

Segurança e monitoramento com RFID

IoT aplicada à segurança e ao monitoramento

Aubani Júnio Teixeira Cândido
Universidade de Brasília - UnB
Brasília-DF, Brasil
junio@aluno.unb.br

Elpidio Cândido De Araujo Bisneto
Universidade de Brasília - UnB
Brasília-DF, Brasil
elpidio.araujo@hotmail.com

Resumo— Uso de conceitos de Internet das Coisas, Internet of Things (IoT), para a implementação de um sistema de segurança e monitoramento de um ambiente através de acesso controlado de RFID e sincronizado em nuvem.

Keywords—monitoramento, segurança, IoT, RFID

I. JUSTIFICATIVA

Tendo em vista a necessidade de sistemas de segurança que controlem o acesso a um ambiente, exemplo de laboratórios, é necessário fazer o gerenciamento do acesso. Uma forma eficiente de realizar essa tarefa é usando a tecnologia de RFID aliada aos conceitos de IoT e armazenamento em nuvem. Proporcionando um controle daqueles que tiveram acesso e daqueles cadastrados para acessarem o ambiente.

II. OBJETIVO

A. Garantir segurança no acesso

Somente pessoas autorizadas por um dos administradores do ambiente, teriam acesso ao ambiente, impedindo que outros não convidados ou pessoas de fora tenham acesso ao local, garantindo integridade e sigilo. Tudo isso com a tecnologia de RFID.

B. Monitoramento do acesso

Mesmo tendo pessoas autorizadas acessando o ambiente é necessário que haja um monitoramento, informando quem acessou o ambiente, que horas acessou e quanto tempo ficou dentro do ambiente. Todos esses dados serão sincronizados em nuvem de forma criptografada, possibilitando consulta em outros dispositivos. Nessa parte também entra o conceito de IoT, fazendo uso de sensores que serão *Things* para interação com a internet via módulo Wi-Fi num servidor controlado.

III. TABELA DE MATERIAIS UTILIZADOS

UND	MATERIAIS	FABRICANTE
01	MSP430G2553LP	TEXAS INSTRUMENT
01	RFID MÓDULO RC522	NXP
03	CARTÕES COM RFID	-
01	MODULO WIFI - ESP8266 ESP-01	ESPRESSIF SYSTEMS
-	JUMPERS	-
01	PROTOBOARD	HIKARI
01	MODEM DSL-2401HN-T1C-NV	MITRASTAR
01	DISPLAY LCD COM MÓDULO 12C	-
01	BUZZER 5V	-
01	TRANSISTOR BC547 (NPN)	FAIRCHILD SEMICONDUCTOR
01	SERVOMOTOR 9G SG-90	TOWER PRO
01	RESISTOR 1KΩ	-
01	RESISTOR 10KΩ	-

IV. HARDWARE

Para a realização desse projeto utilizamos os pinos (colocar os pinos) para fazer a comunicação entre o RFID e a MSP430, utilizamos as entradas para SPI e UART para realizar a comunicação com o módulo RFID (SPI) e módulo Wi-Fi

Assim que o usuário aproxima o cartão no RFDI, o sistema enviará esses dados para a nuvem, informando que alguém está tentando acessar o local. Se o cartão for cadastrado o acesso é liberado, caso contrário, será mostrado acesso negado e registrado na nuvem.

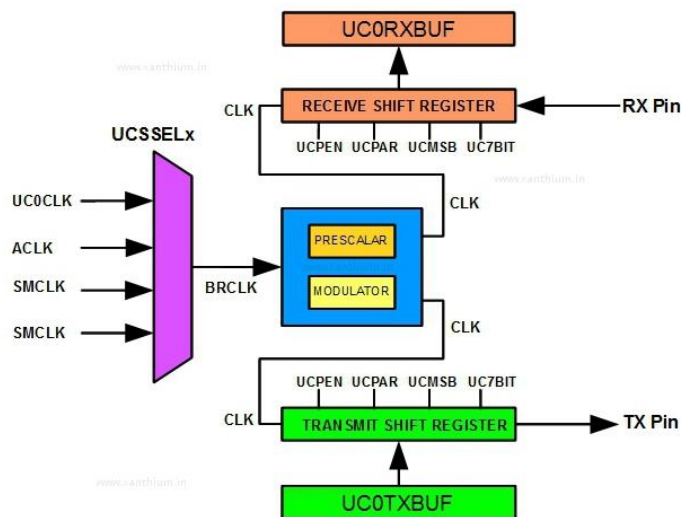


Figura 3 – Diagrama dos registradores para UART [3].

A figura 3 mostra os registradores utilizados para fazer a manipulação do UART no MSP430. O algoritmo consiste em usar esses registradores fazendo interrupção no TX e RX, ou seja, ao receber ou enviar pelo UART, executar um ISR

```
interrupt(USCIB0TX_VECTOR) USCIB0TX_ISR(void)
{
    //Pega os valores passados nos parâmetros da função
    byte* buf = Serial_UART.TxBuffer;
    volatile int* i = &Serial_UART.iTx;
    volatile byte* lock = &Serial_UART.lockTX;
    *i++;
    // Compara se o valor é maior ou igual com o tamanho que deve ler
    // >= caso o lenght seja 0, ele lê 1 vez da mesma forma.
    if (*i >= Serial_UART.lenghtTX)
    {
        *lock = 0;
        UC0IE &= UCA0TXIE;
    }

    else //Coloca o valor para ser enviado no BUF do TX
        UCA0TXBUF = buf[*i];

    //Desabilita a flag do TX
    //IFG2 &= ~UCA0TXIFG;
}
```

Figura 4 – Código de interrupção TX – UART

A figura 4 mostra o trecho de código de interrupção para envio de dados pelo Serial UART. Ele usa parâmetros dentro da classe “Serial_UART” para fazer o envio correta das informações. Mais trechos do código se encontram no link: <https://github.com/Metalus/Microcontroladores/tree/master/Ponto%20de%20controle/Ponto%20de%20controle%203>.

A outra parte do sistema de monitoramento utiliza o RFID, nos primeiros testes foi utilizado bibliotecas do próprio Energia para avaliar o funcionamento do módulo. Após, foi realizado o código para o SPI utilizando os registradores do próprio MSP430, entretanto, diferentemente do código para UART,

nesse não foi utilizado rotinas de interrupção. Considerando que o SPI tem modo de operação *full-duplex*, o código se torna mais eficiente escrevendo uma função para ler e escrever ao mesmo tempo, a figura 5 mostra o trecho de envio e recebimento do SPI.

```
unsigned char Serial_SPIClass::transfer(unsigned char data)
{
    UCB0TXBUF = data; // enviar o byte
    while (UCB0STAT & UCBUSY); //aguardar o registrador
    return UCB0RXBUF; // ler o valor recebido
}
```

Figura 5 – Código SPI.

VI. REQUISITOS

Uma placa MSP430, um módulo Wi-Fi, um módulo de RFID, Cartão RFID e uma nuvem para armazenamento dos dados. Estudo sobre protocolo TCP, conceitos de IoT, funcionamento de RFID e comunicação com a plataforma em nuvem utilizada.

VII. BENEFÍCIOS

O sistema irá se beneficiar de tecnologia mais atual para a autorização de pessoas como o RFID, sendo cada cartão único para ter acesso ao ambiente. Uso de sistema em nuvem para monitoramento, tornando-o eficaz para análise de controle ao ambiente. Uso de conceitos de IoT, para uma nova ponta de tecnologia, onde temos sensores como *Things*, mantendo essa abstração o sistema pode ser complementado para interação com outros. Baixo custo de projeto e de consumo.

VIII. RESULTADO

A dupla conseguiu passar os códigos do Energia para o CCS, escrevendo todos os códigos que estão em anexo, a MSP está comunicando bem com o modulo RFID, já o Modulo WiFi não houve uma Comunicação boa, a dupla tentou encontrar algum erro de código ou de montagem na protoboard, porém não foi encontrado nenhum erro nesses quesitos, o mais provável é que houve algum problema com o modulo. Também está sendo montado o circuito simulador da porta para ser apresentado no último ponto de controle. Posteriormente será substituído o modulo WiFi ou por um novo ou por uma comunicação diretamente com o computador.

REFERÊNCIAS

- [1] fmilburn3. Biblioteca RC522 para MSP430. Disponível em <https://github.com/fmilburn3/CardReader_RFID_RC522>
- [2] Energia. Mapeamento dos pinos. Disponível em <<http://energia.nu/pin-maps/>>
- [3] Xanthium. Imagem do diagrama dos registradores para UART no MSP430. Disponível em <<http://www.xanthium.in/sites/default/files/site-images/serial-com-msp430-uart/uart-msp430-block-dia.jpg>>