스마트센서와 IoT

MATLAB 과제 #1

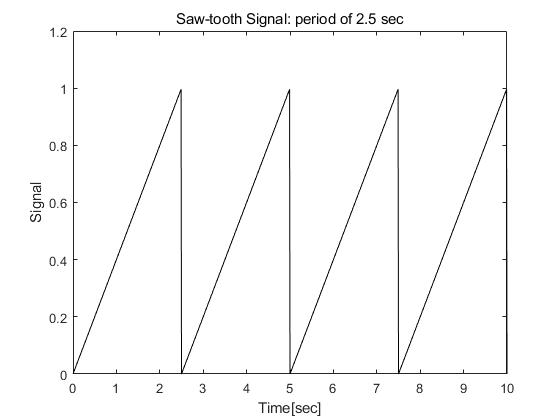
Problem 1. 조건에 맞는 데이터 수 세기

데이터 변수에 대하여 임의의 조건문을 입력했을 때 조건을 만족하는 데이터의 수를 출력하는 m-file 함수를 작성하라. 아래는 동작 예시이다.

|  |
| --- |
| Example |
| >> data = rand(5)  data =  0.4218 0.0357 0.7431 0.0318 0.6948  0.9157 0.8491 0.3922 0.2769 0.3171  0.7922 0.9340 0.6555 0.0462 0.9502  0.9595 0.6787 0.1712 0.0971 0.0344  0.6557 0.7577 0.7060 0.8235 0.4387  >>count\_var(data < 0.3)  ans =  7 |
| m-file function |
| function k=count\_var(data,number)  idx=find(data<number); %find안의 식을 만족하는 index 배열화  k=length(idx); %배열의 크기 측정  end  data= rand(5);  i=count\_var(data, 0.3);  disp(i) |

Problem 2. Saw tooth signal 만들기

1) 다음과 같은 plot을 그리는 script m-file을 작성하라.*(Hint. 함수rem( )을 사용할 것)*



|  |
| --- |
| m-file script |
| period=1.5; %주기  peak=2; %진폭  t=0:0.01:10;%time domain  my\_sawtooth(period,peak,t); %함수 |

2)period와 peak, time vector를 입력으로 받아 saw-tooth signal을 그려주고 데이터를 출력하는 function m-file을 작성하라. 다음은 동작 예시이다.

|  |
| --- |
| Example |
| >> period = 1.5; peak = 2; t = (0:0.01:10)’;  >> my\_sawtooth(period, peak, t); |
| m-file function |
| function my\_sawtooth(period,peak,t)  y=peak/period\*rem(t,(period)); %y값에 t에 진폭/주기를 나눈 나머지 값을 넣어준다.  plot(t,y);  xlabel('Time[sec]'); ylabel('Signal'); %x축과 y축 라벨링  title(['Saw-tooth Signal:period of ',num2str(period),' sec'] ); %plot창의 제목  len=length(t); %t의 길이  xmin=t(1); xmax=t(len); %x축의 최소값, 최대값 지정  ymin=0; ymax=peak+1; %y축의 최소값, 최대값 지정 (가시성을 위해 +1 하였음)  axis([xmin xmax ymin ymax]); %축의 범위 지정  end |

Problem 3. PWM 만들기

1)다음과 같은 plot을 그리는 script m-file을 작성하라.



|  |
| --- |
| m-file script |
| t=0:0.001:1;  dutyratio=0.6; frequency=5;  my\_pwm(dutyratio,frequency,t); |

2)Duty ratio와 frequency, time vector를 입력으로 받아 pwm signal을 그려주고 데이터를 출력하는 function m-file을 작성하라. 다음은 동작 예시이다.

|  |
| --- |
| Example |
| >> duty\_ratio = 0.6; frequency = 5; t = (0:0.001:1)’;  >> my\_pwm(duty\_ratio, frequency, t); |
| m-file function |
| function my\_pwm(dutyratio,frequency,t)  T = 1/frequency; %주기 계산  len=length(t); %t 배열 크기  y= zeros(1,len); %빈 행렬 생성  for i=1:len  if rem(t(i),T) <=(dutyratio\*T) %나머지의 비율이 주기의 duty ratio 보다 작을 시에 1을 반환하는 조건문  y(i)=1;  else  y(i)=0;  end  end  ymin=-0.1; ymax=1.2;  xmin=t(1); xmax= t(len);  plot(t,y);  dutyptg= dutyratio\*100; %duty ratio를 퍼센트 단위로 변환  xlabel('Time [sec]'); ylabel('Signal');  title(['Pulse Width modulation: duty ratio ',num2str(dutyptg),'% ',num2str(frequency),'Hz']);  axis([xmin xmax ymin ymax]);  end |