Enable ubuntu on i.MX8QXP

Yocto BSP: L4.14.98_2.0.0_GA
Rootfs: ubuntu18.04-core
Desktop: weston
Board: i.MX8QXP-MEK

1.Build Yocto L4.14.98_2.0.0_GA
2.Compile the kernel separately to replace yocto kernel
3. Install 18.04 base

```
$ mkdir ~/rootfs && cd ~/rootfs
$ wget http://cdimage.ubuntu.com/ubuntu-base/releases/18.04.2/release/ubuntu-base-18.04.2-base-arm64.tar.gz
$ mkdir ubuntu-rootfs
$ tar -xvf ubuntu-base-18.04.2-base-arm64.tar.gz -C ubuntu-rootfs
$ cd ubuntu-rootfs
$ ls
```

4.Install qemu-user-static on PC to simulate arm64 environment

```
$ apt-get install qemu-user-static
$ cp /usr/bin/qemu-aarch64-static   usr/bin
$ cp -b /etc/resolv.conf etc/
```

5.Simulate arm64 environment

```
$ cd ../

$ vi ch-mount.sh
```

(copy the content to ch-mount.sh)

```
#!/bin/bash
#
function mnt() {
    echo "MOUNTING"
    sudo mount -t proc /proc ${2}proc
    sudo mount -t sysfs /sys ${2}sys
    sudo mount -o bind /dev ${2}dev
    sudo mount -o bind /dev/pts ${2}dev/pts
    sudo chroot ${2}
}
function umnt() {
    echo "UNMOUNTING"
    sudo umount ${2}proc
    sudo umount ${2}sys
    sudo umount ${2}dev/pts
    sudo umount ${2}dev
}
if [ "$1" == "-m" ] && [ -n "$2" ] ;
then
    mnt $1 $2
elif [ "$1" == "-u" ] && [ -n "$2" ];
then
    umnt $1 $2
else
    echo ""
    echo "Either 1'st, 2'nd or both parameters were missing"
    echo ""
    echo "1'st parameter can be one of these: -m(mount) OR -u(umount)"
    echo "2'nd parameter is the full path of rootfs directory(with trailing '/')"
    echo ""
    echo "For example: ch-mount -m /media/sdcard/"
    echo ""
    echo 1st parameter : ${1}
    echo 2nd parameter : ${2}
fi
```

(chroot to arm64 filesystem)

```
$ ./ch-mount.sh -m ubuntu-rootfs/
```

6.Install package and configuration

```
# chmod 777 tmp
#echo nameserver 8.8.8.8 > /etc/resolv.conf   or   #echo nameserver 8.8.4.4 > /etc/resolv.conf
```

```
# apt-get update
```

Install package you need

```
apt-get install \
  language-pack-en-base \
  sudo \
  ssh \
  net-tools \
  network-manager \
  iputils-ping \
  rsyslog \
  bash-completion \
  htop \
  resolvconf \
  dialog \
  vim
```

```
#apt-get install v4l-utils alsa-utils git gcc less resolvconf autoconf autopoint libtool bison flex gtk-doc-tools glib-2.0 libglib2.0-dev libpango1.0-dev libatk1.0-dev kmod pciutils
```

7. Add user

```
$ useradd -s '/bin/bash' -m -G adm,sudo yourusername
$ echo "Set password for yourusername:"
$ passwd yourusername
$ echo "Set password for root:"
$ passwd root
```

3) Set host name

```
$ echo 'ubuntu.yourusername' > /etc/hostname
```

8.Configure the serial port

```
$ sudo cp -ra /home/sk/bld_server2/users/skamiya/yocto/L4.14.98/imx-yocto-bsp-ft/build-8qxpmek-xwayland/tmp/work/imx8qxpmek-poky-linux/systemd-serialgetty/1.0-r5/image/*   ubuntu-rootfs/
```

9. copy imx-gpu-viv

1) check not drm,mesa,libglapi,wayland under ubuntu-rootfs/usr/lib/aarch64-linux-gnu. if exit that to delete it.

```
$ sudo rm -rf ubuntu-rootfs/usr/lib/aarch64-linux-gnu/libdrm*
$sudo rm -rf ubuntu-rootfs/usr/lib/aarch64-linux-gnu/mesa-egl/*
$sudo rm -rf ubuntu-rootfs/usr/lib/aarch64-linux-gnu/libglapi.so.0*
$sudo rm -rf /usr/lib/aarch64-linux-gnu/libwayland-*
```

2)copy libdrm,wayland vpu-firmware from yocto

```
$sudo cp -Pra libdrm/2.4.91.imx-r0/image/* ubuntu-rootfs
$sudo cp -Pra imx-gpu-viv/1_6.2.4.p4.0-aarch64-r0/image/* ubuntu-rootfs
$sudo cp -Pra imx-dpu-g2d-1.7.0/imx-dpu-g2d/1.7.0-r0/image/* ubuntu-rootfs
$sudo cp -r     all-poky-linux/firmware-imx/1_8.1-r0/image/* ubuntu-rootfs
```

10.build weston

1)Build wayland (to remove wayland-egl dependency on mesa)

```
#wget https://wayland.freedesktop.org/releases/wayland-1.16.0.tar.xz
#apt-get install libffi-dev libxml2-dev
#tar -xvf ./wayland-1.16.0.tar.xz
#./configure --disable-documentation prefix=/usr
```

```
#make
#make install
#ldconfig
```

2)Build wayland protocol (because wayland protocol version mismatch)

```
#git clone   https://source.codeaurora.org/external/imx/wayland-protocols-imx.git
#cd wayland-protocols-imx/
#git checkout e05c19d9520f0b1289cf0844d6e2f877114f39d5
#./autogen.sh --prefix=/usr
#make install
#ldconfig
```

3)#apt-get install libudev-dev libinput-dev libxkbcommon-dev libpam0g-dev

```
#mkdir ~/weston

#cd ~/weston
#git clone https://source.codeaurora.org/external/imx/weston-imx.git
#cd weston-imx
#git checkout fb563901657b296c7c7c86d26602a622429e334f

#./autogen.sh --prefix=/usr --disable-silent-rules --disable-dependency-tracking --enable-setuid-install --disable-rdp-compositor --enable-clients --enable-simple-clients --enable-demo-clients-install --disable-colord --enable-egl --enable-simple-egl-clients --enable-fbdev-compositor --disable-headless-compositor --enable-drm-compositor --enable-weston-launch --disable-lcms --disable-libunwind --with-pam --disable-vaapi-recorder --enable-wayland-compositor --without-webp --disable-x11-compositor --disable-xwayland

#make
#make install
#ldconfig
```

11.build gstreamer

1)

```
#git clone https://source.codeaurora.org/external/imx/gstreamer.git
#cd gstreamer/
#git checkout d42548da09724ad8cc1aa4f1944607920be2f4c0
#./autogen.sh --disable-gtk-doc --prefix=/usr/
#make
#make install
#ldconfig
```

2)plugins

```
#git clone https://source.codeaurora.org/external/imx/gst-plugins-base.git
#cd gst-plugins-base/
#git checkout f62f8ab685e549eefaecdb1e700239a5a408fcad
#./autogen.sh --disable-gtk-doc --prefix=/usr/
#make
#make install
#ldconfig

#git clone https://source.codeaurora.org/external/imx/gst-plugins-good.git
#cd gst-plugins-good
#git checkout a31aabb7581d3f491cd31889d44479bb0f34990b
#./autogen.sh --disable-gtk-doc --prefix=/usr/

#make
#make install
#ldconfig

#git clone https://source.codeaurora.org/external/imx/gst-plugins-bad.git
#cd gst-plugins-bad
#git checkout 0191521ba226904e4b2f84c38e5f6ae75169a18a
#./autogen.sh --disable-gtk-doc --prefix=/usr/
#make
#make install
#ldconfig
```

12. replace filesystem

```
$ mkdir mountpoint
$ umount /dev/sdb2
$ mkfs.ext4 /dev/sdb2
$ mount /dev/sdb2   /home/gnar/mountpoint
$ cp -a /home/gnar/rootfs/ubuntu-rootfs /* /home/gnar/mountpoint/
umount /home/gnar/mountpoint
sync
```

13. (Feature) Ethernet

```
$ ifconfig eth0 up
$ dhclient eth0
```

14. start weston

```
export XDG_RUNTIME_DIR=/run/user/1000
sudo -E weston --tty=1
```