



Sommaire

- 1 Objectifs
- 2 A quoi sert un signal sinusoïdal ?
- 3 Paramètres du signa
- 4 Génération du signal sinusoïdal en format 8 bits
- 5 Initialisation du port série - Affichage des signaux
- 6 Initialisation des indices des tableaux
- 7 Affichage du signal sinus
 - 7.1 Affichage du signal cosinus
 - 7.2 Mise à jour des indices
- 8 Programme Arduino
- 9 Sujet connexes



Objectifs

1. Savoir générer un signal sinus & cosinus
2. Savoir convertir un signal en format 8 bits non signés
3. Savoir déphaser les signaux
4. Savoir générer plusieurs signaux déphasés sans location mémoire
5. Etc.

A quoi sert un signal sinusoïdal ?

1. Test des circuits électroniques
2. Mesure de la fonction de transfert, diagramme de Bode, etc.
3. Onduleurs monophasés & triphasés (SPWM)
4. Oscillateurs, signal d'horloges RF
5. Modulateur/Démodulateur
6. Etc.

Paramètres du signal

```
#define N      256          // N=2^8, 8 bits
#define Phi    N/2         // 90*356/360
#define pi     3.1415926535897932384626433832795
#define f0     1.00 // La fréquence
```



Génération du signal sinusoïdal en format 8 bits

```
for(i=0;i<N;i++)  
{  
    sin_x=sin(2.00*pi*f0*(float)i/((float)N-1.00));  
    SinCos[i]=0.5*(sin_x+1.00)*(N-1.00); //[-1,1]=> [0,2^8-1]  
}
```

Initialisation du port série - Affichage des signaux

```
Serial.begin(9600);
```

Initialisation des indices des tableaux

```
i_sin=0; // Déphasage de 0  
i_cos=Phi; // Déphasage de T/4
```



Affichage du signal sinus

```
Serial.print( SinCos[i_sin]);  
Serial.print(", "
```

Affichage du signal cosinus

```
Serial.println( SinCos[i_cos]);
```

Mise à jour des indices

```
i_sin++; i_sin=i_sin%(N-1);  
i_cos++; i_cos=i_cos%(N-1);
```

Programme Arduino

```
/*  
256 -----> 360  
x -----> 90  
-----  
x=90*256/360=256/4=N/4  
*/
```



```
#define N      256          // N=2^8, 8 bits
#define Phi    N/2         // 90*356/360
#define pi     3.1415926535897932384626433832795
#define f0     1.00

unsigned int i_sin=0,i_cos, i;
double sin_x, SinCos[N];

void setup()
{
  // Génération du signal sinusoidal en format 8 bits
  for(i=0;i<N;i++)
  {
    sin_x=sin(2.00*pi*f0*(float)i/((float)N-1.00));
    SinCos[i]=0.5*(sin_x+1.00)*(N-1.00); //[-1,1]=> [0,2^8-1]
  }
}
```



```
// Initialisation du port série - Affichage des signaux
Serial.begin(9600);

// Init indices des tableaux
i_sin=0;
i_cos=Phi;
}

void loop()
{
  // Affichage du signal sinus
  Serial.print( SinCos[i_sin]);
  Serial.print(",");

  // Affichage du signal cosinus
  Serial.println( SinCos[i_cos]);

  // Mise à jour des indices
  i_sin++; i_sin=i_sin%(N-1);
  i_cos++; i_cos=i_cos%(N-1);
}
```



```
}
```

Sujet connexes

- générateur de signaux
- générateur sinusoïdal avec [arduino](#)
- générateur signal carré arduino
- générateur de fréquence arduino
- générateur de fréquence arduino uno
- générateur de fonction avec arduino,
- transformer un signal carré en sinusoïdale
- générateur hf arduino

[Total : 1 Moyenne : 5/5]