

ESTUFA AGRÍCOLA AUTOMATIZADA UTILIZANDO MICRO CONTROLADOR ARDUÍNO E COMUNICAÇÃO SERIAL

Bruna Campos Guedes, Gabriel Felipe Vieira de Sousa
UFU - Universidade Federal de Uberlândia
Uberlândia – MG, Brasil
brunacampos.g1998@gmail.com, gabrielfelipe9@hotmail.com

Resumo - A automação no campo apresentou um crescimento de mais de 40% nos últimos anos, esse fato deve-se à eficiência na produtividade automatizada. Tanto para os produtores quanto para o setor de engenharia de controle e automação, o investimento em eficiência propiciou benefícios para ambos os setores. (Nardez, 2016)

Nesse campo, as estufas automatizadas merecem destaque. Devido ao aquecimento do planeta, a necessidade em proteger o cultivo por parte dos produtores os levaram a aderir esse tipo de cultura. (Rural News, 2016)

Esse projeto, busca desenvolver um protótipo de uma estufa inteligente de baixo custo que permite monitorar e informar ao produtor a situação de sua plantaçao possa otimizar o desenvolvimento da mesma. A comunicação serial, permitirá o produtor a inserir os parâmetros nos quais o micro controlador deve executar. A plataforma utilizada será o Micro Controlador Arduino.

Palavras-Chave – Micro Controlador Arduino, Estufa, Monitoramento.

GREENHOUSE AUTOMATED USING MICROCONTROLLER ARDUINO AND SERIAL COMMUNICATION

Abstract - Automation in the field grew over 40% in recent years, this fact is due to the efficiency of automated productivity. Both for producers and for the control and automation engineering sector, investment in efficiency led to benefits for both sectors. (Nardez, 2016)

In this field, automated greenhouses are worth mentioning. Due to global warming, the need to protect the crop by producers led them to join this way of culture. (Rural News, 2016)

In this field, automated greenhouses are worth mentioning. Due to global warming, the need to protect the crop by producers led them to join this way of culture. (Rural News, 2016)

This project intent to develop a prototype of a low-cost intelligent greenhouse that allows the producer to monitor and be informed about the situation of his plantation to

optimize its development. The serial communication will allow the producer to enter the parameters that the micro controller should run.

The platform used will be the Arduino.

Keywords – Microcontroller Arduino, Greenhouse, monitor.

I. INTRODUÇÃO

Muitas plantações necessitam de cuidados especiais durante o cultivo, cuidados como controle de pragas, temperatura e umidade. Entretanto, a variação climática em muitas regiões acabam afetando o desenvolvimento e até mesmo a perda total da produção.

Em regiões de clima muito seco, não é possível desenvolver certos tipos de produtos, portanto deve-se recorrer à utilização de estufas para que a plantaçao se desenvolva sem grandes problemas.

Estufas agrícolas são estruturas responsáveis por proteger as plantas contra agentes externos e também são capazes de absorver o calor proveniente do sol, dessa maneira o solo é aquecido e conseqüentemente é liberada radiação infra vermelho. Essa radiação é capaz de aquecer o ar nas camadas inferiores da estufa formando correntes de ar quente e ar frio.

Quando a camada de ar quente no interior da estufa for maior que a camada de ar frio, a umidade pode ser comprometida sendo necessário ativar um sistema de refrigeração para que o desenvolvimento das plantas não seja afetado. Caso a intensidade do sol não seja suficiente para liberar a radiação infra vermelho é necessário que a estufa seja aquecida através de um aquecedor elétrico.

De modo geral, uma estufa permite a criação de um micro-clima caracterizado por um conjunto de variáveis tais como temperatura, umidade, concentração de CO₂, entre outros. Esses fatores favorecem o desenvolvimento dos vegetais cultivados.

A utilização de uma estufa automatizada é de grande importância para a otimização do cultivo, pois a inferência humana será mínima e dessa forma o clima no interior da estufa estará adaptado rigorosamente às condições necessárias para o melhor desenvolvimento da plantaçao.



II. ESTUFA AUTOMATIZADA

O projeto foi dividido em 3 partes: solução do problema, implementação dos códigos e resultados da implementação.

A. Solução do problema

Para o desenvolvimento do protótipo será utilizado o micro controlador Arduino Uno. Este é o principal módulo onde será feita a aquisição e processamento dos dados. O Arduino fará comunicação serial com um computador onde será gerado uma lista contendo data e hora e os valores de temperatura e umidade naquele instante.

A comunicação serial é feita pela transferência de dados bit a bit entre o micro controlador e o computador. Foi utilizada esse tipo de comunicação para que a interação com o usuário seja realizada de forma simples e sucinta, - apesar do desenvolvimento do código ser considerado de maior complexidade - O computador irá receber e enviar dados para o Arduino, onde ocorrerá o processamento e análise dos valores recebidos.

Se a temperatura no interior da estufa for abaixo do padrão estabelecido, o simulador do aquecedor elétrico(LED) será ativado. Caso contrário será acionado o mecanismo de resfriamento.

Serão acoplados no Arduino os seguintes componentes (vide figuras 1-6):



Figura 1 - Sensor DHT11



Figura 2 – Visor LCD (16x2)

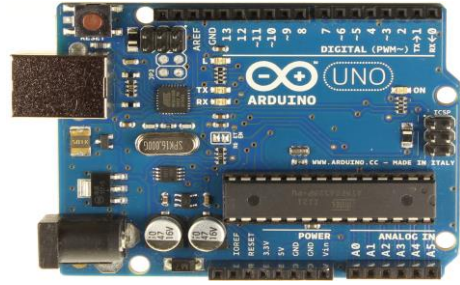


Figura 3 – Micro controlador Arduino UNO



Figura 4 – Cooler



Figura 5 – LED's

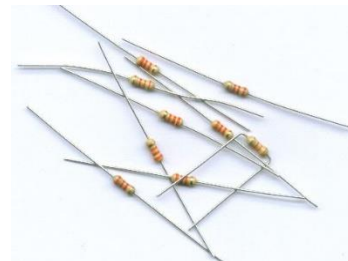
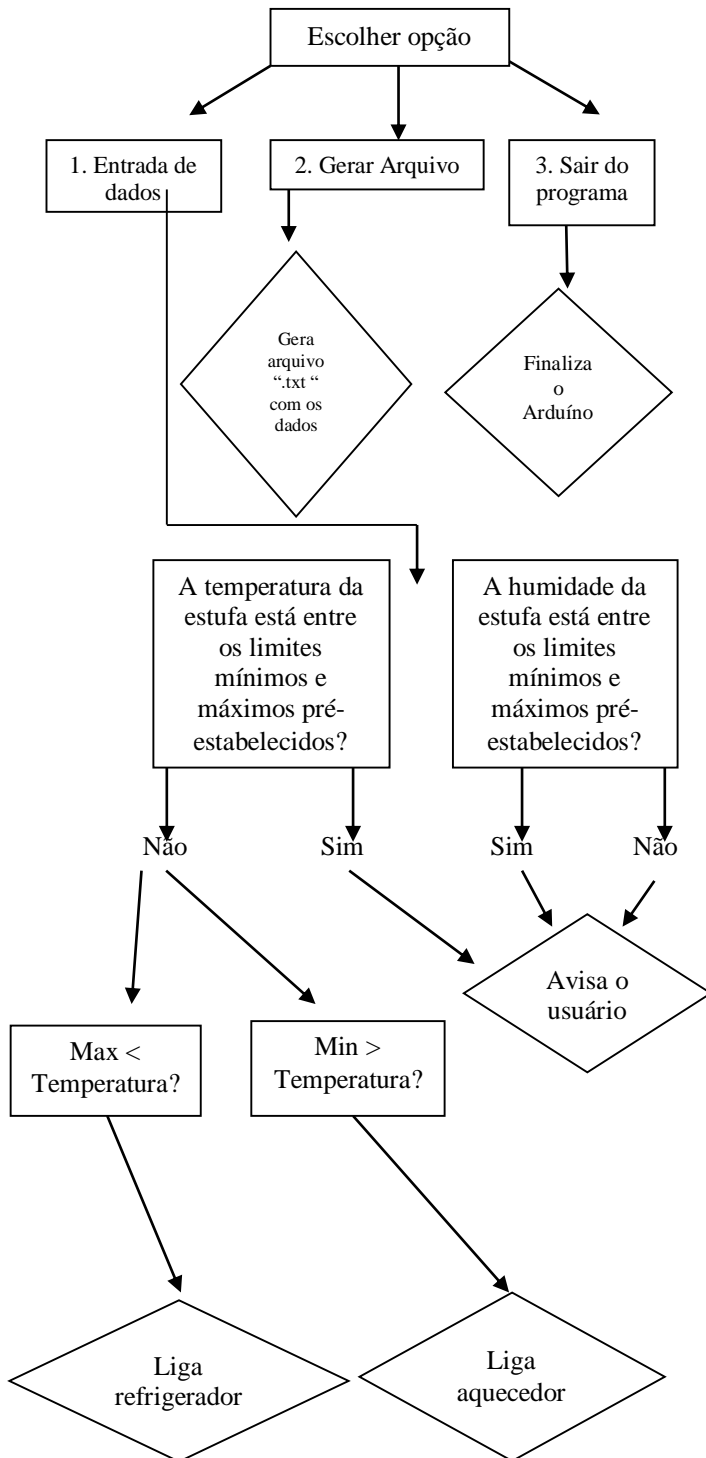


Figura 6 – Resistores de 220 ohms

O fluxograma apresentará, em alto nível, os comandos a serem realizados pelo computador/arduino:



B. Implementação dos códigos

Para o desenvolvimento do código para o computador foram utilizados os conceitos de vetores, ponteiros, listas encadeadas e arquivos.

Os valores de entrada informados pelo usuário são números inteiros e são armazenados em uma string. Essa string é enviada para o arduino através da seguinte função:

```

void SerialPuts(HANDLE *hCom, char *txstring)
{
    BOOL bWriteRC;
    static DWORD iBytesWritten;
    bWriteRC = WriteFile(*hCom, txstring,
        strlen(txstring), &iBytesWritten, NULL);
}
  
```

Figura 7 – Trecho de código da comunicação serial.

O menu foi desenvolvido usando um switch contendo três casos.

A primeira opção permite o usuário informar os limites de temperatura e umidade adequados para o desenvolvimento do cultivo. A segunda opção gera um arquivo contendo o *status* da estufa. Para isso foi implementado uma função para adicionar um novo elemento na fila (*addToHistory(int8_t temp, int8_t hum)*) e outra função para armazenar o elemento no arquivo (*saveToFile(int8_t temp, int8_t hum)*):

```

void
addToHistory(int8_t temp, int8_t hum)
{
    if (historicoEstufa == NULL)
    {
        historicoEstufa = (estufa *)
        malloc(sizeof(estufa));
        if (historicoEstufa == NULL)
        {
            printf("Erro ao criar elemento na
            lista");
        }
        currentDate(historicoEstufa->date);
        historicoEstufa->temp = temp;
        historicoEstufa->hum = hum;
        historicoEstufa->next = NULL;
    }
    else
    {
        estufa *atual = historicoEstufa;

        while (atual->next != NULL)
        {
            atual = atual->next;
        }

        atual->next = (estufa *)
        malloc(sizeof(estufa));
        if (atual->next == NULL)
        {
            printf("Erro ao criar elemento na
            lista");
        }
        currentDate(atual->next->date);
        atual->next->temp = temp;
        atual->next->hum = hum;
    }
}
  
```

```

        atual->next->next = NULL;
    }
}

```

Figura 8 – Função para adicionar o elemento na fila.

Função para salvar o elemento no arquivo:

```

saveToFile(int8_t temp, int8_t hum)
{
    FILE *informa;

    if ((informa = fopen("estufa.txt", "w")) ==
    NULL)
    {
        printf("Erro ao criar arquivo de log!");
        return;
    }

    if (historicoEstufa != NULL)
    {
        estufa *atual;
        atual = historicoEstufa;
        while (atual != NULL)
        {
            fprintf(informa,
            "%s\nTemperatura atual: %d\nHumidade atual: %d\n\n",
            atual->date, atual->temp, atual->hum);
            atual = atual->next;
        }
    }

    fclose(informa);
}

```

Figura 9 – Trecho de código utilizado para a geração do arquivo.

No código desenvolvido para o Aduíno foi implementado uma função para iniciar o display (*void initDisplay(void)*). Além de iniciar o display, a função verifica se os valores de entrada do sensor foram recebidos:

```

void initDisplay(void)
{
    lcd.begin(16, 2);
    lcd.clear();
    lcd.setCursor(5, 0);
    lcd.print("ESTUFA ");
    lcd.setCursor(1, 2);
    lcd.print(" AUTOMATIZADA ");
    delay(3000);
    lcd.clear();
    lcd.print("INICIANDO...");

    float currentTemp, currentUmi;

    while (isnan(currentTemp = dht.readTemperature()) ||
    isnan(currentUmi = dht.readHumidity()));

    temperatura = currentTemp;
}

```

```

umidade = currentUmi;

lcd.createChar(1, temperatureIcon);
lcd.createChar(2, humidityIcon);
lcd.clear();
lcd.print("Status: Ok");
printTemperature(temperatura);
printHumidity(umidade);
}

```

Figura 9 – Inicialização do display no arduíno.

Após a coleta de valores limitantes feita no computador e a leitura da temperatura e umidade atual o arduíno então executa os processos de automatização como visto na figura abaixo:



Figura 10 - Execução do comando pelo Arduíno e informação da temperatura e umidade, respectivamente.

Em anexo, estão os códigos implementados completos.

C. Resultados da implementação

Após o usuário inserir os valores limitantes, o Arduíno imprimiu no LCD que o recebimento foi bem sucedido. A partir disso o micro controlador faz as comparações e realiza os comandos para controlar a temperatura. Da mesma forma ocorre a verificação da umidade no interior da estufa, entretanto apenas uma mensagem de alerta é exibida na tela. O usuário pode através do computador pedir para que seja enviado e salvo o estado no interior da estufa em um arquivo contendo data e hora.



Figura 11 - Protótipo finalizado e em execução.

A qualquer momento o usuário pode sair do programa, e dessa forma o monitoramento da estufa é

finalizado, para reativar o monitoramento basta reiniciar o arduíno.



Figura 12 - Arduíno conectado ao computador.



Figura 13 - Vista superior do protótipo.

III. CONCLUSÕES E TRABALHOS FUTUROS

Por meio desse projeto foi possível realizar a implementação da automação de uma estufa agrícola. Esta que, além do baixo custo, permite ao usuário que sempre que necessário é possível obter informações instantâneas das condições de temperatura e umidade no interior da estufa.

O código foi implementado e compilado sem erros, portanto, verificou-se ao término da construção do protótipo a necessidade de adicionar um relé para fornecer corrente suficiente ao cooler.

Como trabalhos futuros pretende-se automatizar o processo de irrigação no interior da estufa, e conseqüentemente fazer o monitoramento da umidade do solo, assim como o controle de CO_2 . Também pretende-se melhorar a comunicação serial e realizar esta através de comandos remotos, via internet ou bluetooth.

IV. AGRADECIMENTOS

Os autores agradecem pelo apoio do professor Renato Carrijo da disciplina de Métodos e Técnicas de Programação da Faculdade de Engenharia Elétrica da UFU.

V. REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Nardez, M. (28 de Maio de 2016). *Tecnologia para produzir*. Fonte: Diário de Cuiabá: <http://www.diariodecuiaba.com.br/detalhe.php?cod=473740>
- [2] Rural News. (05 de 08 de 2016). *Estufas - produção agrícola em ambientes controlados*. Fonte: Rural News: <http://www.ruralnews.com.br/visualiza.php?id=202>
- [3] Neto, L. M. (04 de Julho de 2016). *Artigo: A automação agropecuária*. Fonte: Agro de precisão: <http://www.senar.org.br/agricultura-precisao/artigo-a-automacao-agropecuaria/>
- [4] ARDUINO WEBSITE. Disponível em < <https://www.arduino.cc/> > . Acesso em 20 de junho de 2016.
- [5] Microsoft. (22 de Junho de 2016). *Serial Communications*. Fonte: <https://msdn.microsoft.com/en-us/library/ff802693.aspx>
- [6] Cplus plus. (19 de Junho de 2016). Fonte: <http://www.cplusplus.com/reference/cstdlib/atoi/?kw=atoi>
- [7] Farias, J. M. (27 de Junho de 2016). *Sistemas Embarcados: A computação invisível*. Fonte: Sistemas embarcados: <http://www.sistemasembarcados.org/artigo-sistemas-embarcados-o-que-e-definicao.html>
- [8] Souza, F. (30 de Agosto de 2016). *Arduino - Comunicação Serial*. Fonte: Embarcados: <http://www.embarcados.com.br/arduino-comunicacao-serial/>
- [9] Tonietto, J. (06 de Setembro de 2016). *Processamento em Clima Temperado*. Fonte: Embrapa: <https://sistemasdeproducao.cnptia.embrapa.br/FontesHTML/Uva/UvaAmericanaHibridaClimaTemperado/clima.html>