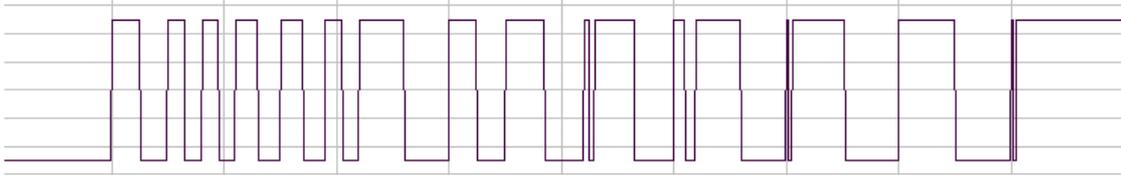


試験波形を作る その2

試験波形としてパルス列を生成する方法について説明します。

<パルスの立ち上がりデータ番号を設定し、デューティ比 50%で任意パルス波形を作る>
 例では、サンプリング周期=1ms、データ個数=1000点 (1sec) としています。



```

step 1 : サンプリング周期を定義する
def sampl_period 1e-3 "sec" /* サンプリング 1ms で定義 */

step2 : パルス列の元になる配列を用意する
assign $1 = 1000<0> /* 1000 点の 0 の配列を生成 */

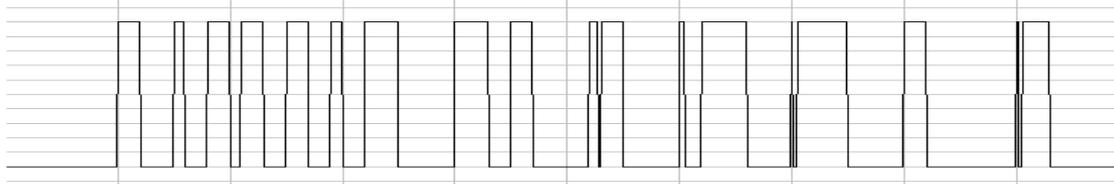
step3 : パルスの立ち上がりデータ番号配列を生成する
パルスの立ち上がり地点のデータ番号配列を作成します。作成に当たって、昇順並びの必要があります。
又、立ち上がり地点のデータ番号は、用意したパルス列の元配列のデータ番号範囲内の必要があります。
assign $2 = 100,150,180,210,250,290,320,400,450,520,530,600,620,700,705,800,900,905

step4 : パルス数分のパルス論理"1"配列を用意する
$3 = LEN($2) /* パルスの立ち上がりデータ個数を取得 */
assign $3 = $3<1> /* 立ち上がり個数分の論理"1"配列生成 */

step5 : パルス波形を生成する
パルス波形の生成は、埋め戻し関数(SBV 関数)を使用して、パルス元配列のパルス立ち上がり地点を"1"にした後、パルス間引き整形関数(TPC 関数)を使用してデューティ比を 50%に整形することで行います。
$4 = TPC(0,SBV($2,$3,$1))

step6 : パルス波形を格納する
def file_id %1 "パルス列" wav /* 格納ファイル名を定義 */
save wave %1 $4 /* データの格納 */
  
```

<パルスの立ち上がりと立下りデータ番号を設定して任意パルス波形を作る>
 例では、サンプリング周期=1ms、データ個数=1000点 (1sec) としています。



```

step 1 : サンプリング周期を定義する
def sampl_period 1e-3 "sec" /* サンプリング 1ms で定義 */

step2 : パルス列の元になる配列を用意する
assign $1 = 1000<0> /* 1000 点の 0 の配列を生成 */

step3 : パルスの立ち上がりデータ番号配列を生成する
① 立ち上がり地点と立下り地点の 2 種の配列を使用する方法
パルスの立ち上がり地点のデータ番号配列と、立下り地点のデータ番号配列を作成します。作成に当たって、昇順並びの必要があります。又、立ち上がり地点と立下り地点のデータ番号は交互に出現する必要があります。
assign $2 = 100,150,180,210,250,290,320,400,450,520,530,600,620,700,705,800,900,905 /*立ち上がり*/
assign $5 = 120,160,200,230,270,300,350,430,470,528,550,605,660,701,750,820,902,930 /*立ち下り*/
② 立ち上がりと立下りを一つの配列で生成した後、立ち上がりと立下りを分離して使用する方法
パルスの立ち上がりデータ番号と立下りデータ番号を一つの配列で定義します。
assign $2 = 100,120,150,160,180,200,210,230,250,270,290,300,320,350,400,430,450,470,530,528,530,550
  
```

```
,600,605,620,660,700,701,705,750,800,820,900,902,905,930  
次に、配列を立ち上がりと立下りに分離します。  
$5 = SEP(1,ERC(1,LEN($2)-1,$2)) /* 立下りデータ番号配列*/  
$2 = SEP(1,$2) /* 立ち上がり配列*/
```

step4 : パルス数分のパルス論理"1"配列を用意する

```
$3 = LEN($2) /* パルスの立ち上がりデータ個数を取得 */  
assign $3 = $3<1> /* 立ち上がり個数分の論理"1"配列生成 */
```

step5 : パルス波形を生成する

パルス波形の生成は、埋め戻し関数(SBV 関数)を使用して、パルス元配列のパルス立ち上がり地点を"1"に、パルスの立下り地点を-1にした後、累積関数(ACC 関数)を使用してパルス波形を生成します。

```
$4 = ACC(SBV($5,0-$3,SBV($2,$3,$1)))
```

step6 : パルス波形を格納する

```
def file_id %1 "パルス列" wav /* 格納ファイル名を定義 */  
save wave %1 $4 /* データの格納 */
```