

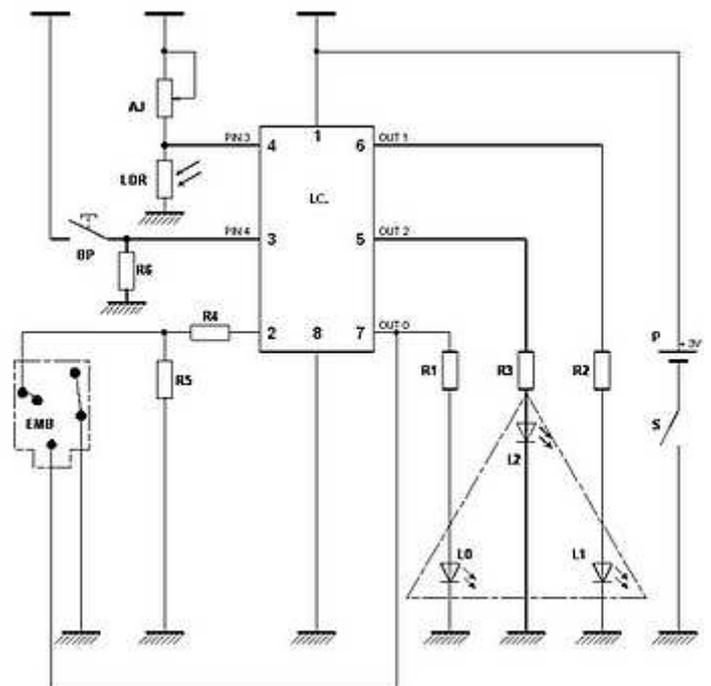
Unité de travail de programmation sur FLASHProg

(les élèves disposent d'un boîtier de programmation par îlot)

A – PRESENTATION DU PROJET

Description du produit
Utilisation pédagogique

A partir du dossier technique de chez A4 TECHNOLOGIE



Description du produit

FlashProg est un triangle de sécurité portatif programmable. Il est destiné à signaler la présence d'un piéton la nuit. Le module électronique fait flasher 3 DEL haute luminosité visibles à plus de 100 mètres. Il dispose de :

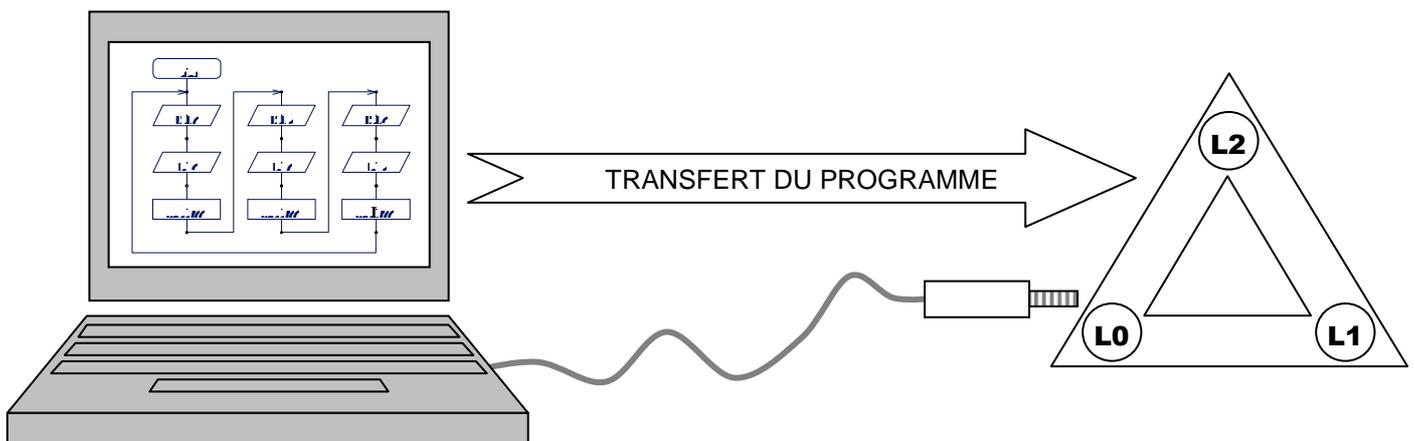
- Trois sorties effecteurs indépendantes constitués par trois DEL.
- Deux entrées constituées par un capteur photosensible (LDR) et un bouton poussoir

La programmation du module permet de créer une grande variété de séquences d'allumage des DELs. Le module accueille un bouton poussoir et une LDR qui permettent de sélectionner différentes séquences d'allumage des DELS en fonction d'un évènement extérieur (appui sur le bouton poussoir, seuil de luminosité ambiante).

La variété du jeu d'instructions disponibles pour la programmation permet de créer des séquences d'allumage des DEL des plus simples aux plus complexes comme par exemple :

- le clignotement simultané des DEL,
- le clignotement alterné des DEL (effet « chenillard »),
- le chaînage de séquences différentes,
- des tests conditionnels sur des actions extérieures (réaction au niveau lumineux ambiant, réaction à l'appui sur le bouton poussoir),
- le comptage d'un nombre de cycles (pour répéter l'exécution d'une séquence plusieurs fois),
- l'exécution d'un programme principal qui fait appel à des sous programmes
- ...

La programmation du module FlashProg est faite à l'aide du logiciel Programming Editor qui permet de créer des diagrammes de programmation qui seront transférés dans le module en connectant l'ordinateur au FlashProg à l'aide d'un câble approprié. Le module reçoit le programme dans sa mémoire (Flash Prom) et est reprogrammable à volonté.



INTRODUCTION

FlashProg est un petit système automatisé dont la simplicité de fonctionnement permet une approche à la fois ludique et sereine de l'automatisme. Ce chapitre est organisé en trois rubriques de difficulté croissante.

- Automatisme niveau 1 : l'élève dont le produit fonctionne va choisir un programme dans une liste et le charger dans son module. Les programmes proposés sont classés par ordre de difficulté croissante.
- Automatisme niveau 2 : l'élève va modifier quelques paramètres d'un programme existant et le tester sur son module.
- Automatisme niveau 3 : l'élève va créer un programme simple et le tester sur son module.

Notions de base

Qu'est ce qu'un microcontrôleur

Un microcontrôleur est un circuit intégré qui contient :

- de la mémoire de programme (qui contient elle-même un programme),
- un microprocesseur (qui exécute les instructions du programme),
- des entrées pour connecter des capteurs (bouton poussoir, LDR...)
- des sorties pour connecter des effecteurs (par exemple des DEL).

Qu'est ce qu'un programme

Un programme est constitué d'une série d'instructions qui sont interprétées et exécutées les unes après les autres par le microcontrôleur. Pour fonctionner, le programme doit être écrit dans un langage compris par le microcontrôleur. Ce langage obéit à des règles strictes de syntaxe propres au microcontrôleur.

L'exécution du programme est cadencée par une horloge interne au microcontrôleur. On peut retenir comme ordre de grandeur que le microcontrôleur utilisé dans le module FlashProg peut exécuter un million d'instructions de base par seconde.

Le langage de base du microcontrôleur est composé d'instructions rudimentaires (langage machine ou assembleur). Afin de rendre la programmation plus conviviale, l'environnement «Programming Editor» permet de définir un diagramme de programmation. Ce diagramme est dans un premier temps converti en un langage évolué (basic), puis dans un deuxième temps en langage machine qui est chargé dans le microcontrôleur.

Environnement de programmation « Programming Editor »

Dans un souci de clarté et de prise en main rapide du système, nous limitons volontairement les explications qui concernent l'environnement de programmation aux éléments utiles à la programmation du FlashProg.

Le logiciel «Programming Editor» permet d'écrire le programme qui va piloter le module FlashProg.

- Le mode Diagramme permet de créer des programmes à l'aide de blocs
- Le contenu d'un diagramme est converti en programme basic.
- Le programme en basic est transféré dans le module pour être exécuté dès la mise sous tension du module électronique.

Mise en oeuvre du logiciel de programmation « Programming Editor »

Pour procéder à la programmation du module du FlashProg, vous devez disposer du CDROM «Programming Editor (Réf. : CD-FPPE), d'un ordinateur (Windows 95, 8Mb RAM mini) et du câble spécifique (Réf. : CABLE-FP) permettant de transférer vos programmes dans le module FlashProg. L'installation du logiciel est décrite en Annexe A de ce dossier.

AUTOMATISME NIVEAU 1 (en suggestion d'utilisation en classe)

Choisir un programme dans une liste et le charger

L'élève charge un programme existant et vérifie que le FlashProg se comporte conformément à sa description. Chaque programme est accessible à partir du dossier « Automatismes niveau 1 » du CDROM CD-FP.

Sommaire des fiches élèves :

Page		Description	Fichier programme
19	FICHE N°1	Pilotage des sorties : Clignotement simultané des trois DEL.	F1-CLIGNO DEL.cad
20	FICHE N°2	Pilotage des sorties : Clignotement alterné des trois DEL (effet chenillard).	F2-CHENILLARD.cad
21	FICHE N°3	Pilotage des sorties : Bâton tournant. Les DEL s'allument deux à deux en tournant.	F3-BATON TOURNANT.cad
22	FICHE N°4	Enchaînement de séquences : Alternance du clignotement simultané des trois DEL et d'un effet chenillard.	F4-CLIGNO-CHENILLARD.cad
23	FICHE N°5	Utilisation du capteur photosensible : Mise en route automatique à la tombée de la nuit.	F5-DEMAR AUTO NUIT.cad
24	FICHE N°6	Utilisation du capteur photosensible : Détection d'un phare de voiture dans la nuit et changement automatique de mode de clignotement.	F6-DETEC PHARE.cad
25	FICHE N°7	Utilisation du capteur Bouton Poussoir : Sélection alternée de 2 effets par le bouton poussoir.	F7-SELECTION PAR BP.cad
26	FICHE N°8	Utilisation du capteur photosensible et Bouton Poussoir : Démarrage automatique dès qu'il fait nuit ou dès que l'on active le bouton poussoir.	F8-DEMAR AUTO LDR-BP.cad
27 28	FICHE N°9	Comptage de séquences : Répétition de deux séquences différentes.	F9-REPET 2 SEQUENCES.cad
29 30	FICHE N°10	Utilisation d'un temps de pause variable : Effet d'accélération d'un chenillard suivi par un clignotement.	F10-ACCELERATION.cad
31	FICHE N°11	Utilisation d'un temps d'attente long : Exécution d'une séquence chenillard une fois toutes les 3 secondes le jour et en permanence la nuit.	F11-CHENILLARD JOUR-NUIT.cad
32	FICHE N°12	Utilisation d'une variable aléatoire : Clignotement aléatoire.	F12-HASARD.cad
33 34	FICHE N°13	Utilisation d'un sous programme : Accélération d'un chenillard commandée par le bouton poussoir.	F13-ACCELERATION PAR BP.cad

Travail élève

- 1** → Consulter la description des programmes dans les fiches suivantes et choisir celui que tu souhaites essayer.
- 2** → Lancer le logiciel Programming Editor et ouvre ton programme (chaque programme est accessible à partir du dossier « Automatismes niveau 1 » du CDROM CD-FP).
- 3** → Convertir le diagramme en Basic (F5). Transférer le programme dans le FlashProg (F5) et vérifier que son exécution correspond à sa description.

FICHE N°1 : programme F1-CLIGNO.cad

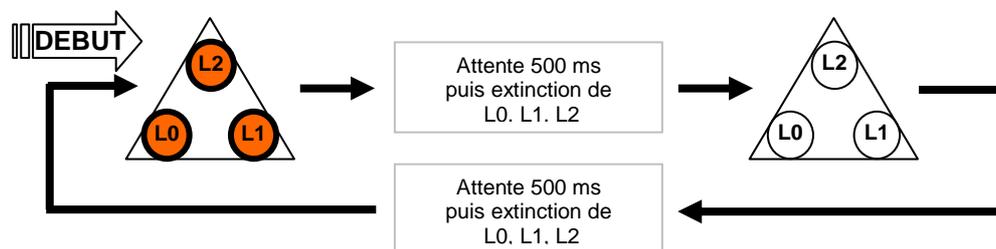
Pour une première exploitation du module, vous devez **charger le fichier de programme à disposition** sur le site du collège ou encore sur votre dossier personnel en partage sur SambaEdu **et ensuite l'exécuter**.

Intervenez ensuite sur le programme afin de respecter le **temps d'attente de 1500ms pour voir reboucler** le programme d'animation indéfiniment.

But du programme : faire clignoter simultanément L0, L1 et L2 (effet clignoteur).

Notion de programmation abordée : activation / désactivation de plusieurs sorties, utilisation d'un temps d'attente

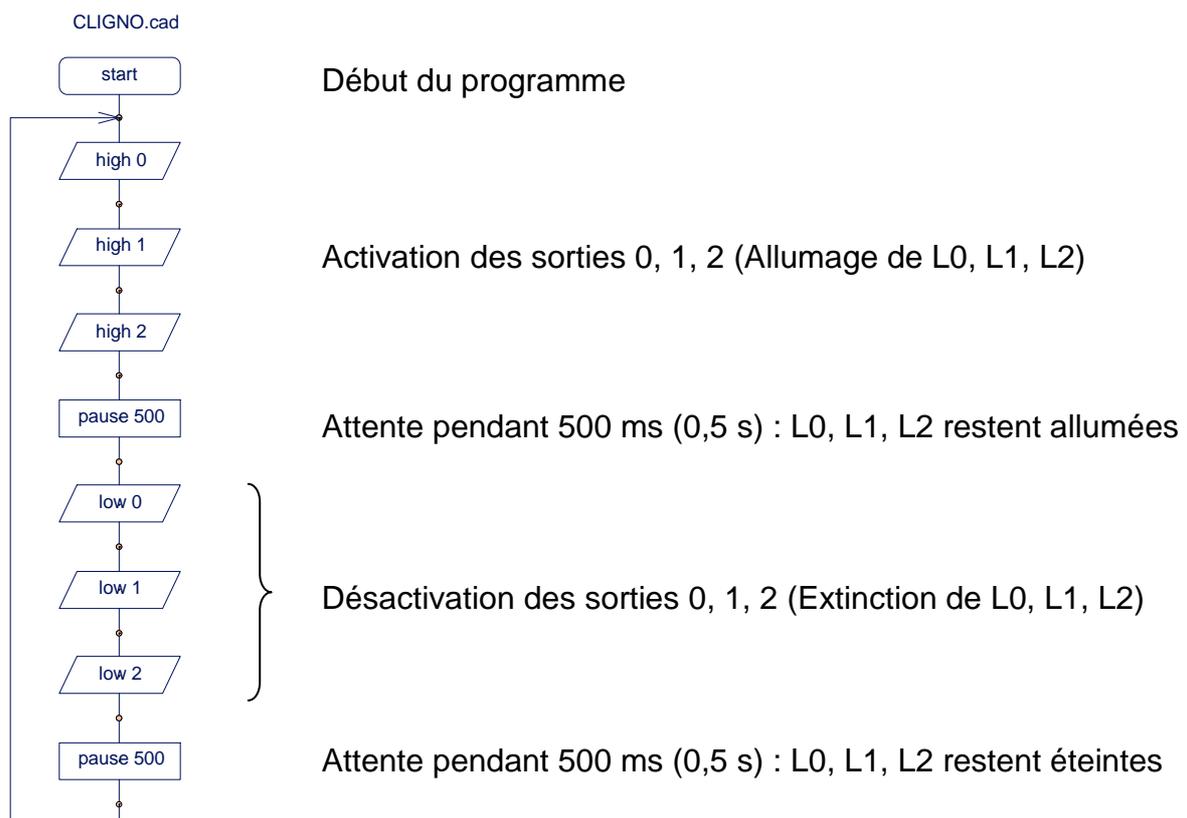
Synoptique :



Commentaire :

Dès la mise sous tension L0, L1, L2 sont allumées. Au bout de 500 ms elles sont éteintes. Un temps d'attente de 1500 ms sera introduit avant de les allumer de nouveau. Le programme reboucle indéfiniment sur lui même.

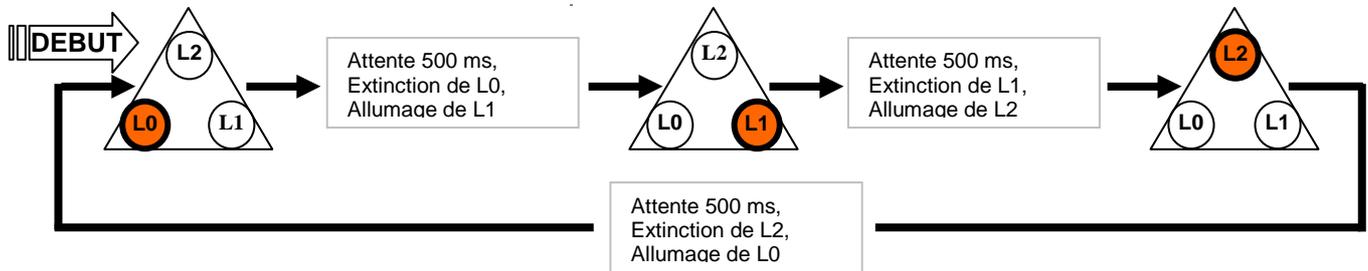
Diagramme de programmation :



But du programme : faire clignoter successivement L0, L1 et L2 (effet chenillard).

Notion de programmation abordée : activation / désactivation de plusieurs sorties, utilisation d'un temps d'attente

Synoptique :



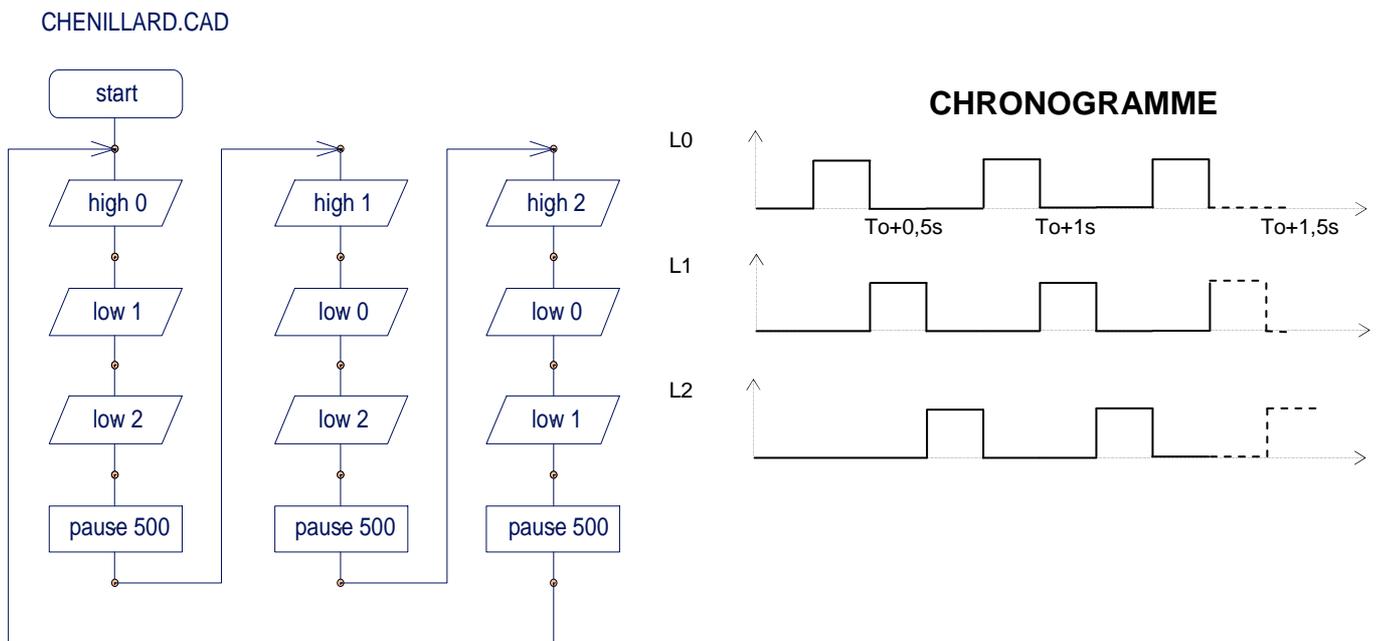
Commentaire :

Dès la mise sous tension L0 est allumée, L1, L2 sont éteintes. Au bout de 500 ms L0 est éteinte et L1 est allumée. Au bout de 500 ms L1 est éteinte et L0 est allumée.

Un temps d'attente de 500ms est introduit avant d'allumer de nouveau L0. Le programme reboucle indéfiniment sur lui-même ; cette séquence donne l'effet d'un chenillard.

Vous disposez du diagramme de programmation ci-dessous et il ne vous reste plus qu'à l'éditer à partir du logiciel "Programming Editor" pour ensuite le charger dans le module FLASHProg et enfin en vérifier le bon déroulement...

Diagramme de programmation :



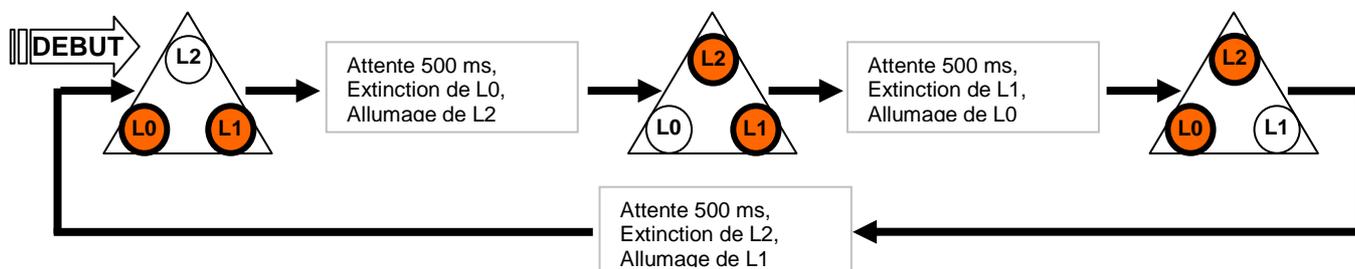
Note : A la mise sous tension du module (première exécution du programme), toutes les sorties sont initialisées à l'état bas (L0, L1, L2 sont éteintes). Une sortie qui est activée à l'état haut (ex. « high0 ») reste active jusqu'à ce que le programme rencontre une instruction qui la désactive (ex. low0).

FICHE N°3 : programme F3-BATON TOURNANT.cad

But du programme : allumer momentanément une paire de DEL et faire une rotation de cet effet en allumant momentanément la paire adjacente.

Notion de programmation abordée : activation / désactivation de plusieurs sorties, utilisation d'un temps d'attente

Synoptique :



Commentaire :

Dès la mise sous tension L0 et L1 sont allumées, L2 est éteinte. Au bout de 500 ms L0 est éteinte et L2 est allumée. Au bout de 500 ms L1 est éteinte et L0 est allumée. Le programme reboucle indéfiniment sur lui-même.

Travail à effectuer :

==> Constituer le Diagramme de programmation

==> Imaginer le Chronogramme correspondant