

EC 020 – TÓPICOS DE COMPUTAÇÃO I

Felipe Godinho de Almeida  
Gabriel Fernando Cunha Santos

Prof. Edielson Prevato Frigieri

**Abstract** – *This article presents a brief introduction of a project using the LPCXpresso Base Board, a microcontroller board composed of sensors of luminosity, accelerometer, temperature sensor, A/D converter, among other resources. The developed design uses the temperature sensor to read the ambient temperature, printing this value on the OLED display and sending by serial for a better control.*

**Resumo** – Este artigo apresenta uma breve introdução de um projeto utilizando o LPCXpresso Base Board, uma placa microcontroladora composta por sensores de luminosidade, acelerômetro, sensor de temperatura, conversores A/D, entre outros recursos. O projeto desenvolvido utiliza o sensor de temperatura para leitura da temperatura ambiente, imprimindo este valor no *display* OLED e enviando via serial para um melhor controle.

**Palavras-Chave:** LPCXpresso Base Board, Sensor de Temperatura, *display* OLED.

## I. INTRODUÇÃO

O LPCXpresso Base Board é uma placa baseada em microcontrolador que permite desenvolver projetos para variadas aplicações pois possui uma vasta quantidade de funcionalidades. A placa é composta por uma série de periféricos para facilitar a aprendizagem e o desenvolvimento de um programa, suas características variam desde entradas analógicas e digitais, barramentos seriais, alimentação, etc.

## II. CARACTERÍSTICAS INICIAIS DO PROJETO

Foi desenvolvido o projeto para leitura da temperatura em qualquer ambiente, seja esse interno ou externo. O sistema foi implementado utilizando *super loop* e linguagem C. A Figura 1 mostra seu funcionamento através de um fluxograma.

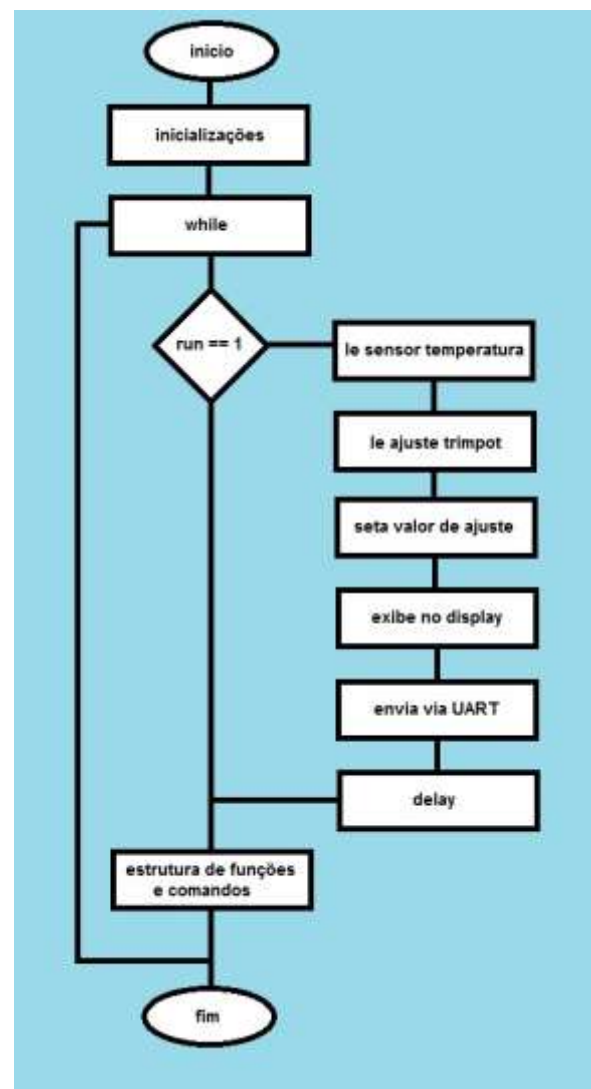


Figura 1 - Fluxograma de funcionamento do projeto

### III. DESCRIÇÃO DO SENSOR E SUAS PRINCIPAIS CARACTERÍSTICAS

O MAX6576/MAX6577 é um sensor de temperatura integrado a placa LPCXpresso e conectado ao LPC1769 através de um pino GPIO digital (PIO0\_2 ou PIO1\_5), dependendo de como o jumper J25 for utilizado. Sua faixa de temperatura varia de -40°C à +125°C e opera entre 2,7V e 5,5V.

A Figura 2 mostra a localização do sensor de temperatura destacada em amarelo.

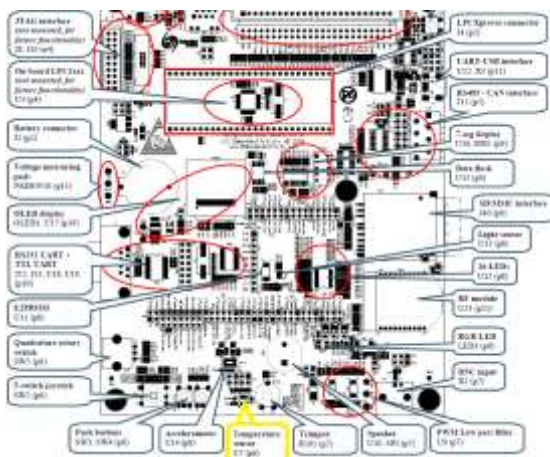


Figura 2 - Localização do sensor de temperatura

#### IV. DESCRIÇÃO DO DISPLAY OLED

O *display* OLED pode ser conectado utilizando tanto o barramento SPI quanto I2C, para permitir que o GPIO\_29 controle a voltagem do OLED sempre se deve inserir um jumper em J44. Para o projeto em questão foi utilizada a interface I2C. A Figura 3 mostra as devidas ligações destacadas em amarelo.

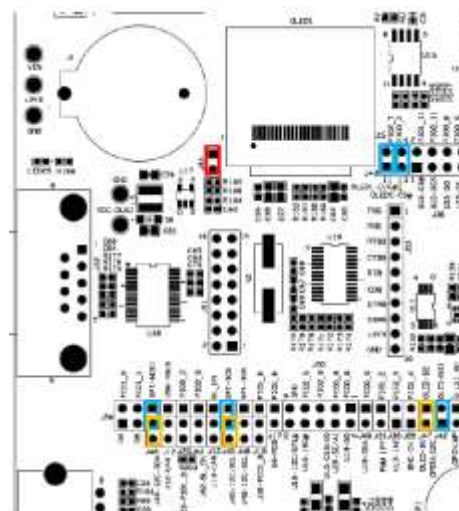


Figura 3 - Localização dos jumpers para o funcionamento do OLED

## V. DESCRIÇÃO DE FUNCIONAMENTO DO SOFTWARE

O software foi desenvolvido para efetuar a leitura constante da temperatura de qualquer ambiente, um limite de temperatura máximo e mínimo é escolhido pelo usuário via interface serial. Caso a temperatura lida pelo sensor for maior que o limite máximo, o *led* vermelho acenderá. Se a temperatura medida estiver abaixo do limite mínimo o *led* azul será acionado, e por fim, se o valor de temperatura estiver dentro do limite máximo e mínimo o *led* branco permanecerá aceso. Além disso, o usuário ainda consegue ajustar o sensor de temperatura utilizando um *trimpot*, o valor da temperatura lida pelo sensor e do ajuste feito é exibido no *display* como mostra a Figura 4.



Figura 4 – OLED display exibindo a temperatura e ajuste

O software possui um menu inicial com as seguintes opções:

- Pressionar ‘P’ para pausar o sistema
- Pressionar ‘I’ para iniciar o sistema
- Pressionar ‘+’ para aumentar a velocidade de leitura do sensor
- Pressionar ‘-’ para diminuir a velocidade de leitura do sensor
- Pressionar ‘A’ para exibir a lista de comandos
- Pressionar ‘1’ para alterar o limite máximo da temperatura
- Pressionar ‘2’ para alterar o limite mínimo da temperatura.
- Pressionar ‘espaço’ para resetar a velocidade de leitura

O usuário pode escolher entre as opções e receber informações através de comandos enviados ou recebidos pela comunicação serial via UART.

A Figura 5 mostra o código para leitura do comando recebido pela porta serial alterando as funcionalidades do sistema.

```
//UART = receive
len = UART_Receive(UART_DEV, &data, 1, MODE_BLOCKING);

if(len > 0){
    if((data == 'I') || data == 'I') && run == 0){
        run = 1;
        UART_SendString(UART_DEV, (uint8_t*)"SISTEMA INICIADO - \n\n");
    }
    if((data == 'p') || data == 'P') && run == 1){
        run = 0;
        UART_SendString(UART_DEV, (uint8_t*)"SISTEMA PARADO - \n\n");
    }
    if((data == '+' || data == '+') && run == 1){
        if(++inc > 5000){
            UART_SendString(UART_DEV, (uint8_t*)"VELOCIDADE AUMENTADA - \n\n");
            while UART_SendString(UART_DEV, (uint8_t*)"VELOCIDADE MAXIMA - \n\n");
        }
    }
    if((data == '-' || data == '-') && run == 1){
        if(--inc < 500){
            UART_SendString(UART_DEV, (uint8_t*)"VELOCIDADE DIMINUIDA - \n\n");
            while UART_SendString(UART_DEV, (uint8_t*)"VELOCIDADE MINIMA - \n\n");
        }
    }
    if((data == '2') && run == 1){
        inc = 50;
        UART_SendString(UART_DEV, (uint8_t*)"VELOCIDADE RESETA - \n\n");
    }
    if((data == '1') && run == 1){
        inc = 5000;
        UART_SendString(UART_DEV, (uint8_t*)"VELOCIDADE MINIMA - \n\n");
    }
    if((data == 'a') || data == 'A'){
        run = 0;
        UART_SendString(UART_DEV, (uint8_t*)"LISTA DE COMANDOS - \n\n");
        UART_SendString(UART_DEV, (uint8_t*)"Para PAUSAR o sistema aperte 'P' \n\n");
        UART_SendString(UART_DEV, (uint8_t*)"Para INICIAR o sistema aperte 'I' \n\n");
        UART_SendString(UART_DEV, (uint8_t*)"Para AUMENTAR a velocidade aperte '+' \n\n");
        UART_SendString(UART_DEV, (uint8_t*)"Para DIMINUIR a velocidade aperte '-' \n\n");
        UART_SendString(UART_DEV, (uint8_t*)"Para RESETA a velocidade aperte '2' \n\n");
        UART_SendString(UART_DEV, (uint8_t*)"Para MUDAR o limite minimo aperte '1' \n\n");
        UART_SendString(UART_DEV, (uint8_t*)"Para MUDAR o limite maximo aperte '2' \n\n");
    }
}
```

Figura 5 – Leitura do UART e seleção do modo

## VI. RESULTADOS PRÁTICOS

Ao ligar a placa é necessário aguardar um tempo para estabelecimento da conexão serial via UART, para ser levado ao menu inicial da aplicação, como mostra a Figura 6.



Figura 6 – Inicialização da placa

A temperatura pode ser ajustada a qualquer momento através do *trimpot*, esse ajuste é devido a possíveis anormalidades (desgaste sensor, ambientes adversos ou necessidade do usuário). A Figura 7 mostra o *trimpot* da placa.



Figura 7 – Trimpot de ajuste

Com base nos dados lidos, o sensor detecta os valores de temperatura do ambiente e exibe no *display* e em um terminal externo, o sistema também exibe no *display* o ajuste.

Para iniciar o sistema é necessário que o usuário envie a letra ‘i’ para a placa, podendo pausar a qualquer momento pressionando a letra ‘p’ (como mostrado na Figura 8).

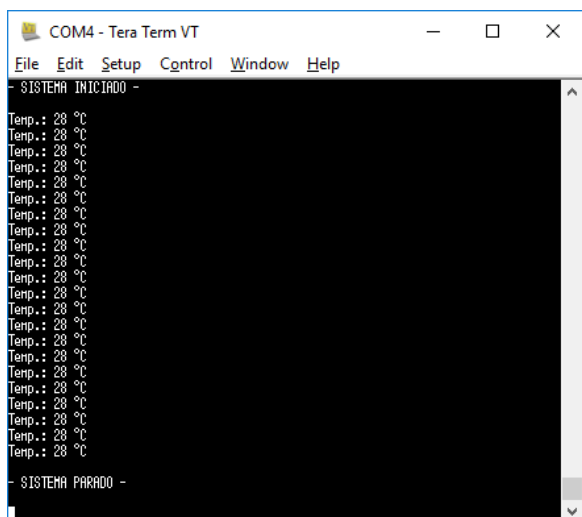


Figura 8 - Resultado da leitura do sensor e ações do sistema

## VII. DIAGRAMAS DO SISTEMA

Foram criados alguns diagramas para facilitar desenvolvimento e o entendimento do projeto. O Diagrama em Blocos, Diagrama de Camadas e Diagrama de Controle auxiliam na identificação dos blocos que serão utilizados no projeto e a relação entre eles, além de fornecer uma visão geral do funcionamento do projeto, facilitando também na manutenção do sistema.

A Figura 9 mostra o Diagrama em Blocos do sistema.



Figura 9 - Diagrama em Blocos

A Figura 10 apresenta o Diagrama de Controle do sistema.



Figura 10 - Diagrama de Controle

A Figura 11 apresenta o Diagrama de Camadas do sistema.



Figura 11 - Diagrama de Camadas

## VIII. CONCLUSÃO

Sendo assim, conclui-se que o *LPCXpresso Base Board* é uma ferramenta com múltiplas aplicações. Pode ser usada em vários meios, por ser bastante completa e com muitos recursos para trabalho. Além disso, se levarmos em consideração a parte de software, é bastante simples para implementação e para o desenvolvimento do código. A parte de programação pode ser considerada simples, o que representa uma boa qualidade para quem irá desenvolver projetos com essa placa microcontroladora.

## IX. REFERÊNCIAS

[1]  
FRIGIERI, E. P., D'ÂNGELO, V. I., RAMOS, R. F. M., JUNIOR, L. C. A., TEIXEIRA, P. S. F. L. Portando Uma Aplicação de Sistema Embarcado Com Arquitetura Super Loop Para Operar Com Sistema Operacional De Tempo Real. IFMG campos Formiga, v. 2, n. 1, p. 47-67, jan./jun. 2014.

[2]  
Link do projeto no GitHub:  
[https://github.com/FeelipeeF30/INATEL\\_EC020](https://github.com/FeelipeeF30/INATEL_EC020)

[3]  
EMBEDDED ARTISTS AB. *LPCXpresso Base Board: User's Guide*. Malmo, Suécia, 2010. PDF. 13 Fevereiro 2017.