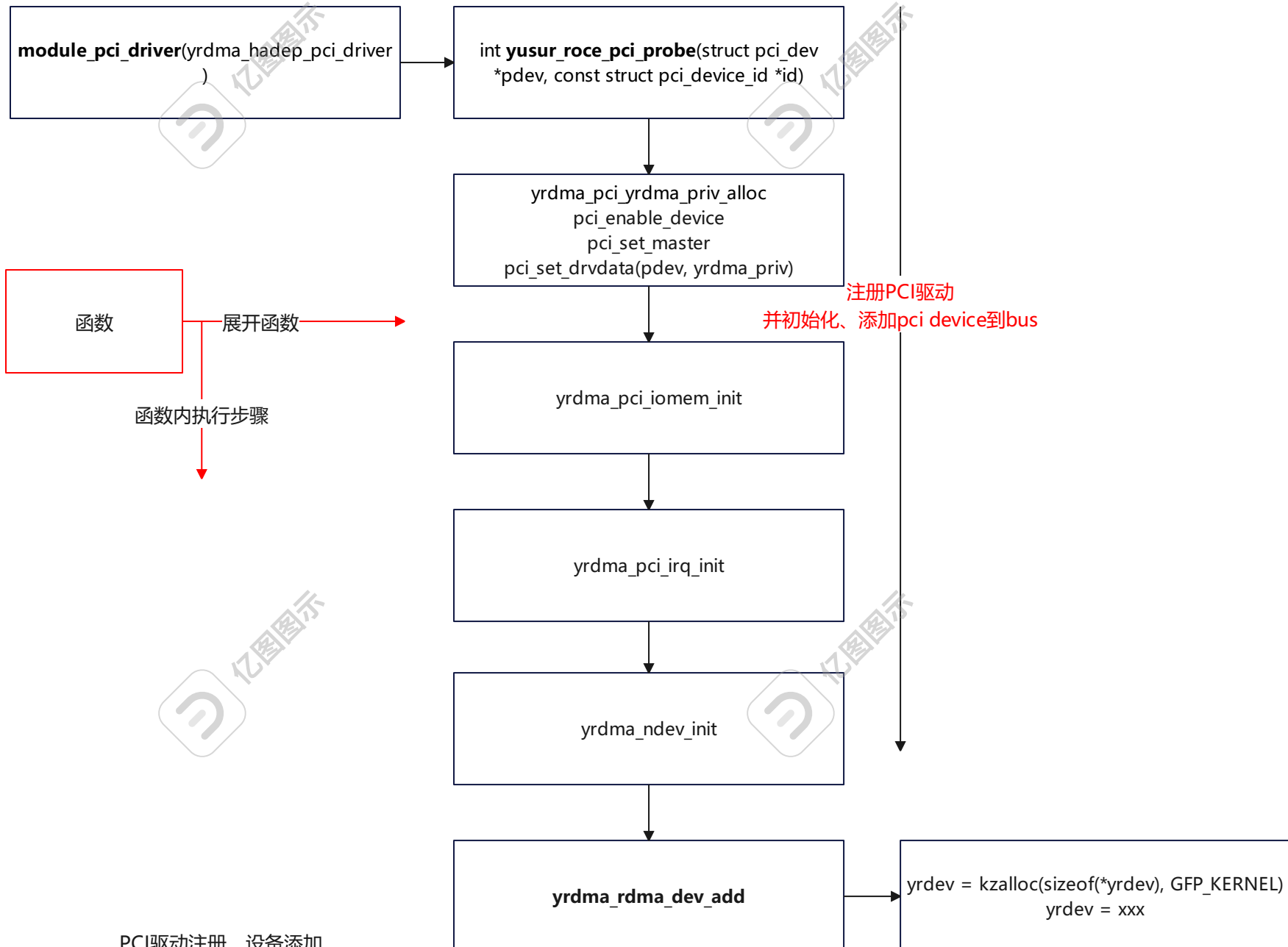


bus register



注意: 总线、驱动和设备最终都会落实为sysfs中的1个目录, 因为进一步追踪代码会发现, 它们实际上都可以认为是kobject的派生类, kobject可看作是总线、设备和驱动的抽象基类, 1个kobject对应sysfs中的1个目录。

经过仿真添加打印, 我们梳理出bus_register, bus_probe, PCIe设备添加, yrdma驱动注册的执行顺序如下:

```
[160449.318405] func_order check, yrdma_intf_init
[160402.950974] func_order check, yusur_roce_pci_probe
[160625.114939] func_order check, yusur_rdma_init
[160625.115130] func_order check, yrdma_match
[160625.115131] func_order check, yrdma_match_id
[160625.115154] func_order check, yrdma_match
[160625.115154] func_order check, yrdma_match_id
[160625.115164] func_order check, yrdma_bus_probe
[160625.115165] func_order check, yrdma_match_id
[160625.115166] func_order check, yusur_ib_probe
```

bus_register()
PCI设备添加, 此时并没有触发yrdma_bus的match操作
yrdma驱动添加
不知道为啥这里执行了两次match
bus_probe接着match执行
调用具体驱动的probe函数, 执行初始化与注册于驱动对应的设备

PCI驱动注册, 设备添加
yrdma驱动注册
PCI设备和yrdma驱动绑定bus

yrdma设备添加

