**Universidade Tecnológica Federal do Paraná**

Héber Miguel dos Santos

Thiago dos Santos Cavali

**COELE – Coordenação do Curso de Engenharia Eletrônica**

Campo Mourão, Paraná, Brasil

Sistemas Embarcados

heber.eng.eletronica@gmail.com

tsantosc123@gmail.com

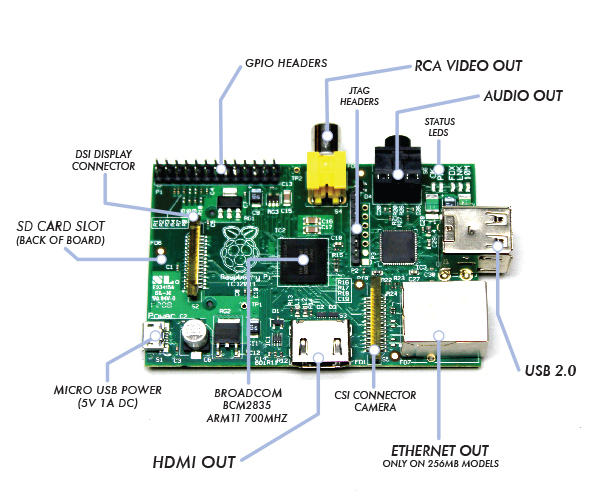
**Raspberry Pi Configurações básicas:**

**Sumario:**

1. **Inicialização da Raspberry Pi:**
   1. **Software de simulação RaspBerry Pi para Windows;**
   2. **Instalação:**
   3. **Configurações básicas:**
   4. **Configurações para Programar:**
2. **PiFace:**
   1. **Instalação:**
   2. **Software de simulação:**
3. **Exemplos de softwares em C:**
4. **Códigos utilizados:**
5. **Inicialização da Raspberry Pi:**
   1. **Software de simulação RaspBerry Pi para Windows:**

Download do software no link abaixo:

<http://sourceforge.net/projects/rpiqemuwindows/>

**1.2. Instalação:**

**1.2.1. Preparando seu Micro SD:**

Realizar o tutorial disponivel em:

<http://elinux.org/RPi_Easy_SD_Card_Setup>

**1.2.2. Conexões fisicas:**

Realizar a Conexão dos dispositivos a serem utilizados nas seguintes portas:

-Mouse e Teclado: USB 2.0;

-Monitor: HDMI OUT;

-Rede Ethernet: ETHERNET OUT;

-Micro SD: SD CARD SLOT;

-Fonte de Alimentação: MICRO USB POWER;

Após as conexões sua Raspberry estará com as seguintes conexões:

****

**1.3. Configurações básicas:**

Após tudo instalado, iniciando a Raspberry, realizaremos as seguintes configurações:

* Códigos iniciais:

1. Código: apt-get update

Objetivo: para atualizar a data.

1. Código: apt-get install mc

Objetivo: Instalação do mcedit (editor de texto).

1. Código: apt-get upgrade

Objetivo: Atualiza o sistema da Raspberry

Anotações: Processo demorado execute somente se necessário.

* Configura Profile Pi:

1. Código: cd /root/etc

Objetivo: entra no diretório /root/etc.

1. Código: mcedit profile

Objetivo: através do mcedit abre o arquivo profile para edição.

-- No final do arquivo aberto insira:

alias ll='ls -l --color=auto' #insere o atalho do comando ll

alias la='ls -la --color=auto' #insere o atalho do comando la

export EDITOR=mcedit #configura mcedit como editor padrão

export HISTSIZE=1000 #configura o tamanho max. do history

export HISTFILESIZE=1000

--

Anotações: comando ll, lista os arquivos do diretório atual.

* Configurações do root:

1. Código: cd /root

Objetivo: entra no diretório /root

1. Código: mcedit .bashrc

Objetivo: através do mcedit abre o arquivo .bashrc para edição.

-- No final do arquivo insira:

alias ll='ls -l --color=auto' #insere o atalho do comando ll

alias la='ls -la --color=auto' #insere o atalho do comando la

export EDITOR=mcedit #configura mcedit como editor padrão

export HISTSIZE=2000 #configura o tamanho max. do history

export HISTFILESIZE=2000

--

Anotações: comando ll, lista os arquivos do diretório atual.

-> Para confirmar as configurações realizadas é necessário reiniciar o sistema, utilizando o comando abaixo.

1. Código: shutdown -r now

Objetivo: Reinicia o Sistema.

* Configurações Vídeo (tela cheia):

1. Código: cd /boot

Objetivo: entra no diretório /boot

1. Código: mcedit config.txt

Objetivo: abre o arquivo para edição

-- Descomente tirando o caracter '#' antes de:

"disable\_overscan=1"

--

-> Para confirmar as configurações realizadas é necessário reiniciar o sistema, utilizando o comando abaixo.

1. Código: shutdown -r now

Objetivo: Reinicia o Sistema.

* Para retirar a necessidade de login iniciado o raspberry execute:

1. Código: mcedit /etc/inittab

Objetivo: abre o arquivo para edição

-- Dentro deste arquivo insira abaixo de #1:2345:respawn:/sbin... o seguinte comando:

#1:2345:respawn:/bin/login -f pi tty1 </dev/tty1 >/dev/tty1 2>&1

--

-> Para confirmar as configurações realizadas é necessário reiniciar o sistema, utilizando o comando abaixo.

1. Código: shutdown -r now

Objetivo: Reinicia o Sistema.

* 1. **Configurações para Programar:**
* Instalar a biblioteca para compilar programas em C.

1. Código: Wget www.lt38c.hturbo.com/bcm2835-1.33.tar.gz -o /tmp/bcm2835.tar.gz

Objetivo: Download da biblioteca.

1. Código: tar -zvxf bcm2835-1.33.tar.gz
2. Código: cd bcm2835-1.33
3. Código: ./configure
4. Código: make
5. Código: make check
6. Código: make install
7. Código: cp -R /tmp/bcm2835-1.33/examples/ /root/bcm2835

-- Para compilar os programas dentro da pasta examples utilize:

1. Código: gcc -o blink blink.c -lbcm2835

Objetivo: compila o código blink.c.

* Instalar o servidor apache, php e mysql:

1. Código: apt-get install apache2 php5 php5-mysql mysql-server
2. Código: sh -c 'echo "<?php phpinfo(); ?>" > /var/www/phpteste.php'

-- Para testar se a instalação foi bem sucedida, entre no modo gráfico e usando o Midori acesse o phpteste.php

* Instalar a bliblioteca wiringPi:

-- Usando do modo gráfico, acesse o link: http://wiringpi.com/download-and-install/

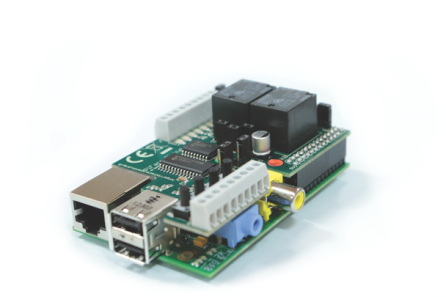
-- Execute os passos do “Plano B”, após baixado mova a pasta para o local desejado e descompacte com:

tar xfz wiringPi-f18c8f7.tar.gz

**2. PiFace:**

**2.1.Instalação:**

A PiFace Será conectada fisicamente como demonstra a seguinte Fig.

****

**2.2. Software de simulação:**

O software em que o desenvolvimento foi baseado esta disponível em:

<http://www.piface.org.uk/guides/Install_PiFace_Software/Installing_PiFace_Digital_modules/>

Para instalar siga as instruções disponíveis no link acima, e execute o app que será instalado a partir do modo gráfico.

1. **Exemplos de softwares em C:**

Foram utilizados alguns exemplos de programas, para conhecimento da programação em C para a RaspBerry como por exemplo o Blink.c disponível nos arquivos presentes no bcm2835, e sua utilização esta disponível no link abaixo:

<http://wiringpi.com/examples/blink/>

Dica para compilar:

1. Código: gcc -o blink blink.c -lbcm2835

Objetivo: compila o código blink.c.

Para a aquisição de novos exemplos foi instalado uma nova biblioteca utilizando o tutorial disponível em: <http://wiringpi.com/download-and-install/>

Dica para compilar:

1. Código: gcc -o blink blink.c -lwiringPi -lwiringPiDev

Objetivo: compila o código blink.c.

1. **Instalação MySQL server:**
2. Código: apt-get install mysql-server

Objetivo: instala o MySQL.

1. Código: mysql -u root –p

Objetivo: Acessa o MySQL.

1. Executar no console: GRANT ALL ON \*.\* TO 'root'@'localhost' IDENTIFIED BY 'password\_mysql\_here';

e

FLUSH PRIVILEGES;

1. Objetivo: Garante acesso do root ao MySQL.

* Para Ativar o acesso ao MySQL em C (rodar uma vez pelo menos):

1. sudo apt-get update
2. sudo apt-get upgrade
3. sudo apt-get install libmysqlclient-dev
4. sudo apt-get install libmysqlclient-dev

//////

1. Código: mcedit myClient.c ???????????
2. Código: g++ myClient.c -o cliente ???????????????
3. Código: ./cliente ?????????????
4. **Compilar o código C que foi desenvolvido:**
5. Código: gcc interfaceSql.c -o roda -lmysqlclient -lwiringPi -lwiringPiDev

Objetivo: compila o código da interface em C.

Anotações: interfaceSql.c é o arquivo que foi criado.

1. **Página Web:**

**1 - Colocar os arquivos da pasta [InterfaceFinal] na pasta do raspberry: /var/www**

**2 – Acessar o console do mysql e criar a base de dados: “Dispositivos” no console do mysql – create database Dispositivos;.**

**3 – Alterar a senha do banco de dados no arquivo conexão.php**

**4 – Rodar no cliente ou no navegador local o arquivo: cria\_tabelas.php**

**5 – Rodar o programa: “**roda**” no raspberry, compilar o programa roda conforme o item 5.**

**6 – Abrir o arquivo no navegador (cliente / rasp.): mostraTela.html**

1. **Códigos utilizados:**
2. Código: gcc -o blink blink.c -lbcm2835

Objetivo: compila o código blink.c.

Anotações:

1. Código: history > nomedoarquivo.txt

Objetivo: exportar o histórico

Anotações: o mesmo deve ser enviado por e-mail

1. Código: halt

Objetivo: desliga a raspberry

1. Código: cd [diretório]

Objetivo: muda diretório

Anotações: algumas variações: cd .. (volta um diretório), cd / (vai para diretório raiz).

1. Código: startx

Objetivo: Inicia o modo gráfico.

Anotações: executar somente no usuário PI.

1. Código: ./nomedoexecutavel

Objetivo: executar um programa compilado em C.

1. Código: mcedit nomedoarquivo

Objetivo: abre o editor de texto do arquivo

1. Código: exit

Objetivo: retorna ao usuário PI.

1. Código: clear

Objetivo: limpa os códigos da tela.

1. Código: ps aux

Objetivo: lista os processos em execução

1. Código: kill numerodoprocesso(ex:3046)

Objetivo: encerra o processo descrito

1. Código: find –name nomedoqueprocura

Objetivo: lista todos os arquivos com o nome descrito.

1. Código: ifconfig

Objetivo: mostra o as configurações de rede.