

각각의 문제에 대해 TI에서 제공하는 sysconfig와 C-code로 구현하라.

최종 평가는 구현 결과의 동작 여부와 각 문제를 어떻게 해결하였는지 발표자료 작성 및 발표를 통해 평가를 한다.

최종 평가일정은 12/17일 14:00부터 평가할 예정이고, 12/10일 13:00부터 실험실에서 조별로 과제 준비를 할 수 있도록 제공할 예정이고, 조별로 질문을 받을 예정.

시스템 기본 동작은 다음과 같다.

- 1) SYSCLK을 120MHz
- 2) Main Loop : 20ms using Polling Timer 0
- 3) Sub Loop : 1ms using Interrupt Timer 1

1. 시스템이 동작되고 있음을 표현할 수 있도록 RED LED(LED4)와 YELLOW LED(LED5)가 500ms마다 서로 다르게 Toggle 되도록 설정하라. (5점)
 - A. RED LED(LED4) : ON→OFF→ON→OFF
 - B. YELLOW LED(LED5) : OFF→ON→OFF→ON
2. Debugging을 위해 CAN Tx, Rx Message를 각각 아래와 같이 설정한다.(5점)
 - A. CAN 통신 Bit rate : 500kbps
 - B. Debugging용 CAN Tx Message : 3개
 - i. ID : 0x100, 0x110, 0x120
 - ii. 송신 주기 : 20ms
 - iii. Data Length는 8 Bytes
 - C. Debugging용 CAN Rx Message : 2개
 - i. ID : 0x700, 0x710
 - ii. Data Length는 8Bytes

3. PWM Port와 부저를 이용하여 다음의 악보를 연주해보자. (20점)

- A. 작은별, 4분의 2박자 → 한 마디가 0.5초 기준, 4 옥타브 기준

작은 별

윤석중 요
모짜르트곡

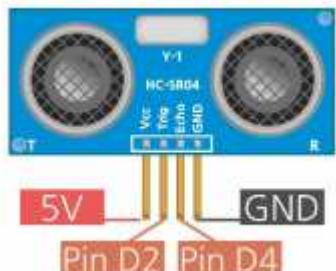
The musical score consists of four staves of music. Each staff has a G-clef, a 2/4 time signature, and a 4-octave range. The lyrics are written below each staff. The first staff starts with '반짝' (C), followed by '반짝' (F), '작은' (C), '별' (F), '아름' (C), '답게' (C). The second staff starts with '비치' (G), followed by '네' (C), '동쪽' (C), '하늘' (F), '에서' (C), '도' (G). The third staff starts with '서쪽' (C), followed by '하늘' (F), '에서' (C), '도' (G), '반짝' (C), '반짝' (C). The fourth staff starts with '작은' (F), followed by '별' (C), '아름' (F), '답게' (C), '비치' (G), '네' (C). The score ends with a double bar line.

4. ADC를 2CH 사용할 수 있도록 설정하고, 각 CH에 입력되는 신호의 전압 값을 읽고 그 값을 Debugging용 CAN으로 송신한다. (20ms 주기) (20점)

- A. System_loop_counter는 20ms마다 1씩 증가하도록 설정한다.
- B. 1CH : ADC Result → 전압으로 변환→ 소수점 첫째자리까지
- C. 2CH : ADC Result → 전압으로 변환→ 소수점 둘째자리까지
- D. 0x100 Data
- [0]~[1] : system_loop_counter
 - [2]~[3] : ADC 1 CH 결과
 - [4]~[5] : ADC 2 CH 결과
 - 소수점 아래 숫자를 보내는 방법을 제시하라.

5. 초음파 센서를 이용하여 거리를 측정해보자. (30점)

A. 초음파 센서



Pin D2는 Trigger 신호 → GPIO → Trigger신호는 20us 유지

PIN D4는 Echo 신호 → ECAP

B. 거리 연산



$$\text{속력} = \frac{\text{거리}}{\text{시간}}, \quad \text{시간} = \frac{\text{거리}}{\text{속력}}, \quad \text{거리} = \text{속력} \times \text{시간}$$

$$\text{왕복 거리} = \text{음속} \times \text{왕복 시간}$$

$$\therefore \text{물체와의 거리} = \frac{\text{음속} \times \text{왕복 시간}}{2}$$

C. 0x110 Data

- i. [0] ~ [1] : 측정 거리, mm단위로

6. CAN 수신 결과에 대한 연산결과 Return하기 (20점)

A. X1, X2, X3, X4와 Y1, Y2, Y3, Y4의 Data는 모두 16bit의 데이터이며, 각각의 데이터는 CAN 통신을 통해 전송된다.

B. 0x700 Data

- i. [0] : X1 Low Byte
- ii. [1] : X1 High Byte
- iii. [2] : X2 Low Byte
- iv. [3] : X2 High Byte

- v. [4] : X3 Low Byte
- vi. [5] : X3 High Byte
- vii. [6] : X4 Low Byte
- viii. [7] : X4 High Byte

C. 0x710 Data

- i. [0] : Y1 Low Byte
- ii. [1] : Y1 High Byte
- iii. [2] : Y2 Low Byte
- iv. [3] : Y2 High Byte
- v. [4] : Y3 Low Byte
- vi. [5] : Y3 High Byte
- vii. [6] : Y4 Low Byte
- viii. [7] : Y4 High Byte

D. 연산 결과 Return 하기 0x120 Data

- i. [0] ~ [1] : X1+Y1
- ii. [2] ~ [3] : X2-Y2
- iii. [4] ~ [5] : X3*Y3
- iv. [6] ~ [7] : X4/Y4 → 소수점 둘째짜리까지 보내기