

Sistema operacional em tempo real LiteOS

Miguel Gut Seara, UCPEL

Resumo—Com o crescimento do interesse na utilização de dispositivos embarcados na onda atual de IoT (internet das coisas), diferentes sistemas operacionais surgem com a proposta de melhor compatibilidade com os requisitos do mercado. A partir da baixa potência, memória de tamanho reduzido, interconexão através de diversos protocolos de comunicação nasce uma gama de sistemas voltados para uso comercial, pessoal e industrial. O Huawei LiteOS vêm se mostrando um sistema operacional em tempo real e open-source, mantido pela gigante da tecnologia Huawei, muito potente e valioso para uso geral e específico de desenvolvimento e manutenção de aplicações IoT. Seu valor fica notável ao ser utilizado para a criação de novos dispositivos das mais variadas funcionalidades, como automóveis, celulares, eletrodomésticos, dispositivos vestíveis como relógios, até mesmo máquinas de uso industrial.

Palavras-Chave—RTOS, Huawei, LiteOS, IoT.



1 Introdução

Os objetos usados no dia a dia estão sendo equipados com dispositivos eletrônicos e suítes de protocolos para torná-los interconectados e conectados à Internet. De acordo com um estudo recente do Gartner [1], 50 bilhões de objetos conectados seriam implantados em cidades inteligentes até 2020. Falando de IoT, estima-se que trilhões de dispositivos estejam em uso em um futuro próximo [2]. Para a construção de dispositivos IoT devemos avaliar o uso de recursos limitados, com mais restrições do que o normal para sistemas embarcados tradicionais.

Recentemente, para definir a diferença entre esse sistema e um computador desktop, introduziu-se o termo “*constrained device*” (tradução literal para dispositivo restrito) [3]. Isto implica em uma redução significativa do consumo de energia, menor potência computacional e processamento, assim como a quantidade de memória significativamente reduzida. Tendo em mente que os dispositivos restritos são baseados principalmente em micro-controladores, que geralmente não têm memória unidades de gestão, seu software de sistema exclui soluções que são comuns em sistemas embarcados.

Outra dimensão também abordada pelo IoT é a necessidade exigida por algumas aplicações de

trabalhar em tempo real, que ainda mais restringe nossa escolha de dispositivos, sistemas operacionais e software. Consequentemente, sistemas operacionais em tempo real (RTOS) devem satisfazer as demandas de desempenho, oferecer resposta de tempo real e lidar com restrições de memória, mas cada vez mais eles também precisam fornecer recursos exigidos pelo novo mundo de máquinas e redes altamente conectadas, preocupadas com a segurança e gerenciadas remotamente. Como é dito comumente na computação, “não existe bala de prata” para se referir que dificilmente (ou nunca) é possível que exista uma solução ótima que seja aplicável a todos os contextos práticos.

Para sistemas RTOS a afirmação é válida: para escolha do melhor sistema devemos levar em conta os seguintes obstáculos:

- Desejamos trabalhar com uma distribuição linux?
- A aplicação é para utilização industrial ou para consumidor final?
- Qual deve ser o grau de escalabilidade do sistema?
- Desejamos trabalhar com modularidade?
- O quão importante é a conectividade para o funcionamento do sistema?
- Há problema em utilizar uma solução open source?

Dentre os inúmeros sistemas operacionais em tempo real populares e disponíveis atualmente,

• **Miguel Gut Seara:** Engenharia de Computação. Universidade Católica de Pelotas - UCPEL.
E-mail: miguel.seara@sou.ucpel.edu.br

destacam-se [4] na categoria open source *RIOT*, *Nano-RK*, *FreeRTOS*, *Apache Mynewt*, *ARM mbed OS*, *Raspbian* e *uOS*. Já as soluções comerciais populares atualmente, podemos citar como exemplo os sistemas *TI RTOS*, *Microsoft Azure (ThreadX)*, *QNX*, *VxWorks*, *SafeRTOS (Commerical FreeRTOS)*, *SCIOPTA*, *RTX*, dentre outros.

Neste trabalho, será avaliado mais profundamente o funcionamento do sistema operacional LiteOS, criado pela empresa telefônica Huawei. Este sistema surgiu como uma alternativa leve e de baixa potência, disponível para inúmeras aplicações de mundo real, desde smart homes (casas inteligentes), wearables (dispositivos de vestir), Internet of Vehicles (focado para automóveis) a manufatura inteligente, que propõe remover barreiras e facilitar o desenvolvimento de aplicações [5].

2 O sistema LiteOS

2.1 História

Em 2012, a empresa Huawei desenvolveu sistemas embarcados para melhor compatibilidade com seus eletrônicos [5]. Em 2014, o sistema *LiteOS* foi implementado em wearables e telefones celulares, das séries *Mate*, *P* e *Honor*. Um ano depois, em 2015, o sistema operacional *Huawei LiteOS IoT* foi lançado e revelou uma solução IoT “1+2+1” no *Huawei Network Congress* (HNC). O termo “1+2+1” indica **uma** plataforma IoT + **dois** modos de acesso + **um** sistema operacional leve IoT.

2.2 Características técnicas

Huawei LiteOS é um sistema operacional leve e de código aberto projetado para dispositivos IoT, dispositivos wearable e outros sistemas de computação sensíveis ao tempo e energia. O tamanho básico do kernel é menor que 10K e o tempo de resposta está dentro de 100 µs em plataformas típicas de processadores embarcados. Os principais componentes do kernel básico do Huawei LiteOS são mostrados na Figura 1.

Seu kernel leve permite que parceiros e desenvolvedores consigam rapidamente iniciar o processo de desenvolvimento de hardware baseado no LiteOS, criando produtos altamente competitivos que tenham inicialização rápida e baixo consumo de energia.

Huawei LiteOS Kernel

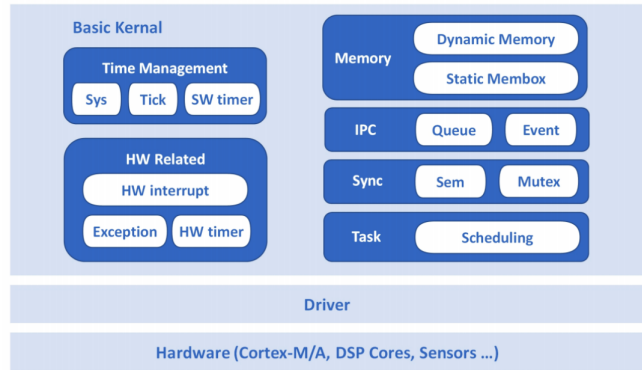


Figura 1. Kernel básico do LiteOS

O Huawei LiteOS fornece um framework de sensores para fornecer uma rede de baixo atraso e alta precisão de sensoriamento. Desde as últimas versões do sistema operacional, o atraso de resposta foi reduzido em 50% e sua precisão aumentou em mais que o dobro. Além disso, algoritmos de coleção simples foram substituídos por algoritmos inteligentes.

Sua engine de conectividade oferece suporte a um vasto conjunto de tecnologias necessárias para comunicação, como *Wi-Fi*, *Bluetooth*, *Zigbee*, *Ethernet*, e *NB-IoT* para interação entre máquinas e dispositivos IoT. Com ele é possível interagir com protocolos e cenários distintos, desde smart homes, dispositivos wearable e até cenários industriais.

Um grande diferencial do LiteOS são suas APIs e definição de perfis de serviço na camada de descrição de serviço, ajudando desenvolvedores a criar suas aplicações de maneira mais ágil, permitindo a interoperação entre dispositivos. Ao disponibilizar em sua interface o uso de APIs, o LiteOS fornece um ecossistema unificado que pode ser utilizado em diversos domínios IoT.

Sua engine Operacional para sustentar a uma máquina virtual leve e de alto desempenho baseada em JavaScript e máquinas virtuais JavaScript. Sua ROM é pequena e utiliza pouca memória, fornecendo espaço distinto para utilização de usuário e aplicação, garantindo maior segurança. Com suporte à autenticação de dois fatores, transmissão criptografada DTLS, e upgrades remotos para terminais fracos em cenários LPWA (*Low-Power Wide-Area*, traduzindo literalmente para Baixa Potência Alta Área). Por terminais fracos, podemos citar medidores de água, gás ou veiculares



Figura 2. Ampla espectro de utilizações do LiteOS e suas interconexões

que possuem capacidade e recursos limitados para memória, armazenamento e CPU, e requisições restritas em custo e consumo de potência.

3 Considerações Finais

Com a crescente demanda por aparelhos IoT em uso em um futuro próximo, um desafio fundamental para implantar e manter um número tão grande de dispositivos é como alimentá-los de maneira eficiente, econômica e sustentável. O rápido desenvolvimento da tecnologia de hardware e software de computação de potência ultra-baixa tornaram possível construir dispositivos IoT que atendessem a essa necessidade.

Também podemos destacar o novo paradigma chamado “Computação Intermitente”, cuja proposta tem chamado atenção nos últimos anos: Um serviço que possa funcionar considerando que a energia não estará sempre disponível, para que ele possa ser interrompido sem aviso prévio e retomado a partir de algum ponto na história de sua execução sem prejuízo significativo. Em um sistema de captação de energia, as falhas de energia podem ocorrer com muita frequência (por exemplo, centenas de vezes por segundo). Portanto, as pessoas devem projetar o sistema de maneira ciente da intermitência para que ele possa fazer um progresso significativo na presença de falhas de energia frequentes. A chave para viabilizar a computação intermitente é o sistema operacional, que fornece serviços fundamentais para suportar a execução intermitente correta de

aplicativos sob fonte de alimentação imprevisível, motivo que faz com que o sistema LiteOS funcione em excelente simbiose com esta proposta, através do reaproveitamento do seu kernel, devido ao seu tempo de resposta veloz e memória base muito pequena [2].

Podemos observar com as diversas aplicações possíveis que o sistema LiteOS é um sistema operacional multi-uso e de grande relevância para o universo IoT e também outros paradigmas da computação. Suas facilidades de programação devido às suas engines completas em termos de conectividade fácil e integrada, memória de baixa potência e APIs representam uma diferencial atraente para programadores que desejam embarcar rapidamente no mundo da programação de dispositivos interoperáveis IoT.

Referências

- [1] Badis Hammi, Rida Khatoun, Sherali Zeadally, Achraf Fayad, and Lyes Khoukhi. Iot technologies for smart cities. *IET Networks*, 7(1):1–13, 2017.
- [2] Nan Guan and Qiulin Chen. Liteos for intermittent computing. *RTSS@ WORK 2019*, page 5, 2019.
- [3] Aleksandar Milinković, Stevan Milinković, and Ljubomir Lazić. Choosing the right rtos for iot platform. *Infoteh Jahorina*, 14:504–9, 2015.
- [4] Carsten Gregersen. How to choose the best rtos for iot devices, 2020.
- [5] Ltd. Huawei Technologies Co. Huawei liteos, 2020.



Miguel Gut Seara é graduando em Engenharia de Computação da Universidade Católica de Pelotas. Tem experiência na área de computação na nuvem e programação, com ênfase em desenvolvimento web, banco de dados e utilização de frameworks CakePHP e Drupal.