

# Capteurs

- Saisies
- Compteur
- I2C
- USB

# Saisies

## Le démarrage de chaque bloc

Le **démarrage de chaque bloc** permet d'exécuter un programme si une condition est remplie. Il fonctionne donc comme une distinction de cas, non seulement une fois, mais chaque fois que la condition est remplie, tout au long du programme. Le **démarrage de chaque bloc** :



Est une abréviation de la construction suivante :



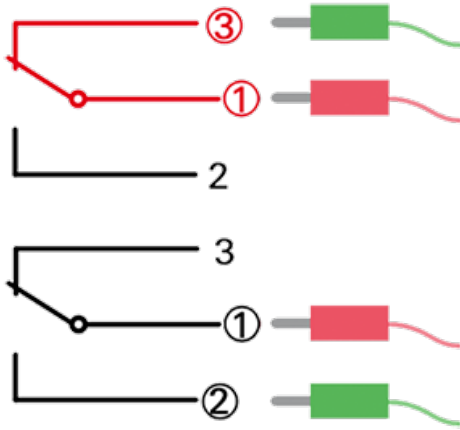
On peut insérer dans le **Démarrage de chaque bloc** de la catégorie Saisies toutes les conditions de cette même catégorie.

**Remarque :** La section du programme à l'intérieur du démarrage de chaque bloc doit être courte et ne pas comporter d'appels bloqués ou de boucles de fin de session, afin que cette partie du programme puisse être traitée rapidement.

## Touche de palpation



La touche de palpation est un capteur numérique qui distingue les états « Le courant circule » et « Le courant ne circule pas ». Le flux de courant dépend à la fois du câblage et du fait que la touche de palpation soit enfoncée. On peut donc utiliser le bouton de deux façons différentes :



Comme « fermeture » :  
les contacts 1 et 3 sont raccordés.

Touche de palpage enfoncée : Le courant circule.  
Touche de palpage non enfoncée : Le courant ne circule pas

Comme « ouverture » :  
Les contacts 1 et 2 sont raccordés.

Touche de palpage enfoncée : Le courant ne circule pas.  
Touche de palpage non enfoncée : Le courant circule.

## Afficher

Avec **Récupérer le statut de la mini touche de palpage**, on obtient des informations sur le fait que du courant circule dans la touche de palpage ou non. Si du courant circule, le signal **1** est retourné, en l'absence de courant, le signal est de **0**.

## Interroger

Pour vérifier si le bouton est dans un état donné, le bloc **Bouton ...** est utilisé. Le menu déroulant (petit triangle) permet d'indiquer l'état interrogé. Ce bloc peut être utilisé comme une condition.

## Capteur à ultrasons



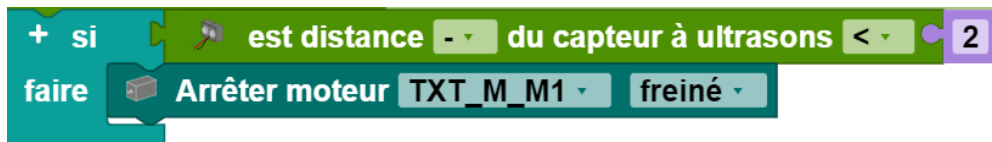
Le capteur à ultrasons est utilisé pour mesurer des distances.

## Afficher

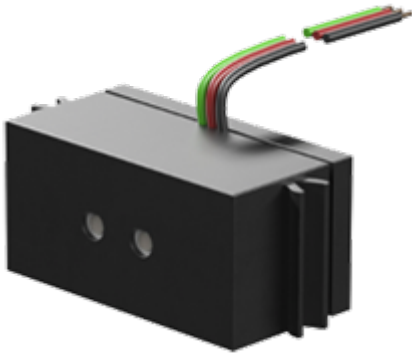
Avec **Récupérer la distance du capteur à ultrasons**, on obtient des informations sur la distance entre le capteur et l'objet le plus proche. La distance est retournée en cm.

## Interroger

Pour déterminer si le capteur est situé à une certaine distance de l'objet le plus proche, le bloc **Distance du capteur à ultrasons ...** est utilisé. Le menu déroulant (petit triangle) permet de sélectionner comment comparer la distance mesurée à une valeur saisie (<, ?, =, ?, ?, >). La valeur de comparaison est saisie dans le champ numérique à la fin du bloc. Ce bloc peut être utilisé comme une condition. Dans l'exemple, le moteur est arrêté lorsque la distance mesurée est inférieure à 2 cm.



## Capteur de couleurs



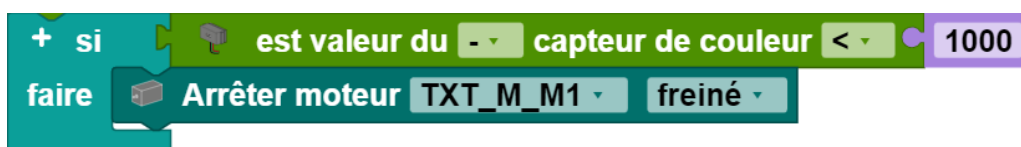
Le capteur de couleur émet de la lumière rouge et mesure la quantité de lumière réfléchiée. En fonction de l'intensité de la réflexion, le capteur de couleur renvoie des valeurs de 0 à 2000. Il convient parfaitement pour reconnaître les couleurs préalablement calibrées.

## Afficher

La fonction **Récupérer la valeur du capteur de couleur** permet d'obtenir des informations sur l'intensité de réflexion de la lumière par une surface.

## Interroger

Pour déterminer si le capteur a une certaine couleur devant lui, le bloc **Valeur du capteur de couleurs ...** est utilisé. Le menu déroulant (petit triangle) permet de sélectionner comment comparer la valeur de couleur mesurée à une valeur saisie (<, ?, =, ?, ?, >). La valeur de comparaison est saisie dans le champ numérique à la fin du bloc. Ce bloc peut être utilisé comme une condition. Dans l'exemple, le moteur est arrêté lorsque la valeur de couleur mesurée est inférieure à 1000.



## Détecteur de piste infrarouge



Le détecteur de piste infrarouge est un détecteur numérique qui détecte une piste noire sur un fond blanc et qui fonctionne à une distance de 5 à 30 mm d'un capteur à l'autre.

## Afficher

Avec **Récupérer le statut du détecteur de piste infrarouge** on obtient 0 si le capteur ne détecte pas de piste. Si le capteur détecte une piste, 1 est retourné.

## Interroger

Pour savoir si le détecteur de piste infrarouge détecte une piste, il faut comparer l'état actuel de la piste à 0 ou 1. Le bloc **Statut du détecteur de piste infrarouge** [] ... convient pour cela . Le menu déroulant (petit triangle) permet de sélectionner comment comparer la valeur de piste mesurée à une valeur saisie (<, ?, =, ?, >). La valeur de comparaison est saisie dans le champ numérique à la fin du bloc. Ce bloc peut être utilisé comme une condition. Dans l'exemple, le moteur est arrêté lorsque la valeur de piste est à 0.



## Phototransistor



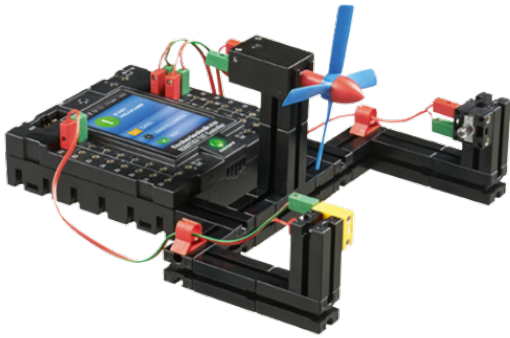
Le phototransistor est un capteur numérique qui distingue le clair du foncé.

## Afficher

Avec **Récupérer le statut du phototransistor** on obtient 0 si le capteur ne détecte pas de lumière. Si le capteur détecte une lumière, 1 est retourné..

## Interroger

Pour savoir si le phototransistor détecte du clair ou du foncé, il faut comparer l'état de luminosité à 0 ou 1. Le bloc **Statut du phototransistor** []. convient pour cela Le menu déroulant (petit triangle) permet de spécifier s'il faut interroger le clair ou le foncé.



Une utilisation possible d'un phototransistor est une barrière lumineuse, comme ce modèle.

## Photorésistance



La résistance de la photorésistance diminue si elle est exposée à une luminosité plus importante. La valeur de la photorésistance est donc une mesure de la luminosité.

### Afficher

La fonction **Récupérer la valeur de la photorésistance** permet de savoir à quel point la luminosité est importante. Plus la valeur émise est faible, plus elle est claire.

### Interroger

Pour savoir si la photorésistance mesure une valeur de luminosité spécifique, le bloc **Valeur de photorésistance** ... est utilisé. Le menu déroulant (petit triangle) permet de sélectionner comment comparer la valeur de luminosité avec la valeur saisie (<, ?, =, ?, >). La valeur de comparaison est saisie dans le champ numérique à la fin du bloc. Ce bloc peut être utilisé comme une condition. Dans l'exemple, le moteur est arrêté lorsque la valeur de luminosité mesurée est inférieure à 3000.



## Résistance NTC



La résistance NTC est un capteur de température non binaire. Sa résistance électrique diminue lorsque la température augmente, ce qui en fait une mesure de la température.

## Afficher

La fonction **Récupérer la résistance NTC** [] permet d'obtenir une valeur de résistance ou la température déduite. Vous pouvez sélectionner ce qui doit être retourné en utilisant le menu déroulant (petit triangle).

## Interroger

Pour savoir si la résistance NTC mesure une certaine valeur, le bloc **Résistance NTC** [] ... est utilisé. Les menus déroulants (petit triangle) permettent de sélectionner ce que vous souhaitez comparer et avec quel opérateur de comparaison. La valeur de comparaison est saisie dans le champ numérique à la fin du bloc. Ce bloc peut être utilisé comme une condition. Dans l'exemple, le moteur est arrêté lorsque la température est inférieure à 20.



# Compteur



Le moteur encodeur peut être utilisé comme compteur. Le nombre de tours est compté, non seulement lorsqu'il tourne en tant que moteur, mais aussi lorsqu'il est entraîné mécaniquement de l'extérieur.

## Le démarrage de chaque bloc

Le **démarrage de chaque bloc** permet d'exécuter un programme si une condition est remplie. Il fonctionne donc comme une distinction de cas, non seulement une fois, mais chaque fois que la condition est remplie, tout au long du programme. Le **démarrage de chaque bloc** :



Est une abréviation de la construction suivante :



On peut insérer dans le **démarrage chaque bloc** de la catégorie Compteur toutes les conditions de cette même catégorie.

**Remarque :** La section du programme à l'intérieur du démarrage de chaque bloc doit être courte et ne pas comporter d'appels bloqués ou de boucles de fin de session, afin que cette partie du programme puisse être traitée rapidement.

## Afficher

La fonction **Récupérer la valeur du compteur** permet d'obtenir la valeur comptée par le compteur.

## Interroger

Pour vérifier si le compteur a compté une certaine valeur, ce bloc est





utilisé. Le menu déroulant (petit triangle) permet de sélectionner comment comparer la valeur comptée à une valeur saisie (<, ?, =, ?, ?, >).

## Réinitialiser

Le compteur recommence à zéro lorsque le bloc **Réinitialiser le compteur** est exécuté.

# I2C

Les capteurs I2C décrits dans ce chapitre sont reliés au contrôleur TXT 4.0 par un câble plat approprié.

## Le démarrage de chaque bloc

Le **démarrage de chaque bloc** permet d'exécuter un programme si une condition est remplie. Il fonctionne donc comme une distinction de cas, non seulement une fois, mais chaque fois que la condition est remplie, tout au long du programme. Le **démarrage de chaque bloc** :



Est une abréviation de la construction suivante :



On peut insérer dans le **démarrage chaque bloc** de la catégorie I2C toutes les conditions de cette même catégorie.

**Remarque :** La section du programme à l'intérieur du démarrage de chaque bloc doit être courte et ne pas comporter d'appels bloqués ou de boucles de fin de session, afin que cette partie du programme puisse être traitée rapidement.

## Capteur combiné



Le capteur combiné combine trois fonctions en un seul élément : le capteur d'accélération, le gyroscope et le capteur de compas.

## Capteur d'accélération

Afficher

La fonction **Récupérer l'accélération du capteur combiné dans []** permet d'obtenir l'accélération dans les directions. La direction souhaitée peut être sélectionnée dans le menu déroulant (petit triangle). L'accélération est indiquée en g.

## Interroger

Pour savoir si l'on mesure une certaine accélération, le bloc **Capteur combiné Accélération dans [] [] ...** est utilisé. Les menus déroulants (petit triangle) permettent de sélectionner la manière de comparer l'accélération à la valeur saisie (<, ?, =, ?, ?, >), ainsi que la direction à explorer. La valeur de comparaison est saisie dans le champ numérique à la fin du bloc. Ce bloc peut être utilisé comme une condition. Dans l'exemple, le moteur est arrêté lorsque l'accélération dans la direction x est supérieure à 10.



## Gyroscope

### Afficher

La fonction **Récupérer la rotation du capteur combiné dans []** permet d'obtenir la rotation dans une direction. La direction souhaitée peut être sélectionnée dans le menu déroulant (petit triangle). La rotation est indiquée en °/s.

## Interroger

Pour savoir si l'on mesure une certaine vitesse d'angle, le bloc **Capteur combiné Rotation dans [] [] ...** est utilisé. Les menus déroulants (petit triangle) permettent de sélectionner la manière de comparer la rotation à la valeur saisie (<, ?, =, ?, ?, >), ainsi que la direction à explorer. La valeur de comparaison est saisie dans le champ numérique à la fin du bloc. Ce bloc peut être utilisé comme une condition. Dans l'exemple, le moteur est arrêté lorsque la rotation dans la direction x est supérieure à 10.



## Capteur à compas

### Afficher

La fonction **Récupérer le flux magnétique du capteur combiné dans []** permet d'obtenir le flux magnétique dans une direction. La direction souhaitée peut être sélectionnée dans le menu déroulant (petit triangle). Le flux magnétique est indiqué en °T.

## Interroger

Pour savoir si l'on mesure un certain flux magnétique, le bloc **Capteur combiné Flux magnétique dans [] [] ...** est utilisé. Les menus déroulants (petit triangle) permettent de sélectionner la manière de comparer le flux magnétique à la valeur saisie (<, ?, =, ?, ?, >), ainsi que la direction à explorer. La valeur de comparaison est saisie dans le champ numérique à la fin du bloc. Ce bloc peut être utilisé comme une condition. Dans l'exemple, le moteur est arrêté lorsque le flux magnétique dans la direction x est supérieure à 10.



# Capteur environnemental



Le capteur environnemental combine en un seul élément les quatre fonctions suivantes : capteur de qualité de l'air, capteur d'humidité de l'air, baromètre et thermomètre.

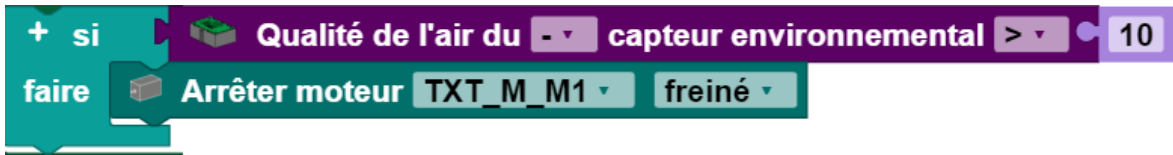
## Capteur de la qualité de l'air

### Afficher

Avec le bloc **Récupérer la qualité de l'air du capteur environnemental comme []** , on peut mesurer la qualité de l'air. Le menu déroulant (petit triangle) permet de choisir si la qualité de l'air doit être restituée en tant que valeur numérique (de 0 à 500) ou en tant que texte.

### Interroger

Pour savoir si l'on mesure une certaine qualité de l'air, le bloc **Capteur environnemental qualité de l'air[] ...** est utilisé. Le menu déroulant (petit triangle) permet de sélectionner comment comparer la qualité de l'air à la valeur saisie (<, ?, =, ?, ?, >). La valeur de comparaison est saisie dans le champ numérique à la fin du bloc. Ce bloc peut être utilisé comme une condition. Dans l'exemple, le moteur est arrêté lorsque la qualité de l'air est supérieure à 10.



## Baromètre

### Afficher

Avec le bloc **Récupérer la pression de l'air du capteur environnemental**, on peut mesurer la pression de l'air.

### Interroger

Pour savoir si l'on mesure une certaine pression de l'air, le bloc **Capteur environnemental pression de l'air[] ...** est utilisé. Le menu déroulant (petit triangle) permet de sélectionner comment comparer la pression de l'air à la

valeur saisie (<, ?, =, ?, ?, >). La valeur de comparaison est saisie dans le champ numérique à la fin du bloc. Ce bloc peut être utilisé comme une condition. Dans l'exemple, le moteur est arrêté lorsque la pression de l'air est supérieure à 10.



## Thermomètre

### Afficher

Avec le bloc **Récupérer la température du capteur environnemental**, on peut mesurer la pression de l'air.

### Interroger

Pour savoir si l'on mesure une certaine température, le bloc **Capteur environnemental température[] ...** est utilisé. Le menu déroulant (petit triangle) permet de sélectionner comment comparer la température à la valeur saisie (<, ?, =, ?, ?, >). La valeur de comparaison est saisie dans le champ numérique à la fin du bloc. Ce bloc peut être utilisé comme une condition. Dans l'exemple, le moteur est arrêté lorsque la température est supérieure à 10.



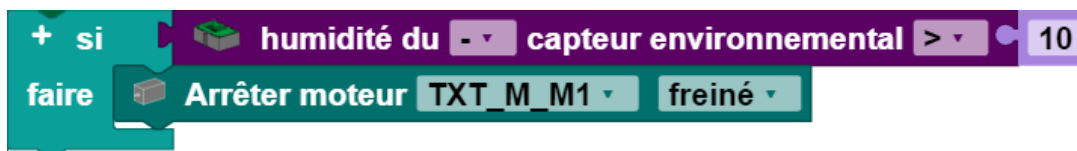
## Capteur d'humidité de l'air

### Afficher

Avec le bloc **Récupérer l'humidité de l'air du capteur environnemental**, on peut mesurer l'humidité de l'air.

### Interroger

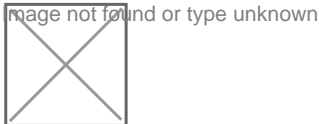
Pour savoir si l'on mesure une certaine température, le bloc **Capteur environnemental humidité de l'air[] ...** est utilisé. Le menu déroulant (petit triangle) permet de sélectionner comment comparer l'humidité de l'air à la valeur saisie (<, ?, =, ?, ?, >). La valeur de comparaison est saisie dans le champ numérique à la fin du bloc. Ce bloc peut être utilisé comme une condition. Dans l'exemple, le moteur est arrêté lorsque l'humidité de l'air est supérieure à 10.



# USB

Le port USB permet de connecter la caméra au microphone intégré. La caméra et le microphone sont ici observés séparément. Pour utiliser les fonctions de la caméra, il faut d'abord les configurer dans l'outil de configuration de la caméra. La méthode d'accès à l'outil de configuration de la caméra est expliquée [ici](#).

## Caméra



La caméra peut notamment servir de détecteur de mouvement, de détecteur de couleur, de détecteur de balle et de détecteur de ligne.

### Détecteur de mouvement

Pour utiliser la caméra comme détecteur de mouvement, il faut faire glisser le symbole de bonhomme dans la surface encadrée dans le configurateur de la caméra, puis une fenêtre s'ouvre à droite, dans laquelle sous l'inspecteur

- on peut définir la zone de pixels dans laquelle les mouvements sont contrôlés
- la position de cette zone (au point indiqué se trouve le coin supérieur gauche de la zone),
- le nom du détecteur de mouvement et
- la tolérance

### Programme du détecteur de mouvement

Le programme du détecteur de mouvement s'exécute lorsqu'un mouvement est détecté. Il est rédigé séparément du programme principal. Les variables fonctionnent globalement sur les deux programmes. Le programme de saisie s'exécute dans le bloc **lorsqu'un mouvement est détecté**.

### Détecteur de couleur

Pour utiliser la caméra comme détecteur de couleur, il faut faire glisser le symbole de bonhomme dans la surface encadrée dans le configurateur de la caméra, puis une fenêtre s'ouvre à droite, dans laquelle sous l'inspecteur

- on peut définir la zone de pixels dans laquelle la couleur est contrôlée,
- la position en pixels de cette zone (au point indiqué se trouve le coin supérieur gauche de la zone),
- le nom du détecteur de couleur et
- le contraste

### Afficher

Avec **Récupérer la couleur comme []**, on obtient la couleur détectée en hexadécimal ou au format RGB. Le format peut être défini dans le menu déroulant (petit triangle).

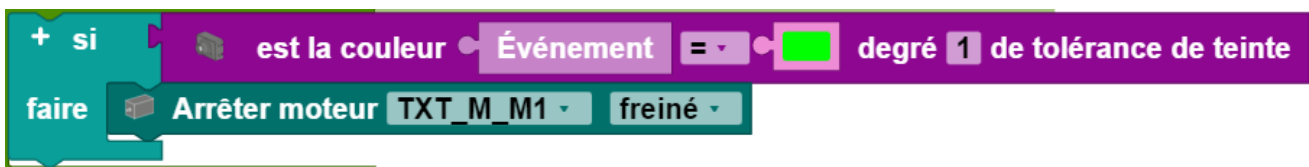
### Interroger

Pour déterminer si le détecteur a détecté une certaine couleur, le bloc **Couleur détectée ...** est utilisé. Ce bloc peut être utilisé comme une condition.

Pour vérifier si le détecteur détecte une certaine couleur, ce bloc est



utilisé. Le bloc permet de comparer la couleur perçue avec une couleur saisie. Le menu déroulant (petit triangle) permet de choisir si la couleur sélectionnée doit être égale ou non à la couleur filmée. Ce bloc peut être utilisé comme une condition. Dans l'exemple, le moteur est arrêté lorsque la couleur détectée est rouge.



## Programme du détecteur de couleur

Le programme de détection de couleur s'exécute lorsqu'une couleur est détectée. Il est rédigé séparément du programme principal. Les variables fonctionnent globalement sur les deux programmes. Le programme de saisie s'exécute dans le bloc **lorsqu'une couleur est détectée**.

## Détecteur de balle

Pour utiliser la caméra comme détecteur de balle, il faut faire glisser le symbole de bonhomme dans la surface encadrée dans le configurateur de la caméra, puis une fenêtre s'ouvre à droite, dans laquelle sous l'inspecteur

- on peut définir la zone de pixels dans laquelle la balle est contrôlée,
- la position en pixels de cette zone (au point indiqué se trouve le coin supérieur gauche de la zone),
- le nom du détecteur de balle,
- la zone dans laquelle se situe le diamètre de la balle,
- la zone de l'axe x,
- la couleur de la balle et
- la tolérance de couleur

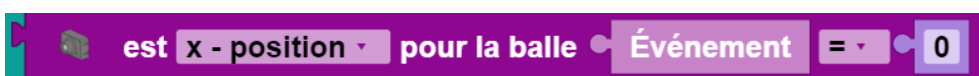
## Afficher

Avec **Récupérer la balle []**, on obtient la position x, la position y, le rayon ou le diamètre de la balle.

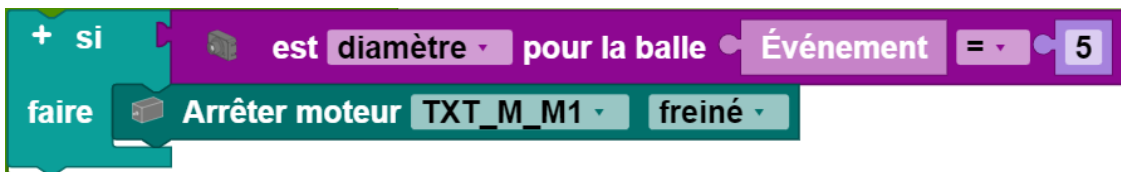
## Interroger

Pour déterminer si le détecteur a détecté une balle, le bloc **Balle détectée ...** est utilisé. Ce bloc peut être utilisé comme une condition.

Pour savoir si le détecteur détecte une balle avec une position x, une position y, un rayon ou un diamètre, ce bloc est



utilisé. Le bloc permet de comparer les spécifications de la balle détectée avec une valeur saisie. Les menus déroulants (petit triangle) permettent de sélectionner ce que vous souhaitez comparer et avec quel opérateur de comparaison. Ce bloc peut être utilisé comme une condition. Dans l'exemple, le moteur est arrêté lorsque le diamètre de la balle détectée est de 5.



## Programme de détecteur de balle

Le programme de détecteur de balle s'exécute lorsqu'une couleur est détectée. Il est rédigé séparément du programme principal. Les variables fonctionnent globalement sur les deux programmes. Le programme de saisie s'exécute dans le bloc **lorsqu'une balle est détectée**.

## Détecteur de lignes

Pour utiliser la caméra comme détecteur de lignes, il faut faire glisser le symbole avec les points sur une ligne dans la zone encadrée dans le configurateur de la caméra, puis une fenêtre s'ouvre à droite, dans laquelle sous l'inspecteur

- on peut définir la zone de pixels dans laquelle les lignes sont contrôlées,
- la position de cette zone (au point indiqué se trouve le coin supérieur gauche de la zone),
- le nom du détecteur de lignes,
- le nombre de lignes à détecter, et
- la zone dans laquelle la largeur des lignes se trouve

## Afficher

Avec **Récupérer [ ] la ligne [ ]** on obtient la position ou la largeur d'une des cinq lignes au maximum.

Avec **Récupérer la couleur de la ligne [ ] comme [ ]**, on obtient la couleur d'une ligne en hexadécimal ou au format RGB. Le format peut être défini dans le menu déroulant (petit triangle).

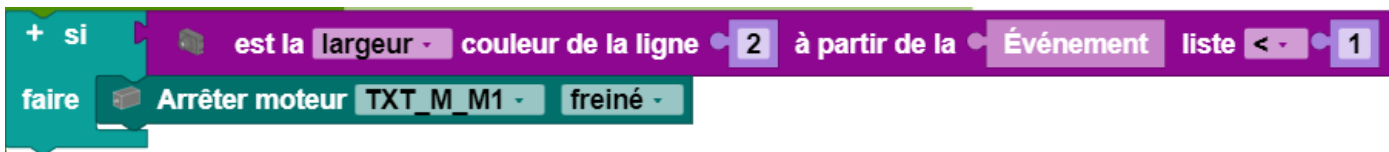
## Interroger

Pour déterminer si le détecteur a détecté une ligne, le bloc **Ligne détectée ...** est utilisé. Ce bloc peut être utilisé comme une condition.

Pour savoir si le détecteur détecte une ligne avec une position ou une largeur définie, ce bloc est



utilisé. Le bloc permet de comparer les spécifications de la ligne détectée avec une valeur saisie. Les menus déroulants (petit triangle) permettent de sélectionner ce que vous souhaitez comparer et avec quel opérateur de comparaison. Ce bloc peut être utilisé comme une condition. Dans l'exemple, le moteur est arrêté lorsque la largeur de la ligne perçue est inférieure à 2.



Pour savoir si le détecteur détecte une ligne avec une couleur définie, ce bloc est





utilisé. Le bloc permet de comparer la couleur de la ligne perçue avec une couleur saisie. Le menu déroulant (petit triangle) permet de choisir si la couleur sélectionnée doit être égale ou non à la couleur filmée. Ce bloc peut être utilisé comme une condition.

## Programme de détecteur de ligne

Le programme de détecteur de ligne s'exécute lorsqu'une ou plusieurs lignes sont détectées. Il est rédigé séparément du programme principal. Les variables fonctionnent globalement sur les deux programmes. Le programme de saisie s'exécute dans le bloc **lorsqu'une ligne est détectée**.

# Microphone

Le microphone intégré à la caméra peut être utilisé comme détecteur de volume.

## Le démarrage de chaque bloc

Le **démarrage de chaque bloc** permet d'exécuter un programme si une condition est remplie. Il fonctionne donc comme une distinction de cas, non seulement une fois, mais chaque fois que la condition est remplie, tout au long du programme. Le **démarrage de chaque bloc** :



Est une abréviation de la construction suivante :



On peut insérer dans le **démarrage chaque bloc** de la catégorie Microphone toutes les conditions de cette même catégorie.

**Remarque :** La section du programme à l'intérieur du démarrage de chaque bloc doit être courte et ne pas comporter d'appels bloqués ou de boucles de fin de session, afin que cette partie du programme puisse être traitée rapidement.

## Détecteur de volume

### Afficher

Avec **Microphone Volume** on obtient le volume en décibels.

### Interroger

Pour vérifier si le détecteur de volume détecte un certain volume, ce bloc est



utilisé. Le bloc permet de comparer le volume perçu avec un volume saisi. Le menu déroulant (petit triangle) permet de sélectionner ce que vous souhaitez comparer et avec quel opérateur de comparaison. Ce bloc peut être utilisé comme une condition.