

A Body Sensor Network Simulation Platform for Medical Applications

R. V. Rodrigues Filho, R. F. Bulcão Neto and I. G. Sene Junior

Abstract— Body Sensor Networks (BSN) is an important technology in providing ubiquitous healthcare services. These sensors are capable of supplying medical professionals with data collected from a patient outside the hospital environment. Remotely acquired information has enormous application potential and can assist medical professionals to provide fast and accurate assistance to injured people at accidents sites, minimizing the risk of further complications by reducing delays in the diagnosis process. Despite the great benefits of such applications, there is a gap in the state-of-art techniques that allow the creation and testing of such high-level and critical applications. This paper presents a body sensor network simulation environment as a response to that need, providing a tool that plays a key role in assisting the development of critical healthcare applications through an emulated healthcare-specific hardware, which can be used to test applications in a controlled environment, considering the demands of the medical field.

Keywords— body sensor networks, healthcare, critical applications, hardware emulator.

I. INTRODUÇÃO

AS Redes de Sensores do Corpo Humano (RSCH) são utilizadas em diversas aplicações envolvendo a coleta de sinais biológicos do indivíduo. Na área da Saúde, a coleta de dados através das RSCH pode ocorrer em diferentes cenários, por exemplo, ao socorrer uma vítima no local do acidente, ao monitorar sinais vitais do paciente para prevenção de doenças, ou mesmo em Unidades de Terapia Intensiva.

O emprego das RSCH nos diversos cenários da Saúde proporciona grandes benefícios para a área dada a sua capacidade e facilidade de cruzar informações de diferentes sinais e o seu baixo custo de implantação [22], [23], [24]. Essas características oferecem ao profissional da saúde, a um custo acessível, dados co-relacionados para apoiar na prevenção de doenças e no acompanhamento do quadro clínico de pacientes.

O auxílio prestado ao profissional da saúde por meio dessa tecnologia contribui na solução de problemas da infraestrutura de prestação de serviço de saúde. Esses problemas têm suas raízes no aumento demográfico, que resulta em um aumento na demanda por serviços de saúde e que contribui na lotação dos hospitais. Outros fatores que agravam a situação é a falta de especialistas em ambientes hospitalares em certas regiões, aparelhos de alto custo necessários na prestação do serviço, dentre outros fatores que são descritos em [7], [6] e [9].

O cenário atual da área de Saúde motiva vários trabalhos na literatura a empregar RSCH no contexto de sistemas ubíquos para solucionar esses problemas. Contudo, apesar do avanço no desenvolvimento de plataformas de hardware e aplicações de RSCH, poucos projetos utilizam plataformas de hardware específicas da área médica para testarem suas soluções, o que faz com que os testes dessas aplicações sejam feitos em cima de plataformas e cenários que não levam em consideração características impactantes da área [8].

O objetivo deste artigo é apresentar uma plataforma emulada de hardware adequada para o teste de aplicações de RSCH na área da Saúde. Esta plataforma é capaz de simular: i) aplicações escritas em qualquer tecnologia; ii) dados de sensores específicos da saúde, como o ECG; e iii) troca de mensagens entre nodos sensores e pacotes de redes mal formatados e maliciosos. Desta forma, o simulador tem como principal contribuição prover um ambiente de teste de aplicações médicas, proporcionando uma ferramenta de apoio à construção de aplicações de qualidade.

Este trabalho está dividido da seguinte maneira: a Seção II traz o referencial teórico tratando do conceito de RSCH, sua aplicação na área da saúde e desafios relacionados; a Seção III descreve os principais trabalhos relacionados apontando os motivos que levaram à construção do simulador; na Seção IV são descritos os detalhes do simulador; a Seção V traz os testes da plataforma; a Seção VI contém as contribuições do artigo, e por fim, a Seção VII conclui o trabalho.

II. REFERENCIAL TEÓRICO

Para melhor contextualizar o trabalho, esta seção discorre sobre as redes de sensores para o corpo humano aplicadas à Saúde, bem como desafios ligados à essa área.

RSCH Aplicadas à Saúde

Uma Rede de Sensores para o Corpo Humano (RSCH) é uma rede ad-hoc composta por pequenos nodos sensores capazes de trocar informações por enlaces sem fio; esses nodos são colocados no corpo humano para coletar os sinais vitais do indivíduo e enviá-los para um servidor.

As RSCH tornam possível a coleta e o envio de sinais vitais pelos nodos sensores por meio de uma infraestrutura de telecomunicação como, por exemplo, a Internet, para serem armazenados e processados por um servidor. Esse servidor provê essas informações recebidas em tempo real (isto é, logo após ser coletado pelos sensores, desconsiderando o tempo de transmissão da rede) para qualquer tipo de dispositivo, como *tablets*, celulares, provendo assim o acesso remoto aos sinais coletados do indivíduo a profissionais autorizados.

Através desta tecnologia torna-se possível o monitoramento remoto de indivíduos, a um baixo custo, durante sua rotina diária em qualquer ambiente, seja esse sua casa, seu local de

R. V. Rodrigues Filho, Lancaster University, Lancaster, UK, r.rodriguesfilho@lancaster.ac.uk

R. F. Bulcão Neto, Universidade Federal de Goiás, Goiânia-GO, Brasil, renato@inf.ufg.br

I. G. Sene Junior, Universidade Federal de Goiás, Goiânia-GO, Brasil, iwens@inf.ufg.br

trabalho, etc. Por ser uma rede composta por nodos sensores que comunicam entre si, essa tecnologia favorece o cruzamento de diversos dados permitindo a composição de informações que necessitam de mais de um sinal vital, prestando melhor auxílio ao profissional da saúde. Por isso, as RSCH tornam-se uma tecnologia importante no provimento de soluções ubíquas para solução dos problemas (descritos adiante) de infraestrutura da área.

Na literatura encontram-se vários trabalhos que ilustram a utilização dessa tecnologia no contexto da área da Saúde, auxiliando a minimizar os problemas enfrentados atualmente. Trabalhos como [2], [3] e [11] ilustram a utilização de sensores para o monitoramento de indivíduos em um ambiente domiciliar, e de idosos em casas de repouso. Esses trabalhos descrevem sistemas que permitem, através de sensores no ambiente e no próprio indivíduo, o monitoramento contínuo de seus sinais vitais. Assim, é possível monitorar, analisar e identificar situações críticas que ocorrem com o indivíduo remotamente em tempo real.

Dessa forma, as RSCH são utilizadas para compor sistemas ubíquos para o monitoramento remoto dos sinais vitais do paciente, e desempenham um papel importante para o acompanhamento do estado de saúde das pessoas sem que elas estejam em um ambiente hospitalar, contribuindo com a redução das lotações nos hospitais, o número de pessoas internadas e, conseqüentemente, o risco de infecções hospitalares, dentre outros problemas relacionados.

Desafios da Área de Saúde

As pessoas que procuram serviços hospitalares emergenciais muitas vezes encontram problemas como: lotação, alto custo de internação, riscos de infecções hospitalares, dificuldade de deslocamento para unidade de tratamento, falta de especialistas no centro de atendimento local, dentre outros problemas.

Entretanto, essa situação nem sempre foi assim, de acordo com [6]; inicialmente no Brasil as pessoas recebiam atendimento domiciliar por médicos de família, e somente casos extremos eram levados para receberem tratamento em hospitais. Esse paradigma, se aplicado atualmente, retiraria a sobrecarga da demanda de atendimento no ambiente hospitalar, o que levaria a uma diminuição dos riscos de infecção hospitalar, problemas no deslocamento para unidades de tratamento, dentre outros. Para embasar essa mudança de paradigma, estudos como [6], indicam que 65% dos pacientes que recebem serviços hospitalares emergenciais poderiam ter sido atendidos em ambulatórios ou pronto-atendimento. Essa informação motiva a utilização de tecnologia de baixo custo para o auxílio de prestação de serviços nessas situações.

Atrelado a esse cenário, algumas situações mais específicas merecem destaque. De acordo com a Organização Mundial de Saúde (OMS), entre os anos de 1950 e 2025, haverá um aumento na população de idosos que nos colocará na sexta posição de países com a maior população de idosos do mundo [4]. No caso do idoso, pensar em uma infraestrutura de atendimento e cuidados desses indivíduos em casas de repouso ou em suas próprias residências é de suma importância. O monitoramento remoto do idoso lhes proporciona mais independência e autonomia, e ao mesmo tempo provê uma estrutura de monitoramento contínua, fazendo-se necessário o

deslocamento desse indivíduo a um serviço de emergência hospitalar somente em caso de real necessidade.

Além da preocupação com envelhecimento da população, uma das prioridades na área da Saúde são as doenças crônicas não transmissíveis (DCNT), responsáveis por 72% das mortes ocorridas em 2007 [9]. Também de acordo com os autores, devido a políticas bem-sucedidas de saúde que diminuíram o tabagismo e aumentaram o acesso à atenção básica de saúde, a incidência de doenças cardiovasculares e respiratórias crônicas vêm diminuindo, mas há um aumento de hipertensão e diabetes. Todavia, o tratamento de pessoas com DCNTs requer o contínuo monitoramento do paciente, e é fundamental contar com uma infraestrutura de monitoramento que reduza a constante necessidade do indivíduo realizar visitas ao hospital.

As situações ilustradas anteriormente mostram cenários em que o uso de RSCH é parte essencial na solução de problemas de infraestrutura da área de Saúde, seja no contexto de prover serviços médicos a pacientes remotos, monitoração de idosos em casas de repouso e até o constante acompanhamento de pacientes com DCNTs. Em todos os casos citados, é imprescindível o desenvolvimento de aplicações de alta qualidade, pois envolvem situações críticas que colocam em risco a vida de pessoas.

As dificuldades intrínsecas do processo de desenvolvimento de aplicação de alto nível para RSCH, e para Redes de Sensores sem Fio (RSSF), em geral, é constantemente discutido na literatura [12], [13] e [14]. Uma solução recorrente na comunidade de RSSF é o desenvolvimento de simuladores. Assim, neste trabalho, apresentamos o simulador da plataforma de hardware SHIMMER, cujo objetivo é auxiliar desenvolvedores no processo de desenvolvimento de aplicações de qualidade, especificamente para RSCH com foco na área médica.

III. TRABALHOS RELACIONADOS

Apesar dos avanços na área de redes de sensores com o desenvolvimento de sensores cada vez menores e de baixo custo, projetos que utilizam essa tecnologia para prover soluções na área de Saúde muitas vezes realizam testes em plataformas de hardware inadequadas.

As plataformas de hardware de propósito geral não levam em consideração algumas características que deveriam levar ao serem utilizadas para a coleta de dados do corpo humano: características como ser leve, pequeno, esterilizável, lavável, feito de material que não causa irritação à pele e não restringir a movimentação do indivíduo são imprescindíveis ao serem utilizados em uma RSCH.

Por conta dessa situação, trabalhos clássicos da área como o projeto CodeBlue [10] projetaram plataformas específicas da área de Saúde, dando origem assim a plataformas como o SHIMMER [1] e o PLUTO [5].

A plataforma de hardware SHIMMER (Sensing Health with Intelligence, Modularity, Mobility and Experimental Reusability) [1] é uma plataforma específica para a Saúde, composta por periféricos integrados, software aberto, hardware para gerência de energia e capacidade de sensoriamento voltado para a área médica. A plataforma conta com microcontrolador MSP430F1116, interface de rádio CC2420 e baterias recarregáveis de Li-Ion. Ressalta-se que a plataforma é compatível com o sistema operacional TinyOS.

Da mesma forma, tem-se a plataforma PLUTO [5], uma plataforma de hardware específica para a área de Saúde, desenvolvida na Harvard University pelo projeto CodeBlue, é composta por um microcontrolador MPS430, uma interface de rádio CC2420, possui duas baterias AA e também uma bateria de lítio, com isso a plataforma funciona por volta de cinco horas; também há uma porta USB Mini-B para programação e recarregamento energético da plataforma. O PLUTO é uma plataforma de hardware baseada no TelosB que, por sua vez, é uma plataforma de redes de sensores de propósito geral voltada para monitoramento de ambientes, e também compatível com o sistema operacional TinyOS.

Entretando, mesmo com a aquisição dessas plataformas, falta um ambiente de testes para garantir a qualidade das aplicações, sendo que, após implantar o código no nodo sensor, a única interface de verificação de execução da aplicação existente são os LEDs, ou quando a plataforma fornece uma interface JTAG, que se limita na depuração do código somente em um único sensor, não sendo possível depurar o comportamento da rede como um todo.

IV. SIMULADOR

O ambiente de teste de aplicações para saúde deve ser capaz de oferecer formas de mais alto nível para depurar código para o desenvolvedor, permitindo: i) a simulação de troca de mensagem entre nodos sensores para detecção de problemas de lógica; ii) montar topologias de rede; e iii) verificar o comportamento da aplicação ao injetar pacotes maliciosos. Esses requisitos são importantes ao considerar a criticidade dos sistemas de Saúde. O simulador deve permitir a emulação de plataformas de hardware que possuem as características já citadas: leve, pequeno, esterilizável, lavável e feito de material que não irrita a pele e não restringe a movimentação.

O simulador deve permitir a emulação de plataformas de hardware que possuem as características já citadas: leve, pequeno, esterilizável, lavável e feito de material que não irrita a pele e não restringe a movimentação.

Com esse intuito, foi desenvolvido um simulador, ilustrado na Fig. 1, para redes de sensores do corpo humano com o objetivo de suprir a necessidade de um ambiente para simular aplicações da área de Saúde utilizando uma plataforma de hardware adequada.

Na Fig. 1 são mostradas duas instâncias do simulador trocando mensagens entre si. À esquerda, há a imagem da placa SHIMMER com seus LEDs, ao centro, estão os gráficos que ilustram a quantidade de bytes na pilha de execução e, por fim, à direita, duas telas que permitem a depuração do código rodando no simulador, sendo possível depurar linha por linha do código. Cada uma das partes que compõem o simulador será apresentada em detalhes na próxima seção.

O simulador de aplicações para RSCH apresenta um ambiente de teste de alto nível, capaz de executar aplicações escritas em diferentes tecnologias, ao contrário de simuladores como TOSSIM (simulador oficial do TinyOS), que só simula aplicações na plataforma MicaZ escritas para TinyOS. Com o propósito do simulador ser independente de tecnologia, o mesmo foi escrito para receber como entrada código em linguagem de máquina e emular todo o comportamento da plataforma de hardware SHIMMER, inclusive o comportamento de um sensor ECG, o conjunto de operações e comportamento do microcontrolador, memória, LEDs, e outros dispositivos que compõem a plataforma real.

O simulador contempla as necessidades daqueles que possuem (ou não) a plataforma SHIMMER real. Pois, apesar da plataforma SHIMMER possuir JTAG para depuração do código, a tarefa de depuração pode ser desgastante dada a sua



Figura 1. Ilustração do Simulador.

complexidade e limitação, por considerar a depuração somente de um nodo sensor, sendo bastante complexa a depuração do código em mais de um sensor ao mesmo tempo trocando mensagens entre si formando uma rede. Com isso, o simulador provê uma forma menos dispendiosa e que demanda menos tempo para teste de aplicações de RSCH em uma plataforma de hardware específica da Saúde, de forma a não precisar gravar o programa na memória *flash* do dispositivo que possui um número limitado de gravações, aumentando o tempo de vida útil do nodo sensor.

A escolha da plataforma SHIMMER se deu pela sua relevância para o desenvolvimento de aplicações para RSCH vestíveis. A plataforma é citada em artigos científicos desde 2007 como em [16], [17] e [18], o que demonstra a relevância da plataforma para desenvolvimento de aplicações que demandam RSCH vestíveis.

Os benefícios proporcionados pelo simulador estão no ambiente de testes de baixo custo e alto nível para aplicações específicas de RSCH na área de Saúde. Esse ambiente de simulação fornece funcionalidades essenciais para que aplicações possam ser devidamente testadas a fim de garantir sua qualidade. É importante enfatizar que o objetivo deste trabalho é contribuir com o amadurecimento do desenvolvimento de aplicações para RSCH para área de Saúde, provendo uma ferramenta que auxilie na construção de aplicações de qualidade, visto que de aplicações na área médica é exigido um alto nível de tolerância a falhas, disponibilidade, segurança, etc.

De forma a proporcionar as características supracitadas, a plataforma de simulação para RSCH para a área de Saúde foi construída com o auxílio da plataforma de simulação MSPSim [15]. O MSPSim é uma plataforma de simulação *open source* escrita na linguagem de programação Java que simula, principalmente, o microcontrolador MSP430, emulando o conjunto de instruções do microcontrolador e todo o seu comportamento ao executar código binário de sua arquitetura.

A partir do emulador do microcontrolador MSP430, o ambiente MSPSim proporciona a criação de plataformas de hardware para RSSF de propósito geral. É possível utilizar o simulador proposto para simulação de código em plataformas de hardware como o Telos, TMoteSky, ESB e WiSMOTE.

Essas plataformas de RSSF foram construídas em torno do emulador do microcontrolador MSP430, foram desenvolvidos seus dispositivos como rádio, memória e sensores, a partir da plataforma do MSPSim que proporciona métodos, classes e interfaces para o desenvolvimento de dispositivos que compõem plataformas de hardware de RSSF, de forma a integrá-los com o microcontrolador.

Utilizando esse ambiente, foi construída a plataforma de hardware SHIMMER. Essa plataforma é composta pelo microcontrolador MSP430F1116, um rádio CC2420, LEDs, memória e sensores da área médica. Aproveitando os recursos e plataformas já existentes do MSPSim, que já traz a implementação de algumas plataformas de hardware, facilitou o desenvolvimento dos componentes centrais da plataforma SHIMMER, como o rádio e o microcontrolador.

A criação de uma nova plataforma de hardware no ambiente do MSPSim consiste na criação dos periféricos e na conexão deles ao microcontrolador, essa conexão via software é análoga a conexão feita no circuito físico.

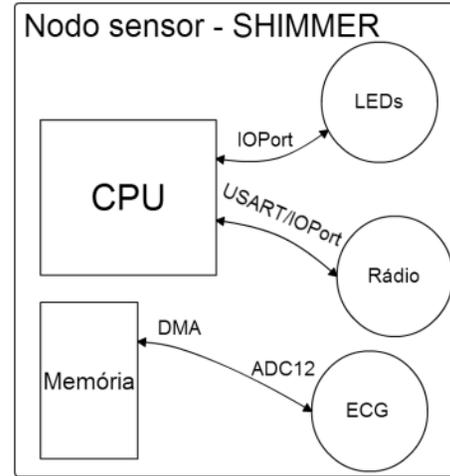


Figura 2. Representação dos periféricos que compõem o nodo sensor SHIMMER na plataforma MSPSim.

Em detalhes, o MSPSim possui um vetor que armazena objetos que representam portas de entrada e saída. É através destas portas que o microcontrolador recebe e envia informações para os periféricos, como por exemplo, pela porta USART (*Universal Synchronous and Assynchronous Recieve and Transmit*). Todas as informações que são enviadas pelo microcontrolador através das portas são recebidas pelo periférico ligado à mesma. O número das portas no MSPSim segue o mesmo padrão do microcontrolador físico, assim, pode-se fazer a ligação dos periféricos que compõem a plataforma SHIMMER seguindo o documento de especificação da própria plataforma física. Em nosso emulador foram conectados os LEDs, o rádio CC2420 e o sensor ECG, como ilustra a Fig. 2.

Antes de entrar nos detalhes de implementação de cada periférico é importante deixar claro que todas as plataformas de hardware implementadas no MSPSim precisam estender a classe *GenericNode*. A classe *GenericNode* é uma classe abstrata cujo principal método é o *setupNode()*, chamado pela *thread* principal do MSPSim para criar e configurar uma instancia da plataforma de hardware escolhida para simulação. No nosso caso, criou-se a classe *ShimmerNode*, que estende *GenericNode*, e no método *setupNode()* conectam-se os periféricos ao microcontrolador. Um trecho de código que ilustra a conexão dos LEDs, rádio e ECG consta na Fig. 3.

Outro ponto importante é entender que a classe *ShimmerNode* implementa os métodos *portWrite()* e *dataReceived()* das interfaces *PortListener* e *USARTListener*, respectivamente. O processo de conexão de um periférico a uma porta consiste em passar uma instancia do *ShimmerNode* a instancia de uma porta, como ilustrado pelo trecho de código *port4.addPortListener(this)*.

Assim, quando surge um dado em uma porta que deve ser enviado para um periférico, a *thread* principal do simulador chama o método *portWrite()* ou *dataReceived()* implementado pela classe *ShimmerNode*, atribuindo a ela a responsabilidade de passar os dados recebidos do microcontrolador, tratá-los e enviá-los para o periférico de interesse.

```

// led
port4 = cpu.getIOUnit(IOPort.class, "P4");
port4.addPortListener(this);

leds = new Leds(cpu, LEDS);

// sensor
ecg = new ECG(cpu);
ADC12 adc = cpu.getIOUnit(ADC12.class, "ADC12");

// serial
USART usart = cpu.getIOUnit(USART.class, "USART1");
usart.addUSARTListener(this);

USART usart0 = cpu.getIOUnit(USART.class, "USART0");
if (usart0 != null) {
    registry.registerComponent("serialio", usart0);
}

//radio
port1 = cpu.getIOUnit(IOPort.class, "P1");
port1.addPortListener(this);

port2 = cpu.getIOUnit(IOPort.class, "P2");
port2.addPortListener(this);

radio.setCCAPort(port2, CC2420_CCA);
radio.setFIFOPort(port1, CC2420_FIFOP);
radio.setFIFOPort(port1, CC2420_FIF0);
radio.setSFDPort(port1, CC2420_SFD);

```

Figura 3. Código do método *setupNode()* da classe *ShimmerNode*, onde os periféricos são conectados a suas respectivas portas.

Na plataforma MSPSim, a classe *Led* mantém o estado de todos os LEDs da plataforma sendo emulada. No caso do SHIMMER, essa classe manterá o estado de cada um dos quatro LEDs que a plataforma física possui. Além de manter o estado dos LEDs, a classe *Led* é também utilizada para refletir o comportamento dos LEDs na interface gráfica do simulador. Para adicionar os LEDs da plataforma SHIMMER ao simulador, é necessário obedecer a descrição do hardware SHIMMER, que contém o número da porta em que os LEDs deverão estar conectados, assim como a sequência de bits que descreve as ações ligar/desligar. Na documentação do SHIMMER, os LEDs são ligados à porta 4.

Para ligar o periférico ao microcontrolador é necessário obter uma instância da classe *IOPort* através do método *getIOUnit()* da classe *CPU* (como ilustrado no primeiro trecho da Fig. 3). Além da conexão com microcontrolador, é necessário definir a quantidade de LEDs na classe *Led*, e definir os valores que espera-se receber da CPU que conterá a informação de qual LED será aceso ou apagado — assim, pode-se tratar a informação recebida pela método *portWrite()* e repassar as ações para a classe *Led*.

O rádio utilizado pela plataforma de hardware física do sensor SHIMMER é o CC2420, periférico bastante utilizado em plataformas de hardware de RSSF que suportam o protocolo de comunicação IEEE 802.15.4. De forma análoga ao periférico LED, uma classe que implementa o comportamento do rádio CC2420 está disponível pelo MSPSim, exigindo a ligação, de acordo com a documentação do SHIMMER, com as portas do microcontrolador. O rádio é conectado ao microcontrolador pelas portas 1, 2 e 5. As informações recebidas pelas portas 1 e 2 são enviadas diretamente para a classe *CC2420*, que é responsável por receber os dados diretamente das portas e realizar as ações esperadas. Em contrapartida, as informações enviadas para a porta 5 são tratadas pelo *ShimmerNode*; é através das

informações enviadas para essa porta que se controla o estado do rádio entre ligado e desligado.

O periférico ECG, característico da plataforma SHIMMER, foi construído com o intuito de realizar testes de aplicações na área médica que fazem uso deste sensor. Esse periférico foi desenvolvido exclusivamente para a plataforma SHIMMER e não há uma implementação deste dispositivo disponível no MSPSim. O eletrocardiograma (ECG) é o registro dos impulsos elétricos resultados da atividade cardíaca. O ECG disponível na plataforma SHIMMER é o ECG de três pontas, isto é, possui três eletrodos para captura dos sinais elétricos resultados da atividade cardíaca; um ligado à esquerda do corpo (LA), um à parte direita (RA) e um conectado à parte esquerda em uma região baixa do corpo (LL); é através do cálculo da diferença de potencial dos sinais coletados dessas pontas que se obtém o sinal do ECG. A classe *ECG* foi criada para gerar valores para cada uma das três pontas e fornecer esses valores ao simulador.

A ligação do ECG à plataforma SHIMMER criada na plataforma MSPSim segue a forma já descrita para os demais periféricos descritos anteriormente. Seguindo a documentação que descreve o hardware SHIMMER, conectamos o ECG ao ADC (conversor analógico-digital), esse componente é responsável por realizar a conversão de um sinal analógico em uma representação digital. Esse conversor é uma classe disponibilizada pelo MSPSim. A integração do ECG ao ADC é análogo às portas (*IOPort* e *USART*, descritas anteriormente). A partir da instância da classe *CPU* pode-se obter uma instância da classe *ADC*, e assim, é possível enviar dados gerados pela classe *ECG* para o conversor, que transformará os dados recebidos para uma representação digital, e enviarão os dados através do DMA (Acesso Direto à Memória) para memória.

V. TESTES

Os testes para verificação do funcionamento da plataforma SHIMMER criada no MSPSim foram feitos utilizando códigos que foram disponibilizados no repositório oficial do TinyOS, na área de exemplos para a plataforma SHIMMER [19]. Os códigos foram compilados e o arquivo executável foi simulado no MSPSim na plataforma SHIMMER criada.

Foram selecionados os códigos exemplos que faziam uso de cada dispositivo periférico principal da plataforma, como por exemplo, o *Blink*, para testar os LEDs; o *RadioCountToLeds*, que utiliza duas instâncias do simulador no qual um nodo sensor envia o valor de um contador para o outro nodo sensor, que acende os LEDs de acordo com o número enviado.

Por fim, foi verificado o funcionamento do ECG, executando código de exemplo que faz uso do ECG físico na plataforma SHIMMER, uma variação do código *AccelECG*, que foi modificado para enviar dados do ECG pelo rádio CC2420 ao invés do Bluetooth.

Ao executar os códigos no ambiente de simulação foi verificado se os periféricos eram ativados e se respondiam de acordo com o esperado; essa verificação se deu por meio da depuração do simulador na IDE Eclipse em modo de depuração. Dessa forma, os testes realizados nos leva a concluir que a plataforma de emulação SHIMMER é capaz de executar códigos escritos para a plataforma real sem a necessidade de nenhuma alteração no código.

VI. CONTRIBUIÇÕES

A principal contribuição deste trabalho é o apontamento da deficiência do desenvolvimento de aplicações de qualidade para RSCH na área médica. Como tratado no artigo [8], existe uma carência de ferramentas que suportam o desenvolvimento de aplicações de qualidade na área de Saúde. Muitos trabalhos, apesar de criarem aplicações específicas da área médica, muitas vezes não testam suas soluções em uma plataforma de hardware que contempla os requisitos da área médica: hardware esterilizável, leve, que não causa irritação à pele, não restringe a movimentação do paciente, etc.

Esses requisitos do hardware para aplicações médicas afetam diretamente em suas características físicas. Por exemplo: um nodo sensor para ser pequeno terá o tamanho de sua bateria reduzida, o que impacta diretamente na sua capacidade energética, que influencia na escolha de microcontroladores, rádios e outros periféricos.

Como solução apresentamos a implementação do nodo sensor SHIMMER utilizando como base a plataforma de simulação MSPSim. Pois, além de oferecer uma plataforma de RSCH que leva em consideração os requisitos da área médica, o simulador oferece um ambiente que facilita a verificação do comportamento das aplicações criadas para área da saúde.

O simulador aqui apresentado é integrado com tecnologias de código aberto amplamente utilizadas para desenvolvimento de aplicação de RSSF de propósito geral. Assim, vê-se necessária a separação dos benefícios obtidos da implementação do próprio nodo sensor SHIMMER como as contribuições obtidas da utilização do ambiente de base, o MSPSim. A implementação de um simulador do nodo sensor SHIMMER, como ferramenta que auxilia o desenvolvimento de aplicações de RSCH na área da saúde, traz as seguintes contribuições:

- Uma ferramenta que possibilita o desenvolvimento de aplicação de RSCH específico para área de Saúde, independente de se ter o nodo sensor real;
- O provimento de um ambiente que permite a depuração de código, dada a criticidade das aplicações da área de Saúde e a dificuldade de verificar código em plataformas de hardware físicas, como o próprio SHIMMER;
- Um simulador capaz de ser integrado com ferramentas de simulação de código aberto que estão sendo vastamente utilizadas, como o MSPSim e o COOJA.

Contribuições para a área de Saúde por utilizar uma plataforma de simulação MSPSim como base da implementação do nodo sensor SHIMMER:

- Um ambiente de simulação não atrelado a tecnologia específica, sendo possível o desenvolvimento de aplicações utilizando qualquer sistema operacional e linguagem, importando somente o código executável compatível com o conjunto de instruções do microcontrolador;
- Um ambiente que provê uma interface de verificação da aplicação de RSCH, provendo mecanismos de depuração do código, ciclo de trabalho do nodo sensor, sendo possível verificar através de gráficos a utilização em termos de porcentagem o uso do microcontrolador e dos dispositivos de nodo sensor como o rádio (transmissão e recebimento), LEDs, etc.

- Facilidade de integração com o COOJA de forma a poder simular o comportamento de redes de sensores completas, definindo topologias, rotas de comunicação, etc.

A utilização do MSPSim, como base de um ambiente de simulação, permite a criação de outras plataformas de hardware para a área de Saúde, tornando-as disponíveis para o teste de aplicações escritas para a plataforma em questão. Ao utilizar o MSPSim, é possível também utilizar os dispositivos já implementados além de implementar novos que possam ser utilizados para criar um novo tipo de plataforma de hardware, e testar aplicações escritas para elas antes de implementar o hardware fisicamente.

Desta forma mostra-se a flexibilidade do ambiente de simulação do MSPSim, que provê uma base para a construção de diversos tipos de plataformas de hardware.

VII. CONCLUSÕES

As RSCH têm um enorme potencial em aberto para aplicações na área médica, sendo uma delas o monitoramento remoto de indivíduos. Todavia, são necessárias ferramentas que auxiliem no desenvolvimento de aplicações de qualidade, principalmente levando em consideração a criticidade de aplicações da área médica.

Neste trabalho propomos um ambiente de simulação com o objetivo de oferecer um ambiente de teste de baixo custo e alto nível para aplicações específicas de RSCH na área de Saúde. Esse ambiente de simulação fornece funcionalidades essenciais para que aplicações possam ser devidamente testadas a fim de garantir a qualidade demandada pela área médica, permitindo, em especial: o teste de aplicações desenvolvidas em diferentes tecnologias (TinyOS, Contiki, etc.), a simulação de dados de sensores específicos da saúde, e a troca de mensagens entre nodos sensores.

O ambiente de simulação emula a plataforma de hardware SHIMMER utilizando como base a plataforma *open source* MSPSim, visando a facilidade da adoção pela comunidade de pesquisadores e desenvolvedores que estão familiarizados com o Contiki e a cadeia de ferramentas que acompanha o sistema operacional. A escolha da plataforma de hardware SHIMMER deu-se por ser uma plataforma amplamente utilizada para aplicações de RSCH e por implementar a maior parte dos requisitos de hardware demandados na área médica, como por exemplo, ser leve, vestível e não restringir os movimentos.

Os testes da aplicação foram feitos utilizando códigos escritos especificamente para a plataforma de hardware real, obtidos diretamente do repositório oficial da plataforma, e executadas na plataforma simulada observando atentamente, através de um depurador, se o comportamento da plataforma emulada coincidia com o esperado na plataforma real. Os testes mostram que códigos escritos para a plataforma real são executados da forma esperada, sem necessidade de alteração.

Desta forma, com o desenvolvimento do simulador da plataforma SHIMMER, espera-se contribuir com o amadurecimento da tecnologia de RSCH, sendo o simulador uma ferramenta para o auxílio na construção de aplicações de qualidade na área médica.

AGRADECIMENTOS

Roberto Vito Rodrigues Filho gostaria de agradecer à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pelo financiamento de sua bolsa de estudos BEX13292/13-7.

REFERÊNCIAS

- [1] A. Bruns, B. R. Greene, M. J. Mcgrath, T. J. O'shea, B. Kuris, S. M. Ayer, F. Stroiescu, and V. Cionca. SHIMMER - A Wireless Sensor Platform for Noninvasive Biomedical Research. *IEEE Sensors Journal*, Vol. 10, p. 1527-1534, 2010.
- [2] S. T. Carvalho, A. Copetti, and O. Loques. Um Sistema Computacional Inteligente de Assistência Domiciliar à Saúde. *Proceedings of the XII Brazilian Congress of Health Informatics*, Porto de Galinhas, Brazil, 2010, p 1-6.
- [3] A. Copetti, J. Leite, O. Loques, A. C. L. Nobrega, and T. P. C. Barbosa. Monitoramento Inteligente e Sensível ao Contexto na Assistência Domiciliar Telemonitorada. *Proceedings of the XXXV Seminário Integrado de Software e Hardware*, Belém, Brazil, 2008.
- [4] E. F. A. Costa, C. C. Porto, and A. T. Soares. Envelhecimento populacional brasileiro e o aprendizado de geriatria e gerontologia. *Revista da UFG*, Vol. 5, No. 2, 2003.
- [5] D. Geer. Pervasive Medical Devices: Less Invasive, More Productive. *IEEE Pervasive Computing*, Vol. 5, No. 2, p. 85-88, 2006.
- [6] A. V. F. Gonçalves. Avaliação do Acolhimento no Serviço de Emergência do Hospital de Clínicas de Porto Alegre na Perspectiva da Pessoa Idosa. *Dissertação de Mestrado em Enfermagem*, UFRGS, 2011.
- [7] M. L. C. Lima, E. R. Souza, R. M. L. Acioli, and E. D. Bezerra. Análise dos serviços hospitalares clínicos aos idosos vítimas de acidentes e violências. *Ciência Saúde Coletiva*. Vol. 15, No. 6, p. 2687-2697, 2010.
- [8] R. V. Rodrigues-Filho, R. F. Bulcão-Neto, B. O. Silvestre, L. L. G. Oliveira, R. O. Oliveira, and I. G. Sene-Júnior. An Evaluation Method of Research on Wearable Wireless Body Area Networks in Healthcare. *International Journal of Computer Science and Information Technology*, Vol. 5, p. 65-78, 2013.
- [9] M. I. Schmidt, B. B. Duncan, G. A. Silva, A. M. Menezes, C. A. Monteiro, S. M. Barreto, D. Chor, and P. R. Menezes. Doenças crônicas não transmissíveis no Brasil: carga e desafios atuais. *The Lancet*. London, p. 61-74, 2011.
- [10] V. Shnayder, B. Chen, K. Lorincz, T. R. F. Fulford-Jones, and M. Welsh. Sensor networks for medical care. *Proceedings of the 3rd international conference on Embedded networked sensor systems*, San Diego, California, USA, p. 2-4, 2005.
- [11] A. Wood, G. Virone, T. Doan, Q. Cao, L. Selavo, Y. Wu, L. Fang, Z. He, S. Lin, and J. Stankovic. Alarm-Net: Wireless sensor networks for assisted-living and residential monitoring. University of Virginia Computer Science Department, *Technical Report*, 2006.
- [12] T. Hasiotis, G. Alyfantis, V. Tsetsos, O. Sekkas, and S. Hadjithymiades. Sensation: A middleware integration platform for pervasive applications in wireless sensor networks. *Proceedings of the Second European Workshop on Wireless Sensor Networks*, Istanbul, Turkey, p. 366-277, 2005.
- [13] K. Tei, Y. Fukazawa, and S. Honiden. Applying design patterns to wireless sensor network programming. *Proceedings of 16th International Conference on Computer Communications and Networks*, Honolulu, USA, p. 1099-1104, 2007.
- [14] M. Serna, C. J. Sreenan, and S. Fedor. A visual programming framework for wireless sensor networks in smart home applications. *Proceedings of the IEEE Tenth International Conference on Intelligent Sensors, Sensor Networks and Information Processing*. Singapore, p. 1-6, 2015.
- [15] J. Eriksson, F. Österlind, N. Finne, N. Tsiftes, A. Dunkels, T. Voigt, R. Sauter, and P. J. Marrón. COOJA/MSPSim: interoperability testing for wireless sensor networks. *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, Brussels, Belgium, p. 1-7, 2009.
- [16] K. Lorincz, B. Kuris, S. M. Ayer, S. Patel, P. Bonato, and M. Welsh. Wearable wireless sensor network to assess clinical status in patients with neurological disorders. *Proceedings of the 6th international conference on Information processing in sensor networks*. Cambridge, MA, USA, p. 563-564, 2007.
- [17] G. Fortino, D. Parisi, V. Pirrone, and G. Di Fatta. BodyCloud: A SaaS approach for community body sensor networks. *Future Generation Computer Systems*. Vol. 35, p. 62-79, 2014.
- [18] C. C. Y. Poon, B. P. L. Lo, M. R. Yuce, A. Alomainy, and Y. Hao. Body Sensor Networks: In the Era of Big Data and Beyond. *IEEE Reviews in Biomedical Engineering*. Vol. 8, p. 4-16, 2015.
- [19] Shimmer research repository for TinyOS applications. Disponível online em <https://github.com/ShimmerResearch/tinyos-shimmer>, 2014.
- [20] V. L. Bezerra, L. B. Leal, M. V. Lemos, C. G. Carvalho, J. Bringel Filho, and N. Agoulmine. A pervasive energy-efficient ECG monitoring approach for detecting abnormal cardiac situations. *Proceedings of the IEEE 15th International Conference on e-Health Networking, Applications & Services*, Lisbon, Portugal, p. 340-345, 2013.
- [21] M. Oliveira, L. O. M. Andrade, M. Santos, R. Alcantra, G. Teles, and N. Agoulmine. Towards a cost-effective homecare for a public health management system in Brazil. *Proceedings of the IEEE 16th International Conference on e-Health Networking, Applications & Services*, Natal, Brazil, p. 206-211, 2014.
- [22] M. Chen, S. Gonzalez, A. Vasilakos, H. Cao, and V. C. M. Leung. Body area networks: A survey. *Mobile networks and applications*. Vol. 16., No. 2, p. 171-193, 2011.
- [23] Y. Hao, and R. Foster. Wireless body sensor networks for health-monitoring applications. *Physiological measurement*. Vol. 29, No. 11, R27-R56, 2008.
- [24] A. Pantelopoulos, and N. G. Bourbakis. A survey on wearable sensor-based systems for health monitoring and prognosis. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, Vol. 40, No. 1, p. 1-12, 2010.
- [25] Y. Lyu, J. Hong, Y. Wei, J. Yang, Y. Tang, W. Wang, and N. Agoulmine. N. Dynamic evaluation model of coronary heart disease for ubiquitous healthcare. *Computers in Industry*. Vol. 69, p. 35-44, 2015.



Roberto Vito Rodrigues Filho

Mestre em Ciência da Computação pelo INF-UFG. Atualmente, é aluno de doutorado na Universidade de Lancaster, Reino Unido. Suas áreas de pesquisa incluem Sistemas Emergentes e Auto-Adaptáveis e Virtualização e Emulação de Arquiteturas de Hardware para Redes de Sensores para o Corpo Humano.



Renato de Freitas Bulcão Neto

Doutor em Ciência da Computação pelo ICMC-USP, é pesquisador do Instituto de Informática da Universidade Federal de Goiás, Brasil. Seus interesses em pesquisa envolvem a intersecção das áreas de Computação Sensível a Contexto, Engenharia de Software e Web Semântica.



Iwens Gervásio Sene Júnior

Doutor em Engenharia Elétrica pela Universidade de Brasília (UnB), é professor do Instituto de Informática da Universidade Federal de Goiás (INF-UFG). Suas áreas de pesquisa incluem Redes de Sensores Sem Fio, Segurança da Informação e Aplicações para Healthcare.