

MINI-MAX/ARM9260-E

Single Board Computer

Quick Start Guide

Document Revision: 1.01

Date: 07 October, 2009



BiPOM Electronics, Inc.

16301 Blue Ridge Road, Missouri City, Texas 77489

Telephone: 1-713-283-9970 Fax: 1-281-416-2806

E-mail: info@bipom.com

Web: www.bipom.com

All trademarked names in this manual are the property of respective owners.
© 2009 BiPOM Electronics, Inc.

Overview

Thank you for your purchase of the MINI-MAX/ARM9260-E Single Board Computer. MINI-MAX/ARM9260-E is a powerful computer board that is capable of running high-level operating systems such as Linux.

This document describes how to quickly get started with MINI-MAX/ARM9260-E and includes the following steps:

- Requirements
- Hardware Setup
- Software Setup

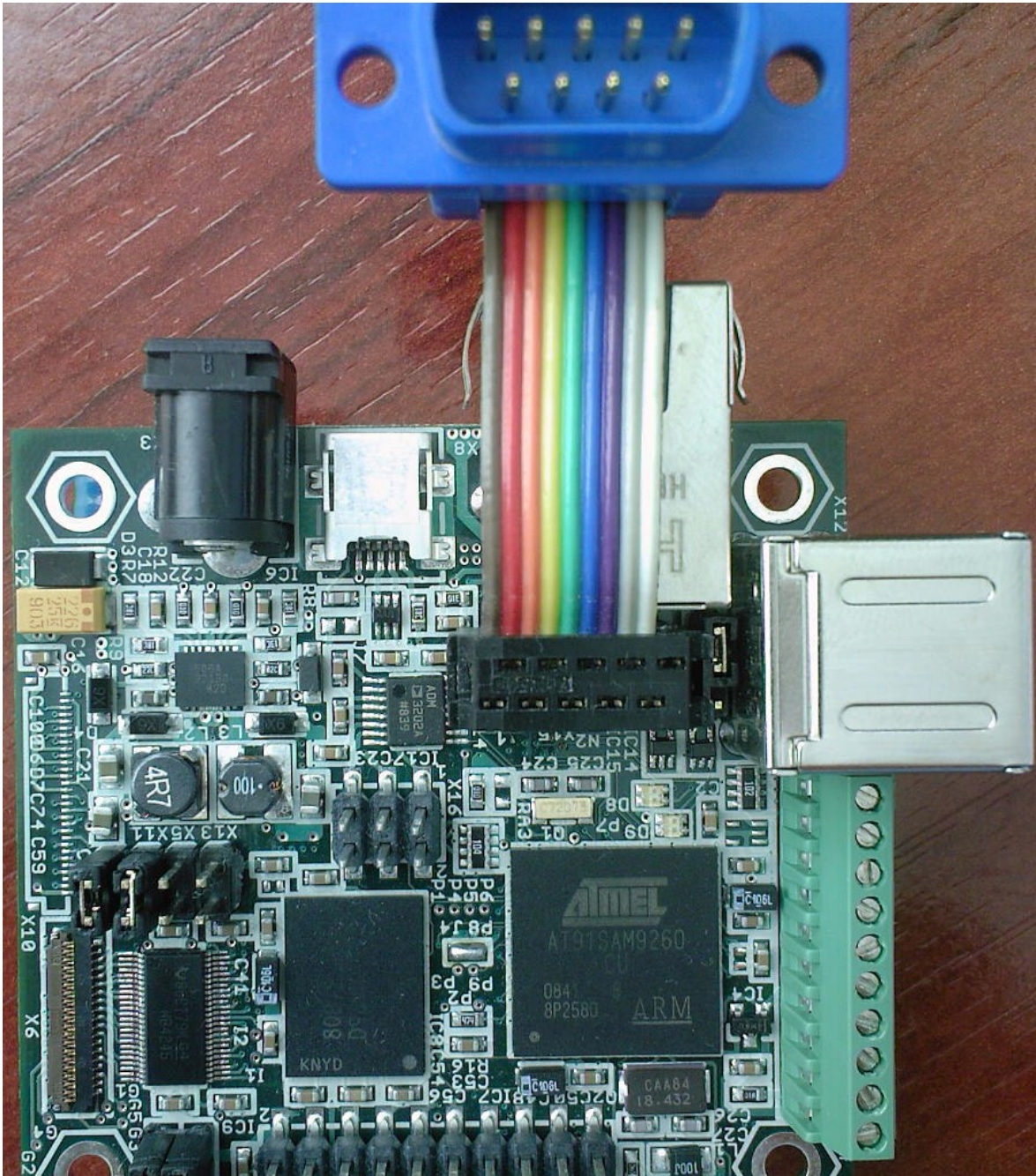
Requirements

MINI-MAX/ARM9260-E can be used for many different applications. The minimum requirements to run Linux are:

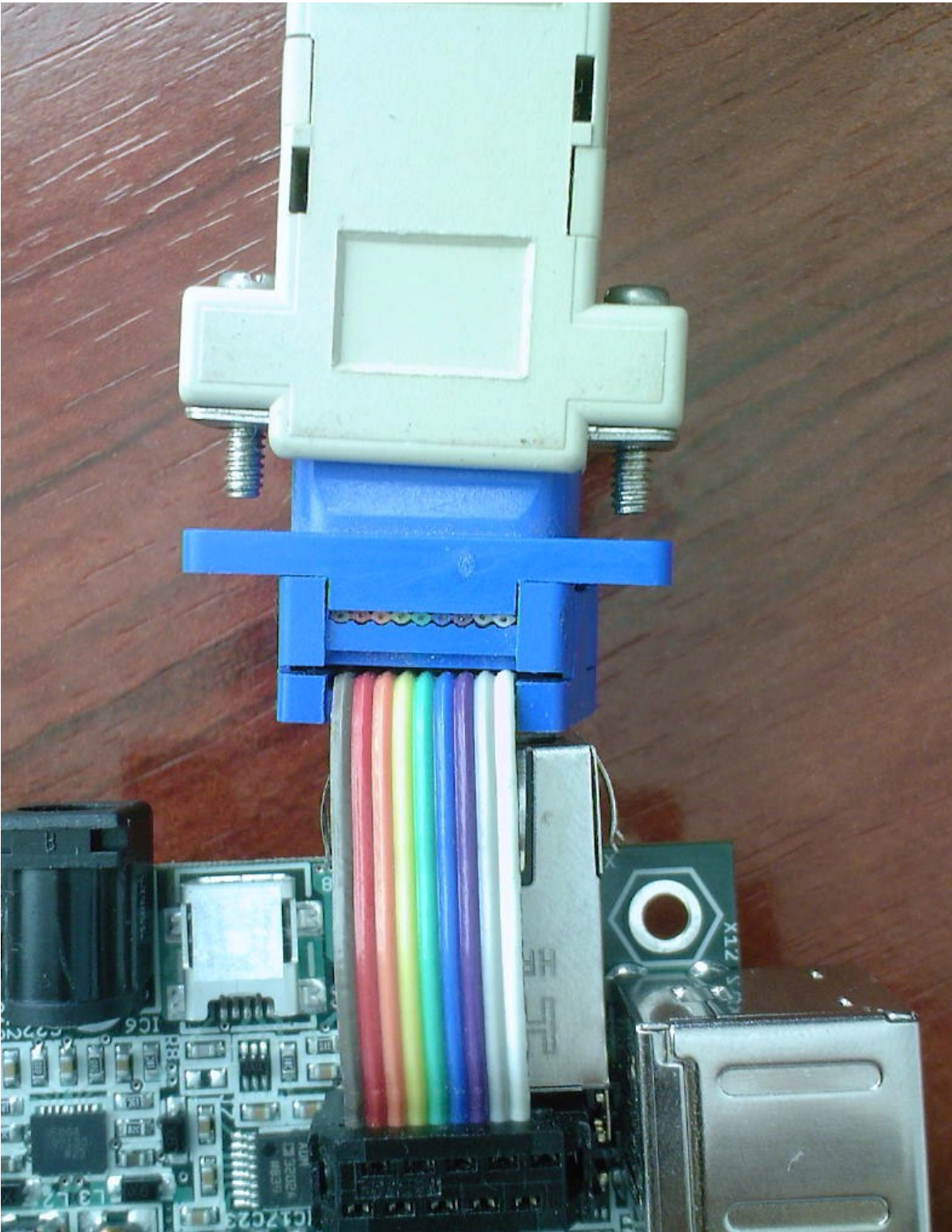
- MINI-MAX/ARM9260-E
- USB Flash Drive
- A Windows PC to install and run MINI-MAX/ARM9260-E software and Linux Control Panel

Hardware Setup

1. Place the MINI-MAX/ARM9260-E board on a clean, non-conductive surface.
2. Connect the provided RS-232 Cable to the 10-pin serial port dual-row header on the MINI-MAX/ARM9260-E.

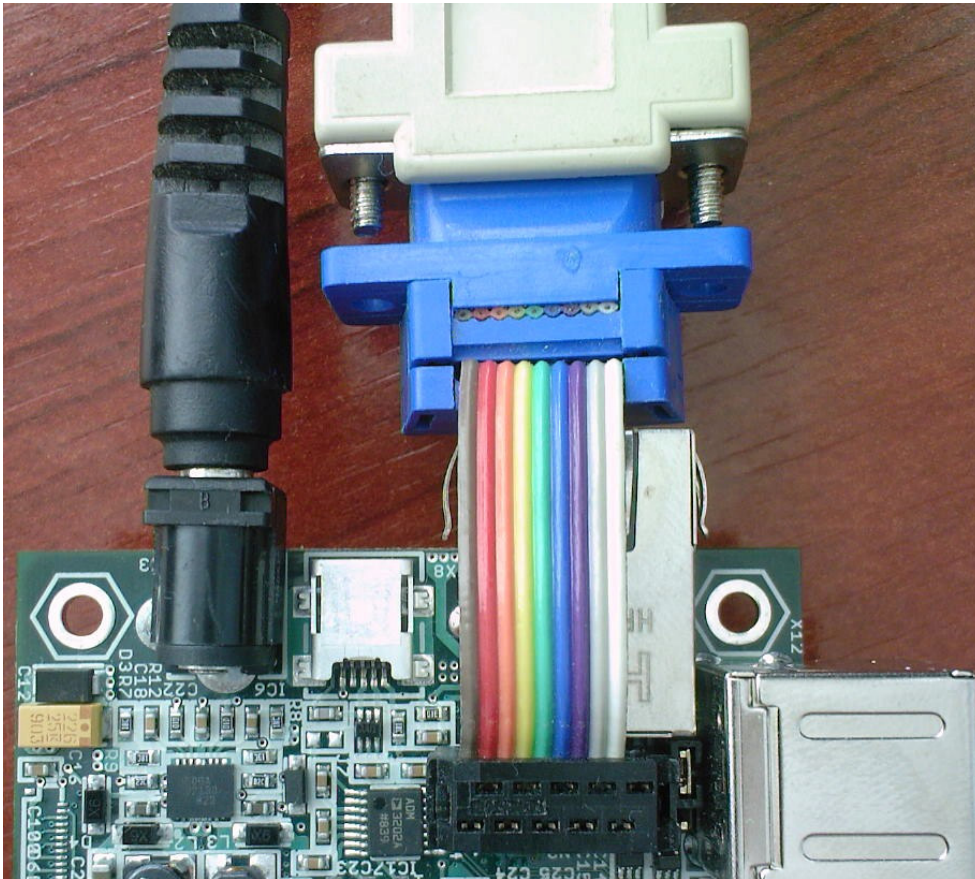


3. Connect the other end of the RS232 cable to a NULL-modem cable.



4. Connect the other end of the NULL-modem cable to a serial port (typically COM1) of your PC.

5. Connect the supplied power adapter to the board.
Don't power the board for now.



Software Setup

MINI-MAX/ARM9260-E first boots up from on-board DataFlash memory. Compressed Linux image is decompressed to MINI-MAX/ARM9260-E's on-board RAM and executed. This loads Linux kernel, drivers, and applications such as BusyBox. Much like the Linux distributions on regular PC's, MINI-MAX/ARM9260-E Linux runs from RAM.

MINI-MAX/ARM9260-E 's Linux distribution includes standard Linux kernel drivers as well as BiPOM-supplied drivers for various popular USB devices.

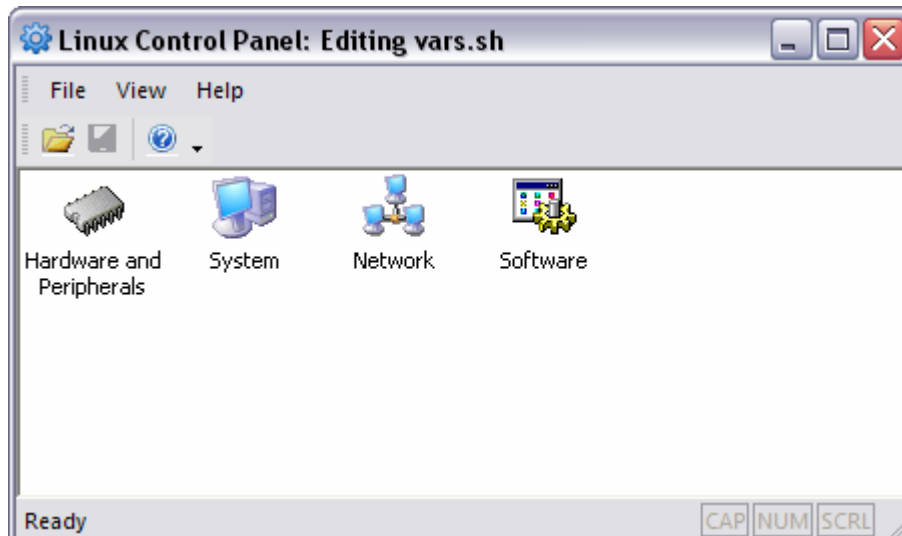
MINI-MAX/ARM9260-E automatically detects USB Flash Drives (for example, Thumb drives). To make MINI-MAX/ARM9260-E run your own applications and any custom commands, place commands to special shell file (Bash Shell) with the name **user_cmds.sh**. This file is in **sh** folder in root directory on USB Flash Drive. Also you can configure the Linux with **Linux Control Panel**. This software allows to change many options like Network settings, Hardware list etc,

Here are the steps to configure and run MINI-MAX/ARM9260-E Linux:

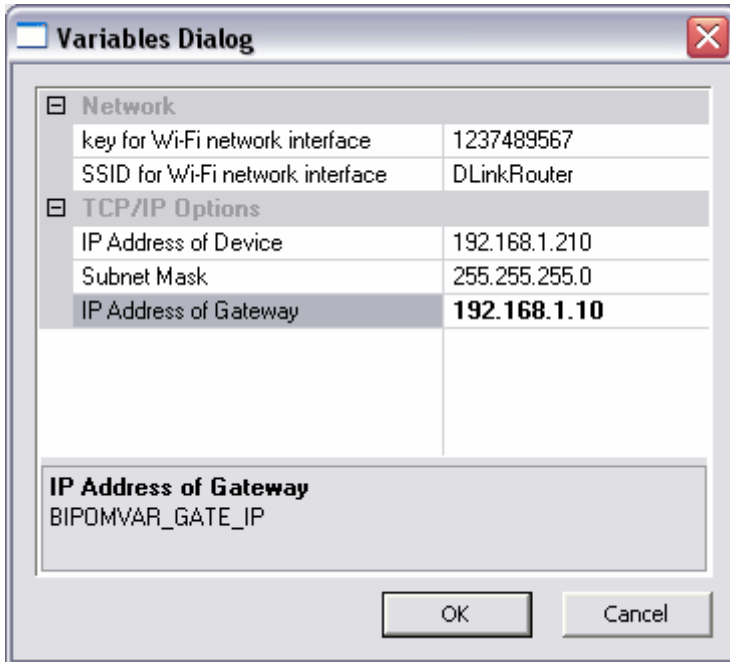
1. Download latest **MINI-MAX/ARM9260-E Linux Release (Windows Installation)** from http://www.bipom.com/web_softwares/2900044.html and install it.
2. In order to configure **MINI-MAX/ARM9260-E Linux**, start **Linux Control Panel** software from **Start -> All Programs -> MMARM9260E -> Linux Control Panel**. After **Linux Control Panel** starts, open **vars.sh** file that is located in **c:\bipom\devtools\MMARM9260E\sh** (if you installed Linux Control Panel to its default path).

Select **File** → **Open** from the menu and open **vars.sh** file in **sh** folder. By default, the program would start from the folder where MINI-MAX/ARM9260-E release was installed.

When you open **vars.sh** file, you will see configuration group icons such as **Hardware & Peripherals**, **Software**, **System** and **Network**.



Now you can double click on any icon to view and edit all possible configuration options for that group.



You can change any options as needed and click the OK button. The options correspond to the configuration variables in **vars.sh** file. You can read the description of each variable in the chapter titled **Linux Configuration File**.

The bottom area of Variables Dialog gives a short description and name of variable from **vars.sh** file. When you change the option, that variable will be changed in **vars.sh**.

In order to save your changes, select **File → Save** from the menu. This saves all the changes back to **vars.sh** on your computer.

The current MINI-MAX/ARM9260-E release has 5 sections:

- **Hardware and Peripherals** – allows enabling/disabling of embedded hardware and peripherals
- **System** - allows configuring of system parameters
- **Network** - allows configuring of network parameters
- **Software** - allows configuring various software packages in MINI-MAX/ARM9260-E Linux
- **User Defined Options** - allows configuring of user defined variables

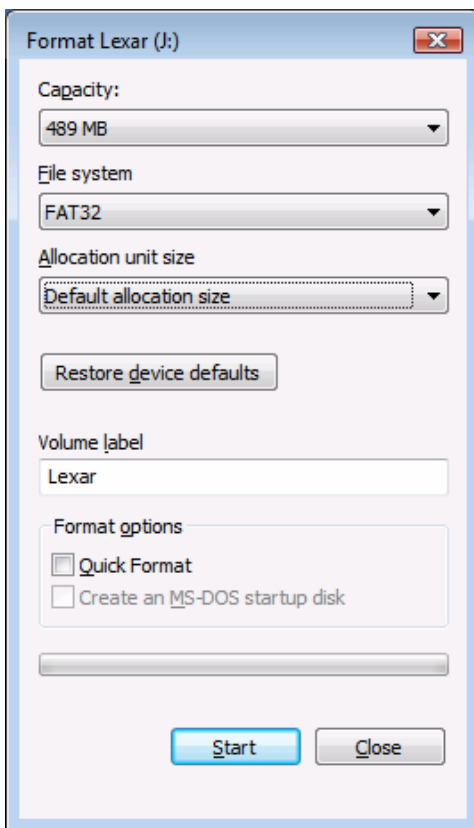
IMPORTANT: Please be careful when editing any **.sh** files (**vars.sh**, **user_cmds.sh**, etc.) manually. These are **UNIX script** files and almost all **Windows** editors will corrupt them. This will cause the files to be not executable under Linux. You can use the **Micro-IDE** editor (**Start -> All Programs -> Micro-IDE->Micro-IDE**) that works with **.sh** files. Also, do NOT rename predefined variables and avoid modifying them manually. Instead, use the **Linux Control Panel** to modify predefined variables.

3. You can edit **user_cmds.sh** in Micro-IDE if you need start your programs at boot time. This file will be executed after Linux boots up. This file also located in **c:\bipom\devtools\MMARM9260E\sh** (if you installed the release to its default path).

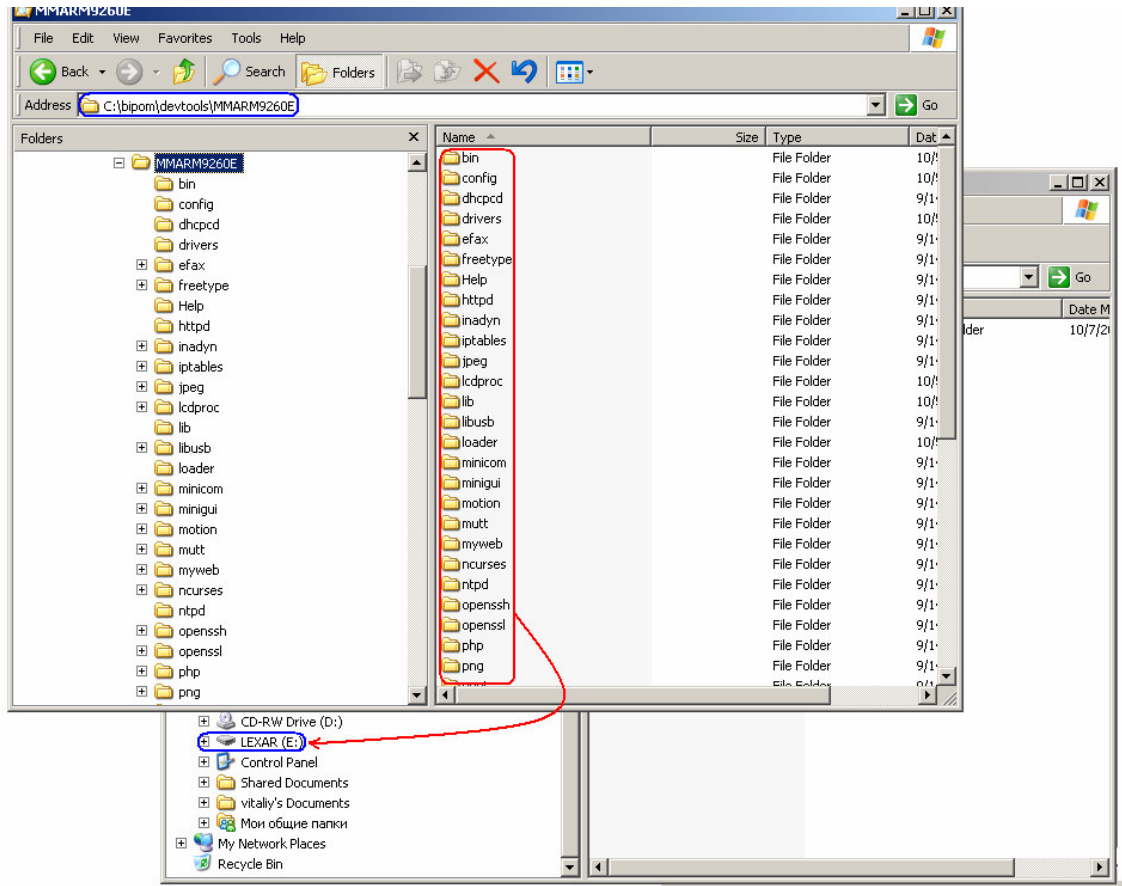
4. Plug the USB Flash Drive to your PC's USB port. Skip to Step 6 if the USB Flash Drive is already formatted.



5. Format the USB Flash Drive with FAT32 format on Windows if the USB Flash Drive has not yet been formatted:



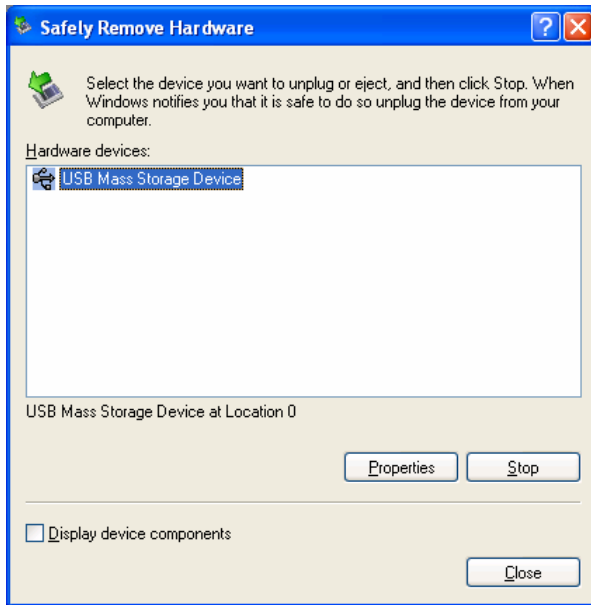
6. Copy all the files/folders from folder where you installed **MINI-MAX/ARM9260-E Linux Release** to the root directory of your USB Flash Drive.



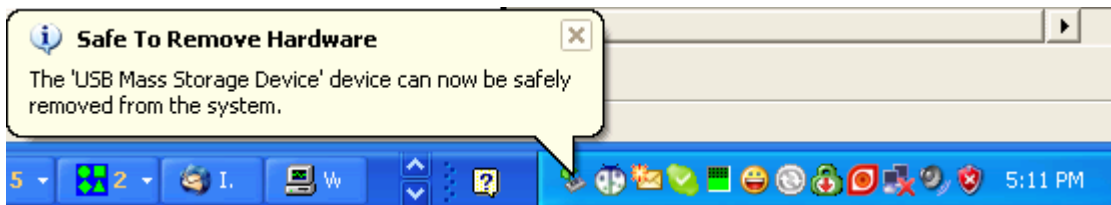
7. Stop your USB Flash Drive using the “Safely Remove Hardware” icon on your Windows taskbar:



When you click “Safely Remove Hardware” icon, the “Safely Remove Hardware” dialog will appear. Click Stop to stop the USB Mass Storage Device (your USB Flash Drive)

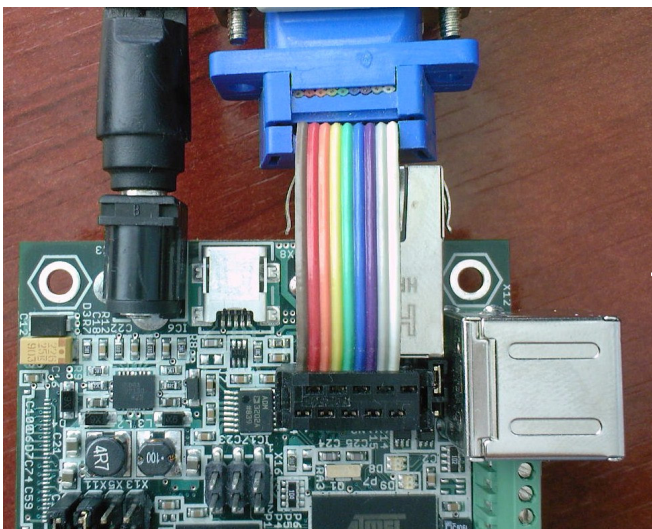


Shortly after you click the Stop button, a message will appear:

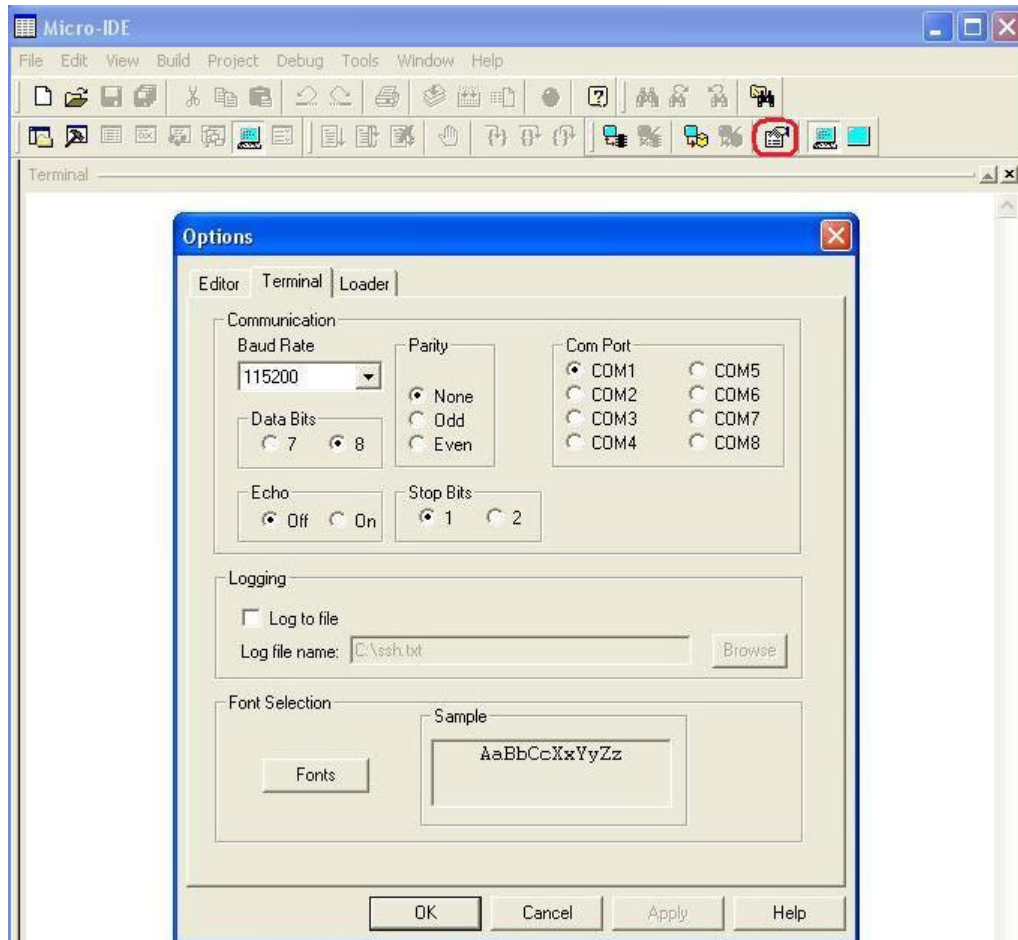


indicating that it is safe to remove the USB Flash Drive.
Remove from USB Flash Drive your PC.

8. Insert the USB Flash Drive to any unused USB port of MINI-MAX/ARM9260-E .



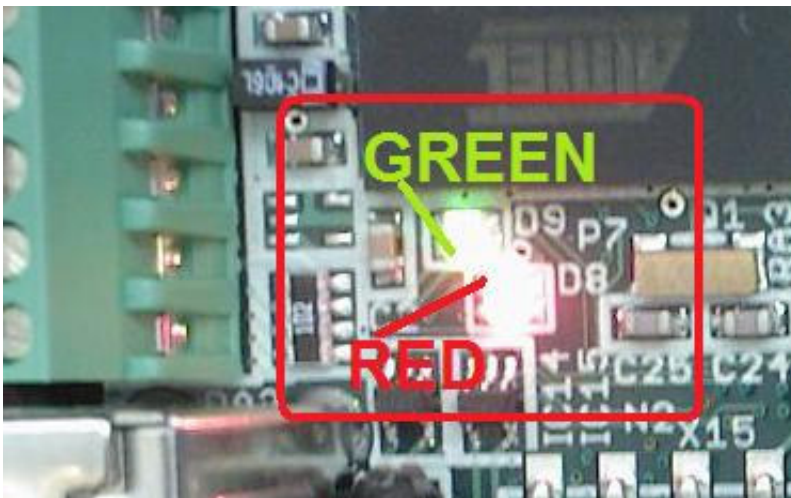
9. Run Micro-IDE (**Start -> All Programs ->Micro-IDE->Micro-IDE**).
Configure the terminal using “Options” icon button.



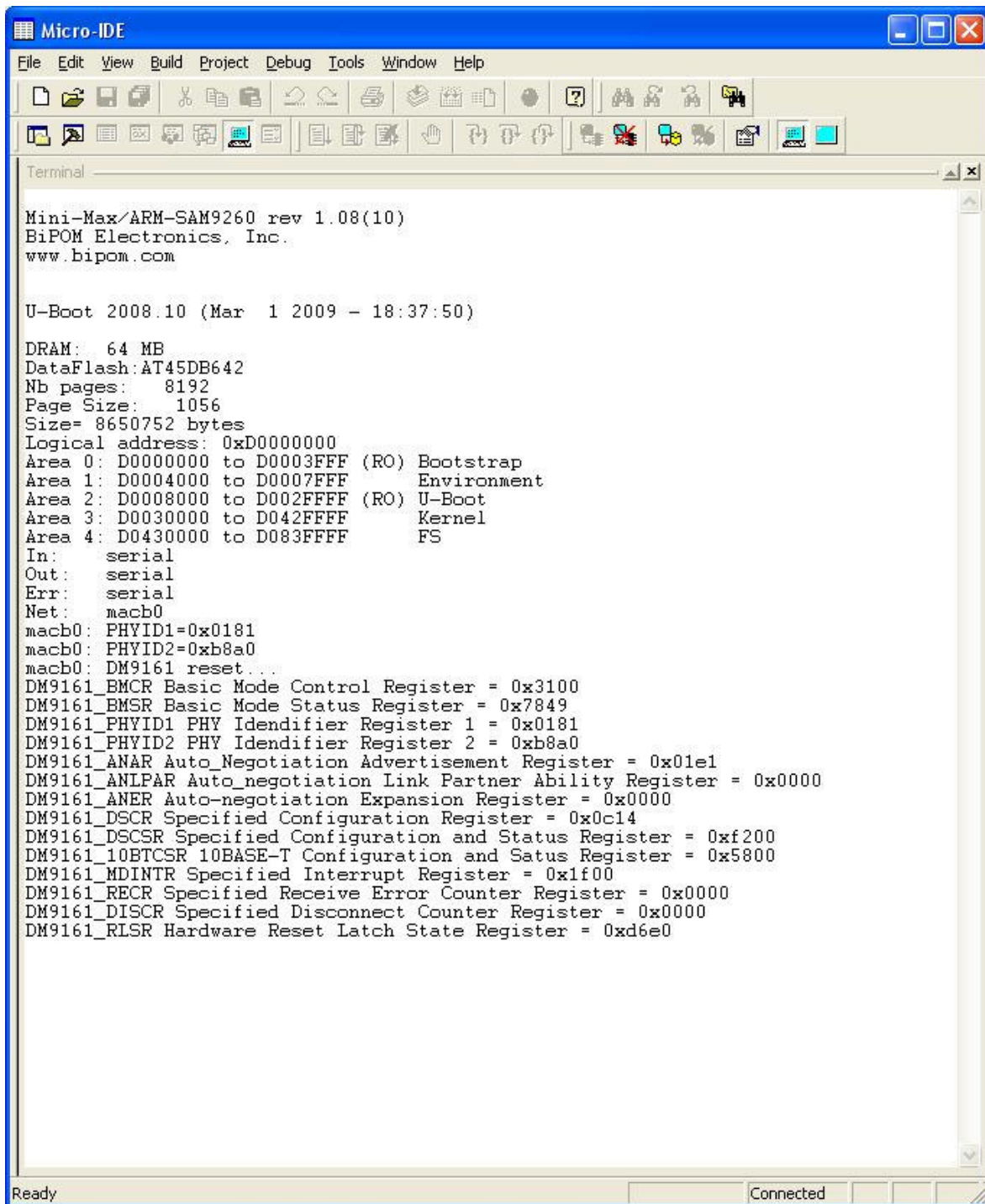
10. Connect the terminal to the board using “Connect” icon button



11. Power the board from a power adapter. 1 RED and 1 GREEN LED's will get ON.



12. MINI-MAX/ARM9260-E will boot Linux and within few seconds it will be ready to use.



```
Micro-IDE
File Edit View Build Project Debug Tools Window Help

Terminal

Mini-Max/ARM-SAM9260 rev 1.08(10)
BiPOM Electronics, Inc.
www.bipom.com

U-Boot 2008.10 (Mar  1 2009 - 18:37:50)

DRAM:  64 MB
DataFlash:AT45DB642
Nb pages:  8192
Page Size:  1056
Size= 8650752 bytes
Logical address: 0xD0000000
Area 0: D0000000 to D0003FFF (RO) Bootstrap
Area 1: D0004000 to D0007FFF (RO) Environment
Area 2: D0008000 to D002FFFF (RO) U-Boot
Area 3: D0030000 to D042FFFF (RO) Kernel
Area 4: D0430000 to D083FFFF (RO) FS
In:  serial
Out: serial
Err: serial
Net: macb0
macb0: PHYID1=0x0181
macb0: PHYID2=0xb8a0
macb0: DM9161 reset...
DM9161_BMCR Basic Mode Control Register = 0x3100
DM9161_BMSR Basic Mode Status Register = 0x7849
DM9161_PHYID1 PHY Identifier Register 1 = 0x0181
DM9161_PHYID2 PHY Identifier Register 2 = 0xb8a0
DM9161_ANAR Auto_Negotiation Advertisement Register = 0x01e1
DM9161_ANLPAR Auto_negotiation Link Partner Ability Register = 0x0000
DM9161_ANER Auto-negotiation Expansion Register = 0x0000
DM9161_DSCR Specified Configuration Register = 0x0c14
DM9161_DSCSR Specified Configuration and Status Register = 0xf200
DM9161_10BTCSR 10BASE-T Configuration and Satus Register = 0x5800
DM9161_MDINTR Specified Interrupt Register = 0x1f00
DM9161_RECR Specified Receive Error Counter Register = 0x0000
DM9161_DISCR Specified Disconnect Counter Register = 0x0000
DM9161_RLSR Hardware Reset Latch State Register = 0xd6e0

Ready Connected
```


13. This is a typical Linux log on MINI-MAX/ARM9260-E board:

Mini-Max/ARM-SAM9260 rev 1.08(10)
BiPOM Electronics, Inc.
www.bipom.com

U-Boot 2008.10 (Mar 1 2009 - 18:37:50)

DRAM: 64 MB
DataFlash:AT45DB642
Nb pages: 8192
Page Size: 1056
Size= 8650752 bytes
Logical address: 0xD0000000
Area 0: D0000000 to D0003FFF (RO) Bootstrap
Area 1: D0004000 to D0007FFF Environment
Area 2: D0008000 to D002FFFF (RO) U-Boot
Area 3: D0030000 to D042FFFF Kernel
Area 4: D0430000 to D083FFFF FS
In: serial
Out: serial
Err: serial
Net: macb0
macb0: PHYID1=0x0181
macb0: PHYID2=0xb8a0
macb0: DM9161 reset...
DM9161_BMCR Basic Mode Control Register = 0x3100
DM9161_BMSR Basic Mode Status Register = 0x7849
DM9161_PHYID1 PHY Identifier Register 1 = 0x0181
DM9161_PHYID2 PHY Identifier Register 2 = 0xb8a0
DM9161_ANAR Auto_Negotiation Advertisement Register = 0x01e1
DM9161_ANLPAR Auto_negotiation Link Partner Ability Register = 0x0000
DM9161_ANER Auto-negotiation Expansion Register = 0x0000
DM9161_DSCR Specified Configuration Register = 0x0c14
DM9161_DSCSR Specified Configuration and Status Register = 0xf200
DM9161_10BTCSR 10BASE-T Configuration and Status Register = 0x5800
DM9161_MDINTR Specified Interrupt Register = 0x1f00
DM9161_RECR Specified Receive Error Counter Register = 0x0000
DM9161_DISCR Specified Disconnect Counter Register = 0x0000
DM9161_RLSR Hardware Reset Latch State Register = 0xd6e0
macb0: Autonegotiation timed out (status=0x7849)
macb0: link down (status: 0x7849)
Hit any key to stop autoboot: 0
(Re)start USB...
USB: scanning bus for devices... 2 USB Device(s) found
scanning bus for storage devices... 1 Storage Device(s) found
reading logo
.....
307200 bytes read
(Re)start USB...
USB: scanning bus for devices... 2 USB Device(s) found
scanning bus for storage devices... 1 Storage Device(s) found
reading uimage
.....

1622943 bytes read
reading mmfs.gz

.....
.....
2606700 bytes read
Booting kernel from Legacy Image at 21000000 ...
Image Name:
Image Type: ARM Linux Kernel Image (gzip compressed)
Data Size: 1622879 Bytes = 1.5 MB
Load Address: 20008000
Entry Point: 20008000
Verifying Checksum ... OK
Uncompressing Kernel Image ... OK

Starting kernel ...

Linux version 2.6.30.4 (mini@mini-max) (gcc version 4.3.2 (Sourcery G++ Lite 2008q3-41)) #66
Mon Oct 5 15:29:43 EEST 2009
CPU: ARM926EJ-S [41069265] revision 5 (ARMv5TEJ), cr=00053177
CPU: VIVT data cache, VIVT instruction cache
Machine: Mini-Max/ARM9260
Memory policy: ECC disabled, Data cache writeback
Clocks: CPU 198 MHz, master 99 MHz, main 18.432 MHz
Built 1 zonelists in Zone order, mobility grouping on. Total pages: 16256
Kernel command line: mem=64M console=ttyS1,115200 root=/dev/ram0 rw
initrd=0x20410000,0x800000 ramdisk_size=8200
NR_IRQS:192
AT91: 96 gpio irqs in 3 banks
PID hash table entries: 256 (order: 8, 1024 bytes)
Console: colour dummy device 80x30
Dentry cache hash table entries: 8192 (order: 3, 32768 bytes)
Inode-cache hash table entries: 4096 (order: 2, 16384 bytes)
Memory: 64MB = 64MB total
Memory: 53408KB available (2816K code, 240K data, 112K init, 0K highmem)
Calibrating delay loop... 99.12 BogoMIPS (lpj=495616)
Mount-cache hash table entries: 512
CPU: Testing write buffer coherency: ok
net_namespace: 296 bytes
NET: Registered protocol family 16
bio: create slab <bio-0> at 0
SCSI subsystem initialized
usbcore: registered new interface driver usbfs
usbcore: registered new interface driver hub
usbcore: registered new device driver usb
cfg80211: Calling CRDA to update world regulatory domain
NET: Registered protocol family 2
IP route cache hash table entries: 1024 (order: 0, 4096 bytes)
TCP established hash table entries: 2048 (order: 2, 16384 bytes)
TCP bind hash table entries: 2048 (order: 1, 8192 bytes)
TCP: Hash tables configured (established 2048 bind 2048)
TCP reno registered
NET: Registered protocol family 1
Trying to unpack rootfs image as initramfs...
rootfs image is not initramfs (no cpio magic); looks like an initrd

Freeing initrd memory: 8192K
JFFS2 version 2.2. (NAND) ?© 2001-2006 Red Hat, Inc.
ROMFS MTD (C) 2007 Red Hat, Inc.
msgmni has been set to 120
alg: No test for stdrng (krng)
io scheduler noop registered
io scheduler anticipatory registered (default)
atmel_usart.0: ttyS0 at MMIO 0xfefff200 (irq = 1) is a ATMEL_SERIAL
atmel_usart.1: ttyS1 at MMIO 0xffffb0000 (irq = 6) is a ATMEL_SERIAL
console [ttyS1] enabled
atmel_usart.2: ttyS2 at MMIO 0xffffb4000 (irq = 7) is a ATMEL_SERIAL
brd: module loaded
ssc ssc.0: Atmel SSC device at 0xc4878000 (irq 14)
Driver 'sd' needs updating - please use bus_type methods
atmel_spi atmel_spi.1: Atmel SPI Controller at 0xffcc000 (irq 13)
ohci_hcd: USB 1.1 'Open' Host Controller (OHCI) Driver
at91_ohci at91_ohci: AT91 OHCI
at91_ohci at91_ohci: new USB bus registered, assigned bus number 1
at91_ohci at91_ohci: irq 20, io mem 0x00500000
usb usb1: configuration #1 chosen from 1 choice
hub 1-0:1.0: USB hub found
hub 1-0:1.0: 2 ports detected
Initializing USB Mass Storage driver...
usbcore: registered new interface driver usb-storage
USB Mass Storage support registered.
udc: at91_udc version 3 May 2006
g_serial gadget: Gadget Serial v2.4
g_serial gadget: g_serial ready
mice: PS/2 mouse device common for all mice
rtc-at91sam9 at91_rtt.0: rtc core: registered at91_rtt as rtc0
IRQ 1/rtc0: IRQF_DISABLED is not guaranteed on shared IRQs
i2c-gpio i2c-gpio: using pins 55 (SDA) and 56 (SCL)
Linux video capture interface: v2.00
AT91SAM9 Watchdog: sorry, watchdog is disabled
at91_wdt: probe of at91_wdt failed with error -5
sdhci: Secure Digital Host Controller Interface driver
sdhci: Copyright(c) Pierre Ossman
Advanced Linux Sound Architecture Driver Version 1.0.20.
at73c213 spi.1.0: at73c213: supported bitrate is 48500 (64 divider)
mmc0: SD card claims to support the incompletely defined 'low voltage range'. This will be ignored.
mmc0: host does not support reading read-only switch. assuming write-enable.
mmc0: new SD card at address b368
mmcbk0: mmc0:b368 SD256 244 MiB
mmcbk0: p1
usb 1-2: new full speed USB device using at91_ohci and address 2
usb 1-2: configuration #1 chosen from 1 choice
scsi0 : SCSI emulation for USB Mass Storage devices
ALSA device list:
#0: AT91SAM9260-EK external DAC on irq 14
TCP cubic registered
NET: Registered protocol family 17
VFP support v0.3: not present
rtc-at91sam9 at91_rtt.0: setting system clock to 2009-10-07 12:56:53 UTC (1254920213)
RAMDISK: gzip image found at block 0
EXT2-fs warning: maximal mount count reached, running e2fsck is recommended

VFS: Mounted root (ext2 filesystem) on device 1:0.
Freeing init memory: 112K

Starting system...

```
scsi 0:0:0:0: Direct-Access   Generic Flash Disk    8.07 PQ: 0 ANSI: 2
sd 0:0:0:0: [sda] 4028416 512-byte hardware sectors: (2.06 GB/1.92 GiB)
sd 0:0:0:0: [sda] Write Protect is off
sd 0:0:0:0: [sda] Assuming drive cache: write through
sd 0:0:0:0: [sda] Assuming drive cache: write through
sda: sda1
sd 0:0:0:0: [sda] Attached SCSI removable disk
Starting user command file...
```

USER.SH - GADGETPC USER COMMAND FILE

Password for 'root' changed

Install ILI9325

ILI9325 Init

ILI9325 Probe

ILI9325 Start:0x10000000 End: 0x104fffff

ILI9325 VER 1.01 05/12/2009

ILI9325 Thread Enter

Install DM9161A Ethernet interface

MACB_mii_bus: probed

eth0: Atmel MACB at 0xfffc4000 irq 21 (00:50:c2:46:90:02)

eth0: attached PHY driver [Davicom DM9161A] (mii_bus:phy_addr=ffffff:00, irq=-1)

HOST IP: NETMASK: GATE IP:

Update all devices

Start Telnet Server

Start VSFTP server

Password for 'ftp' changed

Start Web (HTTP) Server

Start OpenSSH server

Password for 'sshd' changed

Create links to support usbutils

INSTALL MINIGUI LIBRARIES

Run user defined shell commands

USER.SH

Please press Enter to activate this console.

```
[root@Mini-Max/ARM9 /]$cd /
```

```
[root@Mini-Max/ARM9 /]$dir
```

```
bin    etc    linuxrc  proc    sys    var
cmd    home   lost+found  root   tmp
dev    lib    mnt     sbin   usr
```

```
[root@Mini-Max/ARM9 /]$ls -l
```


Linux Configuration File

vars.sh file contains shell variables that are used by other shell files to configure various aspects of MINI-MAX/ARM9260-E Linux. The following describes the various configuration options in more detail. You can edit **vars.sh** and other ***.sh** files to match your Linux system needs.

Lines starting with '#' sign are comment lines and have no effect on configuration. Echo command prints useful information to console. By default, console output goes to MINI-MAX/ARM9260-E 's serial port.

The **vars.sh** is located in the **/sh** directory of USB Flash Drive. This directory also has other configuration files.

Current MINI-MAX/ARM9260-E release has following configuration files (more files may be added as the software is upgraded and more hardware devices and software packages are added):

Filename	Description
ftp_server.sh	Configure and start FTP server
init_at91sam_adc.sh	Configure ADC hardware on MINI-MAX/ARM9 Series Boards
init_dlink_usb_eth.sh	Configure D-Link DUB-E100 Ethernet Adapter
init_dlink_usb_wifi.sh	Configure D-Link WUA-1340 USB Wi-Fi adapter
init_dm9161a.sh	Configure DM9161A Ethernet adapter
init_ftdi_usb_serial.sh	Configure FTDI USB to serial adapter
init_gspca_usb_webcam.sh	Configure GSPCA USB web camera interface
init_i2c_pcf8574.sh	Configure I2C interface
init_microsd.sh	Mount MicroSD card
init_ports.sh	Set unused I/O port
mail_agent.sh	Configure mail agent
motion_server.sh	Configure and start Motion server
ntpd_server.sh	Configure and start NTPD server
sPCA_video_server.sh	Configure and start SCPA Video server
ssh_server.sh	Configure and start SSH server
system.sh	Configure common system parameters
telnet_server.sh	Configure and start TELNET server
usb_utils.sh	Create links to support USB utilities
user_cmds.sh	User can add new commands here.
vars.sh	Set configuration variables which used in other files
web_server.sh	Configure and start HTTP server

You should NOT edit any **.sh** files except **vars.sh** and **user_cmds.sh**

You can modify the variables in **vars.sh** to configure the Linux system using **Linux Control Panel** as described previously.

user_cmds.sh file is called after the system is initialized. You can add here any UNIX shell commands available in current **MINI-MAX/ARM9260-E Linux Release** or call any software developed for **MINI-MAX/ARM9260-E Linux Release**. By default, **user_cmds.sh** is empty.

The following list describes all predefined variables from **vars.sh** available in current MINI-MAX/ARM9260-E Linux Release.

Appendix A: Linux Configuration Variables

Variable Name :	BIPOMVAR_DM9161A_ETHERNET
Value :	ON OFF
Description :	If set to ON then script will configure DM9161 Ethernet adapter. If set to OFF then script will skip configuration code for this device.
Variable Name :	BIPOMVAR_DLINK_USB_ETHERNET
Value :	ON OFF
Description :	If set to ON then script will configure D-Link DUB-E100 Ethernet adapter. If set to OFF then script will skip configuration code for this device.
Variable Name :	BIPOMVAR_GSPCA_USB_WEBCAM
Value :	ON OFF
Description :	If set to ON then script will configure GSPCA USB web camera. If set to OFF then script will skip configuration code for this device.
Variable Name :	BIPOMVAR_DLINK_USB_WIFI
Value :	ON OFF
Description :	If set to ON then script will configure D-Link WUA-1340 USB Wi-Fi adapter. If set to OFF then script will skip configuration code for this device.
Variable Name :	BIPOMVAR_FTDI_USB_SERIAL
Value :	ON OFF
Description :	If set to ON then script will configure FDTI USB to serial adapter. If set to OFF then script will skip configuration code for this device.
Variable Name :	BIPOMVAR_AT91SAM_ADC
Value :	ON OFF
Description :	If set to ON then script will configure ADC hardware on MINI-MAX/ARM9 Series Boards. If set to OFF then script will skip configuration code for this device.
Variable Name :	BIPOMVAR_I2C_PCF8574
Value :	ON OFF
Description :	If set to ON then script will configure PCF8574 I2C Controller on MINI-MAX/ARM9 Series Boards. If set to OFF then script will skip configuration code for this device.
Variable Name :	BIPOMVAR_MICROSD
Value :	ON OFF
Description :	If set to ON then script will mount MicroSD card. If set to OFF then script will not mount this device.
Variable Name :	BIPOMVAR_CONFIGURE_UNUSED_PORT_PINS
Value :	ON OFF
Description :	If set to ON then script will initialize unused I/O ports on the hardware. If set to OFF then I/O ports will be in not initialized state.

Variable Name : **BIPOMVAR_ROOT_PWD**
Value : **Text string**
Description : This parameter sets password for 'root' user on Linux system.

Variable Name : **BIPOMVAR_DRIVER_SOURCE_PATH**
Value : **UNIX style path**
Description : This parameter sets path to directory where system drivers are located.

Variable Name : **BIPOMVAR_DRIVER_INSTALL_PATH**
Value : **UNIX style path**
Description : This parameter sets path to directory where system drivers are copied.

Variable Name : **BIPOMVAR_HOST_IP**
Value : **IP address**
Description : This parameter sets IP address of device.

Variable Name : **BIPOMVAR_NETMASK**
Value : **IP address**
Description : This parameter sets subnet mask.

Variable Name : **BIPOMVAR_GATE_IP**
Value : **IP address**
Description : This parameter sets IP address of gateway.

Variable Name : **BIPOMVAR_WIFI_KEY**
Value : **Text string**
Description : This parameter sets KEY for Wi-Fi network interface (if used).

Variable Name : **BIPOMVAR_WIFI_SSID**
Value : **Text string**
Description : This parameter sets SSID for Wi-Fi network interface (if used).

Variable Name : **BIPOMVAR_TELNET_SERVER**
Value : **YES | NO**
Description : If set to YES then script will run TELNET server.
 If set to NO then script will not run TELNET server.

Variable Name : **BIPOMVAR_WEB_SERVER**
Value : **YES | NO**
Description : If set to YES then script will run HTTP server.
 If set to NO then script will not run HTTP server.

Variable Name : **BIPOMVAR_HTTP_CFG_FILE_PATH**
Value : **UNIX style path**
Description : This parameter sets path to configuration file

Variable Name : **BIPOMVAR_HTTP_PORT**
Value : **INT (integer number)**
Description : This parameter sets HTTP port

Variable Name : **BIPOMVAR_FTP_SERVER**
Value : **YES | NO**
Description : If set to YES then script will run FTP server.
 If set to NO then script will not run FTP server.

Variable Name : **BIPOMVAR_FTP_USER**
Value : **Text string**
Description : This parameter sets FTP username.

Variable Name : **BIPOMVAR_FTP_PWD**
Value : **Text string**
Description : This parameter sets FTP password.

Variable Name : **BIPOMVAR_OPEN_SSH_SERVER**
Value : **YES | NO**
Description : If set to YES then script will run SSH server.
 If set to NO then script will not run SSH server.

Variable Name : **BIPOMVAR_SSH_USER**
Value : **Text string**
Description : This parameter sets SSH username.

Variable Name : **BIPOMVAR_SSH_PWD**
Value : **Text string**
Description : This parameter sets SSH password.

Variable Name : **BIPOMVAR_SSH_HOST_KEY_PATH**
Value : **UNIX style path**
Description : This parameter sets path to HOST KEY file

Variable Name : **BIPOMVAR_SSH_DSA_KEY_PATH**
Value : **UNIX style path**
Description : This parameter sets path to DSA KEY file

Variable Name : **BIPOMVAR_SSH_RSA_KEY_PATH**
Value : **UNIX style path**
Description : This parameter sets path to RSA KEY file

Variable Name : **BIPOMVAR_OPEN_NTPD_SERVER**
Value : **YES | NO**
Description : If set to YES then script will run NTPD server.
 If set to NO then script will not run NTPD server.

Variable Name : **BIPOMVAR_NTPD_USER**
Value : **Text string**
Description : This parameter sets NTPD username.

Variable Name : **BIPOMVAR_NTPD_PWD**
Value : **Text string**
Description : This parameter sets NTPD password.

Variable Name : **BIPOMVAR_NTPD_CFG_FILE_PATH**
Value : **UNIX style path**
Description : This parameter sets path to configuration file of NTPD server

Variable Name : **BIPOMVAR_MAIL_AGENT**
Value : **YES | NO**
Description : If set to YES then script will run mail agent.
 If set to NO then script will not run mail agent.

Variable Name : BIPOMVAR_USB_UTILS
Value : YES | NO
Description : If set to YES then script creates links to support USB utilities.
If set to NO then script will skip configuration of USB utilities.

Variable Name : BIPOMVAR_SPCA_VIDEO_SERVER
Value : YES | NO
Description : If set to YES then script will run SPCA Video Server.
If set to NO then script will not run SPCA Video Server.

Variable Name : BIPOMVAR_MOTION_SERVER
Value : YES | NO
Description : If set to YES then script will run Motion Server.
If set to NO then script will not run Motion Server.

Variable Name : BIPOMVAR_MOTIONSRV_CFG_FILE_PATH
Value : UNIX style path
Description : This parameter sets path to configuration file of motion server