

논문 2011-06-08

# 자동차 전장 시스템에서 주기 및 비주기 태스크를 위한 실시간 스케줄링

(Real-Time Scheduling for Periodic and Aperiodic Tasks on  
Automotive Electronic System)

조수연, 김남진, 이은령, 김재영, 김주만\*

(Su-Yeon Jo, Nam-Jin Kim, Eun-Ryung Lee, Jae-Young Kim, Joo-Man Kim\*)

Abstract : We propose power-saving real-time scheduling method for mixed task sets which consist of both time-based periodic and event-based aperiodic tasks in the automotive operating system. In this system, we have to pursue maximization of power-saving using the slack time estimation and minimization of response time of aperiodic tasks simultaneously. However, since these two goals conflict each other, one has to make a compromise between them according to the given application domain. In this paper, we find the adjustment factor which gives better response time of aperiodic tasks with slight power consumption increase. The adjustment factor denotes the gravity of response time for aperiodic tasks. We apply the ccEDF scheduling for time-based periodic tasks and then calculate new utilization to be applied to the adjustment factor. In this paper, we suggest the lccEDF algorithm to make a tradeoff between the two goals by systematically adjusting the factor. Simulation results show that our approach is excellent for variety of task sets.

Keywords : OSEK, Real-time system, ccEDF, Power-saving, Scheduling

## 1. 서론

최근 들어 자동차에 대한 높은 안정성과 편의성의 요구에 따라 자동차 전장 시스템의 고도화와 지능화를 위한 전장 소프트웨어의 복잡도는 증대되었다. 따라서 유럽 자동차 업체를 중심으로 전장 소프트웨어 개발 시 시스템간의 통합과 응용이나 서버 시스템의 이식성에 대한 표준으로 운영체제, 통신 및 시스템 개발에 관한 표준 규격인 OSEK/VDX 표준안을 발표하였다. OSEK/VDX는 OSEK OS, OSEKtime OS, OSEK COM, OSEK OIL을 포함한 총 8가지 세부 명세를 제공하고 있다 [1].

OSEK/VDX 규격에서 실시간 운영체제는 주기적 태스크로 구성되는 시간기반의 OSEKtime OS와 비

주기적 또는 산발적 태스크로 구성되는 이벤트 기반의 OSEK OS를 규정하고 있으며, 시간 기반과 이벤트 기반의 주기 혹은 비주기 태스크가 혼합되거나 독립적인 수행 환경을 보장하고 있다.

주기(periodic) 태스크는 일정한 주기에 따라 활성화되어 작업을 수행하는 태스크로서 마감시간을 반드시 지켜야 한다. 반면, 비주기(aperiodic) 태스크는 주기가 없이 발생 이벤트에 따라 산발적으로 활성화되며, 마감시간에 대한 구속이 비교적 약한 연성 실시간 스케줄링에 기반하고 있다. 이러한 주기 태스크와 비주기 태스크가 혼합 수행 될 경우, 주기 태스크들의 마감시간은 반드시 지켜져야 하고, 아울러 비주기 태스크들도 우선순위 기반 하에 빠른 응답시간을 보여야 할 것이다 [2][3][6].

일반적으로 주기 태스크의 엄격한 마감시간 준수로 인하여 실제 수행 시간과의 차이로 남은 여유시간의 효과적 활용에 대한 많은 연구, 즉, 여유시간을 이용하여 전력 절감이나 처리기 이용률을 높이는 연구가 진행되었다 [4-8].

\* 교신저자(Corresponding Author)

논문접수 : 2010. 12. 29., 수정일: 2011. 01. 18.,

채택확정 : 2011. 02. 10.

조수연, 김남진, 김주만 : 부산대학교 IT응용공학과

이은령, 김재영 : 한국전자통신연구원

실시간 시스템에서 적용 가능한 전력 절감을 위한 알고리즘인 ccEDF는 주기 태스크의 여유시간을 줄임으로써 주파수 조정을 전력절감 및 여유시간 감소에 따른 에너지 절감 효과를 거둘 수 있다.

본 논문은 자동차 전장 표준 운영체제 규격인 OSEK/VDX의 OSEKtime OS와 OSEK OS의 주기 태스크와 비주기 태스크가 혼합되어 수행될 때, 주기 태스크의 마감시간을 보장하면서 비주기 태스크의 응답성을 높여주는 lccEDF 알고리즘을 제안한다. OSEK/VDX의 주기 태스크에 ccEDF 알고리즘을 적용할 시 전력절감 효과는 거둘 수 있지만, 동시에 수행되는 비주기 태스크에 대한 응답률을 보장할 수 없다.

따라서 본 논문에서는 주기 태스크의 마감시간을 보장하면서 비주기 태스크의 일정 응답성 보장을 전제로 최적의 전력 절감을 달성한다. 이들 두 가지 목적에 대한 상관관계를 규정하여 전력 절감을 위한 조정함수를 적용하는 알고리즘을 제시한다.

본 논문의 2장에서는 OSEKtime과 OSEK OS의 혼합 시스템의 명세 및 특징과 ccEDF 알고리즘에 관해 기술하고 3장에서는 lccEDF 알고리즘을 제안한다. 4장에서 이를 따르는 임의의 태스크 셋의 실험 및 결과를 분석하고 마지막 5장에서는 논문의 결론을 맺는다.

## II. 관련 연구

### 1. 혼합된 OSEKtime / OSEK OS 시스템

OSEK/VDX 표준은 시간기반의 OSEKtime OS와 이벤트 기반의 OSEK OS에 대한 표준을 정의하고 있다. 시간기반 태스크는 일정한 주기마다 알람에 의해서 구동되는 주기 태스크로서 실행 시간과 마감 시간에 대한 엄격한 실시간 특성을 요구한다.

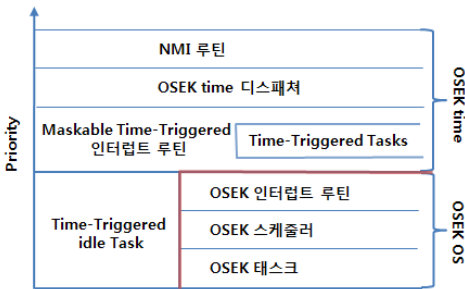


그림 1. OSEK/VDX 태스크 처리 레벨  
Fig. 1. OSEK/VDX task processing level

반면, 이벤트 기반 태스크는 이벤트 발생 시에 활성화 되는 비주기적 혹은 산발적인 태스크로서 연성 실시간 특성을 가진다.

표준에서는 이러한 두 시스템을 단독 또는 혼합한 형태로 수행할 수 있도록 제공 및 정의하고 있다. 그림 1은 OSEK/VDX에서 정의하고 있는 두 시스템의 처리레벨을 보여준다. 두 운영체제의 결합은 OSEKtime OS가 OSEK OS보다 높은 처리레벨을 가짐으로써 OSEKtime OS의 모든 태스크는 OSEK OS의 태스크 및 인터럽트를 선점할 수 있다. 따라서 OSEK OS는 OSEKtime OS의 유희시간 즉, ttitle 일 경우에 수행하게 된다. 시간 기반 태스크는 비 선점 이벤트기반 태스크에 대하여도 선점할 수 있도록 규정하고 있다.

### 2. Cycle-Conserving EDF 알고리즘

일반적으로 태스크는 최악 실행시간(WCET : Worst Case Execution Time)을 가지지만, 실제로 실시간 시스템에서의 태스크 실행시간은 최악 실행시간 보다 훨씬 짧다. 이에 따라 시스템은 많은 슬랙 시간으로 인하여 낮은 시스템 이용률을 가지게 된다. ccEDF 알고리즘은 이러한 슬랙 시간을 이용하여 최악 실행시간까지 실행하도록 주파수와 전압을 조정함으로써 전력 절감 효과와 처리기 이용률을 높이기 위한 알고리즘이다. 즉, 실제 실행시간이 최악 실행 시간보다 짧으면 주파수를 낮추고, 최악의 경우 마감시간이내에 종료할 수 있도록 주파수 및 전압을 높여 실시간성을 보장하는데 있다 [4-5].

```

select_frequency():
    use lowest freq  $f_i \in f_1, \dots, f_n | f_1 < \dots < f_m$ 
    such that  $U_1 + \dots + U_n \leq f_1 / f_m$ 

upontask_release():
    set  $U_i$  to  $C_i / P_i$ ;
    /*  $C_i$  is WCET */
    select_frequency();

upontask_completion():
    set  $U_i$  to  $cc_i / P_i$ ;
    /*  $cc_i$  is the actual cycles used this invocation */
    select_frequency();
    
```

그림 2. ccEDF 알고리즘  
Fig. 2. Algorithm for ccEDF

이 알고리즘에서 주파수가 조정되는 시점은 태

스크 도착 시와 태스크 수행 완료시이다. 태스크 도착 시에는 태스크의 실제 수행시간을 알 수 없으므로, 최악실행시간에 따라 이용률을 계산하여 주파수를 조정하고, 수행 완료 후 태스크는 실제 실행 시간을 반영한 이용률을 계산하여 전압을 재조정한다. 재계산된 이용률은 다음 태스크 도착의 이용률에 반영한다. ccEDF 알고리즘을 적용함으로써, 시스템은 프로세서의 유휴시간을 줄이고 더 낮은 전력으로 시스템을 운영하도록 한다.

### III. 알고리즘 제안

OSEK/VDX 표준에서는 주기 태스크의 마감 시간 보장과 비주기 태스크의 우선순위 기반 선점 스케줄링에 대한 표준 규격만 명시하고 있다. 따라서 이러한 표준을 근거로 구현된 OSEK 운영체제들은 OSEK OS와 OSEKtime OS의 혼합 시스템에서 OSEK OS의 비주기 태스크는 OSEKtime OS의 주기 태스크에 의해 선점되어 실행이 지연되는 비효율적 시스템 운영환경이 된다.

실제 자동차 운행 환경에서 시간 기반 태스크는 미리 정의된 태스크 집합을 가지지만, 비주기 태스크는 임의의 사건에 대응하게 되어 발생 빈도가 비교적 낮다. 따라서 시간 기반 태스크의 유휴 시간을 이용하여 전력 절감 효과를 목적으로 하는 ccEDF 알고리즘을 적용한다. 그러나 ccEDF 알고리즘은 전력 절감 및 시스템 이용률은 높여주지만, 유휴시간에 수행되어야 할 비주기 태스크의 수행시간을 지연하게 함으로써 비효율적 시스템 운영이 되는 문제가 발생한다.

본 장에서는 전력절감 효과와 함께 비주기 태스크에 대한 수행을 보장하기 위한 방법으로 비주기 태스크의 중요도와 긴급성을 감안하는 조정계수를 결정하고, 이 조정값을 ccEDF 알고리즘의 주파수 조정 반영 시에 전력 절감과 비주기 태스크 응답성을 절충하는 새로운 알고리즘인 lccEDF(Loosly ccEDF)를 제안한다.

#### 1. 비주기 태스크의 조정값 계산

주기 태스크의 전력절감을 위한 주파수 사이클인  $1-U_i$ 에 대하여 비주기 태스크의 응답성의 중요도에 따른 조정을 가능하게 해야 한다.

주기 태스크의 전력절감 및 비주기 태스크의 응답성의 목적을 절충하기 위해 비주기 태스크에 할

당하는 조정계수는 비주기 태스크의 현재 상태를 반영할 수 있는 값이어야 한다. 비주기 태스크의 현재 상태를 반영할 수 있는 다양한 값이 있겠지만, 여기서는 비주기 태스크 중 선점된 상태의 태스크와 준비리스트 큐 전체에 대한 대기 중인 태스크의 대기율로 조정계수를 결정한다.

주기 태스크에 의해 선점되어 대기 중인 태스크 중 선점되어 대기 중이거나, 준비리스트에 대기 중인 태스크의 수가 많은 경우 비주기 태스크의 응답률을 보장하기 위해서는 빠른 수행을 요구한다. 따라서 이러한 비주기 태스크의 실행을 보장하기 위한 조정값의 결정은 현재 선점된 태스크의 여부와 전체 큐에 대해 현재 준비큐에 대기 중인 태스크의 대기율로써 결정하도록 한다. 식 (1)과 (2)는 대기 중인 태스크 중 선점 당한 비주기 태스크에 대한 50%와 준비 큐에 비주기 태스크의 대기율에 대한 50%에 의해 결정되는 조정값  $\alpha$ 에 대한 수식이다.

$$\alpha = 0.5 + [queue\ waiting\ rate]_{0 \sim 0.5} \quad (1)$$

$$\alpha = [queue\ waiting\ rate]_{0 \sim 0.5} \quad (2)$$

위의 식은 선점된 태스크에 대하여  $U_a$ 의 남은 이용률의 50%를 증가시키고, 준비 큐의 총 길이에 대한 현재 대기 큐의 길이로써 조정하도록 한다. 선점된 태스크는 준비 큐에서의 대기 시간 이후, 선점에 의한 수행되지 못하는 시간에 대한 가중치를 가지고 이에 대해 빠른 응답률이 필요하기 때문에 선점된 태스크에 대한 가중치를 50%로 지정하였다.

#### 2. lccEDF 알고리즘

제안하는 알고리즘은 주기 태스크가 도착했을 때 ccEDF 알고리즘이 적용된 이용률을 반영하기 전, 현재 상태 비주기 태스크의 조정값을 반영하여 주파수를 조정한다. 수식 (3)은 이러한 주파수 조정을 위한 조정계수를 포함하는 이용률을 구하는 식을 보여준다.

$$U_a = U_i + (1 - U_i) \times \alpha \quad \forall (0 \leq \alpha < 1) \quad (3)$$

ccEDF 알고리즘에 의해 계산된 이용률  $U_i$ 은 0~1사이의 값을 가지는 조정계수  $\alpha$ 를 반영하여 새로운 이용률  $U_a$ 를 주파수 조정에 반영한다. 위 식 (3)에서 보는 것과 같이 조정계수  $\alpha$ 의 값이 1에 가까울수록 전체 이용률이 증가하고 0에 가까울수록 감소하므로, 비주기 태스크의 응답 시간의 중요도가

높을수록 1에 가까운 값을 제공해야 한다.

```

upon task_completion():
    find the adjustment factor  $\alpha$ ;
    set  $U_i$  to  $cc_i/P_i$ ;
    set  $U_\alpha$  to  $(U_i + (1 - U_i) * \alpha)$ ;
    /*  $cc_i$  is the actual cycles used this invocation */
    /*  $\alpha$  is an adjustment factor */
    select_frequency();
  
```

그림 3. lccEDF 알고리즘

Fig. 3. lccEDF algorithm

중요도에 따른 조정 계수의 값은 비주기 태스크가 주기 태스크에 의해 선점 여부 및 횟수, 준비 큐에 대기 중인 태스크 수가 많을 경우 등의 비주기 태스크의 현재 속성을 반영하는 값이어야 한다. 따라서 조정값에 따라 이용률을 재계산하여 비주기 태스크의 수행 중요성을 감안한 새로운 알고리즘인 lccEDF는 그림 3과 같다. 그림 3에서 주기 태스크가 수행 완료 되었을 때, ccEDF에 의하여 다음 이용률 계산 시에 조정값을 반영한 새로운 이용률  $U_\alpha$ 를 주파수 조정값으로 설정함으로써 전력 절감과 비주기 태스크의 수행 보장을 절충할 수 있게 한다. 이는 전력절감률이 커질수록 비주기 태스크의 응답률이 감소하는 반면 응답률 증가는 전력절감 효과의 감소로 이어지는 두 가지의 상충된 목적을 절충할 수 있을 것으로 보인다.

#### IV. 실험 및 평가

이 장에서는 주기 및 비주기 태스크 셋을 통해 제시된 알고리즘을 적용 및 분석한다. 이 실험을 통해 우리는 OSEK/VDX 표준규격, ccEDF 알고리즘 그리고 본 논문에서 제안하는 lccEDF 알고리즘을 적용하여 수행하고 각 비주기 태스크에 대한 응답률과 주기 태스크에 대한 전력을 비교하여 lccEDF가 이 두 가지 목적을 절충할 수 있음을 보인다. 본 실험을 위하여 다수개의 임의 태스크 셋을 사용한 실험을 수행하였다. 실험 결과 태스크 셋의 3가지 특성으로 분류되었고, 따라서 본 논문에서 결과의 명확성을 설명하고, 두가지 목적을 절충할 수 있는 가장 간단한 태스크 셋 3 종류를 선정하여 실험한 결과를 보여준다. 태스크는 임의의 조건 아래 OSEKtime OS에서의 주기 태스크와 OSEK OS에서의 비주기 태스크를 각각 3개씩 정의하고, 주기 태

스크는  $T_1$ ,  $T_2$ ,  $T_3$ 에 대하여 시작시간, 수행시간, 주기(데드라인), 각 주기에 따른 실제 수행시간을 제공하고, 비주기 태스크  $T_4$ ,  $T_5$ ,  $T_6$ 에 대해 시작시간, 수행시간, 준비 큐의 길이, 선점 여부에 대한 정보를 제공한다. 단, 총 준비 큐의 길이는 10이라 가정하였다. 이러한 주기/비주기 태스크의 실험 세트는 아래 표1과 2에 나타내었다.

그림 4는 위의 태스크셋에 대한 수행결과를 제시 하였다. 그림 4-(a)는 OSEK/VDX 규격을 그대로 적용한 경우로서 주기/비주기 태스크의 이용률은 ECU 용량 전부를 사용하는 경우이다. OSEK/VDX 규격에 따르는 수행은 전력절감을 줄 수 없지만, 비주기 태스크의 응답률이 높은 것을 확인할 수 있다.

표 1. 주기 태스크 실험 셋

Table 1. Test sets for periodic tasks

Periodic Task	Start Time	Computing Time	Period	Invocation 1	Invocation 2
$T_1$	1	3	16	2	1
$T_2$	5	5	20	3	2
$T_3$	11	6	25	3	1

표 2. 비주기 태스크 실험 셋

Table 2. Test sets for aperiodic tasks

Aperiodic Task	Start Time	Computing Time	Free mpted	current_Q
$T_4$	2	2	✓	0
$T_5$	7	3	✓	1
$T_6$	7	3		0

반면 ccEDF 알고리즘을 적용한 그림 4-(b)에서는 주기 태스크의 이용률은 0.68에서부터 알고리즘에 의해 점차 감소하며, 4-(a)의 그림보다 유휴시간 및 전력이 줄어든 것을 확인할 수 있다. 하지만, ccEDF 고유의 전력절감 효과는  $T_3$ 가 주기 태스크에 의해 2번 선점됨으로써 비주기 태스크에 대한 응답률이 떨어진 것을 확인할 수 있다. 4-(c)는 본 논문에서 제안하는 lccEDF를 적용한 사례로서,  $T_4$ 가 수행 시  $T_2$ 는 이용률을 계산하기 전,  $T_4$ 에 대한 조정값을 확인하여 이용률을 0.62에서 0.82로 올려준 후,  $T_4$ 에 대한 응답률을 보장하고, 이 알고리즘에 의해  $T_5$  역시 선점횟수를 줄이고 응답률을 보장하였음을 확인할 수 있다.

아래 그림에서 이용률 조정된 lccEDF 알고리즘

은 ccEDF의 전력 절감을 위한 이용률을 조정하여 응용 도메인에서 정하는 가장 효율적인 조정값을 선택적으로 적용함으로써, 사용자에 의한 절충을 할 수 있다는데 목적이 있다.

이에 따라 우리는 위의 태스크 셋에 대한 활성화된 태스크의 총 수행에 걸린 시간과 실제 수행시간을 비주기 태스크의 응답률 및 각 수행에 따른 전력소비량을 도출하였다. 결과적으로 이러한 수행은 그림 5의 그래프에서와 같이 lccEDF 알고리즘은 기존 OSEK/VDX에 비하여 20%의 전력 감소 효과를 얻었고, ccEDF 알고리즘에 비해 약 5~10% 비주기 태스크가 응답 시간을 보상 받았음을 알 수 있다.

크 셋의 특성에 따라 전력 절감과 비주기 태스크의 응답성에 대한 예측한 결과를 나타내었다. lccEDF 알고리즘은 모든 전력을 사용하는 OSEK/VDX보다는 작고 ccEDF 알고리즘을 통해 절감되어 사용된 전력 소비량에 조정계수  $\alpha$ 를 반영한 값만큼 전력 소비량이 감소됨을 확인 할 수 있었다. 이 값은 조정계수 값이 증가함에 따라 비례적으로 증가 하였다. 응답률은 대기 중인 비주기 태스크의 수행시간에 따라 차이가 있었지만 주파수 조정에 따른 슬랙 시간에 따라 변함을 확인하였다.

이러한 결과를 통해 lccEDF 알고리즘은 전력절감과 동시에 비주기 태스크의 응답성을 보장하는 두 가지 목적을 절충할 수 있음을 확인할 수 있었다.

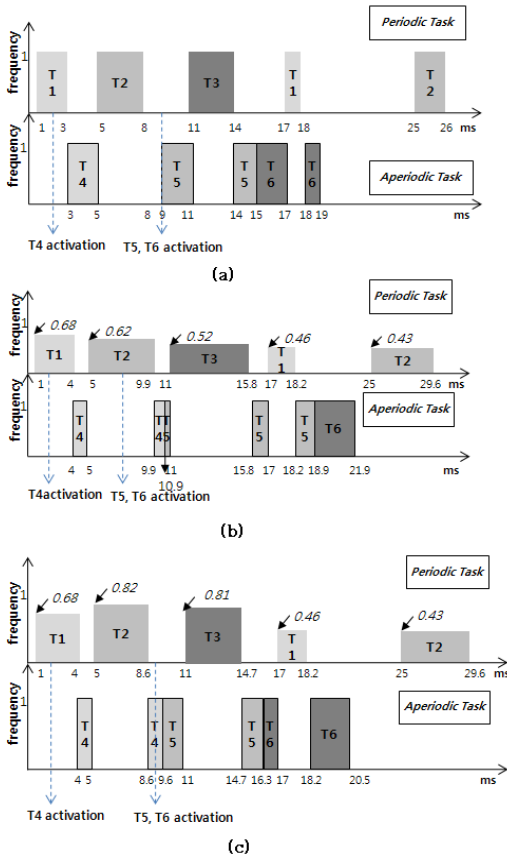


그림 4. 태스크 수행

Fig. 4. Example of task execution : (a) OSEK/VDX (b) ccEDF (c) lccEDF

본 실험을 통하여 여러 개의 임의 태스크 셋으로 실험을 수행하였다. 그 결과는 주기 비주기 태스

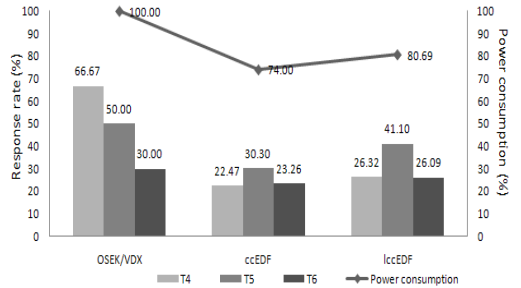


그림 5. 알고리즘에 따른 전력 및 응답률  
Fig. 5. Power and response time by algorithms

## V. 결론

본 논문에서는 OSEK/VDX에서 비주기 태스크와 주기 태스크가 동시에 수행될 때, 주기 태스크의 마감시간을 보장하면서 전력 절감 효과를 얻는 기존의 ccEDF 알고리즘을 이용하여 비주기 태스크의 응답 시간에 대한 중요도를 가늠하는 조정값을 반영하여 전력 절감과 비주기 태스크의 응답성에 대한 상충되는 목적을 절충하는 기법을 제안하였다.

자동차 전장인 ECU의 배치 위치와 응용 프로그램에 따라 조정값의 평가 기준을 정하여 전력 절감과 비주기 태스크 응답성에 따른 절충을 가능하게 함으로써 자동차 전장 시스템의 소프트웨어적 효율성을 높이는 데 있다.

본 연구의 결과는 OSEK/VDX의 표준을 준수하는 OSEK 운영체제 구현 시에 OSEKtime 혹은 혼합 시스템에 적용하는 일차적 과정이 필요하며, 비

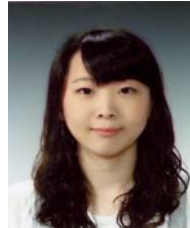
주기 태스크의 응답성에 대한 정밀한 평가를 통한 현실적 평가 기준을 위한 연구가 진행될 것이다.

## 저 자 소 개

### 참고문헌

- [1] OSEK, "OSEK/VDX operating system version 2.2.3", <http://portal.osek-vdx.org>, 2005.
- [2] D. Shin and J. Kim, "Dynamic voltage scaling for mixed task systems in priority-driven systems", Technical Report, Computer Architecture and Embedded Systems Laboratory, Seoul National University, 2004.
- [3] C. Scordino and G. Lipari, "A Resource reservation algorithm for power-aware scheduling of periodic and aperiodic real-time tasks", IEEE TRANSACTIONS ON COMPUTERS, Vol.55, No.12, Dec. 2006.
- [4] P. Pillai and K. G. Shin, "Real-time dynamic voltage scaling for low-power embedded operating systems", in Proc. 18th Symposium Operating Systems.
- [5] 김주만, 이철훈, "EDF 알고리즘을 이용한 경성 실시간 시스템용 온라인 동적 전압 조정", 차세대 PC학회논문지, 제1권, 제2호, pp. 57-65, 2005.
- [6] 권규호, 김주만, "자동차 전장용 실시간 태스크 스케줄링 알고리즘", 대한임베디드공학회논문지, 제5권, 제2호, 2010.
- [7] 공민식, 정근재, "주기성과 산발성 태스크가 혼합된 시스템을 위한 전력절감 스케줄링", 한국콘텐츠학회, 한국콘텐츠학회논문지, 제7권, 제1호, 2007.
- [8] 김희현, 박학봉. "잉여 여유시간을 이용한 연성 비주기 태스크들의 효율적인 스케줄링", 한국정보과학회, 제36권, 제1호, pp. 9-20, 2009.

### 조 수 연(Su-Yeon Cho)



2010년 : 부산대학교  
마이크로정보전자 학사.  
현재, 부산대학교  
마이크로메디컬공학 석사과정.  
관심분야 : 임베디드 시스템,  
Real-time OS.

Email : minu0847@hanmail.net

### 김 남 진(Nam-Jin Kim)



2008년 : 부산대학교  
컴퓨터공학 학사.  
현재, 부산대학교  
IT응용공학과 석사과정.  
관심분야 : 임베디드 시스템,  
Real-time OS.

Email : comets82@hotmail.com

### 이 은 령(Eun-Ryung Lee)



1989년 : 경북대학교  
전자공학과 학사.  
1993년 : 경북대학교  
컴퓨터공학과 석사.  
1995년 : 경북대학교  
컴퓨터공학과 박사수료.

현재, 한국전자통신연구원 선임연구원.  
관심분야 : 임베디드 소프트웨어, RTOS.  
Email : blue4dia@ktr.org

**김 재 영(Jae-Young Kim)**



1991년 : 경북대학교  
컴퓨터공학과 학사.  
1993년 : 경북대학교  
컴퓨터공학 석사.  
1993~1999년 LG정보통신  
(주).

2000년~현재, ETRI 선임연구원.

관심분야 : 차량용 SW플랫폼, 임베디드 소프트웨어,  
유무선 통신.

Email : jaeyoung@etri.re.kr

**김 주 만(Joo-Man Kim)**



2003년 : 충남대학교  
컴퓨터공학과 박사.  
1985~2000년 : ETRI  
책임연구원(운영체제연구팀장).  
1995~1996년 : Novell Inc.  
방문연구원.

2000~현재, 부산대학교 IT응용공학과 부교수.

관심분야 : 임베디드 소프트웨어, Real-time OS,  
분산병렬처리, 고장허용시스템.

Email : joomkim@pnu.edu