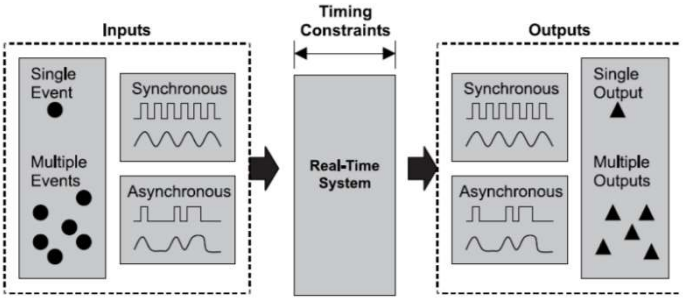
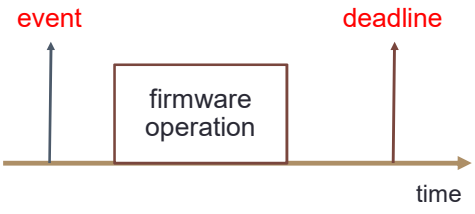
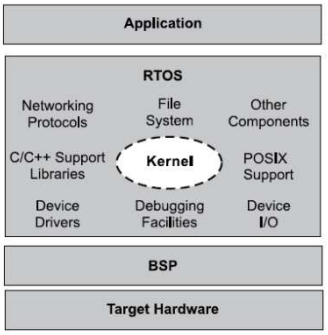
Real-time systems are those systems that respond to external events with respect to timing constraints

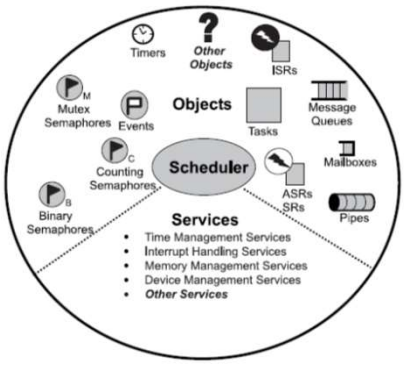


Real-time programs must guarantee response within specified time constraints (deadline)

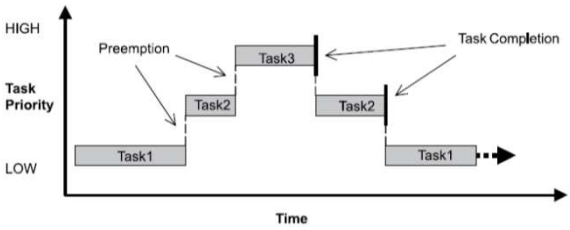


Real-time operating system (RTOS) is a library that schedules execution in a timely manner

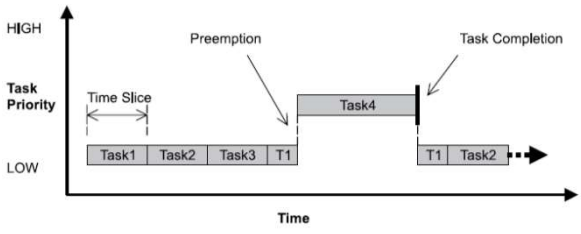




Scheduler determines which task runs by following a scheduling policy

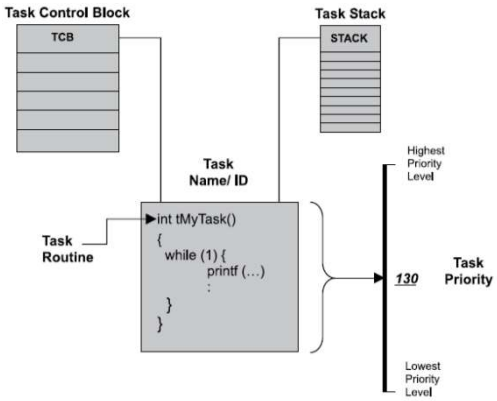


Preemptive priority-based scheduling

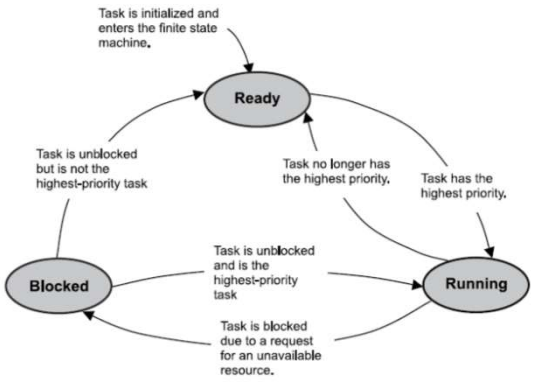


Round-robin scheduling

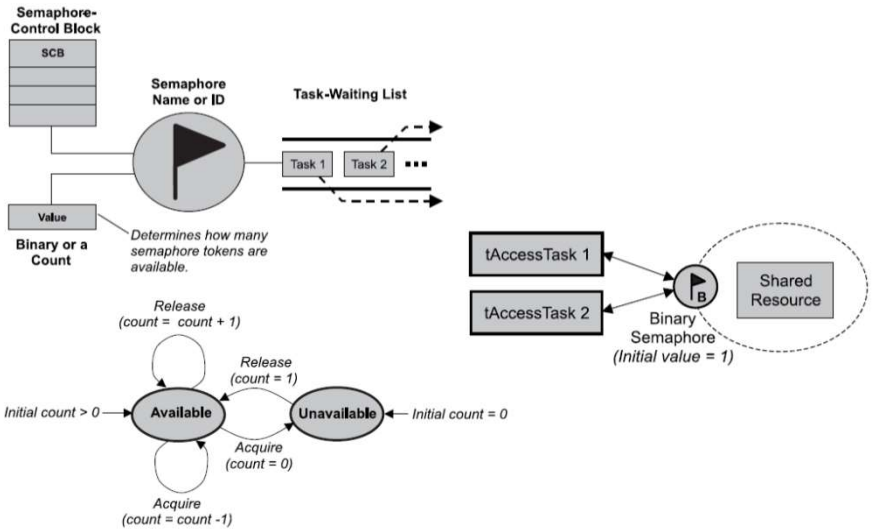
Task is an independent thread of execution that can compete with other concurrent tasks for processor execution time



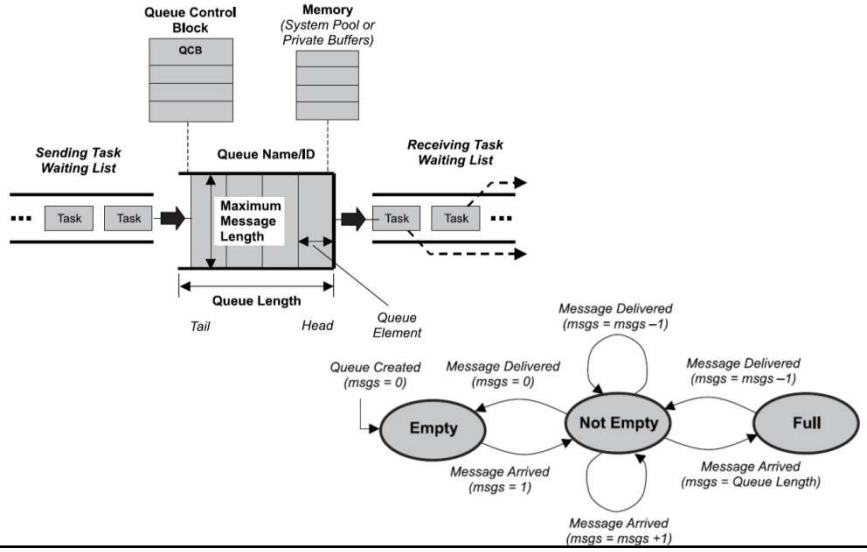
At any time, each task exists in one of task states, including ready, running, or blocked



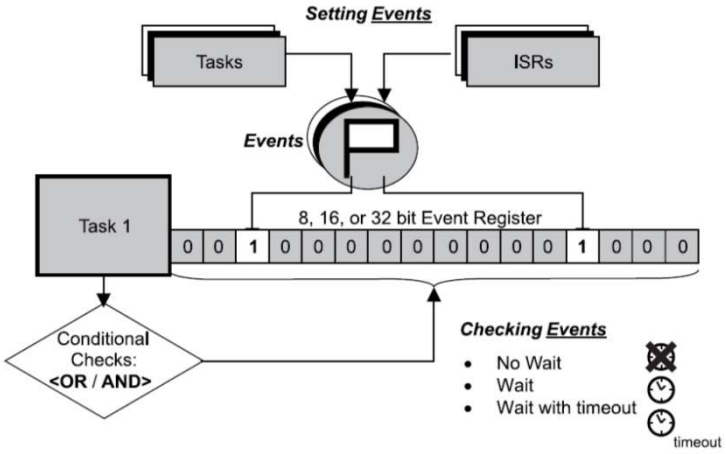
Semaphore is like a key that allows a task to carry out some operation or to access a resource



Message queue is a buffer-like object through which tasks and ISRs send and receive messages to communicate and synchronize with data.



Event register consists of a group of binary event flags used to track the occurrence of specific events



Preparation

1. Add file X.ino with `setupX()` and `loopX()`
2. Add file X.h

```
#ifndef X_H
#define X_H

#endif
```

3. Include RTOS library

```
#include <Energia.h>
#include <xdc/runtime/Error.h>
#include <ti/sysbios/BIOS.h>
#include <ti/sysbios/knl/Event.h>
```

RTOS template

1. Main setup = HW/RTOS initialization

```
void setup() {
    hardwareInit();  // Serial, button, ...
    rtosInit();      // RTOS object
}
```

2. Main loop = event dispatch

```
void loop() {
    if (Serial.available()) {
        Event_post(evtHandle, EVT_SERIAL);
    }
    delay(100);
}
```

```
#define EVT_SERIAL Event_Id_00
```


RTOS template

1. Feature setup

```
void setupX() {
    xInit(); //
}
```

2. Feature loop

```
void loopX() {
    Event_pend(evtHandle, EVT_NONE, EVT_SERIAL,
BIOS_WAIT_FOREVER);
    do {
        uint8_t ch = Serial.read();
        ...
    } while(Serial.available());
}
```