,0x7f};
unsigned char i=0;
unsigned int j=0;

void tempo(unsigned int temps)
{
    for (j=0;j<=temps;j++)
    {
        ;
    }
}

void main(void)
{
    DDRB=0xff;
    while (1)
    {
        for (i=0;i<8;i++)
        {
            PORTB=table[i];
            tempo(40000);
        }
    }
}
```