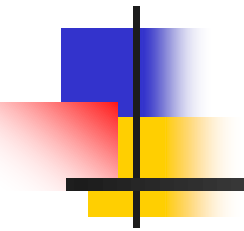
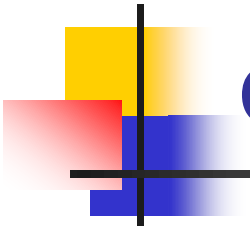


Embedded Software
(MicroC/OS-II: 3)

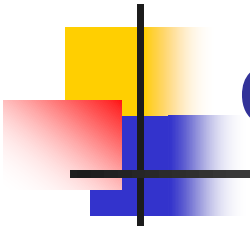
Mailbox and message queue

Fall, 2017



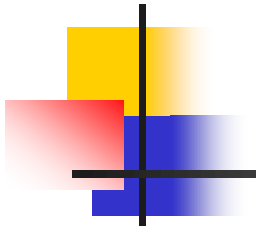
Outline

- Overview
- What to do?
  - 숫자 게임



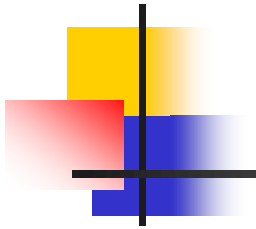
Overview

- 목적
  - Mailbox message queue 활용하기



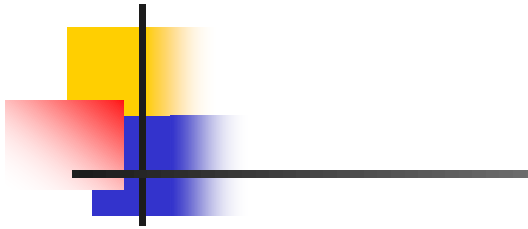
- Mailbox 초기화
  - OS_EVENT *OSMboxCreate(void *value)
    - value – 초기 Mailbox의 값

https://doc.micrium.com/display/osiidoc/Message+Mailbox+Management



- Mailbox 데이터 송신
  - INT8U OSMboxPost(OS_EVENT *pevent, void *msg)
    - Mailbox pevent에 msg 전달
    - pevent – OSMboxCreate()에 의해서 만들어진 Mailbox 정보
    - msg – 전달하려는 정보

https://doc.micrium.com/display/osiidoc/Message+Mailbox+Management

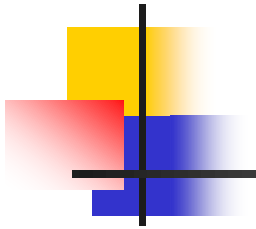


- Mailbox 데이터 수신
  - void *OSMboxPend(OS_EVENT *pevent, INT16U timeout, INT8U *err)
    - pevent – OSMboxCreate()에 의해서 만들어진 Mailbox 정보
    - timeout – 대기시간(클럭 단위, 0: 무한)
    - err – 에러코드
  - 주의
    - 리턴 값으로 0 받을 시, uC/OS-II가 멈출 수 있음
    - If the return value is 0, uC/OS-II could be freezed

https://doc.micrium.com/display/osiidoc/Message+Mailbox+Management

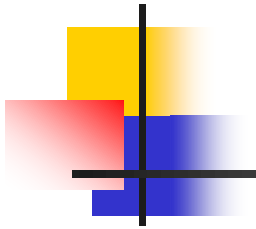
- Message Queue 초기화
  - OS_EVENT *OSQCreate(void **start, INT8U size)
    - start – Data를 저장하기 위한 저장공간의 시작주소
    - size – 저장할 수 있는 데이터의 개수

https://doc.micrium.com/display/osiidoc/Message+Queue+Management



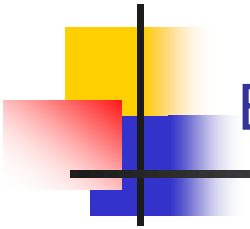
- Message Queue 데이터 송신
  - INT8U OSQPost(OS_EVENT *pevent, void *msg)
    - pevent – OSQCreate()에 의해서 만들어진 Queue 정보
    - msg – 전달하려는 정보

https://doc.micrium.com/display/osiidoc/Message+Queue+Management



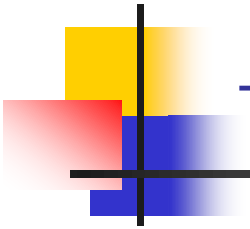
- Message Queue 데이터 수신
  - void *OSQPend(OS_EVENT *pevent, INT16U timeout, INT8U *err)
    - pevent – OSQCreate()에 의해서 만들어진 Queue 정보
    - timeout – 대기시간 (클럭 단위, 0: 무한)
    - err – 에러코드

https://doc.micrium.com/display/osiidoc/Message+Queue+Management



Before implementation, you have to…

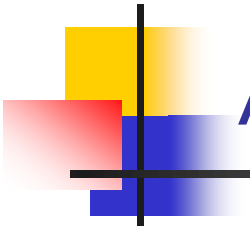
- OS_CFG.h 수정
- Change values in OS_CFG.h
  - #define OS_MAX_EVENTS 10
  - #define OS_MAX_QS 10



There are some hints

- Msg 인자로 값을 직접 넘길 수 있으나, 0값을 넘기게 되면 프로그램이 멈출 수 있음.
  - Passing value itself via msg parameter sometimes works. However, when you pass 0 value, the program could freezes.
- 각 Task 내에서 전달(Post)할 때, 전달하고자 하는 변수의 “주소”를 넘길 것.
  - When you use post function, it’s fine to pass the address of the variable.
- Make Google your best friend.
- Feel free to ask through I-Class.

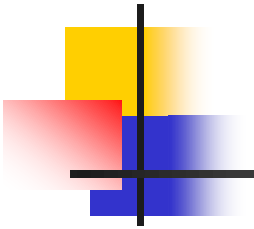
Assignments



- I-Class에 다음 슬라이드의 과제 제출
- 제출 파일
  - 2 x 소스 코드(EX1.C) – Mailbox, MessageQueue
  - 보고서(.pdf)
  - 실행 결과 화면 캡쳐
  - You have to capture and submit the screen on your report
- 파일명
  - 5주차_학번_이름(압축하여 하나의 파일로 제출)
- 제출 기한
  - 다음주 화요일 자정까지

What to do ?

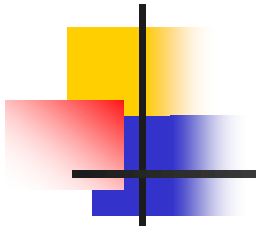
- 1. 5개의 태스크 생성
  - 1개의 Decision task
  - 4개의 Random task
- 2. 4개의 Random task 에서 임의의 숫자를 0부터 63까지 1개 생성할 것
- 3. 4개의 Random task는 해당 숫자를 Decision task 로 보낸다.
  - Mailbox, message queue 각각 활용 2개 파일 제출
- 4. Decision task 는 4개의 태스크 중에서 가장 작은 숫자를 보내온 태스크를 선택하여 'W' 문자를 보내고, 나머지 태스크들에게 'L' 이란 문자를 보낸다.
  - 숫자가 같을 경우 우선순위 높은 태스크에게 'W' 를 보내준다.



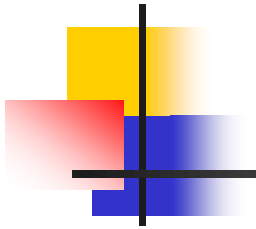
- 6. ‘W’ 를 받은 태스크는 색깔을 칠한다.
  - Random task 우선순위에 따라서 다음의 색을 칠한다.
    - 가장 높은 태스크
      - 빨강
    - 2번째 우선순위 태스크
      - 하늘
    - 3번째 우선순위 태스크
      - 파랑
    - 4번째 우선순위 태스크
      - 녹색
- 7. 모든 태스크는 잠시 쉬고, 위의 과정을 다시 반복한다.

What to do ?

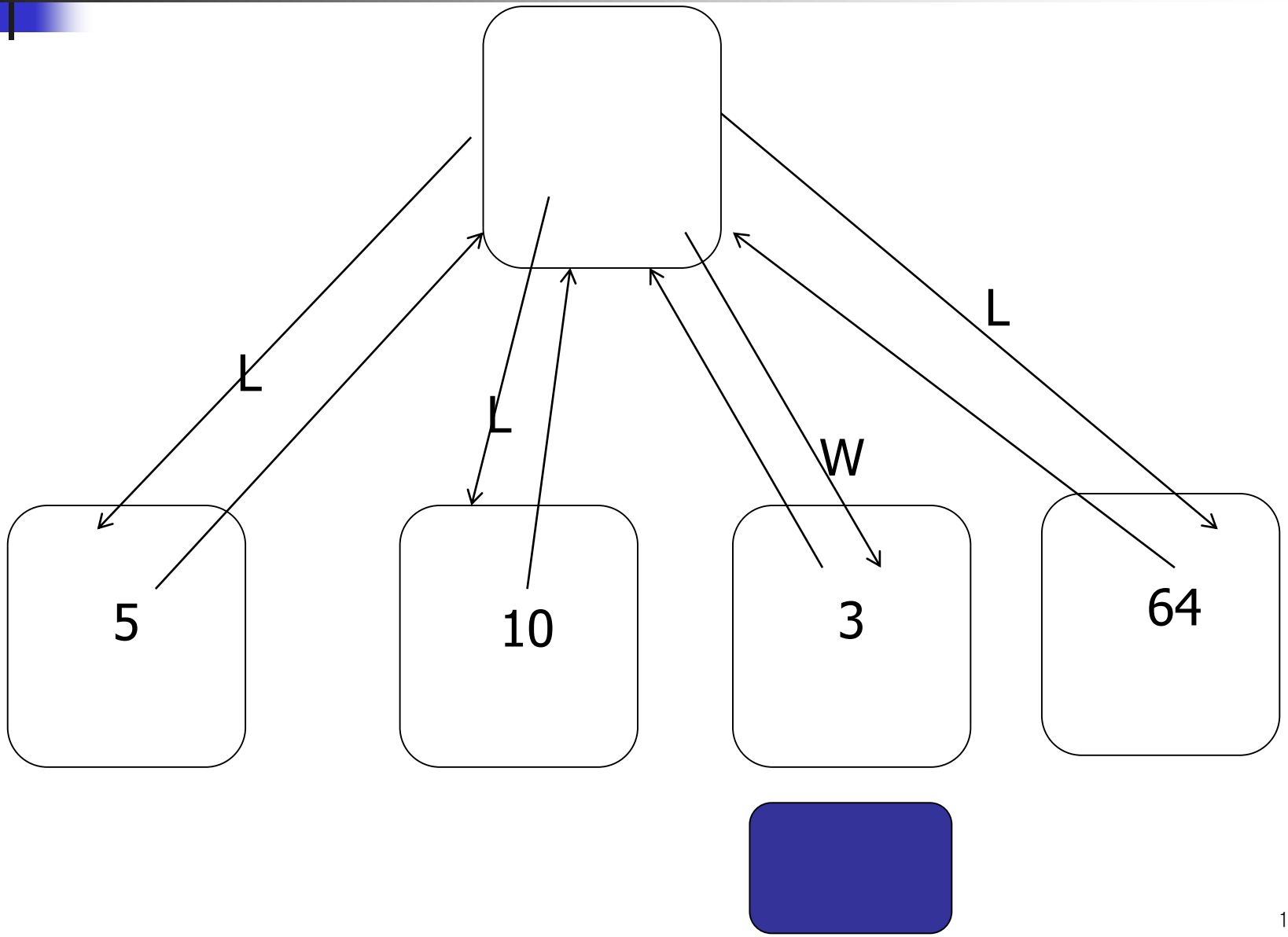
- 1. Create 5 tasks
  - 1 x Decision task
  - 4 x Random task
- 2. Generate a Random number(0-63) on each Random task
- 3. Each Random task send the number to Decision task
  - Implement and submit using Mailbox and Message queue
- 4. Decision task picks the minimum number from them and send back ‘W’ to the task sent the minimum and ‘L’ to the other 3 tasks.
  - If the Decision task gets same minimum numbers from different Random task, send back ‘W’ only to the task which has higher priority.



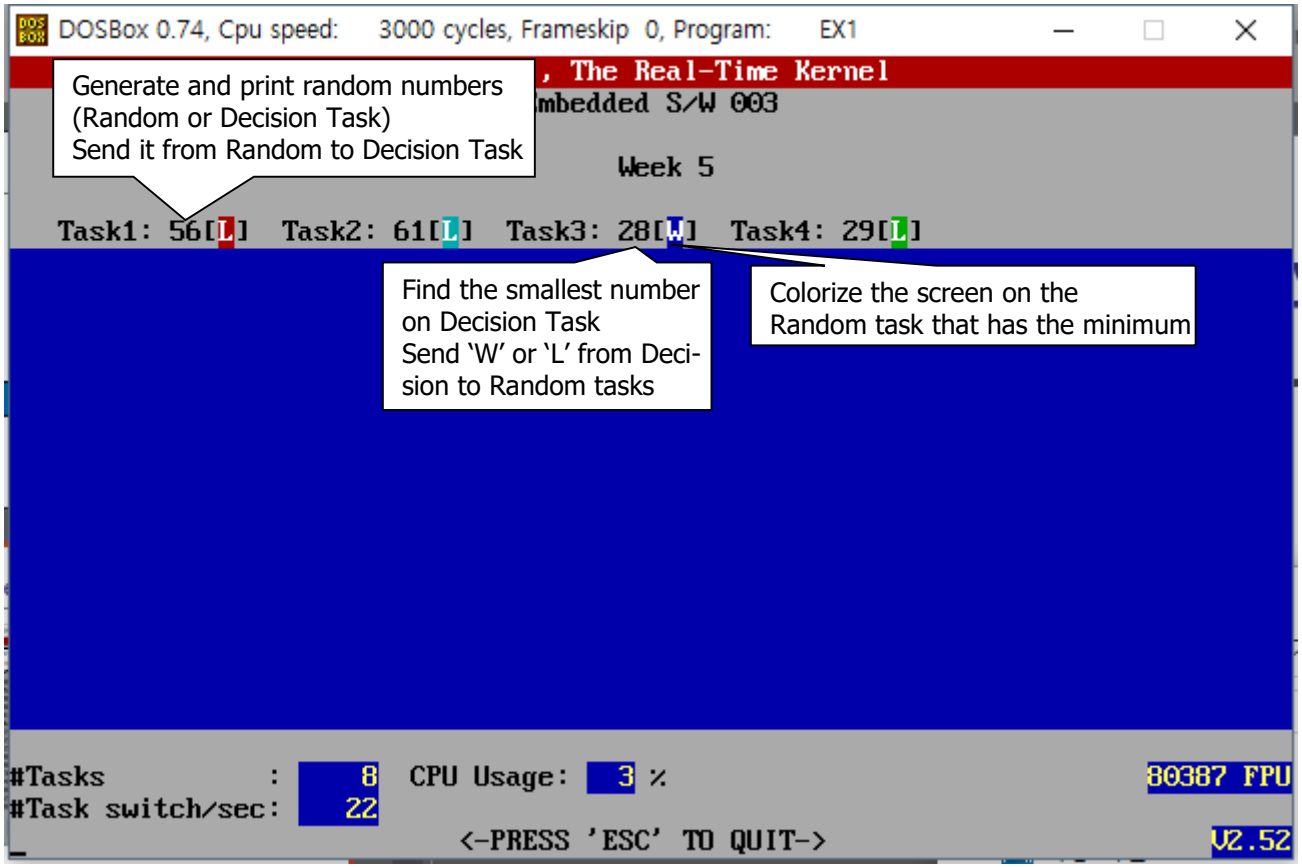
- 6. Task got ‘W’ prints color
  - According to its own priority, each task prints color below
    - The highest priority task
      - Red
    - The second priority task
      - Cyan
    - The third priority task
      - Blue
    - The fourth priority task
      - Green
- 7. Every tasks wait for seconds, iterate every step above.



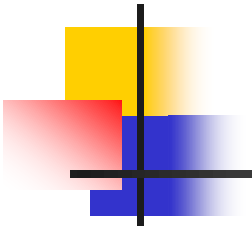
- 최소값 찾기 → 지난 실습 O(1) 알고리즘 or 자유롭게 구현
- It’s fine to use O(1) scheduler from last week or implement as you want


L
L
L
W
5
10
3
64

Assignment example



DON'T COPY AND PASTE THIS IMAGE ON YOUR REPORT!



- 실습 후 C:\SOFTWARE 폴더 삭제해주세요.
- Before you go out,
  please remove C:\SOFTWARE folder.