Suzie Linux https://suzielinux.com/

Suzie Linux was named in memory of my adorable Maine Coon cat Suzie.

Gentoo Linux for YY3568 boards documentation

Author	Date	Project	Revisions
Michel Catudal	2025-06-05	**YY3568 Gentoo Linux Creation**	**1**

REVISION TRACKING SHEET

Rev	Name	Date	Comment
1	Michel Catudal	2025-06-05	

Table of Content

.....1

Suzie Linux https://suzielinux.com/.....1

Suzie Linux was named in memory of my adorable Maine Coon cat Suzie.....1

1. Hardware.....4

1.1. OVERVIEW OF THE YY3568 BOARD.....4

2. Gentoo applications required for chroot.....6

2.1. GENTOO APPLICATIONS REQUIRED.....6
2.2. APPLICATIONS REQUIRED FOR CHROOT ON MAC VMWARE FUSION DEBIAN.....7

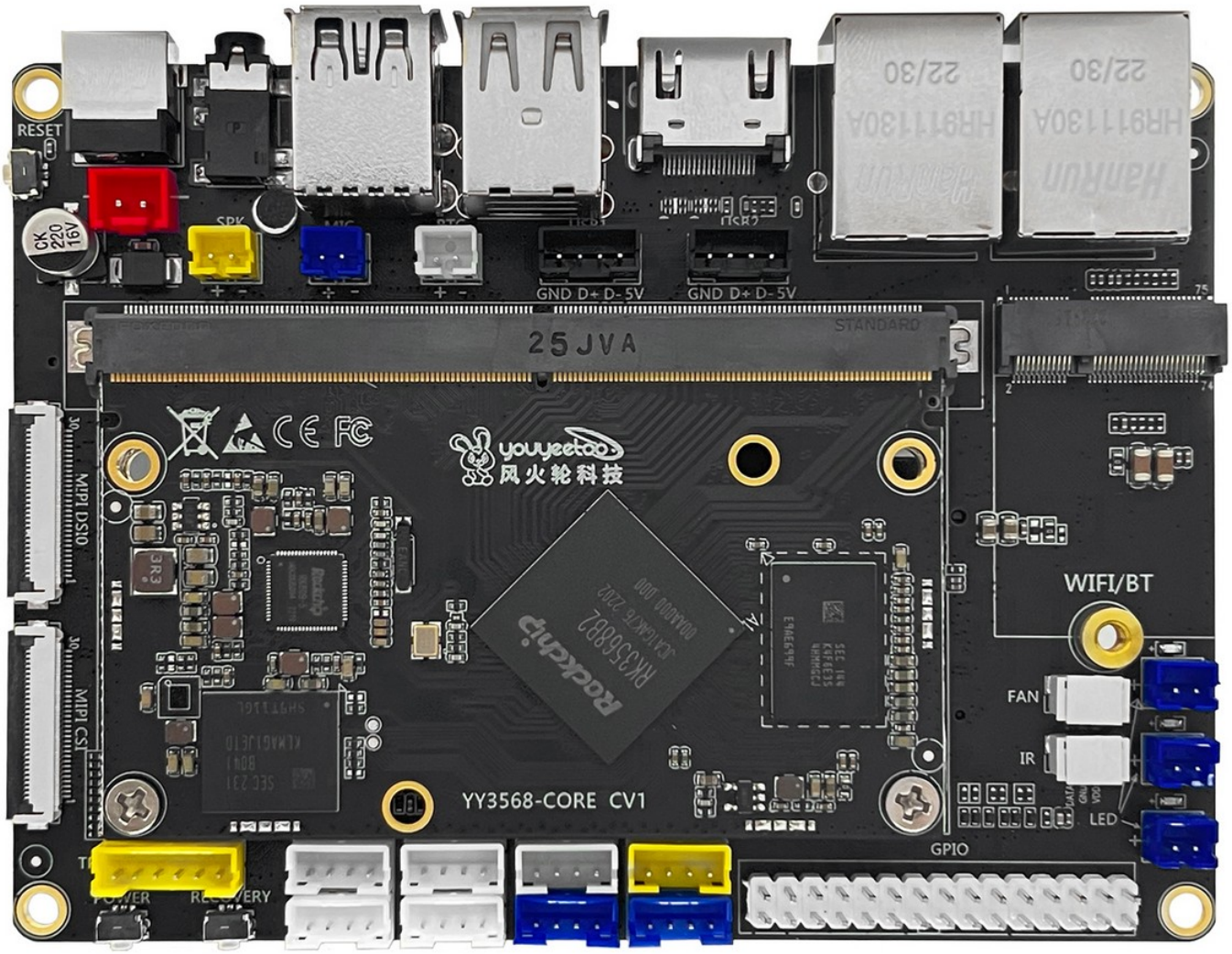
3. Gentoo Linux Root File System.....7

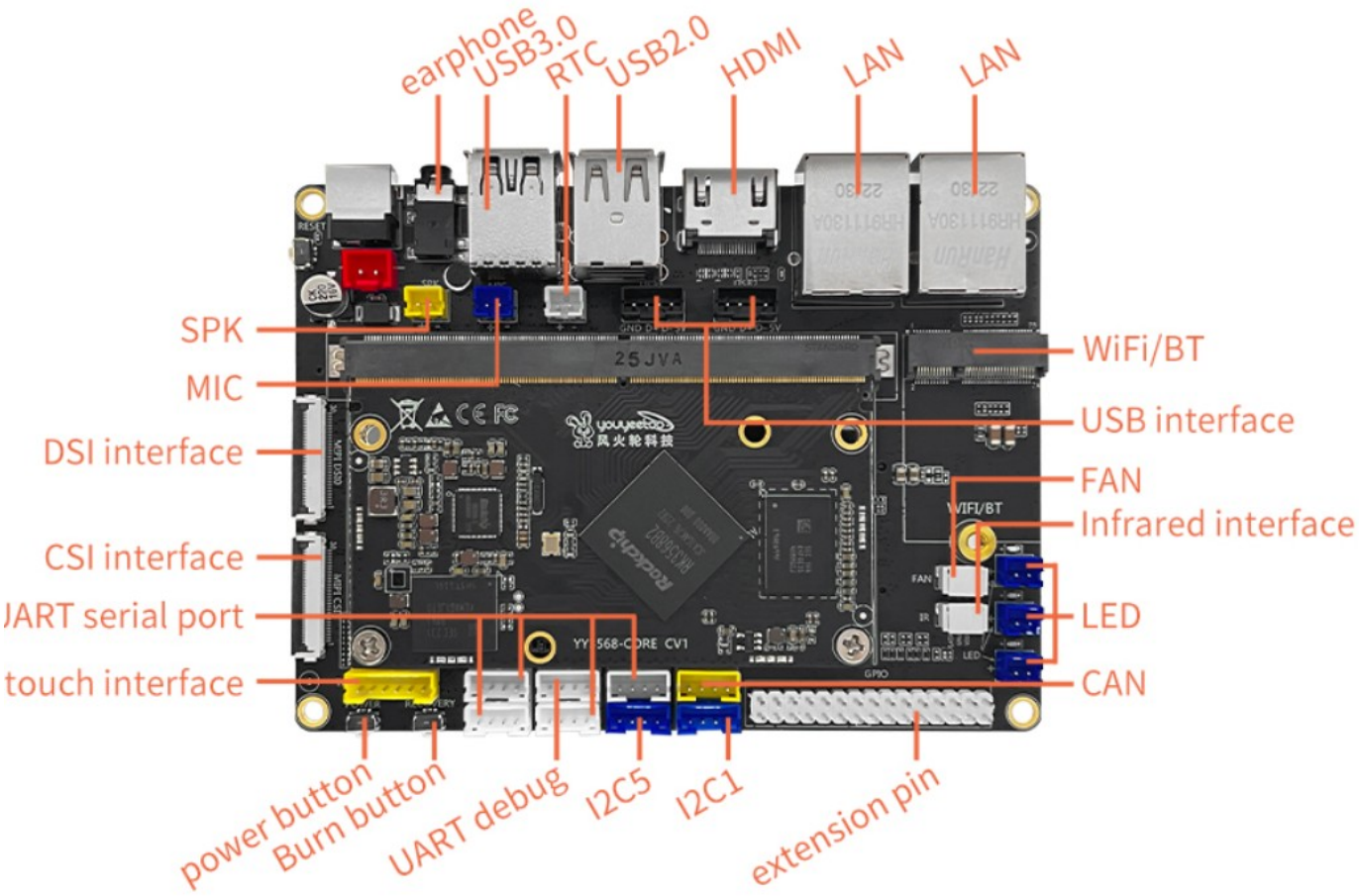
3.1. CREATE A ROOT FILE SYSTEM.....7
3.2. CHROOT INTO GENTOO ROOTFS.....8

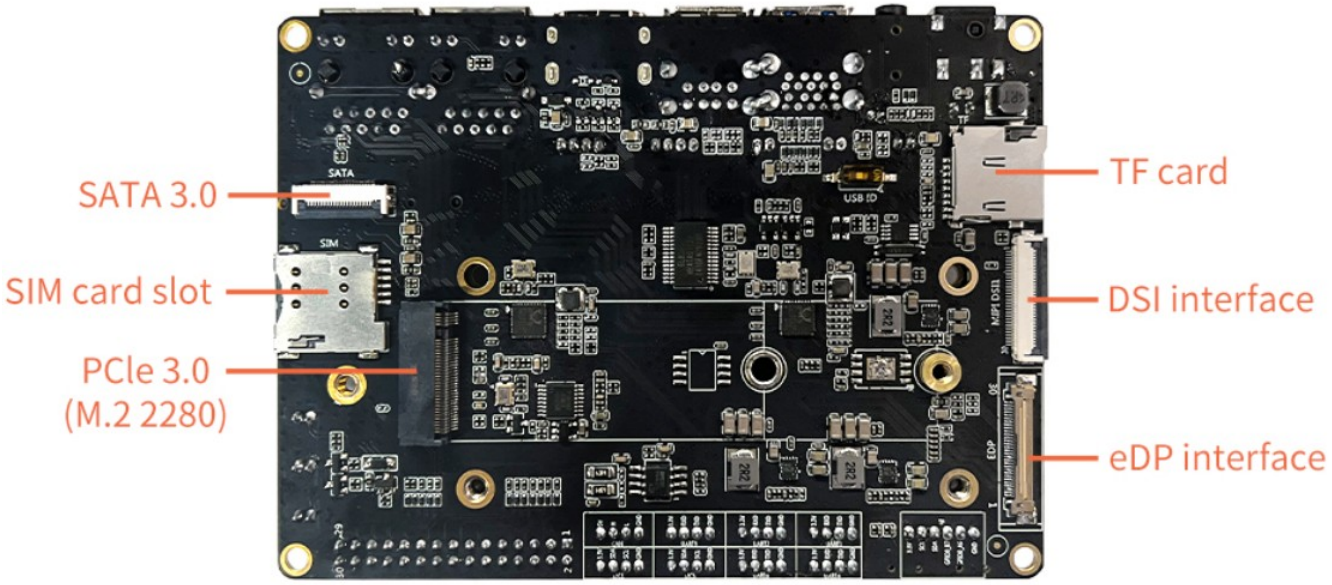
4. Create Gentoo Linux micro SD boot disk.....11

1. Hardware

1.1. Overview of the YY3568 board




earphone
USB3.0
RTC
USB2.0
HDMI
LAN
LAN
SPK
MIC
DSI interface
CSI interface
JART serial port
touch interface
WiFi/BT
USB interface
FAN
Infrared interface
LED
CAN
power button
Burn button
UART debug
I2C5
I2C1
extension pin


SATA 3.0
SIM card slot
PCIe 3.0
(M.2 2280)
TF card
DSI interface
eDP interface

2. Gentoo applications required for chroot

```
cd ~
mkdir yy3568
cd yy3568
export work_directory=$(pwd)
Get misc files needed for the bootloader and rootfs
git clone https://github.com/SuzieLinux/YY3568.git files
```

2.1. Gentoo applications required

```
emerge --ask dev-python/cryptography
emerge --ask dev-python/pyelftools
emerge --ask dev-util/yamllint
emerge --ask dev-libs/libyaml
emerge --ask dev-python/jsonschema
emerge --ask sys-block/bmap-tools
emerge --ask sys-fs/genimage
emerge --ask sys-fs/mtools
emerge --ask gnutls
emerge --ask flex
emerge --ask sys-devel/bc
emerge --ask bison
emerge --ask swig
emerge --ask sys-fs/dosfstools
cd /usr/bin
ln -s mkfs.vfat mkdosfs
emerge --ask sys-apps/arch-chroot

In order to chroot on a arm64 rootfs a few things have to be done.
First you need to make sure that the kernel supports it and emerge needed support
The build system's kernel must support miscellaneous binary formats.
This can be enabled  with CONFIG_BINFMT_MISC=m
or CONFIG_BINFMT_MISC=y in the the kernel's .config file.

A system restart is required after building this module before it can be used.

Enable CONFIG_BINFMT_MISC
Executable file formats  --->
  <*> Kernel support for MISC binaries

USE=static-user needs to be set

Add this to /etc/portage/package.use/qemu :

# Enable static-user and add the arm64 and other targets
app-emulation/qemu static-user QEMU_SOFTMMU_TARGETS: * QEMU_USER_TARGETS: *
# required by app-emulation/qemu::gentoo[static,static-user]
# required by qemu (argument)
dev-libs/glib static-libs
# required by app-emulation/qemu::gentoo[-static,static-user]
# required by qemu (argument)
sys-libs/zlib static-libs
# required by app-emulation/qemu::gentoo[-static,static-user,xattr]
# required by qemu (argument)
```

```
sys-apps/attr static-libs
# required by dev-libs/glib::gentoo
# required by app-emulation/qemu::gentoo[-static,static-user]
# required by qemu (argument)
dev-libs/libpcre2 static-libs

emerge --ask app-emulation/qemu --update --newuse --deep
```

2.2. Applications required for chroot on mac vmware fusion debian

```
sudo apt upgrade
sudo apt install build-essential git vim
sudo apt install gfortran gpc
sudo apt install debhelper fakeroot
sudo apt install python3-cryptography
sudo apt install python3-pyelftools
sudo apt install yamllint
sudo apt install libyaml
sudo apt install libyaml-dev
sudo apt install python3-pyelftools
sudo apt install python3-jsonschema
sudo apt install python-jsonschema
sudo apt install bmap-tools
sudo apt install genimage
sudo apt install dosfstools
sudo apt install mtools
sudo apt install gnutls-dev
sudo apt install flex
sudo apt install bc
sudo apt install bison
sudo apt install swig
sudo apt install arch-chroot-scripts
```

3. Gentoo Linux Root File System

```
export rootfs_dir=$work_directory/gentoo_rootfs
cd $ rootfs_dir

Since this changes often it may be better to go to https://www.gentoo.org/downloads/
and choose the latest arm64 stage 3 openrc

latest_stage3=20250427T235504Z/stage3-arm64-desktop-openrc-20250427T235504Z.tar.xz
wget https://distfiles.gentoo.org/releases/arm64/autobuilds/$latest_stage3
```

3.1. Create a root file System

```
su
mkdir -p $rootfs_dir
tar xfvp stage3-arm64-desktop-openrc-20250427T235504Z.tar.xz -C $rootfs_dir
sync

You may want to edit the locale env and keymaps if your language is not French

cp /usr/bin/qemu-aarch64 $rootfs_dir/usr/bin
cd $rootfs_dir/etc
cp $work_directory/gentoo/files/etc/locale.gen ./
cp $work_directory/gentoo/files/etc/env.d/02locale env.d
```

```
cp $work_directory/gentoo/files/etc/conf.d/keymaps conf.d
cp /etc/resolv.conf ./
cp $work_directory/gentoo/files/scripts/16-set-alias.bash bash/bashrc.d
cp $work_directory/gentoo/files/etc/fstab ./
cp $work_directory/gentoo/files/etc/inittab ./
cd portage
cp $work_directory/gentoo/files/misc/etc/portage/make.conf ./
cd package.accept_keywords
cp $work_directory/gentoo/files/misc/etc/portage/package.accept_keywords/* ./
cd ../package.use
cp $work_directory/gentoo/files/misc/etc/portage/package.use ./
cp -R $work_directory/gentoo/files/misc/etc/savedconfig ./
cd $rootfs_dir/boot
cp -R $work_directory/gentoo/files/misc/boot/extlinux ./
```

3.2. chroot into gentoo rootfs

```
cd $work_directory
It is assumed here that you are still under root

arch-chroot $rootfs_dir
source /etc/profile

This is needed when chroot on gentoo but is not needed on the mac
It actually crashes chroot if you do
export PS1="(chroot) $PS1"

We need a user for later login thru ssh
useradd -m suzie

Here I create simple passwords, after we boot the micro sd we can change them to
more secured password. For all our settings in chroot this approach makes work simple. In
both case it will ask to confirm the password.

For the root password : passwd
For the suzie user password : passwd suzie

emerge-webrsync
eselect profile set 20
emaint --auto sync

For the suzie portage overlay
On this overlay there are two directories:
suzie and metadata

The suzie repository has has two directories:
profile and metadata
Both metadata directories have a file named layout.conf which contains :
masters = gentoo
auto-sync = false

The profiles has a file name repo_name which contains the word suzie

For the time eastern time zone
ln -sf /usr/share/zoneinfo/America/Detroit /etc/localtime
```

```
emerge --ask joe

Setup some links to simulate the cpm-80 wordstar editor
cd /usr/bin
ln -s joe ws
cd /etc/joe
cp jstarrc wsrc

To remove the annoying wordwrap bug delete all mentions of wordwrap in wsrc
It gets to be a pain when you update a script and it cuts a line and you didn't notice
Your script has no chance of working with this ridiculous behavior of the editor.

This part will take quite a bit of time if many programs need to be installed
Which is why it is always important to download the latest
It can be much faster if you chroot on a fast arm64 board or mac with vmware fusion.

locale-gen
emerge --ask --verbose --update --deep --newuse @world
emerge --ask dev-vcs/git subversion
emerge --ask openssh
rc-update add sshd default
gpasswd -a suzie wheel

If you want to be able to ssh as root add this line to /etc/ssh/sshd_config  :
PermitRootLogin yes
emerge --ask wireless-regdb
emerge --ask linux-firmware
emerge --ask lightdm
emerge --ask display_manager
emerge --ask xorg-server
emerge --ask caja libmatekbd mate mate-applets mate-applets-meta mate-common mate-
control-center mate-desktop mate-menus mate-panel mate-session-manager mate-settings-
daemon caja-actions caja-extensions mate-calc mate-indicator-applet mate-media mate-
polkit mate-power-manager mate-screensaver mate-sensors-applet mate-system-monitor
mate-user-share mate-utils
emerge --ask dbus libdbus
emerge --ask dbus-monitorrc-update add dbus default
rc-update add display-manager default
emerge --ask net-misc/ntp
rc-update add hwclock boot
rc-update add ntp-client default

In /etc/init.d/ntp-client add sleep 10, this will delay trying to get the time for 10
seconds.
It could be lowered if the network is on quicker than that.

start() {
        checkconfig || return $?
        # Delay NTP client startup by 10 seconds
        sleep 10

        ebegin "Setting clock via the NTP client '${NTPCLIENT_CMD}'"
        "${NTPCLIENT_CMD}" ${NTPCLIENT_OPTS}
        eend $? "Failed to set clock"
}
```

```
This part could take 8-16 hours if not done on the mac
It will take close to an hour on the mac, maybe a little bit more.

emerge --ask dev-embedded/u-boot-tools
emerge --ask sys-kernel/dracut
emerge --ask sys-kernel/installkernel
emerge --ask kernel-source

cd /usr/src
ln -sf linux-6.15.0 linux
cd linux
cp /etc/savedconfig/sys-kernel/gentoo-kernel .config
make menuconfig   (if you want to make configuration changes)
make
make dtbs         (if dtb are not created, the make might have created them)
make modules_install
make dtbs_install
make install

You may want to delete the compiled files to reduce the size of the image

make mrproper

kernel_version=linux-6.15.0-catu

cd /boot
mkimage -A arm64 -O linux -T ramdisk -C gzip -n uInitrd -d initramfs-linux-6.15.0-
catu.img uInitrd-linux-6.15.0-catu

ln -sf vmlinuz-6.15.0-gentoo-catu Image
ln -sf uInitrd-linux-6.15.0-catu uInitrd
ln -sf System.map-6.15.0-gentoo-catu System.map
ln -sf config-6.15.0-gentoo-catu config
ln -sf dtbs/6.15.0-gentoo-catu/rockchip dtb

To leave chroot type exit

cd $work_directory/gentoo_rootfs
NOW=$(date +"%Y%m%d%H%M")
sudo tar cvfJ $work_directory/gentoo-yy3568-rootfs-$NOW.xz *
```

4. Create Gentoo Linux micro SD boot disk

If chroot was not done on gentoo it needs to be copied to the gentoo disk

```
cd ~/yy3568
export work_directory=$(pwd)

The needed files should be on $work_directory/gentoo/files

mkdir -p input
sudo $work_directory/gentoo/files/scripts/mk_gentoo_rootfs.sh

We create the SD boot disk using a working u-boot image.

uboot_image=$work_directory/gentoo/files/misc/boot/uboot.img
gentoo_rootfs=$work_directory/input/rootfs.ext4

Change sdd to whatever you micro SD is on

su
dd if=$uboot_image of=/dev/sdd bs=512
sync
dd if=$gentoo_rootfs of=/dev/sdd1 status=progress iflag=direct oflag=direct bs=4M
sync
partprobe /dev/sdd
fsck.ext4 -f /dev/sdd1
resize2fs /dev/sdd1

*
```