

Developers Guide:

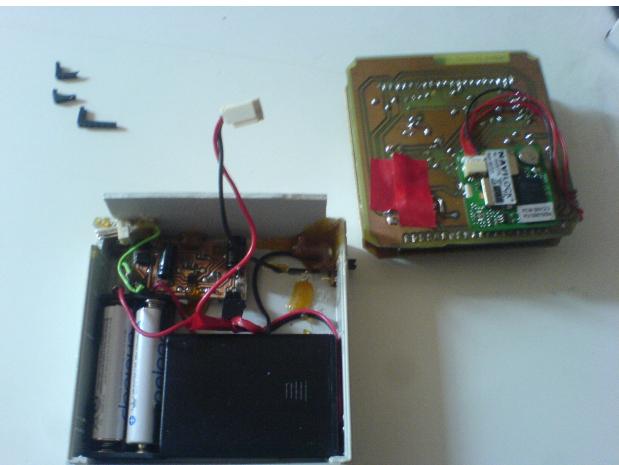
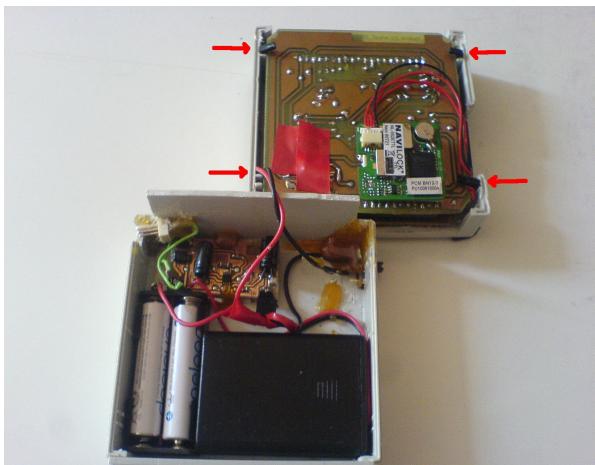
1. Setting up the Hardware:

1. Remove rubber bands



2. Open the device

3. Remove clips and connector



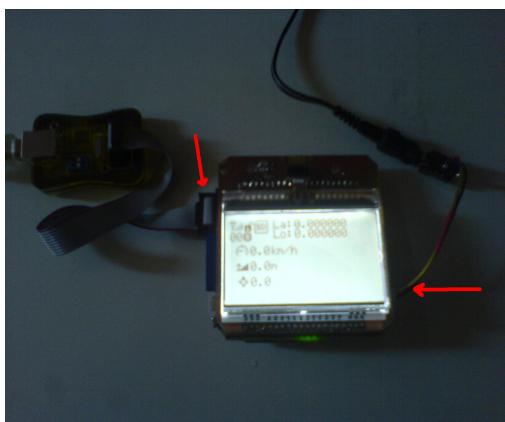
4. plug in external supply voltage and programming device.

Be absolutely sure that the black cable is – and the yellow one is +!

Supply voltages 5.5 V – 9 V.

Next to the SD Slot is the ISP connector. Next to the voltage jack is the JTAG connector.

For JTAG a JTAG controller is needed who is compliant with the ATMega644.



2. Software Setup:

Extract the Linux Image `debian.rar` provided on the DVD

Name ▾ Größe ▾ Typ
debian Dateiordner
debian.rar 1.794.306 KB WinRAR-Archiv

virtual
 debian
 Machine
 debian-5.0-x86
 VDI

Name ▾ Größe ▾ Typ
debian-5.0-x86.vdi 5.497.937 KB VDI-Datei

Get Virtualbox:

<http://www.virtualbox.org/wiki/Downloads>

and set up Virtualbox with the provided Linux Image,

Set up a Filter for the ISP or JTAG Programmer

Oracle VM VirtualBox Manager
Datei Maschine Hilfe

Debian 5.0 ausgeschaltet

Allgemein
Name: Debian 5.0

Debian 5.0 - Ändern

Massenspeicher
TDE-Controller
 debian-5.0-x86.vdi
 VBoxGuestAdditions.iso
SATA-Controller

Attribute
Festplatte: Primärer Master
Information
Typ (Format): normal (VDI)
Virtuelle Größe: 20.00 GB
Wirkliche Größe: 5.29 GB
Ort: D:\virtual\debian\VDI\debian-5.0-x86.vdi
Angeschlossen an: Debian 5.0

Vorschau
Debian 5.0

Debian 5.0 - Ändern

USB
USB-Controller aktivieren
 USB-2.0-Controller aktivieren
Filter für USB-Geräte
 Stange Distribution DX-ISP [0101]

Boot the Linux Image and login with user dev
user: dev, **password:** dev, **root:** linux

You will find the source in the home directory of user dev

src contains the source code

hw contains hardware circuit and layout files

doc contains the documentation of the project

doxygen contains the API documentation

tools contains some tools used

stubs contains some code stubs

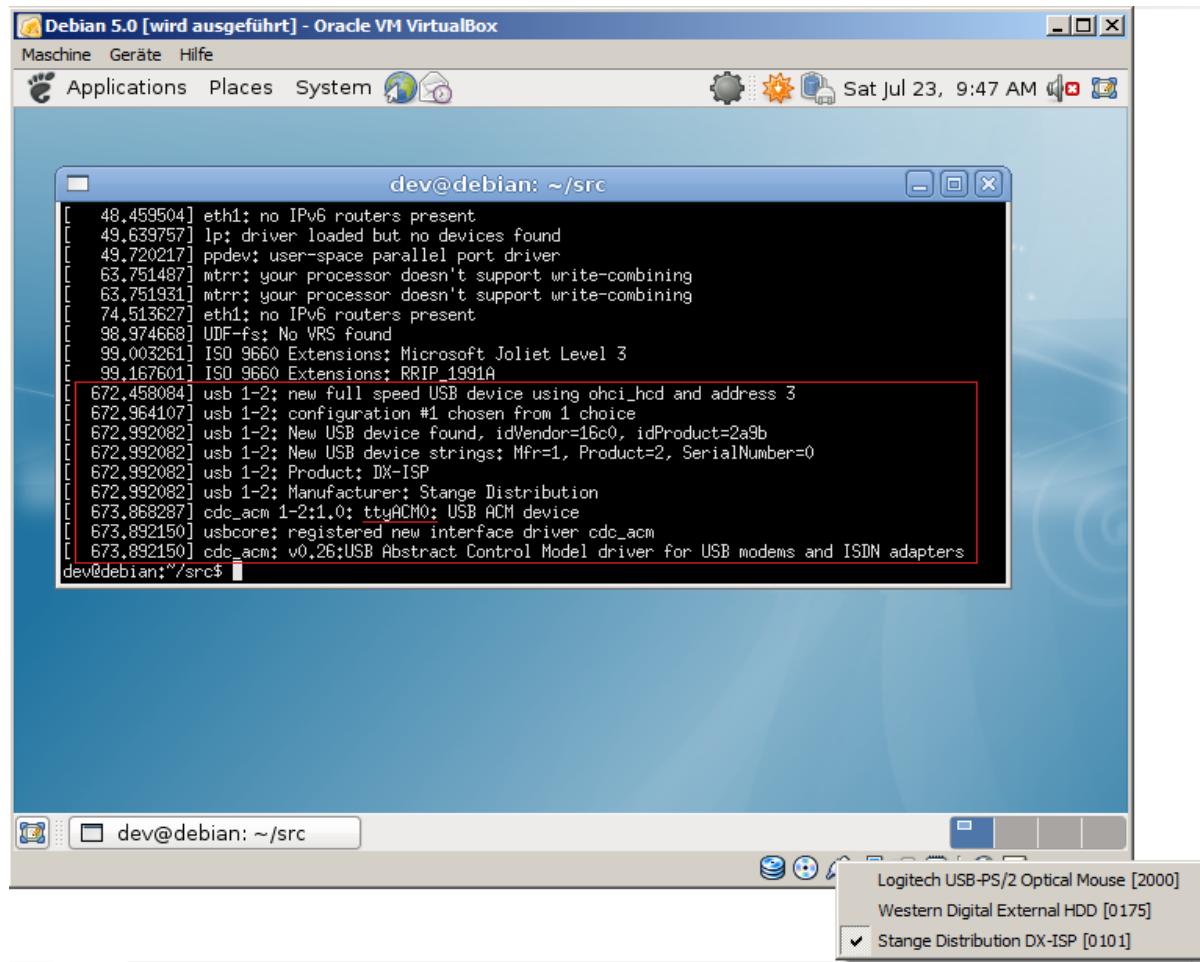
mfile is a program that creates initial Makefiles



Plug in the ISP Programmer

Virtualbox should connect it automatically due to the filter

With the Strange Distribution Diamex ISP Programmer the device is /dev/ttyACM0



If you are not using the Diamex DX-10 ISP Programmer you need to make changes in src/Makefile with the AVRDUDE_PROGRAMMER and the AVR_DUDEPROGRAMMER
Note: See avrdude manpage which protocol you need for your programmer

```
AVRDUDE_PROGRAMMER = stk500v2
AVRDUDE_PORT = /dev/ttyACM0
```

'cd ~/src' and 'make'

The whole source tree is built and if successfully compiled the binary will device will be programmed

After make has sucessfully built the source and programmed the device it gives the following messages

```
avrduude -p atmega644p -P /dev/ttyACM0 -c stk500v2      -U flash:w:main.hex -b 1152
00 -U eeprom:t:w:main.eep

avrduude: AVR device initialized and ready to accept instructions

Reading | ##### | 100% 0.05s

avrduude: Device signature = 0x1e960a
avrduude: NOTE: FLASH memory has been specified, an erase cycle will be performed
          To disable this feature, specify the -D option.
avrduude: erasing chip
avrduude: reading input file "main.hex"
avrduude: input file main.hex auto detected as Intel Hex
avrduude: writing flash (25122 bytes):

Writing | ##### | 100% 5.60s

avrduude: 25122 bytes of flash written
avrduude: verifying flash memory against main.hex:
avrduude: load data flash data from input file main.hex:
avrduude: input file main.hex auto detected as Intel Hex
avrduude: input file main.hex contains 25122 bytes
avrduude: reading on-chip flash data:

Reading | ##### | 100% 6.09s

avrduude: verifying ...
avrduude: 25122 bytes of flash verified
avrduude: reading input file "main.eep"
avrduude: input file main.eep auto detected as Intel Hex
avrduude: writing eeprom (262 bytes):

Writing | ##### | 100% 0.71s

avrduude: 262 bytes of eeprom written
avrduude: verifying eeprom memory against main.eep:
avrduude: load data eeprom data from input file main.eep:
avrduude: input file main.eep auto detected as Intel Hex
avrduude: input file main.eep contains 262 bytes
avrduude: reading on-chip eeprom data:

Reading | ##### | 100% 0.10s

avrduude: verifying ...
avrduude: 262 bytes of eeprom verified

avrduude: safemode: Fuses OK

avrduude done. Thank you.

dev@debian:~/src$
```

If you get an error like this check the permissions of the programmer for your user

```
avrduude: ser_open(): can't open device "/dev/ttyACM0": Permission denied
```

If you get errors like this be sure if the target does not have any short circuits

```
avrduude: AVR device initialized and ready to accept instructions
avrduude: Device signature = 0x000000
avrduude: Yikes! Invalid device signature.
```

If you get errors like this be sure if the target is connected to the supply voltage

```
avrduke: verifying ...
avrduke: verification error, first mismatch at byte 0x0000
    0x0c != 0x00
avrduke: verification error; content mismatch

avrduke: safemode: lfuse changed! Was dc, and is now 0
Would you like this fuse to be changed back? [y/n] █
```

If you get errors like this the ISP Programmer needs to be rest. In order to do that just unplug and reconnect it to the computer

```
avrduke -p atmega644p -P /dev/ttyACM0 -c stk500v2      -U flash:w:main.hex -b 115200 -U eeprom:w:main.eep
avrduke: stk500_2_ReceiveMessage(): timeout
avrduke: stk500_2_ReceiveMessage(): timeout
avrduke: stk500_2_ReceiveMessage(): timeout
avrduke: stk500_2_ReceiveMessage(): timeout
^Cmake: *** [program] Interrupt
```

If you get errors like this make sure the programmer is connected to the computer and the AVRDUDE_PORT is correct in the Makefile

```
avrduke -p atmega644p -P /dev/ttyACM0 -c stk500v2      -U flash:w:main.hex -b 115200 -U eeprom:w:main.eep
avrduke: ser_open(): can't open device "/dev/ttyACM0": Permission denied
make: *** [program] Error 1
```

If you get errors like this make sure the AVRDUDE_PROGRAMMER is correct in the Makefile

```
avrduke -p atmega644p -P /dev/ttyACM0 -c stk200      -U flash:w:main.hex -b 115200 -U eeprom:w:main.eep
avrduke: can't claim device "/dev/ttyACM0": Invalid argument
make: *** [program] Error 1
```

Without errors the device is now successfully programmed

NOTE: We only support the Diamex DX-10 ISP Programmer. Other ISP or JTAG programmers are untested but should work just fine.