



从零开始

DEVELOPING YOUR OWN

自操作系统

OPERATING SYSTEM FROM SCRATCH

开始编码!

Hello kernel world!

OPERATING SYSTEM DEVELOPMENT

TUTORIAL SERIES

EP 5-5



真正的 Hello World——VGA 文本模式 ^[1]

允许我们直接往显存里写我们需要打印的字符

文本缓冲区（显存）起始点： 0xB8000 （注意，物理地址！）

缓冲区大小： $2 \times W \times H$ 字节 (W = 屏幕宽度， H = 高度，以字符为单位)

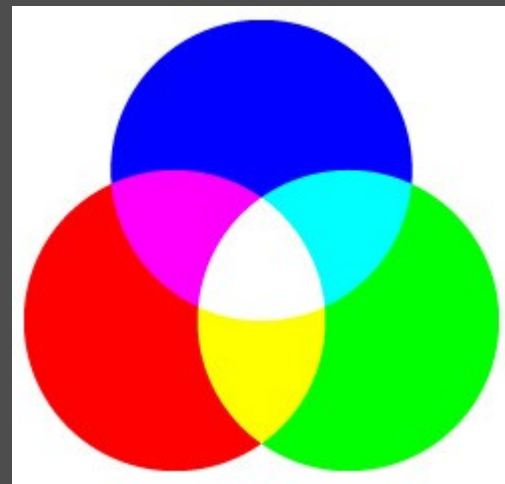
早期的 DOS 系统就是用这个模式的

简单易操作， Hello world 之理想实现



真正的 Hello World——VGA 文本模式 [1]

文本缓冲区的最小单元：用一个字表示显示的字符以及前景背景色



所以，在文本模式下，可选用颜色共有 16 种

CGA 16 色调色板

0 — black	8 — high gray
1 — low blue	9 — high blue
2 — low green	10 — high green
3 — low cyan	11 — high cyan
4 — low red	12 — high red
5 — low magenta	13 — high magenta
6 — low yellow (brown)	14 — high yellow
7 — low gray	15 — white

L	R	G	B
---	---	---	---

? 0 0 0

? 0 0 1

? 0 1 0

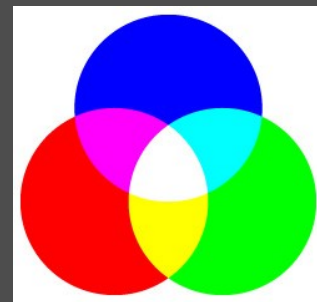
? 0 1 1

? 1 0 0

? 1 0 1

? 1 1 0

? 1 1 1



图片选自：

https://en.wikipedia.org/wiki/List_of_8-bit_computer_hardware_graphics#CGA

Let's code!



新房装修——链接器的配置

可以看到，我们在汇编文件里调用很多 C 里的东西。

在 C 里面也引用了来自别的 C 文件的函数

当全部编译好后，我们需要将这些占位符换成具体的地址。

→ 需要链接器

为了保证我们的 multiboot 是在最开头（至少前 8KB 处）

→ 需要配置链接器行为（Linker Script）

Let's code!



References:

[1] IBM. VGA Technical Reference Manual, DRAFT (1992). pp. 2.12-2.17