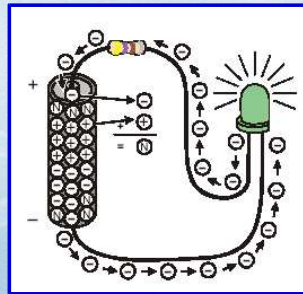




Branchement d'une LED



Connecter une LED

Une tension de 5 volts peut détruire une LED si le courant n'est pas limité (max 30mA).

La LED induit une baisse de tension de 1.4V on a donc:

$$I = V/R = (5V - 1.4V) / R < 0.03 \text{ A}$$

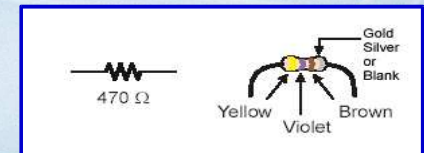
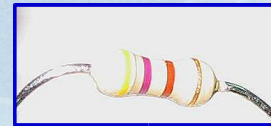
Avec une résistance de 470Ω, on obtient un courant de:

$$(5V - 1.4V) / 470\Omega = 7.7\text{mA}$$



La Résistance

Un code de couleur indique la valeur de la résistance



Schematic Symbol

Part Drawing

- 1^{ère} Bande: 1^{er} Digit
- 2^{ème} Bande: 2^{ème} Digit
- 3^{ème} Bande: facteur multiplicatif
- 4^{ème} Bande (si présente): Tolérance.

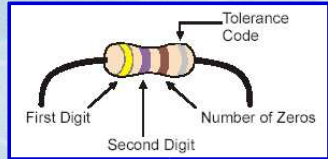


Pour la résistance montrée:
 Jaune = 4, 1st Digit
 Violet = 7, 2nd Digit
 Marron = 1, ajouter 1 zero.
 470 Ω

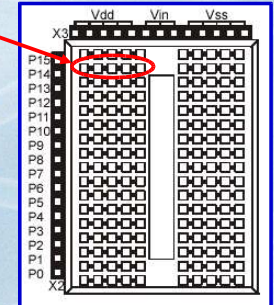
Tolérance :
 Or: 5%
 Argent: 10%
 aucune: 20%

Table 2-1:
 Resistor Color
 Code Values

Digit	Color
0	Black
1	Brown
2	Red
3	Orange
4	Yellow
5	Green
6	Blue
7	Violet
8	Gray
9	White



- Les trous d'une même ligne sont connectées électriquement.
- Les pattes du BASIC Stamp sont reportées sur la carte:
 - **E/S connexions** du BASIC Stamp (P0-P15)
 - **Vdd**: + Voltage
 - **Vss**: - Voltage
 - **Vin**: Alim pile



i Éviter d'utiliser Vin

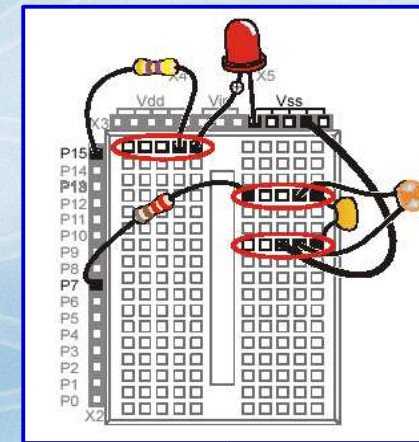
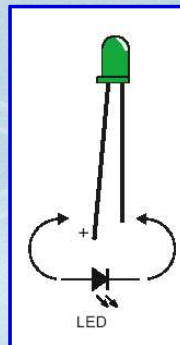


A Diode is light a one-way check valve in that current can flow in only one direction. An LED is a diode that emits light as current is passed through it (Light Emitting Diode).

Note the connections on the LED:

Anode: Connectée au + de l'alim.
 En général broche plus longue.

Cathode: Connectée au - de l'alim.
 Broche courte et méplat de la lentille

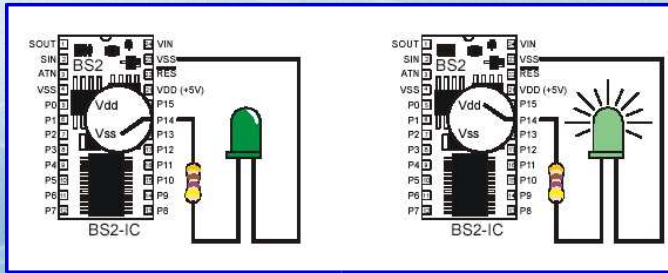




Marche/Arrêt de la LED via le BASIC Stamp

What's a Microcontroller?

Avec le BASIC Stamp les pins Entrées/Sorties (P0-P15) sont contrôlées pour fournir soit une tension Vdd (+) ou Vss (-). Permet de contrôler si le courant passe ou non.



9



Programme pour faire clignoter la LED

What's a Microcontroller?

```

What's a Microcontroller - LedOnOff.bs2
' Turn an LED on and off. Repeat 1 time per second indefinitely.
' {$STAMP BS2}
' {$PBASIC 2.5}
DO
  HIGH 14
  PAUSE 500
  LOW 14
  PAUSE 500
LOOP

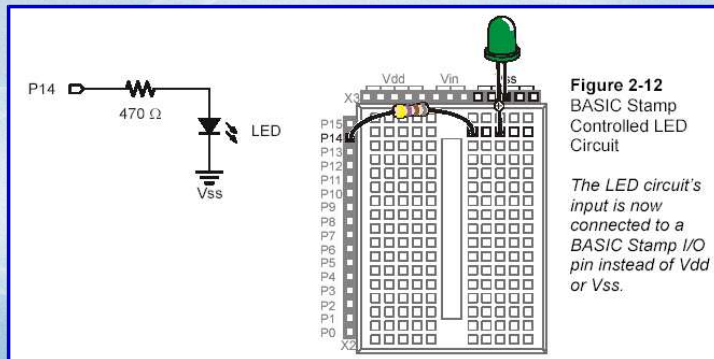
```

11



Exemple de connexion

What's a Microcontroller?



10



What's a Microcontroller?

- La LED doit flasher une fois par seconde.
- Commandes principales:
 - **HIGH 14:** Place l'E/S P14 au niveau haut (correspond à 5V ou Vdd, valeur binaire 1). Le courant entre P14 et Vss alimente la.
 - **PAUSE 500:** Le BASIC Stamp s'arrête pour le temps spécifié en millisecondes.
500 millisecondes = 0.5 seconde
 - **LOW 14:** Places l'E/S P14 au niveau bas (correspond à 0V ou Vss, valeur binaire 0). Le courant ne circule plus entre P14 et Vss, la LED est éteinte.
 - **DO et LOOP:** créent une structure de boucle répétitive.

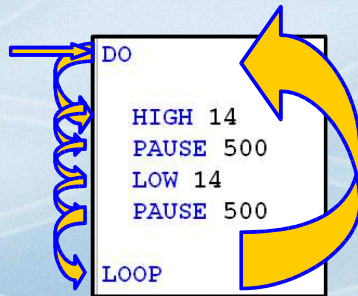
12



Sens d'exécution et boucle

What's a Microcontroller?

Le programme progresse ligne après ligne (haut vers le bas) et exécute chaque instruction.







Boucle: Loop entraîne le branchement vers l'instruction suivant le DO associé au LOOP.



Flowcharts

What's a Microcontroller?

Les Flowcharts utilisent des symboles pour représenter le type d'action qui a lieu à chaque étape du programme afin d'illustrer graphiquement le flot d'instructions.

-  **Ovale - Debut/Fin:** d'un programme ou d'une routine.
-  **Rectangle - Process:** Indique un calcul réalisé par l'ordinateur / controleur.
-  **Parallelogramme – Entrées/Sorties:** Indique que l'on lit une information ou que l'on controle une sortie.
-  **Diamant – Décision:** L'état d'une condition est vérifié et l'exécution se poursuit dans la branche 1 ou 2 en fonction du résultat d'un test: Vrai ou Faux.



Ecriture en Pseudo-Code

What's a Microcontroller?

Exemple de Pseudo-Code pour faire clignoter une LED:

2. Debut
3. Allumer la LED
4. Attendre ½ secondz
5. Eteindre la LED
6. Attendre ½ seconde
7. Retourner au début

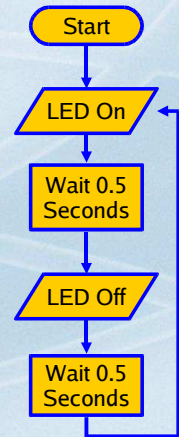
Ce pseudo-code peut être traduit dans n'importe quel langage (ex. JAVA).



What's a Microcontroller?

Le flowchart du programme pour faire clignoter une LED peut être le suivant:

Noter que les flèches indique l'ordre et le sens de l'exécution des instructions.





TP à réaliser :

What's a Microcontroller?

- Faire clignoter une LED sur votre robot
- Maintenant que la LED clignote, testez les points suivants :
- Déconnecter le câble série du BASIC Stamp. Que se passe t'il? Pourquoi? Où est le programme?
 - Eteindre le basic stamp et compter jusqu'à 10. Rallumez le. Que se passe t'il ? Pourquoi?
 - Que se passe t'il si l'argument de PAUSE est changé (tester différentes valeurs)
 - Que se passe t'il si les mots réservés DO et LOOP sont mis en commentaires (mettre une apostrophes juste avant eux)?



Utiliser une boucle DO-LOOP WHILE

What's a Microcontroller?

Une autre manière d'écrire le code est d'utiliser une boucle DO-LOOP WHILE . La boucle continuera de s'écouter tant que le counter <= 10 est vrai. Il s'agit d'une **boucle conditionnelle**.

Le compteur est:

- Initialisé
- Mis à jour
- Testé

```

Counter VAR Byte
Counter = 1
DO
  HIGH 14
  PAUSE 500
  LOW 14
  PAUSE 500
  Counter = Counter + 1
LOOP WHILE (Counter <= 10)
END

```

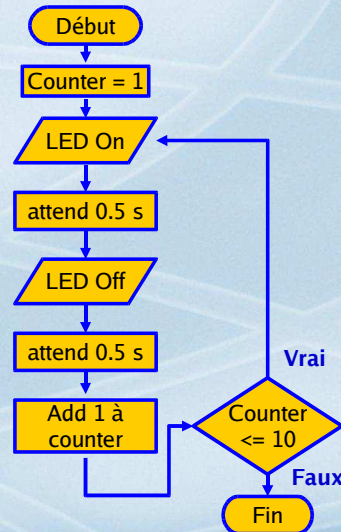
Valeur du compteur à la fin du programme? 11.



Problème: Faire clignoter une LED 10 fois

What's a Microcontroller?

- Début
- Counter est initialisé à 1
- Allume la LED
- Pause ½ seconde
- Eteind la LED
- Pause ½ seconde
- Incrémente counter
- Counter < 10?
 - Vrai, retour à l'étape 4
- Fin du Programme



Les variables

What's a Microcontroller?

Les types de variable sont définies par rapport à la plus grande valeur stockée (espace mémoire nécessaire).

Type de Variable	Valeurs possibles
Bit	0 to 1
Nib (short for Nibble)	0 to 15
Byte	0 to 255
Word	0 to 65535



What's a Microcontroller?

Une boucle FOR-NEXT est une simple structure permettant de compter entre 2 valeurs. L'instruction NEXT renvoie au début de la boucle tant que la valeur finale n'a pas été atteinte.

```

Counter VAR Byte

FOR Counter = 1 TO 10
  DEBUG ? Counter
  HIGH 14
  PAUSE 500
  LOW 14
  PAUSE 500
NEXT

DEBUG "All Done!"
END

```

i Le code est indenté dans la boucle pour faciliter la lecture.



What's a Microcontroller?

Les variables permettent de nommer des positions dans la mémoire RAM (stocker une donnée). Le format général pour définir une variable est: **Nom_variable VAR Type**

Le nom sélectionné doit être prérepresentatif du contenu de la variable. Les limitations sur les noms sont les suivantes:

3. Ne pas utiliser de mots réservés du PBASIC tels que END ou LOOP.
4. Pas d'espace dans les noms.
5. Ne doit pas commencer par un chiffre (mais peut les utiliser).
6. Ne doit pas faire plus de 33 caractères.

Par convention, les variables commencent par une majuscule pour la 1ere lettre de chaque nom, ex:

MaValeur, ValeurIn, Moteur_Gauche



What's a Microcontroller?

```

Counter VAR Byte           ' Variable to hold count

FOR Counter = 1 TO 10     ' Counter starts at 1 up to 10
  DEBUG ? Counter         ' Display Counter value
  HIGH 14                 ' LED on
  PAUSE 500               ' 0.5 Second Wait
  LOW 14                  ' LED Off
  PAUSE 500              ' 0.5 Second Wait
NEXT                      ' Increment for next value of
                          ' Counter
DEBUG "All Done!"        ' Display end message
END                       ' End program

```



What's a Microcontroller?

- Modifier le programme précédent pour commander les 2 leds

