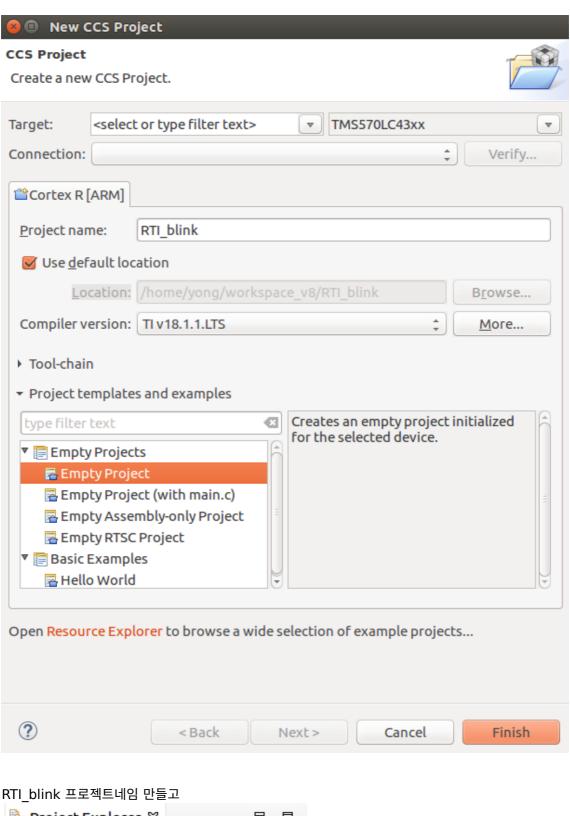
# TI DSP, MCU 및 Xilinx Zynq FPGA

프로그래밍 전문가 과정

강사 - Innova Lee(이상훈)
gcccompil3r@gmail.com
학생 - 하성용
accept0108@naver.com

### CCS 구동



RII\_blink 프로젝트데임 만들고

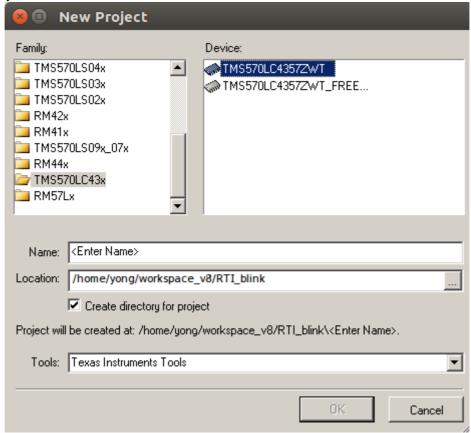
Project Explorer 없 □ □

E 및 ▼

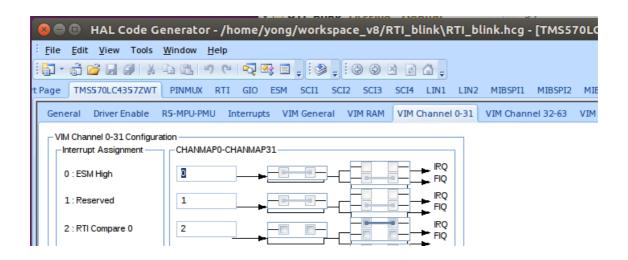
RTI\_blink [Active - Debug]

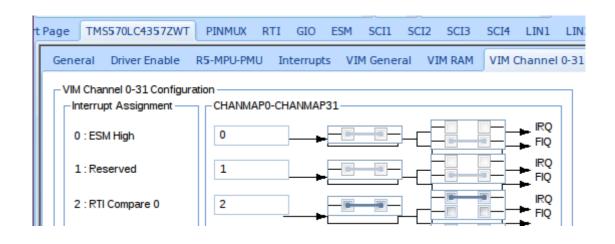
art+enter 들어가서 Resource 에 Location 경로 복사

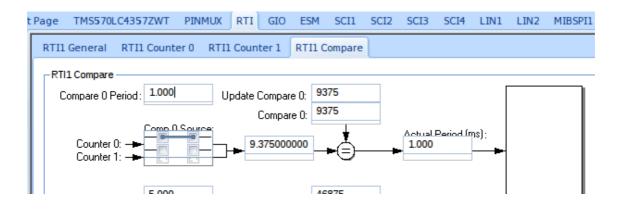
## HALCogen 구동 new project



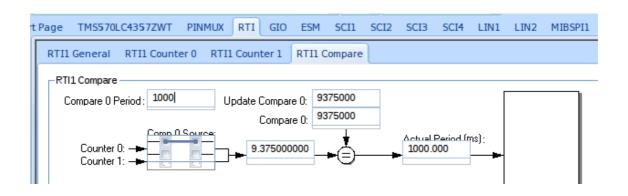
RTI\_blink 프로젝트네임 만들고 Location 에 복사한거 붙여놓고 create 체크해제하고 finish







1000 으로 바꿔서 1 초로바꾸고



file - 제너레이트코드 실행

# ccs 로 가서 HL\_sys\_main.c

```
_ _
Project Explorer 🛭
                                         main.c
                                                     🖟 *HL_sys_main.c 🛭
                                                                                                           41 ^
                            \nabla
                                          42 */
                                          43
   ▶ 🖟 HL_esm.c
                                          44
   ▶ 🖟 HL_etpwm.c
                                          45 /* USER CODE BEGIN (0) */
                                          46 /* USER CODE END */
   ▶ 🖟 HL_gio.c
   ▶ 🖟 HL_het.c
                                          48 /* Include Files */
   ▶ 🕝 HL_i2c.c
                                          50
   ▶ 🖟 HL lin.c
                                          51
                                          52 #include "HL_sys_common.h"
   ▶ 🖟 HL mdio.c
                                          54 /* USER CODE BEGIN (1) */
55 /* USER CODE END */
   ▶ 🖟 HL_mibspi.c
   ▶ 🖟 HL_nmpu.c
                                          56
                                          57 /** @fn void main(void)
   ▶ <a>B</a> HL notification.c
                                          58* @brief Application main function
59* @note This function is empty by default.
   ▶ <a>B HL_phy_dp83640.c</a>
                                          60 *
   ▶ 🖟 HL_phy_tlk111.c
                                          61* This function is called after startup.
62* The user can use this function to implement the applic
   ▶ 🖟 HL_pinmux.c
                                          63 */
   ▶ 🖟 HL_pom.c
                                          64
   ▶ 🖟 HL_rti.c
                                          65 /* USER CODE BEGIN (2) */
                                          66 /* USER CODE END */
   ▶ 🖟 HL_sci.c
                                                     omacAddrocc[611] - Javeell Aveell Aveell
   ▶ <a>B</a> HL_sys_core.asm
                                          60 min+0
   ▶ 🖟 HL sys dma.c
                                         ▶ S HL_sys_intvecs.asm
                                         1 error, 0 warnings, 0 others
   ▶ 🖟 HL_sys_link.cmd
                                         Description
                                                                                         Resource Path
▶ 🖟 HL sys main.c
#include "HL_sys_common.h"
#include "HL_system.h"
#include "HL_sys_core.h"
#include "HL_mibspi.h"
#include "HL_esm.h"
#include "HL_rti.h"
#include "HL_gio.h"
#include "HL_het.h"
추가
```

```
_ _
Project Explorer 🛭
                                                                              main.c
                                                                                                     🖻 *HL_sys_main.c 🛭
                                                      \nabla
                                                                               42 */
                                                                               43
     ▶ 🖟 HL_esm.c
                                                                               44
                                                                               45 /* USER CODE BEGIN (0) */
     ▶ 🖟 HL etpwm.c
                                                                               46 /* USER CODE END */
     ▶ 🖟 HL_gio.c
                                                                               48 /* Include Files */
     ▶ 🖟 HL_het.c
     ▶ 🖟 HL_i2c.c
                                                                               50 #include "HL_sys_common.h"
                                                                               51 #include "HL_system.h"
     ▶ 🖟 HL lin.c
     ▶ 🖟 HL_mdio.c
                                                                               54 #include "HL_sys_core.h" 55 #include "HL_mibspi.h"
     ▶ 🖟 HL mibspi.c
     ▶ 🖟 HL nmpu.c
                                                                               56 #include "HL_esm.h"
57 #include "HL_rti.h"
     ▶ 🖟 HL_notification.c
                                                                             58 #include "HL_gio.h"
59 #include "HL_het.h"
     ▶ 🖟 HL_phy_dp83640.c
     ▶ 🖟 HL_phy_tlk111.c
     ▶ 🖟 HL pinmux.c
                                                                               62 /* USER CODE BEGIN (1) */
                                                                               63 /* USER CODE END */
     ▶ 🖟 HL_pom.c
     ▶ 🖟 HL_rti.c
                                                                               65 /** @fn void main(void)
                                                                               66 *
                                                                                          @brief Application main function
     ▶ 🖟 HL sci.c
                                                                               67 *
                                                                                           @note This function is empty by default.
     ▶ S HL_sys_core.asm
     ▶ 🖟 HL_sys_dma.c
                                                                             ▶ <a>B</a> HL_sys_intvecs.asm
                                                                             1 error, 0 warnings, 0 others
     ▶ B HL sys link.cmd
                                                                              Description
                                                                                                                                                                      Resource
                                                                                                                                                                                                Path
     ▶ 🖟 HL_sys_main.c
                                                             _
Project Explorer 

□
                                                                                                      main.c
                                                      64
                                                                                65 /** @fn void main(void)
                                                                                66 *
                                                                                            @brief Application main function
     ▶ 🖟 HL esm.c
                                                                                67 *
                                                                                            @note This function is empty by default.
     ▶ 🖟 HL_etpwm.c
                                                                                68 *
     ▶ Ic HL gio.c
                                                                                69 *
                                                                                            This function is called after startup.
                                                                                70 *
                                                                                            The user can use this function to implement the applic
     ▶ 🖟 HL het.c
                                                                                71 */
                                                                                72
     ▶ 🖟 HL_i2c.c
                                                                                 73 /* USER CODE BEGIN (2) */
     ▶ I HL_lin.c
                                                                                74 /* USER CODE END */
     ▶ L mdio.c
                                                                                75
                                                                                                                                              {0xFFU, 0xFFU, 0xFFU, 0xFFU, 0
                                                                                76 uint8
                                                                                                     emacAddress[6U] =
     ▶ 🖟 HL_mibspi.c
                                                                                77 uint32 emacPhyAddress =
     ▶ 🖟 HL nmpu.c
                                                                                 79 int main(void)
     ▶ <a>B</a> HL notification.c
                                                                                80 {|
81 /* USER CODE BEGIN (3) */
     ▶ 🖟 HL_phy_dp83640.c
                                                                                82 /* USER CODE END */
     ▶ In the plane is a property of the plane is
                                                                                83
                                                                                             return 0;
     ▶ 🖟 HL pinmux.c
                                                                                85 }
     ▶ 🖟 HL_pom.c
                                                                                86
                                                                                87
     ▶ Ic HL_rti.c
                                                                                88 /* USER CODE BEGIN (4) */
                                                                                89 /* USER CODE END */
     ▶ Ic HL sci.c
     S HL_sys_core.asm
     ▶ 🖟 HL sys dma.c
                                                                              ▶ S HL sys intvecs.asm
                                                                              1 error, 0 warnings, 0 others
     BHL sys_link.cmd
                                                                              Description
                                                                                                                                                                        Resource
                                                                                                                                                                                                   Path
▶ li HL_sys_main.c
           rtiInit();
           gioSetDirection(hetPORT1,0xFFFFFFFF);
           rtiEnableNotification(rtiREG1, rtiNOTIFICATION_COMPARE0);
           enable IRQ interrupt ();
           rtiStartCounter(rtiREG1, rtiCOUNTER BLOCK0);
           while(1);
```

```
Project Explorer 

□
                                                           i main.c i *HL_sys_main.c ⋈ ii HL_reg_het.h
                                                                                                                                                              62/* USEK CUDE BEGIN (1) */
   ▶ 🖟 HL_ecap.c
   ▶ 🖟 HL_emac.c
                                                              64
65/** @fn void main(void)
66* @brief Application main function
67* @note This function is empty by default.
68*
   ▶ 🖟 HL_emif.c
   ▶ 🖟 HL_epc.c
   ▶ 🖟 HL_eqep.c
                                                               69* This function is called after startup.
70* The user can use this function to implement the application.
   ▶ 🖟 HL_errata.c
                                                               71 */
   ▶ 🖟 HL_esm.c
                                                               72
73/* USER CODE BEGIN (2) */
74/* USER CODE END */
   ▶ 🖟 HL_etpwm.c
   ▶ 🖟 HL_gio.c
                                                               76 uint8 emacAddress[6U] =
77 uint32 emacPhyAddress =
                                                                                                        {0xFFU, 0xFFU, 0xFFU, 0xFFU, 0xFFU, 0xF
   ▶ 🖟 HL_het.c
   ▶ 🖟 HL_i2c.c
                                                               79 int main(void)
   ▶ 🖟 HL_lin.c
                                                              80 {
81/* USER CODE BEGIN (3) */
82 rtiInit();
   ▶ 🖟 HL_mdio.c
                                                             81 * USER CODE BEGIN (3) */
82     rtiInit();
83
84     gioSetDirection(hetPORT1,0;
85     rtiEnableNotification(rtiR:
87
88     _enable_IRQ_interrupt_();
89
90     rtiStartCounter(rtiREG1, r:
91
92     while(1);
93
94 /* USER CODE END */
   ▶ 🖟 HL_mibspi.c
                                                                      gioSetDirection(hetPORT1,0xFFFFFFFF);
   ▶ 🖟 HL nmpu.c
   ▶ 🖟 HL_notification.c
                                                                      rtiEnableNotification(rtiREG1, rtiNOTIFICATION_COMPARE0);
   ▶ 🖟 HL_phy_dp83640.c
   ▶ 🖟 HL_phy_tlk111.c
                                                                    rtiStartCounter(rtiREG1, rtiCOUNTER_BLOCK0);
   ▶ 🖟 HL_pinmux.c
   ▶ 🖟 HL_pom.c
   ▶ 🖟 HL_rti.c
                                                               94 /* USER CODE END */
   ▶ 🖟 HL_sci.c
                                                                      return 0;
   ▶ <a>В HL_sys_core.asm</a>
   ▶ 🖟 HL_sys_dma.c
   ▶ S HL_sys_intvecs.asm
                                                             100 /* USER CODE BEGIN (4) */
101 /* USER CODE END */
   ▶ → HL sys link.cmd
▶ 🖟 HL_sys_main.c
```

```
rtiInit();
gioSetDirection(hetPORT1,0xFFFFFFFF);
rtiEnableNotification(rtiREG1, rtiNOTIFICATION_COMPARE0);
_enable_IRQ_interrupt_();
rtiStartCounter(rtiREG1, rtiCOUNTER_BLOCK0);
while(1);
```

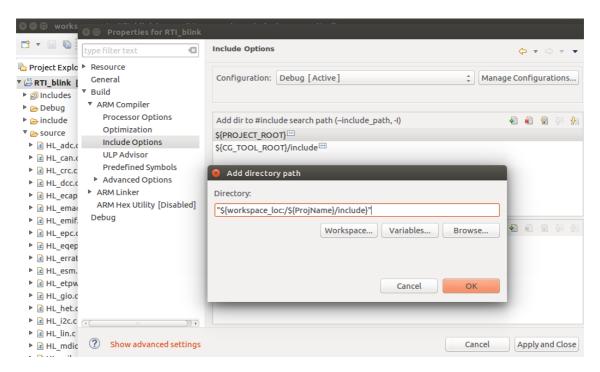
```
Project Explorer 

□
                                              79 /* USER CODE BEGIN (3) */
▼ 🎏 RTI_blink [Active - Debug]
                                                    rtiInit():
▶ ﷺ Binaries
                                                    gioSetDirection(hetPORT1.0xFFFFFFFF):
82
                                              83
 Debua
                                                   rtiEnableNotification(rtiREG1, rtiNOTIFICATION_COMPARE0);
                                              85
▶ (⇒ include)
                                                   _enable_IRQ_interrupt_();
 ▼  source
                                                   rtiStartCounter(rtiREG1, rtiCOUNTER_BLOCK0);
                                              88
  ▶ 🖟 HL adc.c
  ▶ 🖟 HL_can.c
                                                    while(1);
  ▶ I HL_crc.c
                                              92 /* USER CODE END */
  ▶ 🖟 HL_dcc.c
                                              94
  ▶ <a>le HL_ecap.c</a>
  ▶ L emac.c
                                              96
                                              97 void rtiNotification(rtiBASE_t *rtiREG, uint32 notification)
  ▶ 🖟 HL_emif.c
                                                    gioSetPort(hetPORT1, gioGetPort(hetPORT1) ^ 0x000000001);
  ▶ L epc.c
                                              99
                                             100 }
▶ 🖟 HL_eqep.c
                                             101
  ▶ I HL_errata.c
                                             102
                                             103 /* USER CODE BEGIN (4) */
104 /* USER CODE END */
  ▶ 🖟 HL_esm.c
                                             105
  ▶ 🖟 HL_etpwm.c
```

```
void rtiNotification(rtiBASE_t *rtiREG, uint32 notification)
{
    gioSetPort(hetPORT1, gioGetPort(hetPORT1) ^ 0x00000001);
}
```

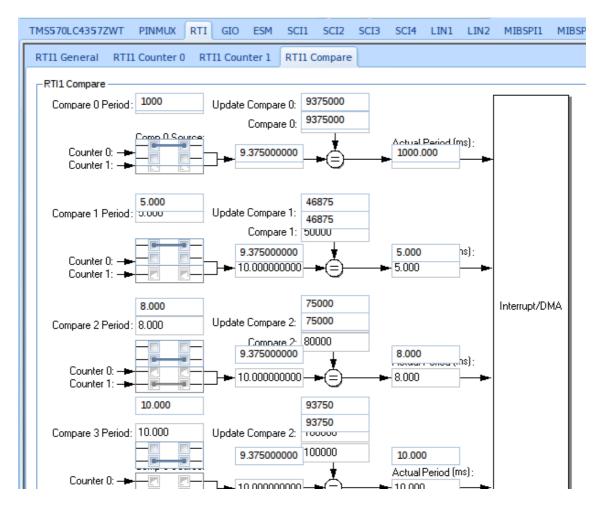
ccs 코드입력하고

바로 망치때리면안되고



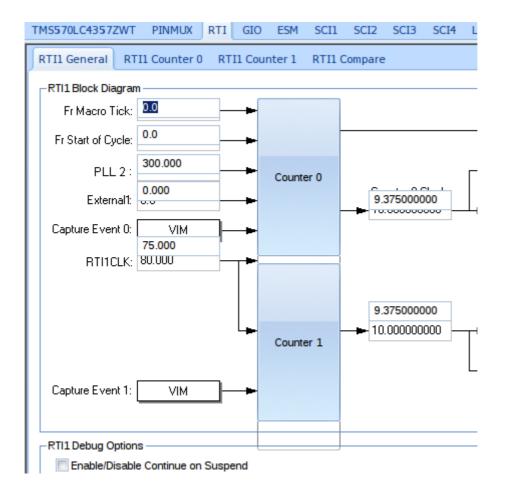
RTI\_blink 우클릭해서 프로폴티 ARM Compiler - Include Options 상단 Add 부분에 +누르고

Directiroy : "\${workspace\_loc:/\${ProjName}/include}" closed apply 하고 다시 망치



9.375 가 걸리는 시간

T = F 분의 1 = 9.375 분의 1 = 9375 분의 1000 = 0.106 ->주기



Mhz 이니 9.375 에 10 의 6 승

```
0.0000001067 * 9375 = 0.0010003125
1 밀리세컨드
1000 곱하기 해주면
1.0003125
```

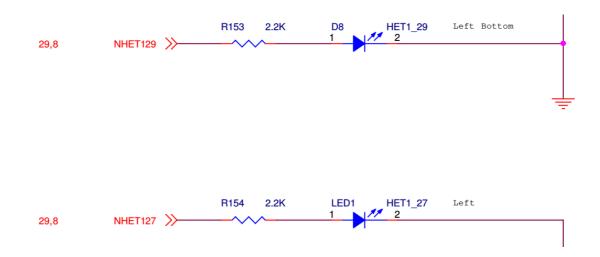
즉 T = f분에 1 = 9375000분에 1 = 1.06(6위에 .) \* 10에 -7승

컴페어값이 9375 이니까 0.0001067 곱하기 9375 를 하면 1.0003125 = 0.000000106 ---- ---- ms us ns

```
HL sys main.c
                                                                                                                  In the last of the last of
                                                  rtiREG1->CMP[3U].UDCPx = 93750U;
    133
    134
    135
                                                   /** - Clear all pending interrupts */
    136
                                                 rtiREG1->INTFLAG = 0x0007000FU;
    137
                                                   /** - Disable all interrupts */
    138
                                                 rtiREG1->CLEARINTENA = 0x00070F0FU;
    139
    140
     141
                                                                                       @note This function has to be called before the driver can
    142
                                                                                                                            This function has to be executed in privileged mode.
    143
                                                                                                                            This function does not start the counters.
   144
```

### **CLEARINTENA**

// 타임베이스가 꺼지는것도 디스에이블 가운데 F 는 0 번부터 3 번 앞 F 는 8 번부터 1 번



# gioGetPort 현재이포트에 들어와있는지 안있는지 있으면 1로 켜지고 0 이면 꺼져있음

실습문제

- 1. N2HET 을 활용해 LED 8 개 전부 켜기
- 2. LED를 뱀처럼 1개 키고 그 다음거 키고(2개 켜짐) 마지막에 8개 키고 다시 반복
- 3. LED 를 순서대로 1 개씩 켜기(다음거 킬때 기존에 켜져 있던 것이 꺼져야함)
- 4. 랜덤으로 LED 가 켜지게됨
- 2, 3, 4 번은 rti 값 1000 을 125 로 바꿔서하면 8 번도는데 1 초걸림

```
void rtiNotification(rtiBASE_t *rtiREG, uint32 notification)
{
    gioSetPort(hetPORT1, gioGetPort(hetPORT1) ^ 0x00000001);
}
```

### 뱀꼬리물기

맨처음에 17을 킨다고 켰을때

- ^ 0x00020000(2 번)
- ^ 0x80000000(31 번)
- ^ 0x0000001(0 번)
- ^ 0x00000020(4 번)-12 15
- ^ 0x02000000(25 는?) 16-
- ^ 0x00040000(18 번) → 14-17,11-13, 8-11,4-7,0-3
- ^ 0x20000000
- ^ 0x08000000(8 이 나오고 다 꺼야함)
- ^ 0xAA060021 (다꺼지기)

# 루프를 8 번만해도 9 번돌수있게

- ^ 0x00020000
- ^ 0x80020000 // 서로 상쇄되니까 전에껀꺼짐 XOR 이니까
- ^ 0x80000001
- ^ 0x00000021
- ^ 0x00000020
- ^ 0x02000020
- ^ 0x02040000
- ^ 0x20040000
- ^ 0x28000000
- ^ 0x08000000

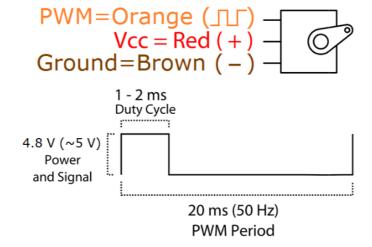
랜덤함수는? 전역변수가있어야함 array 8 개 array[9] = {} 8 개가 다켜지는상황에만 index 가지고 랜덤돌리기

비트파악하고 돌리는게 핵심

NHET 회로

아두이노 서버컨트롤

sg90 datasheet -



Position "0" (1.5 ms pulse) is middle, "90" ( $\sim$ 2 ms pulse) is all the way to the right, "-90" ( $\sim$ 1 ms pulse) is all the way to the left.

9 번이 pwn 연결 브라운은 5 번

등속원운동 미분이란건 고딩때 속도에 대한 미분은 가속도 위치에 대한 미분은 속도 가속도에 대한 미분은 저크 저크는? 가속도에 변화량 가속도에 변화량이 무엇에 영향을 줌? 가속도가 변하면 전자기파가 발생하게됨

전자기학 후반부에 나오는게 저크

미분의 기하학적 의미 접선의 기울기

실질적으로 의미있는해석으로 받아들이려면 신호처리 관점에서 미분이란?

아날로그방식에서 쩜찍고 쩜찍었을때 여기가 x 여기가 x+델타

리밋 델타 x→0

dt 분에 dx

한없이 0 에 가까운 순간을 만들어낼수있게하는

컴퓨터는 할수없음 0은 만들수있어도 0에 가까운숫자를 표현할 방법이없음

9.375Mhz = 0.000000106 100 나노세컨드 어떤값을 측정하는데 걸리는시간이 100 나노세컨드라는소리 엄밀하게는 극한값이 아님

0.00000106 → 샘플링타임

첫번째 계측값

두번째 계측

델타 x 가 무엇이 될까 샘플타임이 되면되는거 샘플타임->줄여서 st 로 약칭

st 분에 f(x+st)-f(x) 신호처리 관점에서의 미분

--개요설명한거

코드를 짜기전의 워밍업

원운동을 다시와서

ex)토르 망치 어떤 속도를 가지고 움직이게될거임 그 속도를 가지고 도니까 같은 분속이니까 무엇을 가지게됨? 등속. 한바퀴도는시간이 일정함 즉 주기가 일정 원주공식 2 파이 r 2 파이 r(원의둘레)을 속도로 나누면 주기가되는거 주기분에 2 파이 r

진동수에 역수가 주기

두개의 값을 잡아놓고 속도는 시간에 따른 위치의 변화량 dt 분에 dx 시간에 따른 각도의 변화량이 각속도 즉 회전속도 각속도가 높다는건 회전속도가 빠르다는소리

가속도의 정의는 시간에 따른 속도의 변화량 속도가 얼마나 빨리 변하는가에 따라 가속도가 결정

시간에 따른 각속도는 각가속도가됨 각속도를 계속가속시킬수있는 매커니즘이있다는소리

ex)박스 차기

dz

F=ma 를 m —

dt

일직선운동에선 f 가 속도를 만들어내는 원천이되는데 회전운동에선 토크가 회전속도(각가속도)를 만들어냄

박스 정지해있다가 발로차서 속도를 가지게됨 ex) 펜을 수직으로 던졌을때 회전을하게됨

각속도는 w = 오메가 가 단위 확장해서보면 한바퀴를 돌때 전체 주기분에 2 파이가 각속도가 표현해진다고볼수있음

s= vt // 이동거리공식, r0 //r 세타

아주 약간의 각도면 거의 직각이되는데 r sin 0

r 0 sin0 r

여기서 테일러곱수가 쓰임

lim  $\sin 0 = 0 \leftarrow$ ----테일러곱수  $0 \rightarrow 0 \quad --\rightarrow r \ 0 (r \ 세타가됨)$ 

테일러곱수 전제조건은 무한미분이 가능해야함

무한미분가능한 함수가 대표적으로 사인 코사인 익스포넨셜?

즉 이걸로 MPU 6050 으로 우리가 해야함 칼만필터를 입혀야함

각도에 변화량 측정하면 각속도를 알수있음

각도를 미분한 각속도도

적분한속도 미분한속도 둘다 100 퍼신뢰는안됨 오차확률가지고있음 미분한 확률분포 서로 종속되있음 종속된 가우시안분포 학교수업에선 독립된 정규분포만 보았을것 적분한결과도 속도 미분한결과도 속도 같은 사건을 얘기하고있으니 이건 서로 종속되었다

두개의 합성된 정규분포를 가지고 계산하면 결과값이 상당히 잘일치함 즉 그 결과값으로 제어를하면 상당히 잘돌아가게됨

가속도는? 시간에 따른 속도의 변화량 dt 분에 d 오메가 합성함수미분 dt 분에 dr 과 w  $\rightarrow$ v (속도?) r 과

rw+0 = v와 r분에 v = r분에 v 제곱 = rw 제곱=r분에 v 제곱 v=rw => w=r분에 v

[T(토크)=I(아이)a(알파)] - 회전 [F=ma] - 병진 (직선)