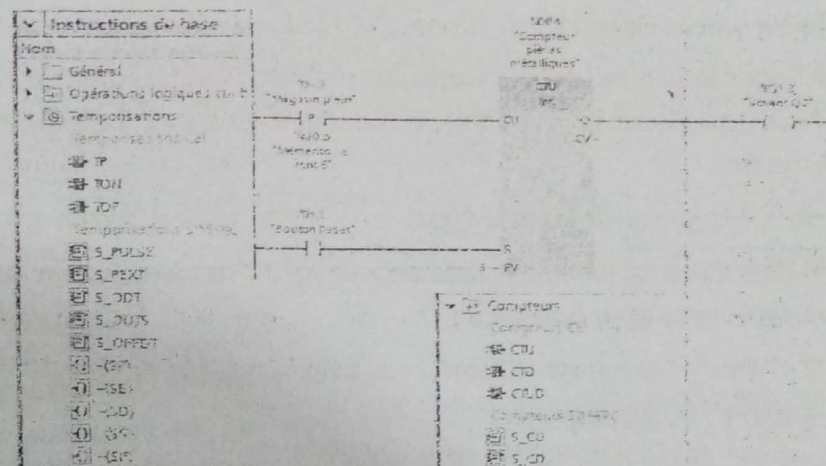


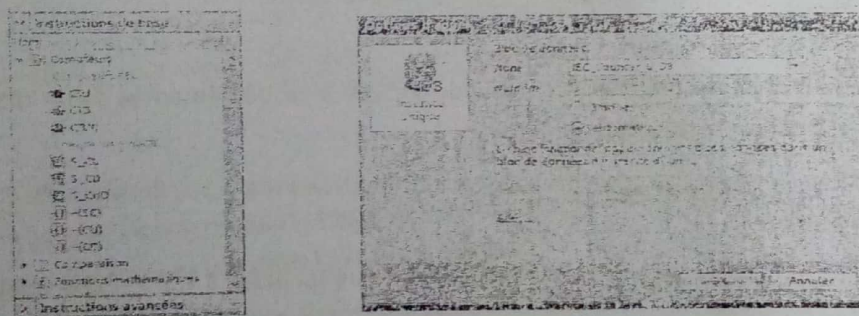
Programmation API 57-300 : Compteurs et Temporisations



1. Les compteurs

Il existe deux types de compteurs utilisables dans S7 : les compteurs S7 et les compteurs SIMATIC.

Les compteurs SIMATIC n'étant plus utilisés avec les nouvelles CPU (374-1B00), ils ne seront pas traités dans cette partie.

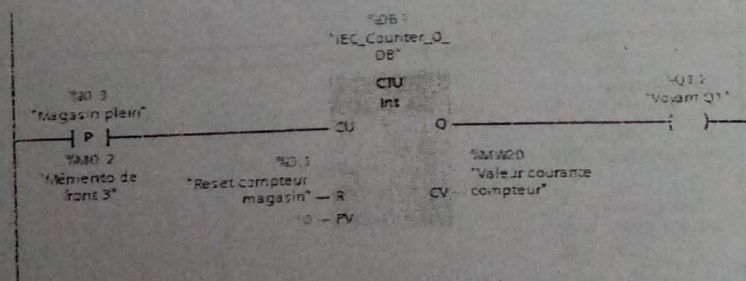


1.1 Les compteurs CEI

Il existe 3 types de compteurs CEI : **CTU** (comptage), **CTD** (décomptage) et **CTUD** (comptage et décomptage). A chaque compteur CEI est associée un DB.

Compteur CTU : comptage

L'instruction « Comptage » permet d'incrémenter la valeur à la sortie CV.



CU → Entrée du compteur. La valeur du compteur est incrémentée de un lorsque l'opérande reliée à cette entrée passe de « 0 » à « 1 ». Le compteur peut être incrémenté jusqu'à ce que la valeur supérieure du type de donnée (INT) soit atteinte.

R → Entrée de réinitialisation. La valeur de sortie CV (valeur de comptage en cours) est remise à zéro lorsque l'entrée R passe à « 1 ». Tant que R est à « 1 », l'entrée CU n'a pas d'effet sur l'instruction.

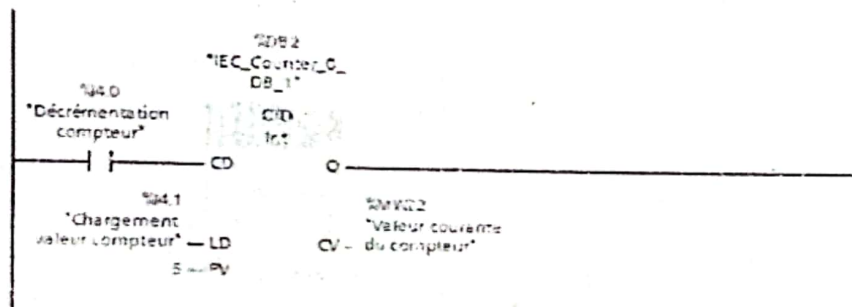
PV → Valeur à laquelle la sortie Q est mise à « 1 ». Cette entrée doit être de type INT.

Q → Etat du compteur. Cette sortie est mise à « 1 » lorsque la valeur actuelle de comptage est supérieure ou égale à la valeur PV.

CV → Valeur de comptage en cours. Cette donnée doit être de type INT.

Compteur CTD : décomptage

La sortie du compteur est mise à « 1 » lorsque la valeur courante du compteur est égale à « 0 ».



CD → Entrée du compteur. La valeur du compteur est décrétementée de un lorsque l'opérande reliée à cette entrée passe de « 0 » à « 1 ».

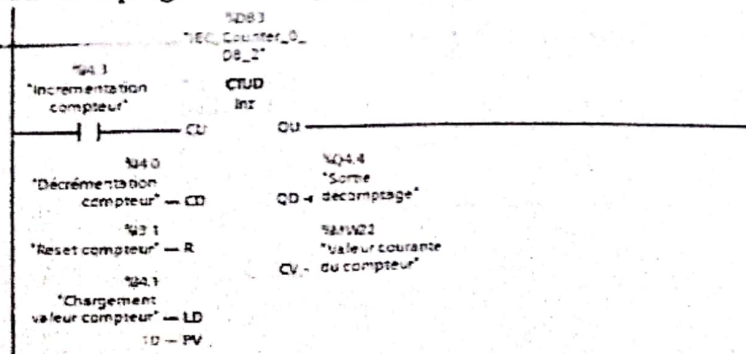
LD → Entrée de chargement. Lorsque cette entrée est à « 1 », la valeur PV est chargée dans le compteur.

PV → Valeur chargée dans le compteur lorsque LD passe à « 1 ». Cette entrée doit être de type INT.

Q → Etat du compteur. Cette sortie est mise à « 1 » lorsque la valeur actuelle de comptage est inférieure ou égale à 0.

CV → Valeur de comptage en cours. Cette donnée doit être de type INT.

Compteur CTUD : comptage et décomptage



| | |
|--------------------------------|------|
| *Compteur pièces dans magasin* | |
| CU | Bool |
| CUO | Bool |
| Q | Bool |
| R | Bool |

Choix de la variable associée

Dans l'exemple ci-dessous, c'est la sortie du compteur « Compteur pièces dans magasin » qui est associée au symbole d'interrogation à 1.

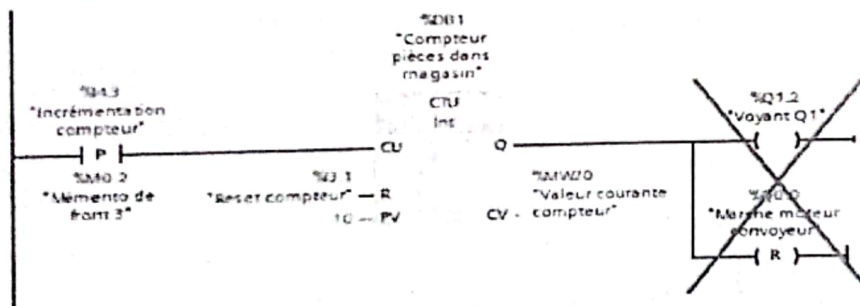
%DB1.DBX4.0
*Compteur
pièces dans
magasin*.Q



1.4 Limites de la connexion des compteurs CEI

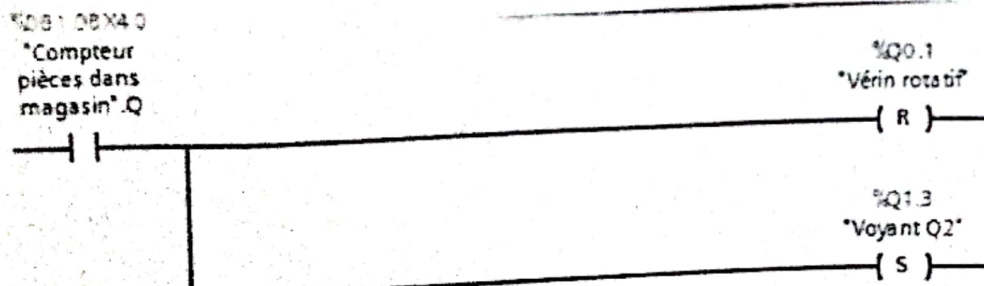
Il faut veiller à respecter la connexion des compteurs. Les règles suivantes s'appliquent à l'utilisation des temporisations CEI et des compteurs CEI sur le système cible :

- La sortie Q ne peut être connectée qu'à une bobine. Il n'est pas possible d'utiliser une bobine SET ou RESET.
- L'entrée CU ne doit pas être connectée par branchement T à une autre entrée CU.



Pour contourner ces restrictions, on peut utiliser des mémentos ou encore des contacts NO ou NF auxquels sont associés des variables liées au compteur.

Dans l'exemple ci-dessous, on met à « 0 » la sortie « Vérin rotation » et on met à « 1 » la sortie « Voyant Q2 » lorsque la sortie du compteur est à « 1 ».



CU → Entrée de comptage. La valeur du compteur est incrémentée de un lorsque l'opérande reliée à cette entrée passe de « 0 » à « 1 ».

CD → Entrée de décomptage. La valeur du compteur est décrémentée de un lorsque l'opérande reliée à cette entrée passe de « 0 » à « 1 ».

R → Entrée de réinitialisation. La valeur de sortie CV est remise à zéro lorsque l'entrée R passe à « 1 ». Tant que R est à « 1 », l'entrée CU n'a pas d'effet sur l'instruction.

LD → Entrée de chargement. Lorsque cette entrée est à « 1 », la valeur PV est chargée dans le compteur.

PV → Valeur chargée dans le compteur lorsque LD passe à « 1 ». Cette entrée doit être de type INT. La sortie comptage (QU) est mise à « 1 » lorsque le compteur atteint la valeur PV.

QU → Sortie comptage. Cette sortie est mise à « 1 » lorsque la valeur actuelle de comptage est supérieure ou égale à la valeur PV.

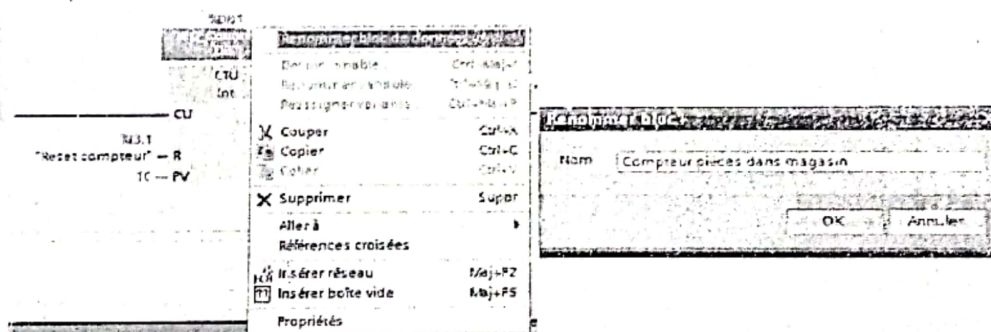
QD → Sortie décomptage. Cette sortie est mise à « 1 » lorsque la valeur actuelle de comptage est inférieure ou égale à « 0 ».

CV → Valeur de comptage en cours. Cette donnée doit être de type INT.

1.2 Renommer le bloc de donnée

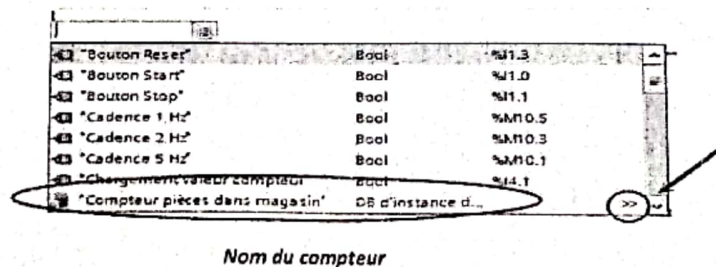
Lorsque l'on insère un compteur CEI dans un bloc de programme, le nom du bloc de donnée qui lui est associé est attribué par défaut. Il est possible de modifier ce nom :

- Lors de l'insertion du compteur, dans la fenêtre « *Option d'appel* »
- Une fois le compteur inséré, en faisant un clic droit sur son nom et en choisissant l'option « *Renommer bloc de donnée* »



1.3 Utilisation des données du compteur dans un programme

Il est possible d'assigner une variable liée au compteur à un symbole d'interrogation à 1 (ou à 0). Pour cela, il faut utiliser le menu permettant d'avoir accès aux variables pouvant être associées au symbole et de choisir le nom du compteur puis la variable associée (CU, CUO, Q, R).



2. Les temporisations

Comme pour les compteurs, il existe deux types de compteur utilisable dans S7 : les temporisations CEI et les temporisations SIMATIC.

Les temporisations SIMATIC n'étant plus utilisées avec les nouvelles CPU (S7 – 1200), elles ne seront pas traitées dans ce module.

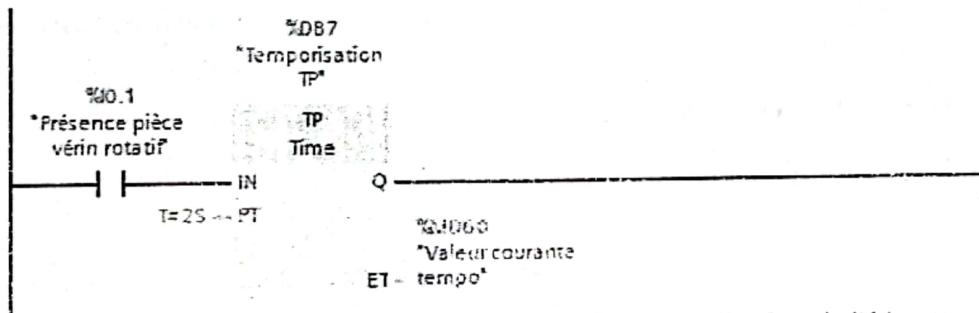
2.1 Les temporisations CEI

Il existe 3 types de temporisations CEI : **TP** (génération d'impulsion), **TON** (retard à la montée) et **TOF** (retard à la retombée). A chaque compteur CEI est associée un DB.

Temporisation TP

L'instruction « TP » met la sortie Q à 1 pour une durée programmée. L'instruction est démarrée lorsque le résultat logique à l'entrée IN passe de « 0 » à « 1 » (front montant du signal). La durée PT programmée commence à s'écouler au démarrage de l'instruction.

La sortie Q est mise à « 1 » pour la durée PT, indépendamment de l'évolution du signal d'entrée. Même la détection d'un nouveau front montant du signal n'a pas d'influence sur l'état logique à la sortie Q tant que la durée PT n'est pas entièrement écoulée.

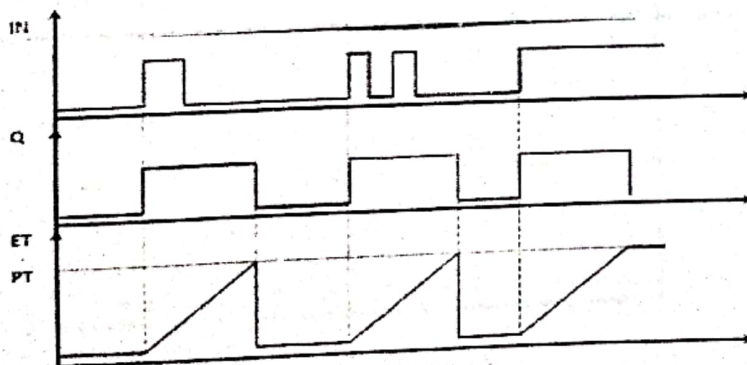


IN → Entrée de démarrage : la temporisation est lancée lorsque l'opérande lié à cette entrée passe de « 0 » à « 1 ».

PT → Durée de l'impulsion. Cette valeur doit être positive.

Q → Sortie de la temporisation. La sortie Q est mise à « 1 » pour la durée PT, indépendamment de l'évolution du signal d'entrée. Même la détection d'un nouveau front montant du signal n'a pas d'influence sur l'état logique à la sortie Q tant que la durée PT n'est pas entièrement écoulée.

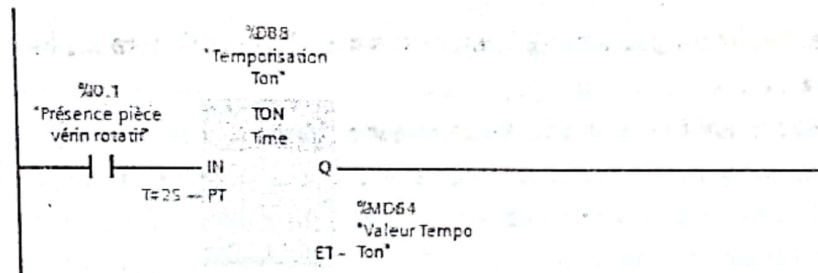
ET → Valeur de temps actuelle : la valeur de temps commence à T#0s et s'arrête lorsque la durée PT est atteinte. Lorsque la durée PT est atteinte et que l'état logique à l'entrée IN est « 0 », la sortie ET est remise à 0. C'est une variable de type « time ».



Temporisation Ton

L'instruction « **Ton** » retarde la mise à « 1 » de la sortie Q de la durée programmée PT. L'instruction est démarrée lorsque le résultat logique à l'entrée IN passe de « 0 » à « 1 ». La durée PT programmée commence à s'écouler au démarrage de l'instruction.

Une fois la durée PT écoulée, la sortie Q fournit l'état logique « 1 ». La sortie Q reste à 1 tant que l'entrée de démarrage fournit "1". Lorsque l'état logique à l'entrée de démarrage passe de « 1 » à « 0 », la sortie Q est remise à 0. La fonction de temporisation est redémarrée lorsqu'un nouveau front montant est détecté à l'entrée de démarrage.



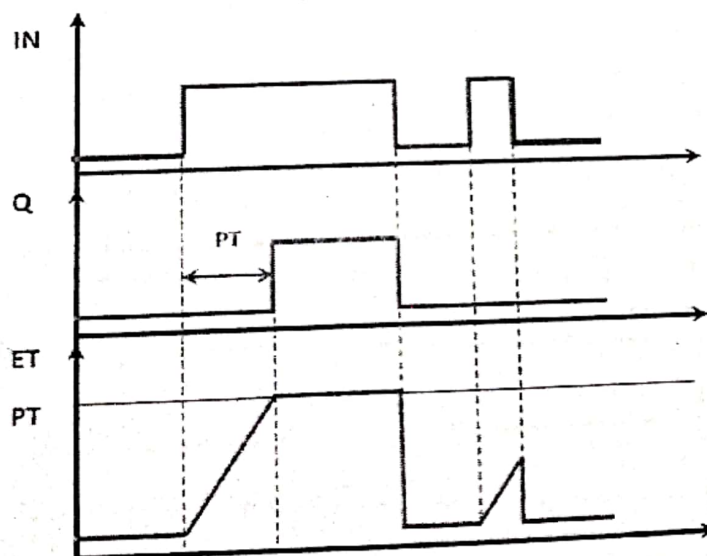
IN → Entrée de démarrage : la temporisation est lancée lorsque l'opérande lié à cette entrée passe de « 0 » à « 1 ».

PT → Durée du retard à la montée. Cette valeur doit être positive.

Q → Sortie de la temporisation. La sortie Q est mise à « 1 » après l'écoulement de la temporisation

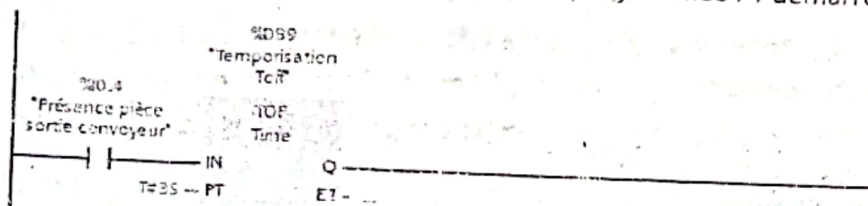
PT. Lorsque l'état logique à l'entrée de démarrage passe de « 1 » à « 0 », la sortie Q est remise à « 0 ». La fonction de temporisation est redémarrée lorsqu'un nouveau front montant est détecté à l'entrée de démarrage.

ET → Valeur de temps actuelle : la valeur de temps commence à T#0s et s'arrête lorsque la durée PT est atteinte. Lorsque la durée PT est atteinte et que l'état logique à l'entrée IN est « 0 », la sortie ET est remise à 0. C'est une variable de type « time ».



Temporisation Tof

L'instruction « Tof » permet de retarder la mise à « 0 » de la sortie Q de la durée programmée PT. La sortie Q est mise à 1 lorsque le résultat logique à l'entrée IN passe de « 0 » à « 1 ». Lorsque l'état logique à l'entrée IN repasse à « 0 », la durée programmée PT démarre.

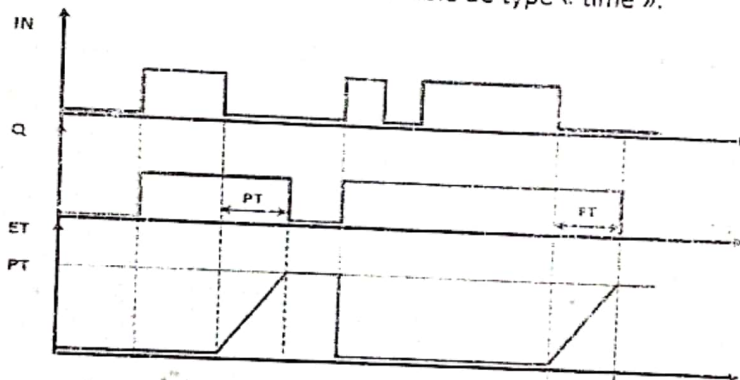


IN → Entrée de démarrage : la sortie Q est mise à « 1 » lorsque l'opérande lié à cette entrée passe de « 0 » à « 1 ».

PT → Durée du retard à la retombée. Cette valeur doit être positive.

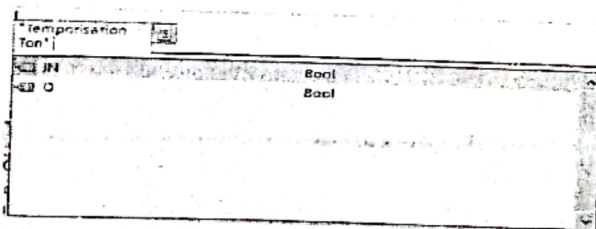
Q → Sortie de la temporisation. La sortie Q est mise à « 1 » lorsque l'entrée IN passe de « 0 » à « 1 ». La sortie Q reste à « 1 » tant que la durée PT s'écoule. Une fois la durée PT écoulée, la sortie Q est remise à « 0 ».

ET → Valeur de temps actuelle : la valeur de temps commence à T#0s et s'arrête lorsque la durée PT est atteinte. Lorsque la durée PT est atteinte et que l'état logique à l'entrée IN est « 0 », la sortie ET est remise à « 0 ». C'est une variable de type « time ».



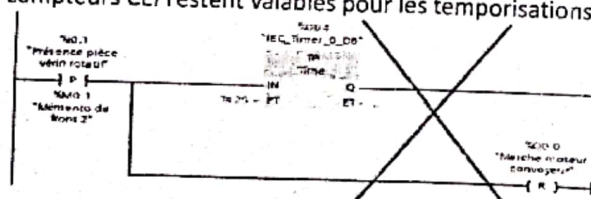
2.2 Utilisation des données des temporisations dans un programme

Comme pour les compteurs, il est possible d'assigner une variable liée à la temporisation à un symbole d'interrogation à 1 (ou à 0). Les variables associées peuvent être « IN » et « Q ».

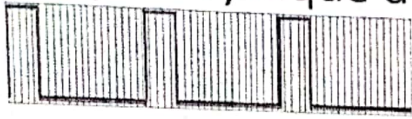


2.3 Limites de la connexion des temporisations CEI

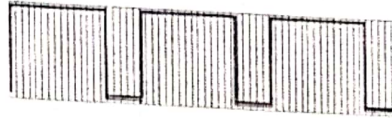
Les remarques faites pour les compteurs CEI restent valables pour les temporisations.



... à l'état haut, 3,3V ou 5V selon l'Arduino, soit un rapport cyclique de 100%.



P.W.M. Signal @ 25%



P.W.M. Signal @ 75%



P.W.M. Signal rising

Utilisation d'un pin en Sortie analogique

Pour émettre un signal analogique, on utilise le code suivant :

analogWrite (Numéro_pin , valeur);

où valeur est un nombre 0 - 255

```
//pin 11,10,9,6,5,3 peut être utilisés en sortie analogique (output)
void setup() {
  pinMode(11, OUTPUT); // Spécifier le numéro de Pin et le mode en sortie
}
void loop() {
  analogWrite(11, 255); // Emettre Analog output 255
  delay(500);          // Temporisation 0.5s
  analogWrite(11, 200); // Emettre Analog output 200
  delay(500);          // Temporisation 0.5s
  analogWrite(11, 150); // Emettre Analog output 150
  delay(500);          // Temporisation 0.5s
  analogWrite(11, 100); // Emettre Analog output 100
  delay(500);          // Temporisation 0.5s
  analogWrite(11, 50);  // Emettre Analog output 50
  delay(500);          // Temporisation 0.5s
  analogWrite(11, 0);   // Emettre analog output 0
  delay(500);          // Temporisation 0.5s
}
```