
임베디드 소프트웨어

- 빛의 세기에 따른 노래의 speed 조절 및 온도 표시 -



학 과 : 컴퓨터 공학과

학 번 : 12161601

제출일 : 2020.12.09

이 름 : 소찬균

1. 기 능

가. 빛의 세기에 따른 노래의 speed 조절

- ✓ 광센서를 이용하여 광량을 측정한다.
- ✓ 측정된 ADC 값을 100으로 나누어 값이 8보다 크면 1배속, $5 < \text{ADC값} \leq 8$ 일 경우 1.2배속, 5미만일 경우 1.4배속으로 speed를 조절한다.
- ✓ DO~UTI의 사이의 값을 8등분하여 LED에 표시한다. 음이 높아질수록 TCNT 값이 커지게 되는데 대략 TCNT 초기값이 21 증가할 때마다 한단계씩 올라간다.
- ✓ buzzer를 통해서 총 3개의 노래를 연주한다.
- ✓ 노래의 speed를 세그먼트에 출력한다.
- ✓ INT4_vect에 해당되는 버튼을 누를 시 다음 곡이 재생된다. (만약 현재 재생중인 노래가 마지막 곡일 경우 첫번째 곡이 재생된다.)
- ✓ 하나의 곡이 완주되면 자동으로 다음곡이 재생된다.(현재 재생이 끝난 노래가 마지막 곡일 경우 첫번째 곡이 재생된다.)

나. 온도 표시

- ✓ 온도 센서를 이용하여 온도를 측정한다.
- ✓ 해당 온도를 세그먼트에 출력한다.
- ✓ INT4_vect에 해당되는 버튼을 누를 시 현재 온도를 기준으로 warm or cold로 표시한다.(다시 누를 시 원래대로 온도를 출력)
- ✓ 20도 초과일 경우 warm, 이하일 경우 cold를 출력한다.
- ✓ 0을 최저온도, 40을 최고온도로 하여 8등분하여 각 레벨에 맞게 LED를 출력한다.(5도 증가할때마다 출력되는 LED가 하나씩 증가)

다. 기능 간 전환

- ✓ INT5_vect에 해당되는 버튼을 클릭할 경우 두 기능 간 전환이 일어난다.

2. Task

가. 빛의 세기에 따른 노래의 speed 조절

1) readLightTask

- ✓ 광센서를 이용하여 현재 ADC값을 읽어온다.
- ✓ 이후 event flag를 통해서 PlaySongTask와 showSpeedTask에게 측정이 완료되었음을 알린다.

2) PlaySongTask

- ✓ 지정된 노래를 연주한다.
- ✓ 광량에 따라서 위에서 언급한 것과 같이 speed가 달라지도록 설정되어 있으며 총 3개의 음악을 연주한다.(징글벨, 고요한 밤, 창밖을 보라)
- ✓ 자동 재생이며 INT4에 해당되는 핀에 인터럽트가 발생하면(버튼을 누를 시) 다음 곡으로 넘어간다.
- ✓ 현재 지정된 음정을 메일박스를 통해서 showToneLEDTask에게 전달한다.

3) showSpeedTask

- ✓ 현재 몇 배속으로 연주하고 있는지 세그먼트에 출력한다.

4) showToneLEDTask

- ✓ 메일 박스를 이용하여 PlaySongTask가 전달한 음정을 받고 이에 반응하는 LED들을 출력한다.

나. 온도 표시

1) TempartureTask

- ✓ 현재 온도를 읽어온다.
- ✓ event flag를 통해서 나머지 Task들에게 측정이 완료되었음을 알린다.

2) showTempTask

- ✓ 온도 측정이 완료되었음을 확인한 후 세그먼트에 현재 온도를 출력한다.

3) showWarmOrColdTask

- ✓ 온도 측정이 완료되었음을 확인한 후 20도 초과일 경우 warm, 20도 이하일 경우 cold를 출력한다.

4) showTempLEDTask

- ✓ 온도 측정이 완료되었음을 확인한 후 0도를 최저온도, 40도를 최고온도로 하여 8등분하여 현재온도에 반응하는 LED들을 출력한다.

3. Synchronization

가. 빛의 세기에 따른 노래의 speed 조절

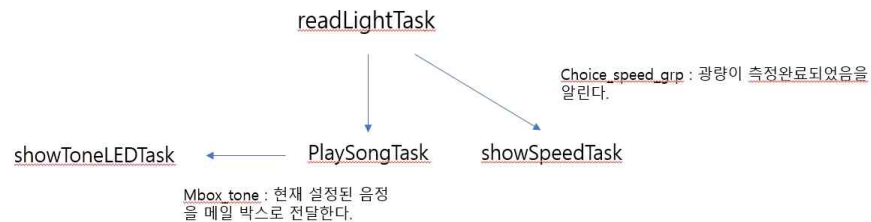


그림 2

- ✓ choice_speed_grp(event flag)를 통해서 PlaySongTask와 showSpeedTask에게 광량 측정이 완료되었음을 알린다. (측정완료 시 0x03으로 flag를 setting한다.)
- ✓ PlaySongTask에서는 0x01이 setting되길 기다리며, showSpeedTask에서는 0x02가 setting되길 기다린다.
- ✓ showToneLEDTask에서는 메일 박스를 통해서 PlaySongTask가 전달한 음정 정보를 받는다.

나. 온도 표시

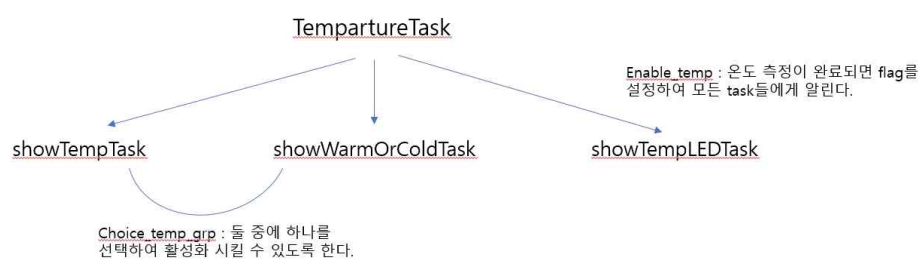


그림 3

- ✓ enable_temp(event flag)를 통하여 온도 측정이 완료되었음을 task들에게 알린다.
- ✓ 온도 측정이 완료되면 flag를 0x07로 setting한다.
- ✓ showTempTask에서는 enable_temp가 0x01로 setting되길 기다리며, setting이 완료되면 세그먼트에 온도를 출력하게 된다.
- ✓ showWarmOrColdTask에서는 enable_temp가 0x02로 setting되길 기다리며, setting이 완료되면 세그먼트에 warm 혹은 cold를 출력하게 된다.

- ✓ showTempLEDTask에서는 flag가 0x04로 setting되길 기다리며, 측정된 값을 가지고
상응하는 LED들을 출력한다.
- ✓ showTempTask와 showWarmOrColdTask간에는 choice_temp_grp(event flag)를 통하
서 동기화시켜준다.
- ✓ choice_temp_grp가 0x01로 setting될 경우 showTempTask가 block해제되고,
choice_temp_grp가 0x02로 setting될 경우 showWarmOrColdTask가 block해제된다.
따라서 둘 중에 하나의 Task만이 동작하도록 한다.

다. 두 기능 간 동기화

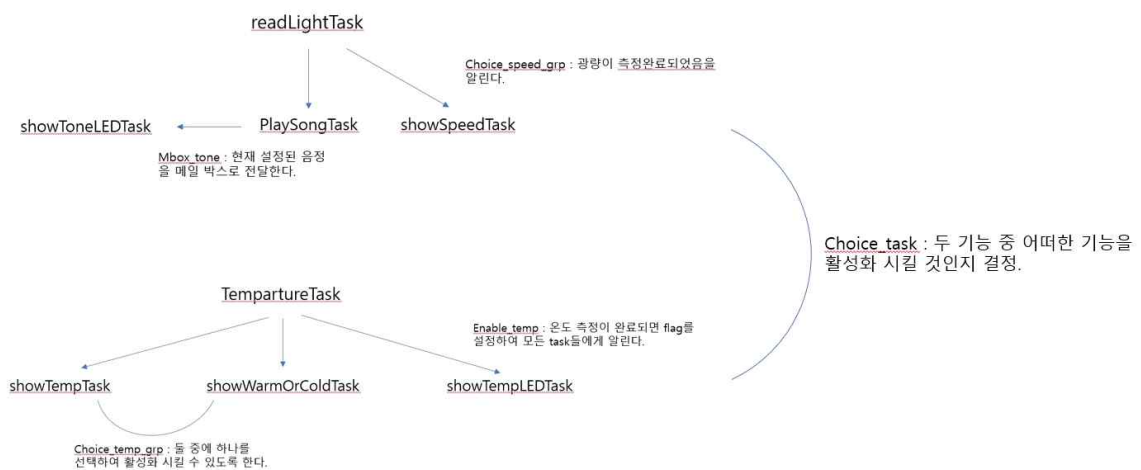


그림 4

- ✓ 위에서 언급한 모든 task에는 choice_task라는 eventflag가 있다.
- ✓ flag가 0x01로 설정될 시 온도 조절에 따른 speed조절과 관련된 기능이 수행되며,
0x02로 설정될 경우 온도 측정 기능이 수행된다.

4. 결과 및 고찰

기존에 계획에서는 timer/counter 0,2를 동시에 사용하여 노래 제목 출력 기능, 주기적인 warm or cold와 온도표시 기능을 구현하려고 하였다. 하지만 timer/counter 0가 task간의 스케줄링에 사용되어 timer/counter 0를 사용할 수 없게 되면서 이 기능들을 구현하지 못하였다. 따라서 노래 제목 대신에 현재 노래의 배속을 출력하였으며, INT4에 해당하는 버튼을 이용해서 현재 온도와 warm or cold 출력 간 전환을 구현하였다.

또한 전역 변수를 사용함으로써 mailbox와 message queue의 사용이 불필요해지며 기존에 mailbox와 message queue를 사용하고자 했던 부분을 eventflag로 동기화하였다.

가. 버튼 오작동 문제

버튼들을 누를 시 task 간 전환이 잘 안되는 경우가 있다. 예를 들어 '징글벨' 다음곡은 '고요한 밤'인데 2번 눌린것으로 인식되어 '창밖을 보라'가 실행되는 경우가 발생한다. 또는 버튼이 눌렸으나 제대로 인식되지 않아 노래가 바뀌지 않는 경우도 있었다.

온도 측정 기능 에서는 버튼을 누를 시 문자열 표시와 숫자 표시 기능 간 전환이 발생하지 않는 경우도 있었다.

이러한 문제점들은 debouncing이 제대로 되지 않아서 발생하는 것으로 추측된다. 하지만 debouncing을 더욱 확실하기 위해 delay를 현재보다 더 오랜 시간을 부여할 경우 task들의 수행에 영향을 주게 된다.

나. task 증가에 따른 각각의 task 실행시간 감소

광량에 따른 노래의 speed 조절의 경우 세그먼트도 출력해야하고 노래도 동시에 출력을 해야 하다보니 task전환을 빠르게 해야하면서 문제가 발생하였다.

세그먼트의 경우 노래 출력 task 실행을 위해 delay를 좀 높게 주다보니 잔상효과가 덜해지는 문제점이 발생하였다.(세그먼트가 좀 흐리게 보임)

노래 출력 task의 경우 우선순위가 가장 낮아(노래 출력 기능과 연관된 task들에서의 우선순위) 나머지 task들을 모두 수행하고 최종적으로 실행되다보니 음이 늘어지는 경우가 발생하였다. 따라서 이 부분을 보정하여 원래의 노래 속도를 맞출 수 있도록 하였다.