### 16.4 在100ASK\_IMX6ULL上机实验

本节视频既适用于“驱动基础”，也适用于“驱动大全”，所以它对应的源码在2个GIT仓库里都有。

驱动基础的GIT仓库：

01\_all\_series\_quickstart\

05\_嵌入式Linux驱动开发基础知识\source\

05\_gpio\_and\_pinctrl\

01\_led

驱动大全的GIT仓库：

doc\_and\_source\_for\_drivers\IMX6ULL\source\07\_GPIO\01\_led

doc\_and\_source\_for\_drivers\STM32MP157\source\A7\07\_GPIO\01\_led

#### 16.4.1 确定引脚并生成设备树节点

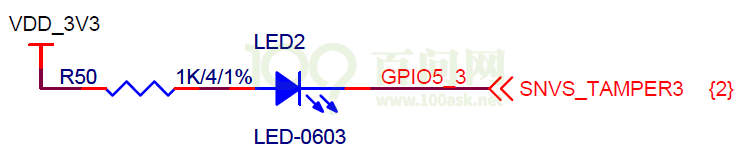
NXP公司对于IMX6ULL芯片，有设备树生成工具。它太大了，我们把它上传到了网盘中，使用GIT命令载后，在这个目录下有下载说明：

doc\_and\_source\_for\_drivers\

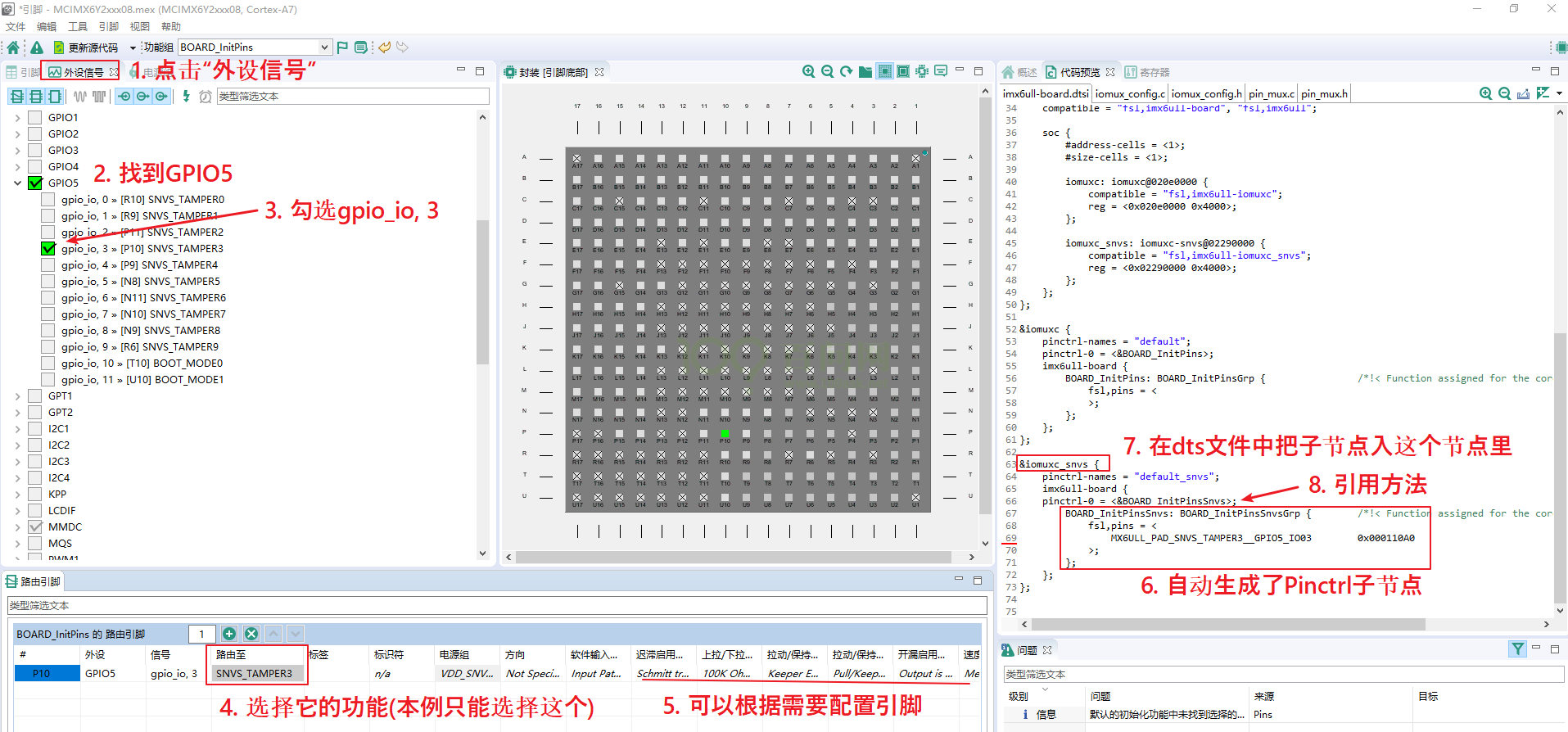
工具\imx

安装“Pins\_Tool\_for\_i.MX\_Processors\_v6\_x64.exe”后运行，打开IMX6ULL的配置文件“MCIMX6Y2xxx08.mex”，就可以在GUI界面中选择引脚，配置它的功能，这就可以自动生成Pinctrl的子节点信息。

100ASK\_IMX6ULL使用的LED原理图如下，可知引脚是GPIO5\_3：



在设备树工具中，如下图操作：



把自动生成的设备树信息，放到内核源码arch/arm/boot/dts/100ask\_imx6ull-14x14.dts中，代码如下：

a. Pinctrl信息：

&iomuxc\_snvs {

……

imx6ul-evk {

myled\_for\_gpio\_subsys: myled\_for\_gpio\_subsys{

fsl,pins = <

MX6ULL\_PAD\_SNVS\_TAMPER3\_\_GPIO5\_IO03 0x000110A0

>;

};

……

}

b. 设备节点信息(放在根节点下)：

myled {

compatible = "100ask,leddrv";

pinctrl-names = "default";

pinctrl-0 = <&myled\_for\_gpio\_subsys>;

led-gpios = <&gpio5 3 GPIO\_ACTIVE\_LOW>;

};

为避免引脚冲突，还要修改arch/arm/boot/dts/100ask\_imx6ull-14x14.dts，在leds节点中如下增加status属性，禁止它：

leds {

compatible = "gpio-leds";

pinctrl-names = "default";

pinctrl-0 = <&pinctrl\_leds>;

status = "disabled";

led0: cpu {

label = "cpu";

gpios = <&gpio5 3 GPIO\_ACTIVE\_LOW>;

default-state = "on";

linux,default-trigger = "heartbeat";

};

};

#### 16.4.2 编译程序

先设置工具链：

export ARCH=arm

export CROSS\_COMPILE=arm-linux-gnueabihf-

export PATH=$PATH:/home/book/100ask\_imx6ull-sdk/ToolChain/gcc-linaro-6.2.1-2016.11-x86\_64\_arm-linux-gnueabihf/bin

编译设备树后，要更新设备树：

cd /home/book/100ask\_imx6ull-sdk/Linux-4.9.88

make dtbs // 得到arch/arm/boot/dts/100ask\_imx6ull-14x14.dtb，把它放到单板/boot目录下

编译驱动程序时，“leddrv\_未测试的原始版本.c”是有错误信息的，“leddrv.c”是修改过的。

测试方法，在板子上执行命令：

# insmod leddrv.ko

# ls /dev/100ask\_led0

# ./ledtest /dev/100ask\_led0 on

# ./ledtest /dev/100ask\_led0 off