

VideoCore IV GPU Driver for rCore on Raspberry Pi 3B+

Zhenbo Sun

Bingjian Huang

2019/12/27

Project Status: **Finished!**

- 完成了学期初的预设目标：在rCore中实现Raspberry Pi 3B+的VideoCore IV GPU驱动，并且能够画出简单的2D/3D图形动画。
- Many Samples!

Performance Comparison

- 为了证明驱动的性能优势，我们对比了使用驱动和不使用驱动两种情况下framebuffer refresh frequency。

app	Frame rate
Bad apple	46
anim	625
cube	455
porygon	440

VC4 Driver

- 主要功能(io control):
 - 创建BO
 - mmap
 - 释放BO
 - 提交任务
- 验证control list
- 在写入BO时, 将offset替换成物理地址

VC4 Driver实现内容

- Bcm2837模块: V3D
- 抽象出GPU Device和GPU BO结构体, 具体实现放在VC4下
- 将GPU作为VFS使用
- 增加系统调用
 - iounmap
 - io control
- 用musl重新编译用户态部分的驱动
- 实现了3D旋转, 平移功能
- 测试样例

C2Rust Tool Analysis

- C2Rust翻译流程

```
intercept-build make
跟踪make过程, 生成compile_commands.json
...

{
    "arguments": [
        "cc",
        "-c",
        "-std=c99",
        "-o",
        "test",
        "test.c"
    ],
    "directory": "/home/huangbj16/Documents/ostrain/buffer",
    "file": "test.c"
},
...

c2rust transpile compile_commands.json
根据json完成翻译。
```

C2Rust Tool Analysis

- 值得借鉴的地方

- 翻译的完整性, 可直接运行的rust代码。
- 函数翻译满足FFI要求,
 - `#[no_mangle]`
`pub extern "C" fn rust_function() {}`
- `include".h"`引用头文件的处理

头文件中的函数声明, 变为extern C

original

```
#include "vc4_cl.h"
```

```
vc4_cl.h:
```

```
void vc4_init_cl(struct vc4_cl *cl);
```

C2Rust Translation

```
vc4_render_cl.rs
```

```
extern "C" {
```

```
    #[no_mangle]
```

```
    fn vc4_bo_create(dev: *mut device, size: size_t,  
                    type_0: vc4_kernel_bo_type) -> *mut vc4_bo;
```

```
}
```

C2Rust Tool Analysis

- 存在的问题
- (1) 大量unsafe函数和raw pointers dereference。

```
bo =  
    &mut *(*vc4).handle_bo_map.offset(handle as isize) as  
        *mut vc4_bo;  
(*bo).size = size;  
(*bo).handle = handle;  
(*bo).paddr = bus_addr;  
(*bo).vaddr = bus_addr as *mut libc::c_void;  
(*bo).type_0 = type_0;
```

```
Arc::new(Mutex::new(gpu_bo {  
    size: size,  
    handle: handle,  
    paddr: paddr,  
    vaddr: vaddr,  
    bo_type: bo_type
```

```
19 results in 4 files  
vc4_bo.rs raspi3\vc4 4  
let mut args = unsafe { &mut *(data as *mut drm_vc4_c...  
let args = unsafe { & *(data as *mut drm_vc4_f...  
let args = unsafe { &mut *(data as *mut drm_vc4_mma...  
let thread = unsafe { current_thread() };  
vc4_gem.rs raspi3\vc4 1  
let args = unsafe { &mut *(data as *mut drm_vc4_submi...  
vc4_render_cl.rs raspi3\vc4 3  
unsafe {  
unsafe {  
unsafe {  
vc4_validate.rs raspi3\vc4 11  
unsafe { *((validated + 14) as *mut u32) = ib_entry.padd...  
unsafe { *((validated as *mut u32) = exec.shader_rec_p + ...  
unsafe {  
unsafe { *((validated as *mut u32) = exec.tile_alloc_offset; }  
unsafe { *((validated + 4) as *mut u32) = vc4.bin_paddr ...  
unsafe { *((validated + 4) as *mut u32) = vc4.bin_paddr; }  
unsafe { *((dst_pkt + i) as *mut u8) = src[src_offset as us...  
unsafe { *((pkt_v + i) as *mut u8) = shader_u[offset + i]; ...  
unsafe { *((pkt_v + 4) as *mut u32) = bo_entry.paddr + f...  
unsafe { *((pkt_v + 8) as *mut u32) = bo_entry.paddr + ...  
unsafe {
```


C2Rust Tool Analysis

- 存在的问题
- (2) 宏定义翻译过于冗杂。
- C函数定义

```
/* Round up to the nearest multiple of n */  
#define ROUNDUP(a, n) ({  
    size_t __n = (size_t)(n);  
    (typeof(a))(ROUNDDOWN((size_t)(a) + __n - 1, __n));  
})
```

- C2Rust翻译

```
let mut __a: size_t =  
    (*bo).size.wrapping_add(__n).wrapping_sub(1i32 as  
    libc::c_ulonglong);  
__a.wrapping_sub(__a.wrapping_rem(__n))
```

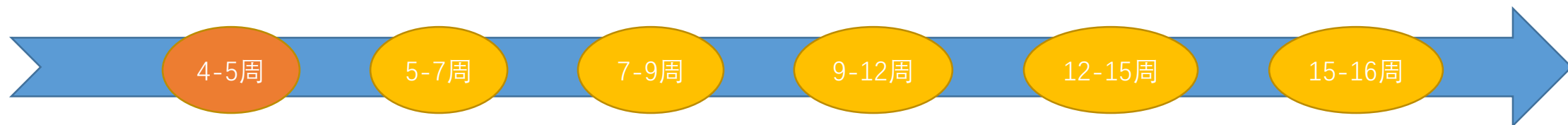
我们的翻译

```
pub fn roundUp(a:u32, n:u32) -> u32 {  
    roundDown(a + n - 1, n)  
}
```

C2Rust Tool Analysis

- (3) Enumeration翻译冗杂;
- (4) 大量使用libc::类型;
- (5) 不支持子项目翻译;
-

Project Timeline



- 确定题目，完成前期文献调研及可行性研究。

huangbj16 / rcore-vc4-opengl

Unwatch 2 Star 0 Fork 1

Code Issues 0 Pull requests 0 Actions Projects 0 Wiki Security Insights Settings

Literature Review

Bran edited this page on 29 Sep · 8 revisions

前期调研

根据我们的调研，前期工作分为三部分。

rcore在Raspberry Pi上的移植 (by 贾越凯)

平台: Raspberry Pi 3

进展:

在Raspberry Pi 3上实现了Framebuffer以及在此基础上的console，可以通过修改Framebuffer的方式在屏幕上画图。

ucore-plus在Raspberry Pi上对VideoCore IV GPU和OpenGL的支持 (by 贾越凯)

平台: Raspberry Pi Zero

成果:

使用Rust实现了VideoCore IV GPU的驱动程序（在内核态和用户态下），补充了需要的syscall；完成了一部分OpenGL的接口（另一部分使用CPU实现尚未合并），可以在用户态调用OpenGL的API画3D图。

1. Raspberry Pi Zero V1.2，片上系统为 Broadcom BCM2835，包含一块 ARM1176JZF-S CPU 和 VideoCore IV GPU(VC4)，包含两个 Micro USB 接口(电源和数据)，一个 Mini HDMI 接口，可以使用串口进行通信

2. Mailbox: ARM CPU和VC4 GPU之间的信息通道，获得framebuffer大小、基址，请求、释放显存

3. framebuffer: 包括地址映射，系统调用等

Pages 3

Find a Page...

Home

Experiment Design

Important Things Logging

Literature Review

Presentation One Review

Presentation Two

vc4 kernel transplant log

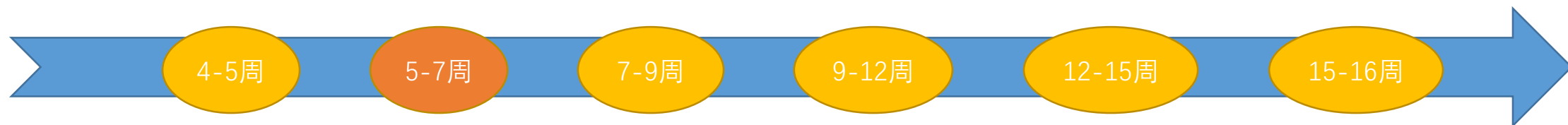
Work Distribution

+ Add a custom sidebar

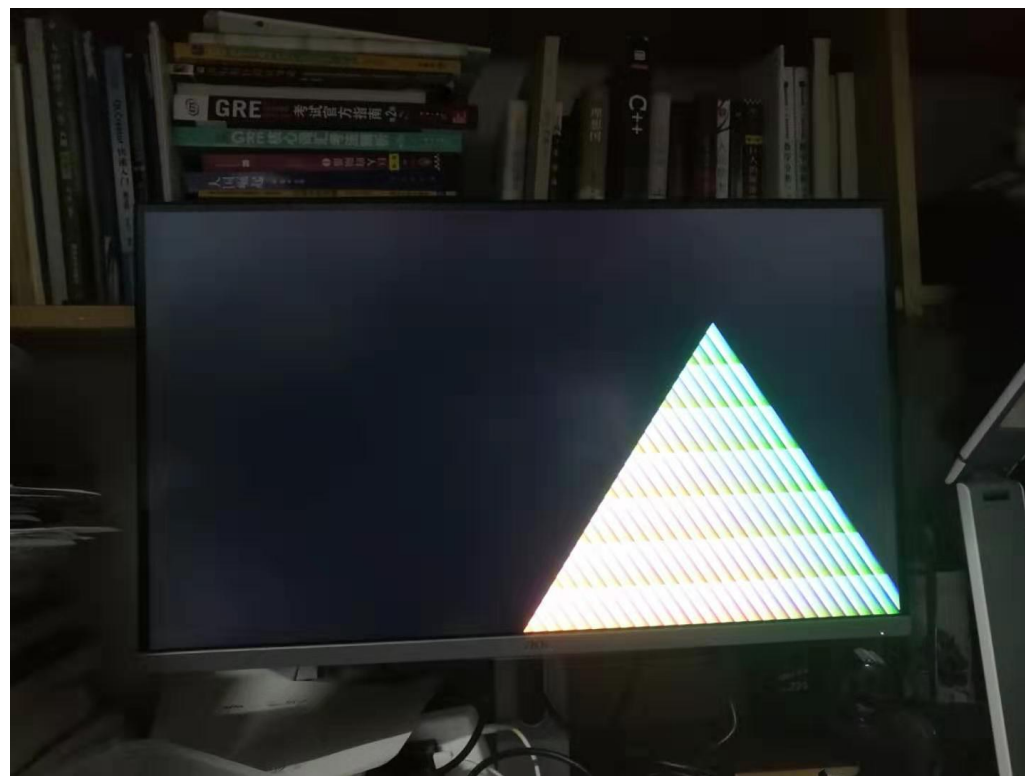
Clone this wiki locally

<https://github.com/huangbj16>

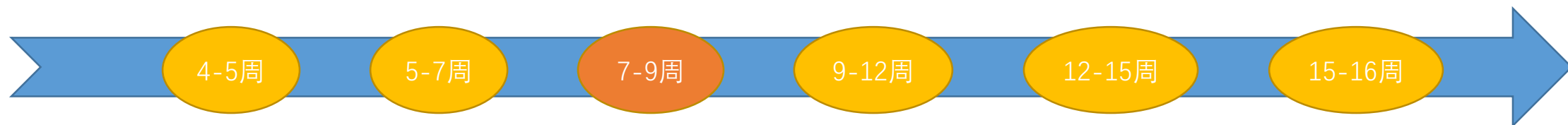
Project Timeline



- 用Rust重写了最简单的Hackdriver，放到内核模块中，虽然有大量unsafe语句，但是成功调用framebuffer实现了在真机上画三角形。

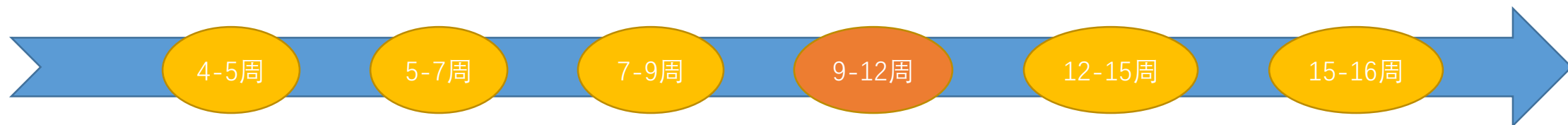


Project Timeline



- 第一阶段移植，阅读 ucore-plus 中的驱动代码，并进行翻译。

Project Timeline

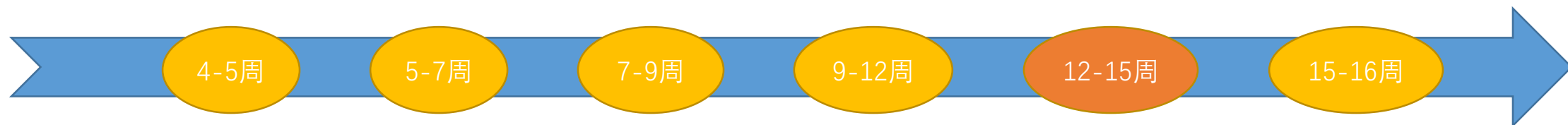


- 忙于申请……

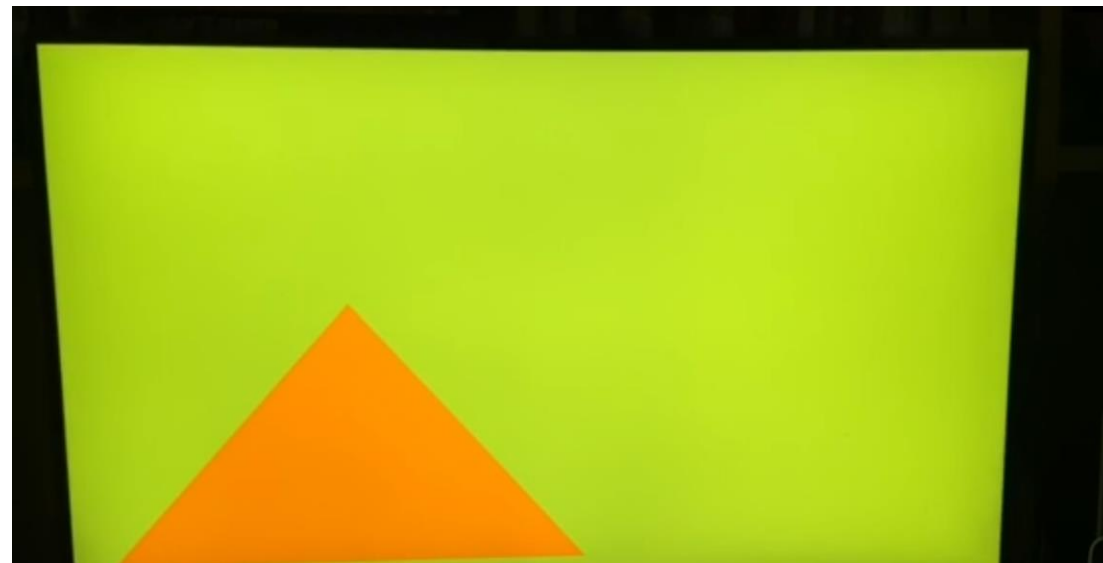


感到蕉绿

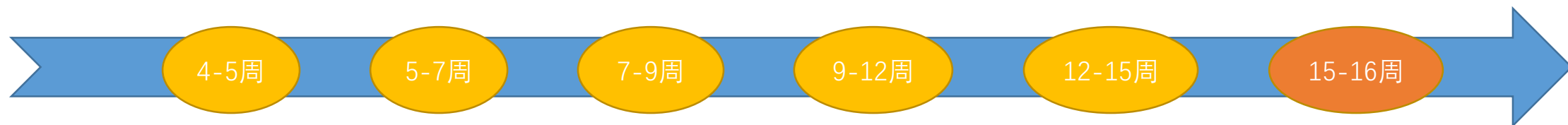
Project Timeline



- 第二阶段移植，生成调用关系图，理解驱动工作方式，根据rust的语言特性和面向对象方法对驱动代码进行重写。
- 使用C2Rust工具直接翻译ucore的驱动代码，对比分析。



Project Timeline



- 修复一些小bug
- 补充了OpenGL API实现。
- 完善功能
- 增加测例
- 编写文档

Project Contribution

- 孙桢波
 - 实现HackDriver
 - 移植内核部分的VC4驱动
 - 移植用户态部分的驱动
 - 补充系统调用
 - 真机调试
 - 补充OpenGL的3D旋转、平移
- 黄冰鉴
 - 翻译内核部分的VC4驱动
 - C2Rust工具分析
 - 调查用户态所需系统调用
 - 真机调试
 - 编写新的测试样例

Future Work

- 完善OpenGL ES库
 - 增加对GLSL的支持
 - 完善其余功能，如光照，纹理
- 移植miniGUI并适配当前的OpenGL