Vga\_kb\_font模块

# 测试数据

## 连续输入——31 31 31 => 存入3个31

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1;

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1;

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //开始位\_ ascii\_1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit2

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit3

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit4

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit5

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit6

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit7

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //校验位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //停止位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //开始位\_ ascii\_1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit2

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit3

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit4

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit5

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit6

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit7

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //校验位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //停止位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //开始位\_ ascii\_1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit2

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit3

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit4

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit5

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit6

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit7

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //校验位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //停止位

#2; ps2\_clk=0;

## 释放按键测试数据——31 F0 31 => 31存入ram

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1;

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1;

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //开始位\_ ascii\_1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit2

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit3

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit4

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit5

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit6

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit7

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //校验位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //停止位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //开始位\_kb\_f0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit2

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit3

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit4

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit5

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit6

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit7

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //校验位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //停止位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //开始位\_ascii\_31

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit2

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit3

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit4

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit5

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit6

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit7

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //校验位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //停止位

#2; ps2\_clk=0;

## Shift按键——12 16 f0 16 f0 12-> 4b的shift按键译码

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1;

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1;

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //开始位\_ kb\_12

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit2

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit3

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit4

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit5

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit6

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit7

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //校验位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //停止位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //开始位\_ ascii\_1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit2

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit3

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit4

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit5

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit6

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit7

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //校验位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //停止位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //开始位\_kb\_f0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit2

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit3

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit4

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit5

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit6

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit7

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //校验位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //停止位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //开始位\_ascii\_31

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit2

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit3

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit4

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit5

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit6

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit7

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //校验位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //停止位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //开始位\_kb\_f0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit2

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit3

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit4

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit5

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit6

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit7

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //校验位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //停止位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //开始位\_ kb\_12

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit2

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit3

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit4

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit5

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit6

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit7

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //校验位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //停止位

#2; ps2\_clk=0;

## E0扩展码测试——E0 72 E0 F0 72

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1;

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1;

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //开始位\_ kb\_E0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit2

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit3

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit4

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit5

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit6

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit7

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //校验位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //停止位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //开始位\_ ascii\_DOWN

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit2

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit3

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit4

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit5

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit6

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit7

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //校验位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //停止位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //开始位\_ kb\_E0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit2

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit3

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit4

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit5

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit6

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit7

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //校验位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //停止位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //开始位\_kb\_f0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit2

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit3

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit4

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit5

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit6

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit7

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //校验位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //停止位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //开始位\_ ascii\_DOWN

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit0

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit1

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit2

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit3

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit4

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit5

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //bit6

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=0; //bit7

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //校验位

#2; ps2\_clk=0;

#2; ps2\_clk=1; ps2\_data=1; //停止位

#2; ps2\_clk=0;

6．连续shift然后再按键