지그비 기반 심전계의 데이터 전송률과 소비 전력 분석
Analysis of Data Transmission Rate and Power Consumption in Zigbee Based Electrocardiography

감길진, 충주한, 이타수
충북대학교 의과대학
Nam-Jin Kim(nkim@ubdoc.re.kr), Joo-Hyun Hong(red76@ubdoc.re.kr),
Tae-Soo Lee(taee@chungbuk.ac.kr)

요약
본 연구에서는, 심전도 송수신 장치의 개발을 위한 지그비(Zigbee) 기반 무선 센서 모듈과 PDA (Personal Digital Assistant)의 데이터 전송률과 전력 소비에 대하여 분석 하였다. 데이터 전송률은 패킷 (Packet) 구조의 빈도수와, 패킷을 개방의 심전도 데이터와 개방의 3축 가속도 센서로 구성하였을 때 초당 300 샘플의 전송률을 나타내었다. 두 개의 AAA 전지를 각각으로 연결하여, 센서 모듈의 동작 시간은 평균적으로 2시간이었다. PDA의 전력 소비는 화면의 ON/OFF 여부와 시리얼 포트의 사용 여부 및 패킷에 의한 전류가 전기적으로 나타난 반응으로 나타난 단순한 모드(Idle)에서 작동되었으며, 이에 PDA는 논 콜로저 모드를 사라져 메모리 전용으로 동작한 결과를 보였다. 결과적으로 본 연구에서 개발된 장치를 24시간 동안 Holter 심전계를 촉응할 경우, 센서 모듈의 전력 소비와 전송 속도는 문제가 없었으나, PDA는 전력 소모량에 문제가 있으며 이는 해결이 어려울 과제이다.

Abstract
In this study, data transmission rate and power consumption issues of Zigbee based sensor module and personal digital assistant(PDA) were addressed to develop ECG telemetry device. PDA processed the data transmitted through serial port using non-blocking method. The transmission rate was dependent on the packet structure. It was 300 ECG samples/sec. when each packet was composed of 2 ECG data and 3-axial accelerometer data. Using two AAA batteries in series, operating time of the wireless sensor module was above 23 hours in average. Power consumption of PDA was dependent on screen ON/OFF condition and serial port usage. In this application, operating time of PDA was 5 hours in average. In conclusion, there was no problem in the power consumption of wireless sensor module and transmission rate, when the developed device was used as 24 hour Holter device. But, PDA has the problem of power consumption, which should be solved.

Keyword : PDA(Personal Digital Assistant), ECG, Zigbee, Three Axial Accelerometer

* 본 연구는 보건복지부 보건의료기술진흥사업의 지원(대이변호 090022)과 한국과학자재단에서 지원하는 생체측정기술연구
구현의 연구과제(R11-2009-194-002030)에 의한 결과입니다.

접수번호 : 090008-001
 접수일자 : 2009년 09월 02일
 심사임의일 : 2009년 11월 23일
 코심사인 : 이타수, e-mail : taee@chungbuk.ac.kr
제1장 서론

현재의 휴대 전화기나 애플리케이션 기능 등의 발전과 기술 인프라의 발달로 인해, 휴대형 컴퓨터와 기술의 발전과 함께 모바일 분야에 대한 관심이 고조되고 있으며, 이 분야의 진전 방법으로써 애플리케이션 (Application) 컴퓨팅 또는 손바닥(Usable) 컴퓨팅 등에 대한 많은 연구가 이루어지고 있다. 이러한 기술들은 산업의 변화 모니터링, 간결 기능성의 조기 발견 시스템 주의를 요구하는 산업에 대한 모바일 시스템 등을 적용하였다. 이와 같은 모바일 시스템에서 주요한 기술적 해결 과제는 센서 기술, 센서 작용이 다양한 애플리케이션에 적용될 수 있는 무선 데이터 전송 기술, 모바일 기기의 처리 및 저장 시스템의 경량화 등을 위한 애플리케이션 기술 등이 요구 된다. 그 밖에 휴대형 장치에 필수적으로 요구되는 것은 센서, 표면 전자장치의 적벽 그리고 신뢰성 있는 데이터 흐름에 대한 문제의 해결이다.

본 연구에서는 무선 송신 센서모듈을 사용하여 사람의 심장에서 발생하는 심전도(ECG Electrocardiography) 신호를 수집하고, 이를 지자 패널 (High Radio Frequency) 통신 방식을 사용하여 수신기에 전달하며, PDA 장치로 사용하여 신체의 데이터를 전송하고 저장하고자 한다. 이와 같은 PDA 패널(Radio Frequency) 송수신 기능을 433 MHz의 저속 빈도에서 2.4 GHz의 고속 송수신 방식까지 다양한 빈도 전송 기술들이 개발되었다. 이들 중에서 433 MHz 패널의 RF 통신 방식은 최대 전송 속도가 73.8 Kbps로써 디지털의 고속 데이터 전송에는 부적합한데, 2.4 GHz 대역의 Bluetooth(Bluetooth) 방식은 속도로 인해 성능으로, 기능 등에서 문제가 없으나 간격의 소모량이나 비교적 큰 문제가 있어 적절한 응용을 제공하지 못한다. 그러나 최근 지자 패널 통신 기술은 송신자 114 mA, 수신자 187 mA의 낮은 전력 소모를 가진 경우 250 Kbps의 비교적 고속 통신이 가능하여 본 연구에서의 센서 데이터 송수신 모듈로 선택하였다.

그러나 느리고 전송 시간이 짧은 적외선(bluetooth) 통신 방식, PDAs에서의 데이터 처리 문제와 관련하여 다음에 설명을 따루고, 결과 나머지에서 데이터
전송률과 전력 소모이 하이퍼 기술할 것이다.

## Ⅱ. 방법

### 1. 시스템 구성

본 연구에서 실험도의 3축 가속도 백터의 횡단과 전송을 위하여 저그림 통신 방식을 사용하는 센서 신호 모듈을 각각 구성하였고, 수신 모듈에 도착한 데이터는 시리얼 포트 출력을 통해 FDC로 데이터를 전송하였다. FDC가 시리얼 포트로 전송된 데이터는 처리 프로그램에 의해 저장되고, 신호를 화면에서 그래프로 출력한다. 본 시스템의 전체 구성은 [그림 1]과 같다.

![그림 1. 시스템 구성과 작동](image1)

센서모듈에 사용된 마이크로컨트롤러(MCU)는 아트 블록(Atmel)사의 Atmega128L이며, 가속도 센서는 카이오닉스(Konix)사의 KX-40 3축 가속도 센서이다. 가속도 센서의 3개의 아날로그 출력은 마이크로컨트롤러의 A/D 변환 채널1,2,3에 각각 연결되었다. 가속도 센서는 4V의 내장 배터리로 작동하며, 측정 전압은 33V 전압 인가치에서 560 mV 전압을 얻었다. 실험도 실험은 1개 실험도 실험을 위하여 3개의 건드편을 표준 유도 방식으로 부착하고, 이를 승상 센서모듈에 연결하였다. 입력되는 신호는 증폭기가 신호 안정성을 위해 마이크로컨트롤러의 A/D 변환 채널 0에 입력하였다. 그림 2는 전송된 신호의 전송이 가능하며 통신 거리가 인체로 가면의 내장된 안테나를 사용할 경우 20 미터의 외장형 사물이 가능할 경우 10 미터까지며, 본 연구에서는 전자기 안테나 방식으로 동작하므로 대중형을 사용하였다.

마이크로컨트롤러의 A/D 변환기에서는 입력된 이날 변 신호를 디지털 신호로 변환한 후 인터럽트 (Interrupt)를 발생 시키고, 프로그램에서는 이메시 A/D 채널을 순차적으로 바꾸어 데이터를 채택하게 된다. 데이터 횟수의 기본 단위는 2개의 전송도와 1개의 3축 가속도 백터 측10매트이며 이들 기준으로 바꾸어 저장된다. 따라서 데이터 횟수의 기본 단위는

\[ n = E_1 \times E_2 \times A_1 \times A_2 \times A_3 \]  

로 하였다. 여기에서, \( E_1, E_2 \)는 부스턴 보너스기에 입력된 심장도 데이터이며, \( A_1, A_2, A_3 \)는 3축 가속도 백터 값이다.


![그림 2. 데이터 횡단 순차적 처리 시스템](image2)

예를 들어, 2개의 심장도와 1개의 3축 가속도 백터를 수집한다고 가정하면 n 번째 패킷에서의 첫 번째 심장도 데이터를 \( E(a) \), 두 번째 가속도 백터는 각각 \( A_1(a), A_2(a), A_3(a) \) 두 번째 심장도 데이터를 \( E(b) \)라고 하고, \( n-1 \) 번째 패킷에서의 첫 번째 심장도 데이터를 \( E(a+n) \).
3축 기속도 벡터는 각각 $A_0(n+1)$, $A_0(n+1)$, $A_0(n+1)$이라고 하며, $E(n)$과 $A_0(n+1)$, $A_0(n+1)$ 그리고 $E(n)$은 이 상의 그림 6에서 ①~③ 단계에서 데이터를 수집하는 주기에서 획득 되므로, 두 센서들의 획득 기간 시간은 허도 데이터 인터럽트 처리 시간과 처리 시간일 것이고, 센서들의 획득 기간 시간은 동일 할 것이다. 그러나 두 번째 케이스의 $E(n)$과 $A_0(n+1)$은 인터럽트 처리 시간에서 데이터 케이스(Parallel 처리 시간과 추가 케이스가 있는 시간에 추가 시간이 발생한다. 그림 3에서 순차적인 획득 방법을 사용했을 경우 3축도 데이터 케이스 획득 주기는 그림 3와 같이 시간적 불균형이 있을 것이다.

그림 3. 순차 처리 구조에서의 비균형 데이터 획득

(그림 3)에서 실시간 데이터의 획득 간격은 $E_0(n)$, $E_0(n)$의 $E_0(n)$, $E_0(n)$의 획득 시간을 비하여 더 간 이길 것이다. 그림에서 160 (Interrupt Delay)는 하드웨어 A/D 데이터 처리를 위해 소요되는 인터럽트 처리 시간과 처리 시간을 의미하며, 30 (Interrupt Delay)는 세 번의 처리 시간을 의미한다. 이러한 문제를 해결하기 위하여 무선 중복 시간을 고려하여 수정된 기본단위 $b'$은

$$b' = b_1', A_x, A_y, A_z, E_x, E_y$$

식이다. 여기에서 $e_0$은 $E_1$, $E_2$의 획득 시간 조절을 위 한 지연요소이다.

$E_1', E_2'$은 우선으로 보내기위해 캡슐링된 실시간 데이터이며, $A_x, A_y, A_z$는 3축 기속도 벡터 한다. 데이터 획득 기간 단위 $b'$의 간격이 조정된 결과 (그림 4)의 같은 균등한 타임 그래프를 볼 수 있었다.

![그림 4. 개선된 타임 차트]

(그림 4)에서 $E_0(n)$, $E_0(n)$간의 시간은 3.39 ms 이며, $E_0(n)$, $E_0(n)$간의 시간은 3.34 ms로 끝내 0.05 ms 차이가 나게 되며, 1.3% 정도의 변화 내에서 균일한 간격이 발생함을 알 수 있다. 데이터 케이스 및 관련 정보 속 1개의 $b'$이 획득 되는 시 간에 이루어진다. 이로서 전송 데이터 처리에 이중 전송 방식을 사용한다. 그림 4는 지시를 통한 무선으로 전송되는 케이스 형식이다.

<table>
<thead>
<tr>
<th>ECG</th>
<th>Delay</th>
<th>Act</th>
<th>Delay</th>
<th>ECG</th>
<th>Delay</th>
<th>Act</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECG</td>
<td>160</td>
<td>Act</td>
<td>Delay</td>
<td>ECG</td>
<td>160</td>
<td>Act</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ECG</th>
<th>Delay</th>
<th>Act</th>
<th>Delay</th>
<th>ECG</th>
<th>Delay</th>
<th>Act</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECG</td>
<td>160</td>
<td>Act</td>
<td>Delay</td>
<td>ECG</td>
<td>160</td>
<td>Act</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ECG</th>
<th>Delay</th>
<th>Act</th>
<th>Delay</th>
<th>ECG</th>
<th>Delay</th>
<th>Act</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECG</td>
<td>160</td>
<td>Act</td>
<td>Delay</td>
<td>ECG</td>
<td>160</td>
<td>Act</td>
</tr>
</tbody>
</table>

그림 5. 자료 본관 무선패킷 구조

(그림 5)에서 전송 데이터의 10 byte의 이진 값이며 패킷 형식은 2개 3축 기속도 벡터 1개(Ax, Ay, Az)의 5개 이진 값이 전송되며, 패킷 간의 구분을 위해 0Byte, 0Byte의 2 byte 헤더를 같이 전송하였다. 그러므로 무선으로 전송되는 데이터의 크기는 모두 12 byte로 이루어진다.

수신 모듈에서는 패킷을 수신하여 직접 시리얼 포트로 전송하는 것이 아니라 PCA6의 처리의 흐름을 위해 ASCII(ASC) 코드로 변경하여 전송한다. 패킷의 구 성은 (그림 5)와 동일하지만 ASCII 코드로 변경되어 모든 데이터의 5 byte의 헤더가 포함된다. 이때 전송 데이터의 크기는 15 byte이고 헤더는 2 byte인 0x00, 0x00이다. 수신 모듈에서 전송된 전송 데이터의 크기는 17 byte가 된다. 데이터 오류 검사는 2 byte의 헤더와 데이터의 끝을 검사를 통해 이루어진다.

자료를 받으면서 불량한 데이터는 PCA6에서 스펙트라 내에서 사양 개발 포트 장치 스탠드를 열어 데이터 수집부에 의한 데이터 포트의 전류 제어에 실패하고 복구 포트로 전송한다. 이때의 총 속도는 31,310 bps이다. 전력 데이터 처리 부분에서 스펙트라 내에서 데이터 패킷 단위로 제어를 분석하여 처리 가능성을 정수형 데이터로 바꾼 후 화면 표시를 한다.
2. 프로그램 구현

본 연구에서의 주요한 처리 과정들은 심전도, 3축 가속도 센서 값의 처리와 송신, PDA로 도착한 데이터의 처리 부분에 관련 것이다. 송신 센서들에서의 심전도 처리의 전체 플로우디어그램은 [그림 6]과 같다.

그림 6. 송신 센서들 처리 플로우 디어그램


그림 7. PDA 처리 플로우 디어그램

이상의 그림에서와 같이 PDA에서의 데이터 처리는 시리얼 포트로 수신된 데이터를 꺼내서 파악(g-buffer)하며 그림에 나타내고, 이를 처리부에서 데이터의 데이터를 분리한 후, 심전도의 3축 가속도 센서 값을 정수 값으로 변환하여 지역 버퍼(buffer)에 저장한다. 전역 버퍼는 수신 데이터를 논 복호화(non-blocking) 방식으로 계속 저장할 수 있지만 버퍼의 크기는 조절하기 때문에 일정 수준 밖으로 수신 버퍼의 초기화가 필요하다. 전역 버퍼 초기화 시점에 이전 처리되지 않은 데이터가 있는 경우 이를 보조 버퍼(buffer)에 저장한다. 전역 버퍼 데이터 처리 전에 보조 버퍼를 먼저 처리하는 방식으로 데이터의 순서를 지정한다. 저장된 지역 버퍼 데이터는 화면에서 그래프로 표시하거나 파일로 저장하였다.

이상과 같은 처리 방식에서 데이터 패킷의 크기에 따른 패킷을 Sampling rate의 변화를 알아내기 위해 패킷의 구성을 각각 달라져서 수신 패킷을 실험하였다. 각각의 패킷 구조는 시간에 따라 달라지며 표 1과 같이 구성하였다. "\(E_1\)"는 패킷의 해답이, "\(E_2\)"는 패킷의 신호를 의미하며, \(A_{\text{g}}\) \(A_{\text{a}}\) \(A_{\text{v}}\)는 3축 가속도 센서의 각축의 값이다. 여기에서 \(E_{\text{g}}\) \(E_{\text{a}}\) \(E_{\text{v}}\) 사이에 시간 조정을 위해 \(\tau\)가 삽입된 모델이다.
表 1. 실험 세부사항

<p>| | | | | | | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
</tr>
<tr>
<td>1</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
</tr>
<tr>
<td>2</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
</tr>
<tr>
<td>3</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
</tr>
<tr>
<td>4</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
</tr>
<tr>
<td>5</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
</tr>
<tr>
<td>6</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
</tr>
<tr>
<td>7</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
</tr>
<tr>
<td>8</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
</tr>
<tr>
<td>9</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
</tr>
<tr>
<td>10</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
<td>E</td>
</tr>
</tbody>
</table>

### III. 결과 및 고찰

1. 데이터 전송률 분석

表 1의 각각의 세부사항을 프로그램하여 데이터의 구성에 따른 전송률의 변화를 비교하였다. 전송률이 데이터의 적은 경우 많은 데이터가 전송되며, 전송이 이루어질 때마다 데이터 전송률(Transmission)과 데이터 전송률(Link layer, Application layer)에서 수신한 데이터의 등록이 높을 때에 전송 효율이 떨어지는 것을 보였다. 또한 하나의 전송 장치의 데이터를 구성하는 경우 그림 2에서와 같이 전송률은 섭정적으로 증가하지 않음을 알 수 있다. 이는 다음의 [표 2]와 같이 부정 전송 유무 시각이 성능의 특정 지수에 따라 증가하기 때문인다.

表 2. 부정 전송시간 증가 결과

<table>
<thead>
<tr>
<th></th>
<th>정상전송시간 (ms)</th>
<th>부정전송시간 (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.92</td>
<td>1.14</td>
</tr>
<tr>
<td>2</td>
<td>2.92</td>
<td>2.88</td>
</tr>
<tr>
<td>3</td>
<td>2.92</td>
<td>3.75</td>
</tr>
<tr>
<td>4</td>
<td>3.08</td>
<td>5.66</td>
</tr>
<tr>
<td>5</td>
<td>3.10</td>
<td>5.94</td>
</tr>
<tr>
<td>6</td>
<td>3.22</td>
<td>6.52</td>
</tr>
<tr>
<td>7</td>
<td>3.32</td>
<td>6.88</td>
</tr>
<tr>
<td>8</td>
<td>3.35</td>
<td>6.78</td>
</tr>
<tr>
<td>9</td>
<td>3.52</td>
<td>7.69</td>
</tr>
<tr>
<td>10</td>
<td>3.59</td>
<td>8.64</td>
</tr>
</tbody>
</table>

2. 전력 소모량 분석

이상의 실험들은 전송 시간이 감소함에 따라 소모량의 변화를 위해서는 전력 소모량에 대한 고찰이 요구된다. 전력 소모량의 경우 실험 영역의 배경에서 사용하는 경우 및 시간 간격 이상의 등록을 하면 전력 소모량에 대한 변화를 고려하여 기록도하던 경우 약 5시간 동안 하였다. 이에 대한 전력 소모량의 변화를 다룹 [표 3]과 같은 결과를 얻었다.

表 3. PDA 장치의 전력 소모량 분석

<table>
<thead>
<tr>
<th>전력 소모량 (W)</th>
<th>장치 소모량 (W)</th>
<th>전력 소모량 (W)</th>
<th>총 전력 소모량 (W)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ON</td>
<td>3.00</td>
<td>3.09</td>
<td>6.09</td>
</tr>
<tr>
<td>OFF</td>
<td>2.90</td>
<td>2.92</td>
<td>5.82</td>
</tr>
<tr>
<td>전력 소모량</td>
<td>0.10</td>
<td>0.12</td>
<td>0.22</td>
</tr>
<tr>
<td>전력 소모량</td>
<td>0.02</td>
<td>0.04</td>
<td>0.06</td>
</tr>
</tbody>
</table>

실에서 사용한 PDA는 전자용량 3.7 V, 1800 mAh의 HP tx4700 시리즈이며, Pocket PC 2003 운영체제 사용하였다. 결과로서의 각각도는 그림 3의 그래프를 사용하여 테스트 하였으며, 전력 소모량은 PDA의 전원을 전 후 사용자 입력에 반응할 수 있는 프로그램을 부가하지 않는 등록 프로그램이 실행되어 있는 상태, 즉 PDA가 부팅된 상태를 의미한다. 정상 실행시 PDA의
전력 소모는 화면을 연 경우 15 mA로 거의 소모하는 것이고 화면을 꺼면 경우 10 mA 정도의 전류 소모량을 보였다. 쓰레드가 사용하지 않는 시스템과 프로그램 실행 상태에서 비교하여 10 mA 정도의 전력만 더 소모한다. 그러나 본 연구에서는 공유의 전력도 메모리 쓰레드를 끄고 메모리 쓰레드가 사용 하였으며 이는 [표 3]에서 같이 보듯 샘플 개수 두 배 정도의 소모량을 보이는 것을 알 수 있다. 전력 측정을 위해 사용할 시스템은 100 sample/sec의 전력도 데이터를 고속 촬영하기 때문에 사용자도외의 짧은 사용 시간을 기록할 수 없었으며 실시간 멀티스트림 모델은 두 개의 AAA 전지를 각각 연속으로 한 시간 동안 동시에 촬영하였다. 또한 수집된 데이터의 처리 및 저장을 위하여 FOD(Personal Digital Assistant)를 사용하였다. FOD의 일반적인 사용 케이스에서는 동작 시간은 사용 형태에 따라 2-3시간이며 실시간 촬영 프로그램 동작 시간은 5 시간, 화면을 전 상태로 3시간 정도만 사용할 수 있어 소소한 쓰레드모드의 업무에 비해 매우 부족한 사용 시간을 나타내었다. 이는 FOD의 사용 형태가 동시에 쓰레드 모드에서 쓰레드 처리의 비효율성 때문이며, 문제 해결을 위해 쓰레드 내에서 쓰레드간 사용자를 사용하여 쓰레드 간에서 사용자와 쓰레드 간으로의 전환을 하여 사용한 경우. PDA의 전력 소모량을 줄일 수는 있지만 필요한 만큼의 데이터를 획득할 수는 없었다. 또한 실제 신호의 코니터링된 과정에서 데이터를 간격적으로 받을 수도 없기 때문에 전력이 가장 동적이고 가능한 처리 시스템의 개발이 요구된다.

IV. 결론

본 연구에서는 전력도 데이터를 측정하고 모니터링 하는 손/수신 쓰레드들의 개발과 처리 시스템들의 실제 환경에 적용하여 있어서의 문제점은 데이터 전송률과 소비 전력에 관한 논의하였다. 유대형 장치는 데이터를 부정으로 전송하는 부분은 유니버설 시리얼 포트를 통해 수집하는 부분으로 이루어져 있으며, 최적화 데이터 처리 시스템은 PDA를 통해 분석되고 기록된다. 이를 쓰레드로 PDA의 저속 및 데이터 전송의 정확한 속도와 사용을 위하여 10개 시나리오로 설정하고, 각 시나리오에 따라 무선 송수 전송 시의 성능을 비교하였다. 실제로 포함한 이론 데이터는 데이터 전송률과 전력과의 관계를 보여 시나리오 종합도 데이터가 확정된 데이터 채널을 위한 데이터 비교시기를 할 수 있다. 본 연구에서는 종합된 애플리케이션과 적합한 쓰레드의 확률을 써서 시나리오 종합도 데이터가 확정된 데이터 채널을 위한 데이터 비교시기를 할 수 있다. 본 연구에서는 종합된 애플리케이션과 적합한 쓰레드의 확률을 써서 시나리오 종합도 데이터가 확정된 데이터 채널을 위한 데이터 비교시기를 할 수 있다.
치 않고 SD 카드(Secure Digital Card) 매체를 이용하여 센서 모듈에서 직접 데이터를 저장하는 방식 등에 관하여 연구할 것이다.


지자소개

김남진(Nam-Jin Kim)

1968년 6월 : 단국대학교 물리학과 (전기기기) 
2000년 8월 : 단국대학교 전자계산기학과 (전기기기)
2003년 8월 : 단국대학교 전자계산기학과 (전기기기)

2004년 6월 ~ 현재 : 큐레이션 디자인 및 기술 개발센터 (사장)
: 실시간 스케줄링 엔데avors 시스템
홍 주현 (Joo-Hyun Hong)  정회원
- 2001년 2월 : 단국대학교 전자공학과 (공학사)
- 2003년 2월 : 충북대학교 의용생명
  치공학과 (공학석사)
- 2005년 3월 : 충북대학교 의용생명
  치공학과 (박사수료)

- 2004년 5월 ~ 현재 : 촉매형진단체로기기 개발센터
  <논문분야> : 임베디드 시스템, 신체영역 통신

이 대수 (Tae-Soo Lee)  정회원
- 1961년 2월 : 서울대학교 전자공학과 (공학사)
- 1963년 2월 : 서울대학교 전자공학과 (공학석사)
- 1968년 8월 : 서울대학교 제어계측공학과 (공학박사)

- 1961년 1월 ~ 현재 : 충남대학교 의학과 교수, 촉매형
  진단체로기기개발센터 이사, 충남대학교병원 의공학
  과장

<논문분야> : 빅데이터 헬스케어 시스템, 컴퓨터 의학응용