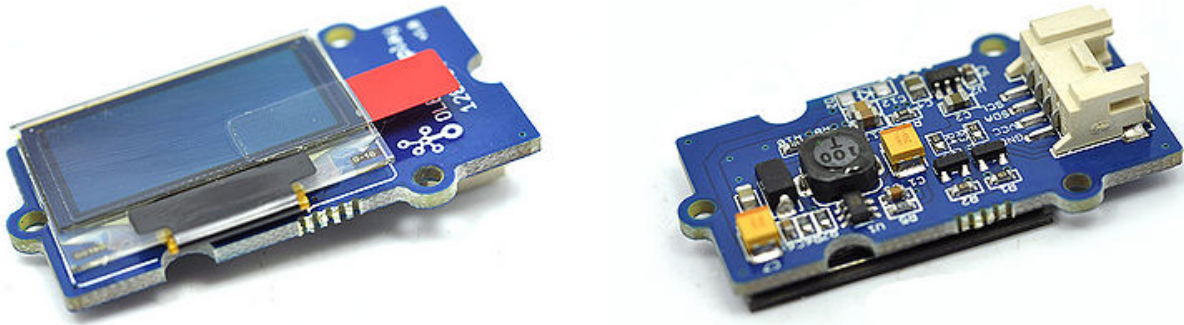


Grove - OLED Display 0.96 inch SKU: 104030008

tags: grove_i2c, io_3v3, io_5v, plat_duino, plat_bbg, plat_pi, plat_wio, plat_linkit



Grove - OLED Display 0.96" 模块是一款带有 Grove I2C 接口的 OLED 单色 128×64 点矩阵显示模块。与 LCD 相比，OLED 屏幕更具竞争力。它具有高亮度，可发光，高对比度，外形小巧，视角广，温度范围宽，功耗低等特点。同时它的屏幕比较大，可以显示比 OLED 96×96 更多的内容。

产品特性

- Grove 接口
- 通行协议: I2C
- 低功耗
- 显示颜色: 白
- 工作温度: -20°C~70°C

!!!warning 请注意: OLED 承受过大压力会导致屏幕碎裂。

!!!Tip 关于 Grove 模块的更多信息请点击 [Grove System](#)

规格参数

项目	最小值	参数	最大值	单位
电源电压 (VCC)	3.3	5.0	5.5	V
驱动 IC	-	SSD1308Z	-	-
显示颜色	-	白	-	-
点矩阵	-	128×64	-	-
尺寸	-	26.7(W)×19.26(H)	-	mm

项目	最小值	参数	最大值	单位
显示区域	-	21.74(W)×11.175 (H)	-	mm
点距	-	0.17(W)×0.175 (H)	-	mm
点大小	-	0.15(W)×0.15 (H)	-	mm
工作温度	-	-20~70	-	°C

Platforms Supported

入门指导

与 Arduino 一起使用

连接

OLED128*64 使用 SSD1308 芯片的所有引脚，默认原点位于左上角。您可以通过更改程序来更改原始点，以显示所需的图案。更多详情，请参阅 [SSD1308_1.0.pdf](#) 和 [LY190-128064.pdf](#)。

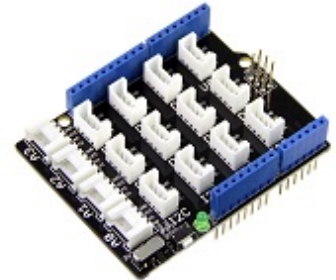
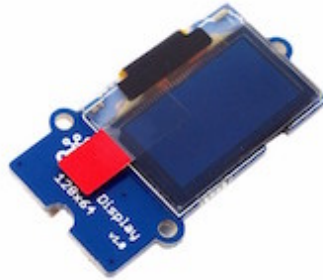
这里我们演示如何在屏幕上显示 "Hello World" 。

首先请准备以下器材：

Seeeduino V4

Grove - OLED Display 0.96inch

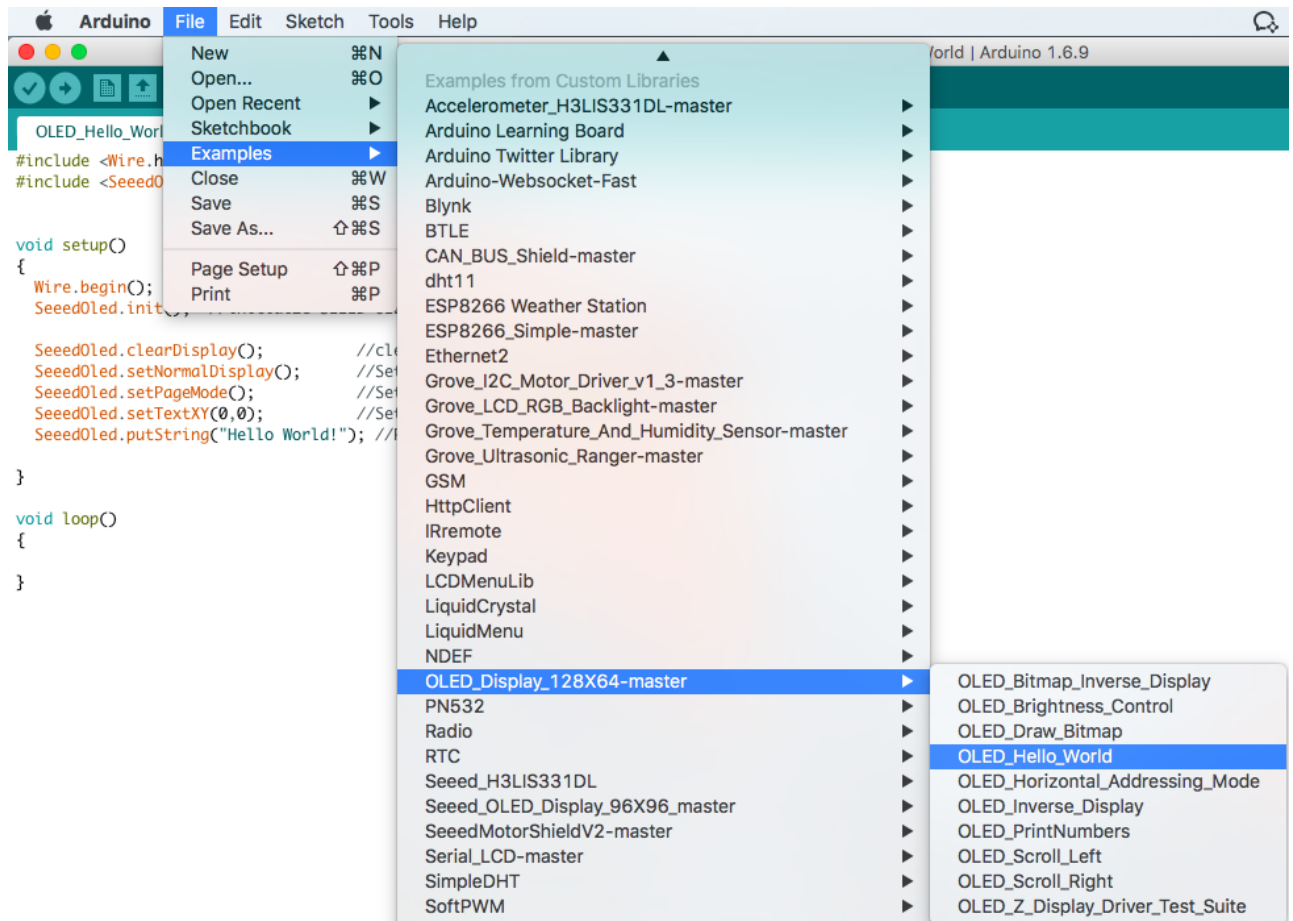
Base Shield



- 将 Grove - OLED Display 128*64 插入 Grove - Base Shield 上的 I2C 端口，然后将 Grove - Base Shield 堆叠在 Seeeduino 上；

软件

- 下载 [Seeed OLED Display 128*64 库文件](#)。
- 请按照 [这里](#) 安装库。
- 通过路径：**File(文件) -> Example(示例) -> OLED_Display_128X64-master->OLED_Hello_World** 打开代码。



```

#include <Wire.h>
#include <SeeedOLED.h>

void setup()
{
  Wire.begin();
  SeeedOled.init(); //initialize SEEED OLED display

  SeeedOled.clearDisplay(); //clear the screen and set start position to
top left corner
  SeeedOled.setNormalDisplay(); //Set display to normal mode (i.e non-inverse
mode)
  SeeedOled.setPageMode(); //Set addressing mode to Page Mode
  SeeedOled.setTextXY(0,0); //Set the cursor to Xth Page, Yth Column
  SeeedOled.putString("Hello World!"); //Print the String
}

void loop()
{
}

```

- 上传代码

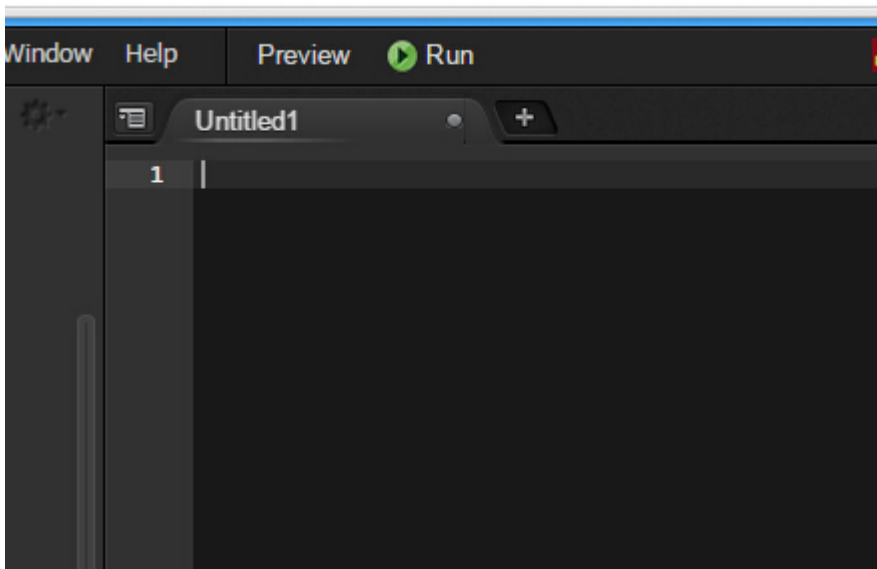
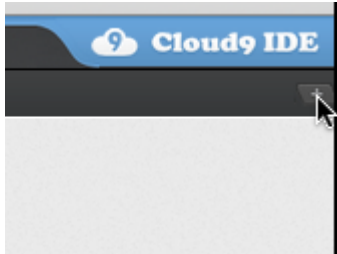
- 屏幕上将显示 "hello world"。

与 Beaglebone Green 一起使用

要编辑 BBG 上的程序，您可以使用 [Cloud9 IDE](#) 并参阅 [Beaglebone Green Wiki](#)。

下面是在 OLED 屏幕上显示 "Hello World" 的步骤：

步骤 1: 点击顶部右端的 "+" 创建新文件。



步骤 2: 复制并粘贴以下代码至 Cloud9 IDE 窗口。

```
from Adafruit_I2C import Adafruit_I2C
import time
import math

Oled = Adafruit_I2C(0x3c)
Command_Mode=0x80
Data_mode=0x40

grayH= 0xF0
grayL= 0x0F
Normal_Display_Cmd=0xA4

BasicFont = [[0 for x in xrange(8)] for x in xrange(10)]
BasicFont=[[0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00],
```

```

[0x00,0x00,0x5F,0x00,0x00,0x00,0x00,0x00],
[0x00,0x00,0x07,0x00,0x07,0x00,0x00,0x00],
[0x00,0x14,0x7F,0x14,0x7F,0x14,0x00,0x00],
[0x00,0x24,0x2A,0x7F,0x2A,0x12,0x00,0x00],
[0x00,0x23,0x13,0x08,0x64,0x62,0x00,0x00],
[0x00,0x36,0x49,0x55,0x22,0x50,0x00,0x00],
[0x00,0x00,0x05,0x03,0x00,0x00,0x00,0x00],
[0x00,0x1C,0x22,0x41,0x00,0x00,0x00,0x00],
[0x00,0x41,0x22,0x1C,0x00,0x00,0x00,0x00],
[0x00,0x08,0x2A,0x1C,0x2A,0x08,0x00,0x00],
[0x00,0x08,0x08,0x3E,0x08,0x08,0x00,0x00],
[0x00,0xA0,0x60,0x00,0x00,0x00,0x00,0x00],
[0x00,0x08,0x08,0x08,0x08,0x08,0x00,0x00],
[0x00,0x60,0x60,0x00,0x00,0x00,0x00,0x00],
[0x00,0x20,0x10,0x08,0x04,0x02,0x00,0x00],
[0x00,0x3E,0x51,0x49,0x45,0x3E,0x00,0x00],
[0x00,0x00,0x42,0x7F,0x40,0x00,0x00,0x00],
[0x00,0x62,0x51,0x49,0x49,0x46,0x00,0x00],
[0x00,0x22,0x41,0x49,0x49,0x36,0x00,0x00],
[0x00,0x18,0x14,0x12,0x7F,0x10,0x00,0x00],
[0x00,0x27,0x45,0x45,0x45,0x39,0x00,0x00],
[0x00,0x3C,0x4A,0x49,0x49,0x30,0x00,0x00],
[0x00,0x01,0x71,0x09,0x05,0x03,0x00,0x00],
[0x00,0x36,0x49,0x49,0x49,0x36,0x00,0x00],
[0x00,0x06,0x49,0x49,0x29,0x1E,0x00,0x00],
[0x00,0x00,0x36,0x36,0x00,0x00,0x00,0x00],
[0x00,0x00,0xAC,0x6C,0x00,0x00,0x00,0x00],
[0x00,0x08,0x14,0x22,0x41,0x00,0x00,0x00],
[0x00,0x14,0x14,0x14,0x14,0x14,0x00,0x00],
[0x00,0x41,0x22,0x14,0x08,0x00,0x00,0x00],
[0x00,0x02,0x01,0x51,0x09,0x06,0x00,0x00],
[0x00,0x32,0x49,0x79,0x41,0x3E,0x00,0x00],
[0x00,0x7E,0x09,0x09,0x09,0x7E,0x00,0x00],
[0x00,0x7F,0x49,0x49,0x49,0x36,0x00,0x00],
[0x00,0x3E,0x41,0x41,0x41,0x22,0x00,0x00],
[0x00,0x7F,0x41,0x41,0x22,0x1C,0x00,0x00],
[0x00,0x7F,0x49,0x49,0x49,0x41,0x00,0x00],
[0x00,0x7F,0x09,0x09,0x09,0x01,0x00,0x00],
[0x00,0x3E,0x41,0x41,0x51,0x72,0x00,0x00],
[0x00,0x7F,0x08,0x08,0x08,0x7F,0x00,0x00],
[0x00,0x41,0x7F,0x41,0x00,0x00,0x00,0x00],
[0x00,0x20,0x40,0x41,0x3F,0x01,0x00,0x00],
[0x00,0x7F,0x08,0x14,0x22,0x41,0x00,0x00],
[0x00,0x7F,0x40,0x40,0x40,0x40,0x00,0x00],
[0x00,0x7F,0x02,0x0C,0x02,0x7F,0x00,0x00],
[0x00,0x7F,0x04,0x08,0x10,0x7F,0x00,0x00],
[0x00,0x3E,0x41,0x41,0x41,0x3E,0x00,0x00],
[0x00,0x7F,0x09,0x09,0x09,0x06,0x00,0x00],
[0x00,0x3E,0x41,0x51,0x21,0x5E,0x00,0x00],
[0x00,0x7F,0x09,0x19,0x29,0x46,0x00,0x00],
[0x00,0x26,0x49,0x49,0x49,0x32,0x00,0x00],
[0x00,0x01,0x01,0x7F,0x01,0x01,0x00,0x00],
[0x00,0x3F,0x40,0x40,0x40,0x3F,0x00,0x00],
[0x00,0x1F,0x20,0x40,0x20,0x1F,0x00,0x00],

```

```
[0x00, 0x3F, 0x40, 0x38, 0x40, 0x3F, 0x00, 0x00],
[0x00, 0x63, 0x14, 0x08, 0x14, 0x63, 0x00, 0x00],
[0x00, 0x03, 0x04, 0x78, 0x04, 0x03, 0x00, 0x00],
[0x00, 0x61, 0x51, 0x49, 0x45, 0x43, 0x00, 0x00],
[0x00, 0x7F, 0x41, 0x41, 0x00, 0x00, 0x00, 0x00],
[0x00, 0x02, 0x04, 0x08, 0x10, 0x20, 0x00, 0x00],
[0x00, 0x41, 0x41, 0x7F, 0x00, 0x00, 0x00, 0x00],
[0x00, 0x04, 0x02, 0x01, 0x02, 0x04, 0x00, 0x00],
[0x00, 0x80, 0x80, 0x80, 0x80, 0x80, 0x00, 0x00],
[0x00, 0x01, 0x02, 0x04, 0x00, 0x00, 0x00, 0x00],
[0x00, 0x20, 0x54, 0x54, 0x54, 0x78, 0x00, 0x00],
[0x00, 0x7F, 0x48, 0x44, 0x44, 0x38, 0x00, 0x00],
[0x00, 0x38, 0x44, 0x44, 0x28, 0x00, 0x00, 0x00],
[0x00, 0x38, 0x44, 0x44, 0x48, 0x7F, 0x00, 0x00],
[0x00, 0x38, 0x54, 0x54, 0x54, 0x18, 0x00, 0x00],
[0x00, 0x08, 0x7E, 0x09, 0x02, 0x00, 0x00, 0x00],
[0x00, 0x18, 0xA4, 0xA4, 0xA4, 0x7C, 0x00, 0x00],
[0x00, 0x7F, 0x08, 0x04, 0x04, 0x78, 0x00, 0x00],
[0x00, 0x00, 0x7D, 0x00, 0x00, 0x00, 0x00, 0x00],
[0x00, 0x80, 0x84, 0x7D, 0x00, 0x00, 0x00, 0x00],
[0x00, 0x7F, 0x10, 0x28, 0x44, 0x00, 0x00, 0x00],
[0x00, 0x41, 0x7F, 0x40, 0x00, 0x00, 0x00, 0x00],
[0x00, 0x7C, 0x04, 0x18, 0x04, 0x78, 0x00, 0x00],
[0x00, 0x7C, 0x08, 0x04, 0x7C, 0x00, 0x00, 0x00],
[0x00, 0x38, 0x44, 0x44, 0x38, 0x00, 0x00, 0x00],
[0x00, 0xFC, 0x24, 0x24, 0x18, 0x00, 0x00, 0x00],
[0x00, 0x18, 0x24, 0x24, 0xFC, 0x00, 0x00, 0x00],
[0x00, 0x00, 0x7C, 0x08, 0x04, 0x00, 0x00, 0x00],
[0x00, 0x48, 0x54, 0x54, 0x24, 0x00, 0x00, 0x00],
[0x00, 0x04, 0x7F, 0x44, 0x00, 0x00, 0x00, 0x00],
[0x00, 0x3C, 0x40, 0x40, 0x7C, 0x00, 0x00, 0x00],
[0x00, 0x1C, 0x20, 0x40, 0x20, 0x1C, 0x00, 0x00],
[0x00, 0x3C, 0x40, 0x30, 0x40, 0x3C, 0x00, 0x00],
[0x00, 0x44, 0x28, 0x10, 0x28, 0x44, 0x00, 0x00],
[0x00, 0x1C, 0xA0, 0xA0, 0x7C, 0x00, 0x00, 0x00],
[0x00, 0x44, 0x64, 0x54, 0x4C, 0x44, 0x00, 0x00],
[0x00, 0x08, 0x36, 0x41, 0x00, 0x00, 0x00, 0x00],
[0x00, 0x00, 0x7F, 0x00, 0x00, 0x00, 0x00, 0x00],
[0x00, 0x41, 0x36, 0x08, 0x00, 0x00, 0x00, 0x00],
[0x00, 0x02, 0x01, 0x01, 0x02, 0x01, 0x00, 0x00],
[0x00, 0x02, 0x05, 0x05, 0x02, 0x00, 0x00, 0x00]]
```

```
def oled_init():
```

```
    sendCommand(0xFD) # Unlock OLED driver IC MCU interface from entering command.
```

```
    i.e: Accept commands
```

```
    sendCommand(0x12)
```

```
    sendCommand(0xAE) # Set display off
```

```
    sendCommand(0xA8) # set multiplex ratio
```

```
    sendCommand(0x5F) # 96
```

```
    sendCommand(0xA1) # set display start line
```

```
    sendCommand(0x00)
```

```
    sendCommand(0xA2) # set display offset
```

```
    sendCommand(0x60)
```

```
    sendCommand(0xA0) # set remap
```

```

sendCommand(0x46)
sendCommand(0xAB) # set vdd internal
sendCommand(0x01)
sendCommand(0x81) # set contrast
sendCommand(0x53) # 100 nit
sendCommand(0xB1) # Set Phase Length
sendCommand(0X51)
sendCommand(0xB3) # Set Display Clock Divide Ratio/Oscillator Frequency
sendCommand(0x01)
sendCommand(0xB9)
sendCommand(0xBC) # set pre_charge voltage/VCOMH
sendCommand(0x08) # (0x08);
sendCommand(0xBE) # set VCOMH
sendCommand(0X07) # (0x07);
sendCommand(0xB6) # Set second pre-charge period
sendCommand(0x01)
sendCommand(0xD5) # enable second precharge and external vs1
sendCommand(0X62) # (0x62);
sendCommand(0xA4) # Set Normal Display Mode
sendCommand(0x2E) # Deactivate Scroll
sendCommand(0xAF) # Switch on display
time.sleep(0.1)
# delay(100);

# Row Address
sendCommand(0x75) # Set Row Address
sendCommand(0x00) # Start 0
sendCommand(0x5f) # End 95

# Column Address
sendCommand(0x15) # Set Column Address
sendCommand(0x08) # Start from 8th Column of driver IC. This is 0th Column
for OLED
    sendCommand(0x37) # End at (8 + 47)th column. Each Column has 2
pixels(segments)

# Init gray level for text. Default:Brightest White
grayH= 0xF0
grayL= 0x0F

def sendCommand(byte):
    Oled.write8(Command_Mode,byte)

def sendData(byte):
    Oled.write8(Data_mode,byte)

def multi_comm(commands):
    for c in commands:
        sendCommand(c)

def oled_clearDisplay():
    for j in range (0,48):
        for i in range (0,96):

```

```

        sendData(0x00)

def oled_setNormalDisplay():
    sendCommand(Normal_Display_Cmd)

def oled_setVerticalMode():
    sendCommand(0xA0)    # remap to
    sendCommand(0x46)    # Vertical mode

def oled_setTextXY(Row,Column):
    sendCommand(0x15)    # Set Column Address
    sendCommand(0x08+(Column*4)) # Start Column: Start from 8
    sendCommand(0x37)    # End Column
    # Row Address
    sendCommand(0x75)    # Set Row Address
    sendCommand(0x00+(Row*8)) # Start Row
    sendCommand(0x07+(Row*8)) # End Row

def oled_putChar(C):
    C_add=ord(C)
    if C_add<32 or C_add>127:    # Ignore non-printable ASCII characters
        C=' '
        C_add=ord(C)

    for i in range(0,8,2):
        for j in range(0,8):
            c=0x00
            bit1=((BasicFont[C_add-32][i])>>j)&0x01
            bit2=((BasicFont[C_add-32][i+1])>>j)&0x01
            if bit1:
                c=c|grayH
            else:
                c=c|0x00
            if bit2:
                c=c|grayL
            else:
                c=c|0x00
            sendData(c)

def oled_putString(String):
    for i in range(len(String)):
        oled_putChar(String[i])

if __name__=="__main__":
    oled_init()
    oled_setNormalDisplay()
    oled_setTextXY(0,0)
    oled_putString("Hello")
    time.sleep(10)
    #Oled.write8(Command_Mode,0xFD)
    #sendCommand(0xFD)
    print 'hello world'

```

步骤 3: 点击带有 .py 扩展名的磁盘图标保存文件。

步骤 4: 把 Grove - OLED 连接到 BBG 上的 Grove I2C 接口。

步骤 5: 运行代码。Grove - OLED 将显示 "Hello World"。

资源下载

- **[Eagle 文件]** [Grove-OLED128x64](#)
- **[原理图 PDF]** [Grove-OLED128x64 Schematic](#)
- **[PCB 图 PDF]** [Grove-OLED128x64 PCB](#)
- **[库文件]** [GitHub Library for OLED](#)
- **[芯片数据手册]** [Resources of SSD1308_1.0.pdf](#)
- **[芯片数据手册]** [Resources of LY190-128064.pdf](#)
- **[其他资源]** [Beaglebone Green Wiki](#)