

无线电

合订本
63周年版·下

科普 创新
实作 分享



智能控制电子设计手册

2018年 第7期~第12期

《无线电》编辑部 编

www.radio.com.cn

Since 1955



 中国工信出版集团

 人民邮电出版社
POSTS & TELECOM PRESS

无线电

合订本
63周年版·下

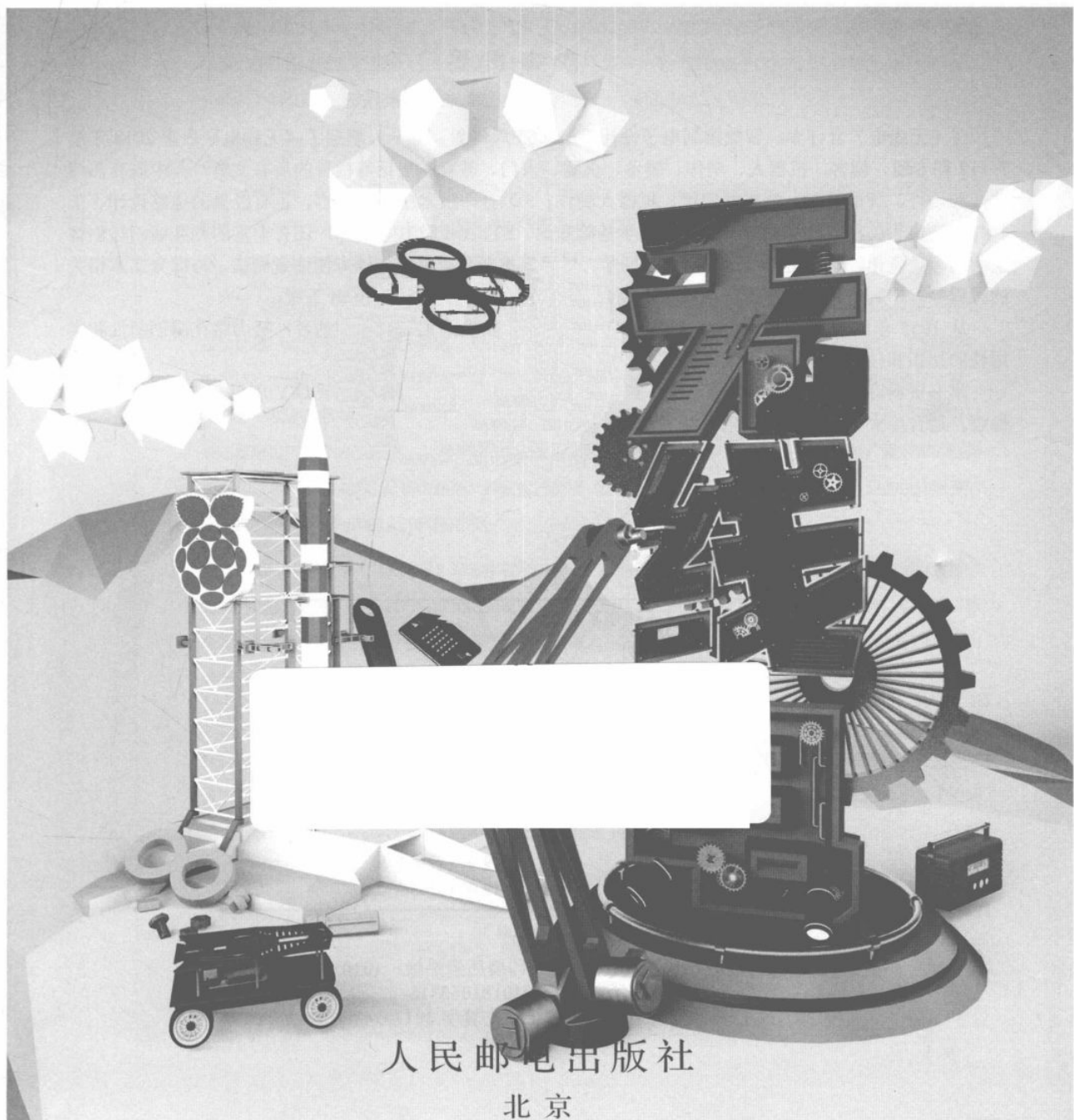
智能控制电子设计手册

2018年 第7期~第12期

《无线电》编辑部 编

www.radio.com.cn

Since 1955



人民邮电出版社
北京

图书在版编目(CIP)数据

《无线电》合订本：63周年版. 智能控制电子设计手册. 下 / 《无线电》编辑部编. — 北京：人民邮电出版社，2019.3

ISBN 978-7-115-50724-2

I. ①无… II. ①无… III. ①无线电技术—丛刊②智能控制—手册③电子电路—电路设计—手册 IV. ①TN014-55

中国版本图书馆CIP数据核字(2019)第028028号

内 容 提 要

《〈无线电〉合订本：智能控制电子设计手册（63周年版·下）》囊括了《无线电》杂志2018年第7~12期专题、创客、机器人、制作、装备、火腿、入门、教育、史话等栏目的所有文章，其中既有热门的开源硬件、智能控制、物联网应用、机器人制作、3D打印技术应用等内容，也有经典的电路设计、工具与仪器仪表使用方法和面向初学者的电学基础知识、STEAM教育等内容，还有丰富的创客活动与创客空间的相关资讯。这些文章经过整理，按期号、栏目等重新分类编排，以方便读者阅读。与部分文章相关的源程序、印制电路板图等资料请到《无线电》杂志网站 www.radio.com.cn 下载。

从1955年创刊起，《无线电》杂志一直是电子领域发展的见证者和推动者，努力将充满创新性和实用性的知识和技能传递给广大读者。

本书资料容量庞大，内容丰富，500多页内容中包含100多个创客与电子DIY项目，文章精练，实用性强，适合广大电子爱好者、电子技术人员、创客及相关专业师生阅读。

-
- ◆ 编 《无线电》编辑部
责任编辑 周 明
责任印制 彭志环
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京盛通印刷股份有限公司印刷
 - ◆ 开本：787×1092 1/16
印张：31.5 2019年3月第1版
字数：1129千字 2019年3月北京第1次印刷
-

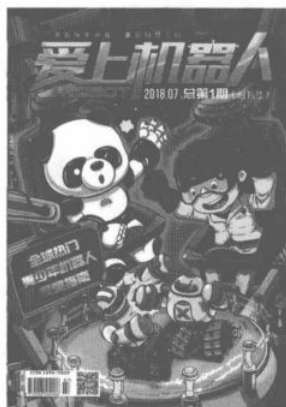
定价：99.00元

读者服务热线：(010)81055339 印装质量热线：(010)81055316

反盗版热线：(010)81055315

广告经营许可证：京东工商广登字20170147号

不忘初心，砥砺前行



最近有位新作者对我们说：“看完十几本《无线电》杂志，发现里面有不少创客的制作项目，对于我这样喜欢挖掘和尝试新东西的人，一天看10本都不腻！”或许，不少喜欢电子DIY的新读者都有同样的感觉，光从名字很难得知《无线电》杂志的真正内容，因此有种“相见恨晚”的感觉。

那么，是《无线电》杂志“名不副实”吗？是，也不是。

从表面看，创客、电子制作的内容确实和“无线电”三个字给人带来的无线电波通信的第一印象不太相符，但实在《无线电》杂志创刊至今的63年历史中，我们始终没有忘却初心，那就是封面上总结的“科普、创新、实作、分享”。

在《无线电》杂志刚刚诞生的那个年代，以收发无线电波为代表的电子通信技术，可以算是最先进的科技，地位一点也不比如今的大数据、云计算、机器学习、深度学习等人工智能技术差。在那时，《无线电》杂志就在传播前沿的电子信息科普知识，同时将种种创新的电子装置，以实际制作项目的形式分享给大家。很多老一辈读者都有看杂志学习组装收音机乃至电视机的经历。可以说，在没有互联网的时代，《无线电》杂志便承担了如今的创客空间的职能，一方面传授动手制作的必备知识与经验，另一方面推广爱好者的创新发明与改良。而在杂志上分享电路图、解读设计原理，也和如今的开源硬件理念不谋而合，目的是让更多的人可以充分学习、利用前人的科技成果。

随着时代的变迁，《无线电》杂志也在与时俱进，不断跟踪新技术、新热点、新趋势，关注爱好者与学习者的需求，积极响应国家方针政策的新要求，在杂志内容、栏目设置和表现形式上不断突破，这才有了如今大家看到的创客项目、电子制作、智能硬件、机器人、3D打印、物联网、音响DIY等内容。我们介绍的内容虽然变化了，但不变的是坚守多年的办刊理念，是对电子和信息技术的普及与推广。

2017年7月，国务院印发了《新一代人工智能发展规划》，部署构筑我国人工智能发展的先发优势，加快建设创新型国家和世界科技强国。这又给我们指明了新的方向，今后，读者们应该会在杂志上看到更多与人工智能相关的应用。事实上，我们以前涉及的图像识别、语音识别、神经网络等内容已经属于这个范畴了。随着技术的不断进步，普通人运用这些先进技术的门槛会越来越低。

2018年，《无线电》杂志参与组织了青少年编程与人工智能应用方面的赛事，同时杂志的青少版《爱上机器人》也与读者见面了。这是一本专门面向青少年的科普杂志，介绍机器人技术，普及关于机器人的科学知识，分享硬件制作和软件编程技巧，同时用科幻作品中的机器人为青少年插上想象的翅膀，希冀他们飞得更高、更远。

《无线电》杂志将继续保持初心，砥砺前行，也请读者们与我们一起前行。



合订本：智能控制电子设计手册
(63周年版·下)

目 录

创客 | MAKER

- ESP32音乐盒 周传林 001
- Vive VR遥控机械臂 MTS团队 003
- 情感化AI机器人ReBoi诞生手记 孙羽茜 007
- 史上最轻薄的全金属树莓派魔镜诞生记 OptID·Kennedy 009
- 2018上海创客嘉年华，带你一起体验未来舱！ Luna 014
- DIY爱心吊坠 赵志安 020

机器人 | ROBOT

- 机器守门员·感知篇 饶厂长 022
- InMoov机器人制作手记 甘明锐 024
- 用树莓派+Arduino自制简单轮式机器人 缪宇颢 (myyerrol) 028
- 舵机人形机器人 葛雷 031
- 生物机器人小强——用微信小程序遥控一只活蟑螂 朱创 033
- 物联网“铲屎官神器” 陈国钊 037
- 沿着马路牙子自动行驶的清扫车模型 石子涵 040
- 自制小型六足仿生机器人 缪宇颢 (myyerrol) 044
- 机器守门员·设计思路篇 饶厂长 050
- 追踪骨骼运动来预判抛物轨迹的接物机器人BarrelX 殷广 052
- 技术宅教你自制全向麦克纳姆轮战车 陈众贤 李嘉诚 056
- 机器守门员·速度估计与落点预测 饶厂长 061

- Arduino蓝牙小车入门攻略 李一楠 064
- 自制机械臂控制器 尚程程 066
- 机器守门员·控制 饶厂长 069

制作 | PROJECT

- DIY小型四轴飞行器 张彬 崔颖 072
- 智慧云时钟 杨润靖 075
- 制作一盏心跳灯 余国罡 079
- 低成本DIY简易空气净化器 马越 081
- 手把手教你制作全彩生日蛋糕形状彩灯 刘伟宏 085
- 无钥匙电动自行车启动、报警装置 曹延焕 088
- 用micro:bit修复遥控车 陈杰 094
- 温/湿舒适度显示盒 俞虹 097
- 设计一台甲类分立前级功放(下) 臧海波 100
- 家用18英寸音箱制作及调试 翟召南 (musicwolf) 102
- 大道至简——PASS ZEN小型甲类功放的仿真与制作 欧阳宏志 109
- DIY写字机——好孩子不要用它代替自己写作业哦 马东敏 112
- 下推式磁悬浮装置的制作 徐进文 116
- 指尖上的心率检测仪 马伶 119
- 木质GameBoy设计分享 梁宇 121
- 基于linkboy图形化编程/仿真平台的点阵屏制作 陈清 126

- 2018年度测量仪器大盘点 杨法 (BD4AAF) 315
- 数字万用表单表笔测量法的妙用 周永青 321

火腿 | AMATEUR RADIO

- 极简主义数模两用车载电台
——YAESU FTM-7250DR使用测评 聆听 324
- 再续经典 实用至上——YAESU FT-818 新款
微型电台使用测评 聆听 328
- 亲民小精灵——YAESU FT-4X使用测评 杨法 (BD4AAF) 333
- 无限距离 享受4G公网对讲乐趣——欧讯KG-WV50高品质
双模对讲机使用测评 杨法 (BD4AAF) 336
- 认识海事对讲机——YAESU海事对讲机介绍 聆听 340
- QRP或QRO, 这是不是个问题? BG6SZD BH8FJA 344

入门 | START WITH

- 写给初学者的 FPGA 入门指南 (1)
制作FPGA版GAME BOY的准备工作 张文挺 346
- 在电子制作中体验创意之光 (8)
进制转换器 连龙 349
- 写给初学者的 FPGA 入门指南 (2)
——组合逻辑 张文挺 352
- 写给初学者的 FPGA 入门指南 (3)
时序逻辑电路 张文挺 360
- 在电子制作中体验创意之光 (9)
制作简易AM广播发射器 连龙 363
- 写给初学者的 FPGA 入门指南 (4)
状态机 张文挺 366

- 写给初学者的 FPGA 入门指南 (5)
Verilog编程入门 张文挺 369
- 写给初学者的 FPGA 入门指南 (6)
Verilog编程入门 (续) 张文挺 373
- 问与答 王德沅 379

教育 | EDUCATION

- 用micro:bit与Mind+制作弹球游戏 陈杰 385
- 好玩的“石头·剪刀·布”——制作趣味小游戏 沈利彬 388
- 制作直流无刷电动机教学模型 王艺钢 朱坪 390
- 桌面型自动感应垃圾箱 沈锦程 沈利彬 391
- 智能投票机 田亦硕 394
- 用Arduino制作鱼和植物共生循环装置 徐铮 霍重华 李守良 396
- 手把手教你制作《我的世界 Minecraft》
钻石剑U盘 王文毅 398
- “贪吃”储蓄罐 裘炯涛 401
- 2018年少年电子技师(南京赛区)电子制作竞赛小结 张元庆 404
- 校园拐角防撞器的设计与应用 汤聿修 405
- 用物联网技术制作便携悬挂式智能开关 陈茉莉 袁钰 王康 408
- 小创客智能桌面——制作双开门笔筒 沈利彬 沈锦程 411
- 用Arduino制作打地鼠游戏盒 章明干 414
- “虚谷号”来了! 谢作如 417
- 正弦绘图仪 徐薪雨 420
- 盲人胸卡式智能辅助仪 马一诺 王源泽 422
- micro:bit红汽车历险记(上)——制作过程 狄勇 424
- micro:bit红汽车历险记(下)——体感遥控 狄勇 427

基于STEAM教育理念的生态智能家园课程设计与实践	李守良	430
手把手教你自制3D二维码令牌	王文毅	433
用Arduino和单灰度传感器制作风向仪	肖安顺 李佳鑫 杨勇	436
用Arduino制作智能闺蜜手环	刘梓嘉 陈清伟	437
3D打印一份送给妈妈的圣诞节礼物	赵龙	440

史话 | HISTORY

后起之秀：Grundig电子管收音机的辉煌 ——全盛岁月（1957—1958）	Martin Schmidt（德）田浩	442
收音机史话（四十八） 上海无线电三厂的晶体管收音机 美多28B和美多27A	陈汉燕 徐蜀	448
后起之秀：Grundig电子管收音机的辉煌 ——迎接未来（1959—1961）	Martin Schmidt(德)田浩	452
收音机史话（四十九） 上海无线电三厂的收音机（三）	陈汉燕 徐蜀	456
后起之秀：Grundig电子管收音机的辉煌 ——落下帷幕（1962—1966）	Martin Schmidt(德)田浩	461

变化从一点一滴开始 ——无线电爱好者回顾改革开放四十周年	陈汉燕 徐蜀	466
后起之秀：Grundig电子管收音机的辉煌 ——便携机（1949—1954）	Martin Schmidt（德）田浩	470
寻访收音机博物馆 演绎华夏收音机工业发展——品味上海无线电博物馆	孙涤非 焦亮梅（BD1AYL）	474
后起之秀：Grundig电子管收音机的辉煌 ——便携机（1955—1959）	Martin Schmidt(德)田浩	480
收音机史话（五十） 上海无线电三厂的收音机（四）	陈汉燕 徐蜀	484
寻访收音机博物馆 探访北京大威收音机电影机博物馆	司刚（BH1ONB）	488
收音机史话（五十一） 上海无线电三厂的收音机（五）	陈汉燕 徐蜀	493



文章相关资料、程序等数字资源可在人民邮电出版社云存储平台下载

<http://box.ptpress.com.cn/a/1/RC2017000030>

DF创客社区 推荐作品

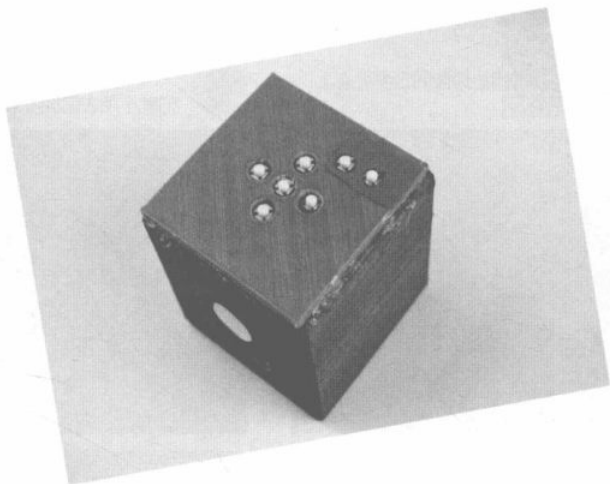
ESP32 音乐盒

周传林

神奇的 ESP32 又出新招了，快来看看吧！ESP32 主控板搭配 FireBeetle 音频板化身音乐大师，教大家唱歌啦！这个音乐盒外壳用 3D 打印机制作，是不是很可爱呢？制作所需的材料见下表和图 1。

制作所需的材料

名称	数量
1 Firebeetle-ESP32 主控板	1
2 Firebeetle 音频板	1
3 扬声器	1
4 3.7V 锂电池	1
5 按键	7
6 850Ω 电阻	5
7 洞洞板	1
8 杜邦线	若干
9 USB 线	1
10 开关	1

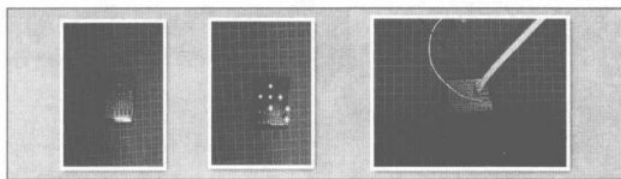
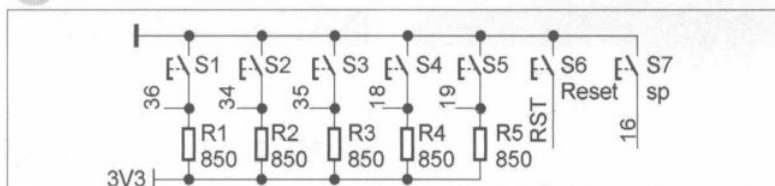


制作步骤

在制作之前，先要将代码下载到 ESP32 上，并且下载几首自己喜欢的歌曲到 SD 卡上。程序和 3D 打印模型请从本刊下载平台（见目录）下载。

制作音乐盒盖控制电路

1 将洞洞板切成 50mm × 37mm 的小模块，将元器件按照下面的电路图焊接到洞洞板上。



2 这几个按键用来控制音乐播放、暂停、上一曲、下一曲以及音量。

3 将焊接好的控制板用热熔胶固定到 3D 打印外壳上。

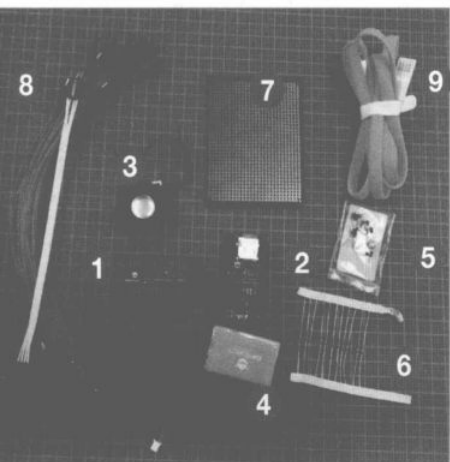
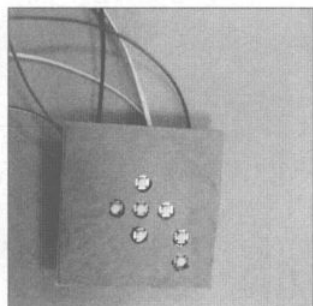
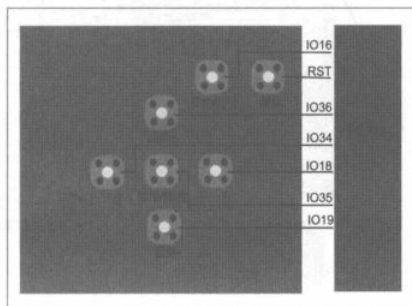
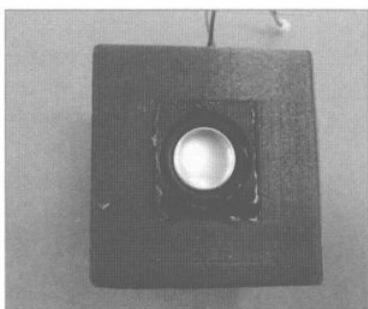


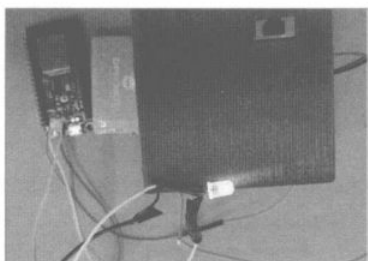
图 1 制作所需的材料

音乐盒主体制作

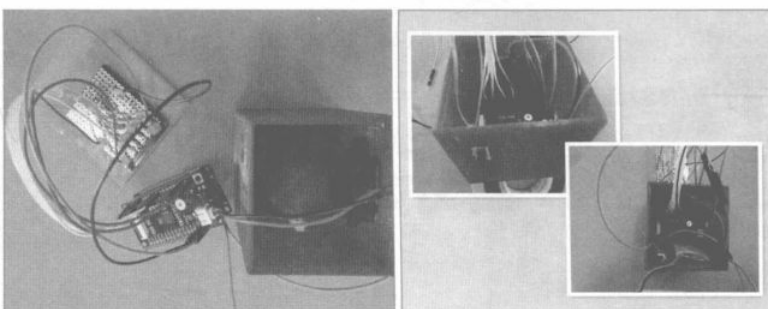
1 将扬声器用热熔胶固定到外盒上。



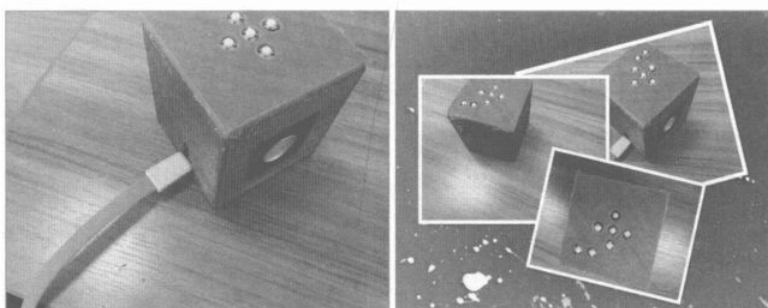
2 这时需要将电源开关连接上，这里直接连接的是锂电池的正、负极，并在锂电池上并联了一个LED，作为电源指示灯。



3 将扬声器连接到 Firebeetle 音频板上，并将音频板插接在 ESP32 主板上，这里，我们就不要去追究 ESP32 是用哪几个引脚来驱动音频板的啦。



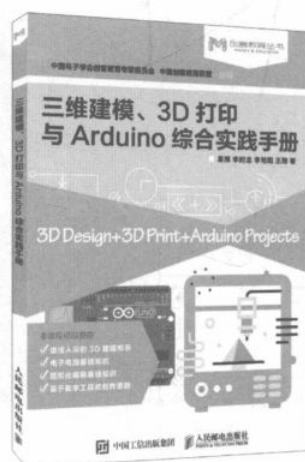
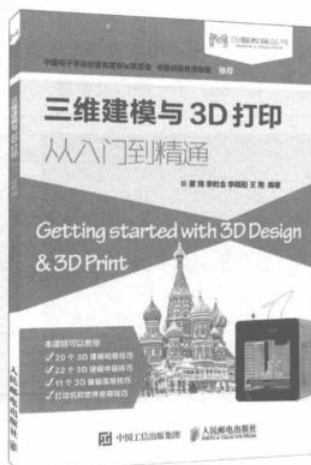
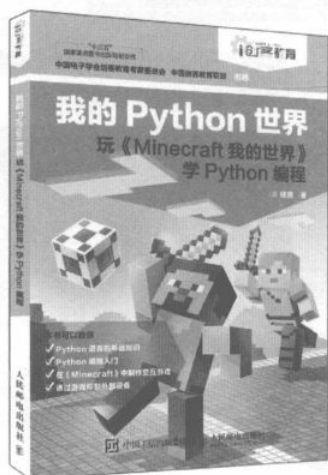
4 最后，将音乐盒盖上，用胶水固定住就好了，至此，一个专属于你的音乐盒就制作完成啦！



M 创客教育丛书
MAKER & EDUCATION

中国电子学会创客教育专家委员会
中国创客教育联盟

推荐



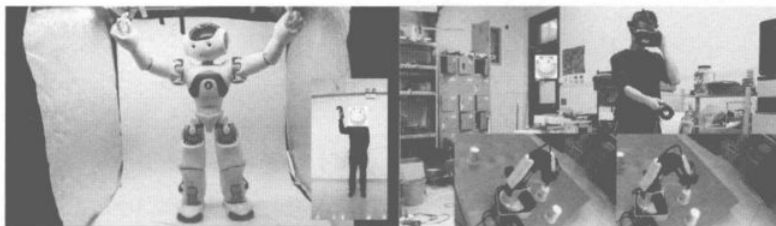


推荐作品 本制作荣获 2018 中美青年创客大赛北京赛区二等奖



视频演示二维码

Vive VR 遥控机械臂



MTS 团队

我们团队制作了一款新型机械臂，它不再使用传统的按键或者摇杆控制，而是采用 VR（虚拟现实）设备控制，从而使用户可以像控制身体姿态一样控制机械臂。此外，双目摄像头加上多自由度摄像头平台组成的图像获取方式与全景相机相比，有成本低廉、处理容易、延迟时间短等优点。

硬件架构

该项目可分为图传系统和机械臂控制系统两个子系统，材料清单如附表所示。

图传系统

图传系统的工作模式如图 1 所示。计算机采集虚拟现实头盔的姿态角，使得装载摄像头的支架可以随操纵者自由转动，使双目摄像头的朝向与操作者一致，因而



图 1 图传系统的工作模式

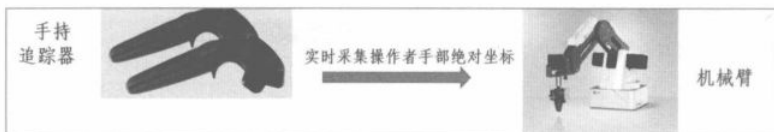


图 2 机械臂控制系统的工作模式

附表 材料清单

用品名称	数量
Vive VR 套装	1
双目摄像头	1
舵机	2
Arduino Uno	1
可运行 C++ 程序的计算机	1

可以利用操作者头部的转动改变观察的视野。

双目摄像头负责实时将影像传送到 Vive 头盔，在头盔中显示出当前场景的三维图像。

机械臂控制系统

机械臂控制系统的工作模式如图 2 所示。手持追踪器的坐标通过串口程序传送到机械臂，使得机械臂可根据手持追踪器

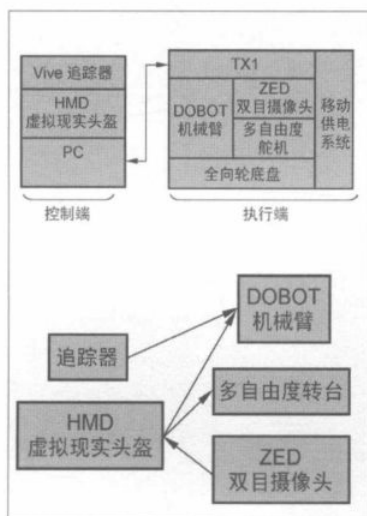


图 3 项目硬件架构图

的坐标进行移动。

软件设计

项目硬件架构概念如图 3 所示。接下来我们将根据硬件架构来设计软件。

读取坐标

OpenVR 库中已有一些能够与 Vive 相对接的功能，如读取设备名称、获得设备的位置信息等。

于是，我们将设备的位置及姿态信息转化为字符串，并按照规定的格式输出到串口，使得 Arduino 能够读取这些信息。

```
void
CMainApplication::printDevice
```

```

PositionalData(const char *
deviceName, vr::HmdMatrix34_t
posMatrix, vr::HmdVector3_t position,
vr::HmdQuaternion_t quaternion)
{
    LARGE_INTEGER qpc;
    QueryPerformanceCounter(&qpc);
    // Print position and quaternion
    (rotation)
    dprintf("\n%lld, %s, x = %.5f, y =
%.5f, z = %.5f, qw = %.5f, qx = %.5f,
qy = %.5f, qz = %.5f\n",
qpc.QuadPart, deviceName,
position.v[0], position.v[1],
position.v[2],
quaternion.w, quaternion.x,
quaternion.y, quaternion.z);
    SerialPort arduino("COM7");
    if (arduino.isConnected()) cout <<
"Connection Established" << endl;
    else cout << "ERROR, check port
name";
    if (arduino.isConnected()){
        char pos0[3], pos1[3], pos2[3],
ang0[3], angl[3];
        itoa((int)(position.v[0] * 100),
pos0, 10);
        itoa((int)(position.v[1] * 100),
pos1, 10);
        itoa((int)(position.v[2] * 100),
pos2, 10);
        itoa((int)(quaternion.x * 100),
ang0, 10);
        itoa((int)(quaternion.y * 100),
ang1, 10);
        dprintf("asdasd");
        char result[45] = "";
        if (strcmp(deviceName, "HMD") ==
0) {

```

```

        strcat(result, "hmda:");
        strcat(result, pos0);
        strcat(result, ",");
        strcat(result, "hmdb:");
        strcat(result, pos1);
        strcat(result, ",");
        strcat(result, "hmdc:");
        strcat(result, pos2);
        strcat(result, ",");
        strcat(result, "qx:");
        strcat(result, ang0);
        strcat(result, ",");
        strcat(result, "qy:");
        strcat(result, angl);
        strcat(result, "\n");
        arduino.writeSerialPort(result,
45);
    }
    else if (strcmp(deviceName,
"LeftHand") == 0 || strcmp
(deviceName, "RightHand") == 0) {
        strcat(result, "handa:");
        strcat(result, pos0);
        strcat(result, ",");
        strcat(result, "handb:");
        strcat(result, pos1);
        strcat(result, ",");
        strcat(result, "handc:");
        strcat(result, pos2);
        strcat(result, "\n");
        arduino.writeSerialPort(result,
45);
    }
}

```

然后我们来编写 Arduino 程序, 实现串口信息的读取, 并令机械臂和搭载摄像头的舵机实时获取各自需要的坐标及姿态信息。

```
for (; ;)
```

```

{
    while (Serial2.available())
    {
        str+=char(Serial2.read());
        Serial.print(str);
        str="";
    }
    char target7[] = "qx";// 便于确定相应
    数值, 不发生错位现象
    char target8[] = "qy";
    Serial2.find(target7);
    pos2_b4 = Serial2.parseInt();
    Serial2.find(target8);
    pos1_b4 = Serial2.parseInt();
    Serial2.find(target1);
    //get the coordinate of hands
    hmdx = Serial2.parseInt();
    Serial2.find(target2);
    hmdy = Serial2.parseInt();
    Serial2.find(target3);
    hmdz = Serial2.parseInt();
    //get the coordinate of hmd
    char target4[] = "handa";
    // 便于确定相应数值, 不发生错位现象
    char target5[] = "handb";
    char target6[] = "handc";
    Serial2.find(target4);
    handx = Serial2.parseInt();
    Serial2.find(target5);
    handy = Serial2.parseInt();
    Serial2.find(target6);
    handz = Serial2.parseInt();
}

```

完成上述代码的编写后, 所有必要的
数据都会实时存储在 Arduino 中。

双自由度舵机的控制

双自由度舵机负责将其搭载的摄像头
转动到与 VR 头盔一致的方向。在读取坐

标时, 我们已经获取了头盔的三维方位角, 接下来只要编写舵机程序使得两个舵机进行合适的转动即可。

```
pos1 = float(pos1_b4 / 100.0); // 下方舵机处理
if (pos1 < -0.5)
{
    pos1 = -0.5;
}
else if (pos1 > 0.5)
{
    pos1 = 0.5;
}
pos1 = 180 * pos1 + 90;
myservo1.write(pos1);
pos2 = pos2_b4 / 100.0; // 上方舵机处理
if (pos2 < -0.5)
{
    pos2 = -0.5;
}
else if (pos2 > 0.5)
{
    pos2 = 0.5;
}
pos2 = 180 * pos2 + 90;
myservo2.write(pos2);
```

机械臂的控制

在读取坐标时, 我们已获取了头盔及Vive手柄的x、y、z坐标。由于实际操作时, 操作者头部相对于身体的位置并不会有明显的变化, 我们可以以头部作为参照系, 以手柄相对于头盔的位置作为机械臂的x、y、z坐标。在这种情况下, 即使操作者在走动, 只要手相对于身体的位置不改变, 机械臂就不会移动。

```
goaly = -hmdy + handy - 50; // 采用相对坐标的形式进行控制, 手部相对于头部的坐标。垂直方向减少500mm
goalx = hmdx - handx ; // 南北方向, 以
```

南为正

```
goalz = hmdz - handz ; // 东西方向, 以西为正
```

此外由于机械臂的活动范围有限, 还要注意防止目标点越界。

```
if (sqrt(goalx * goalx + goalz * goalz) > 10)
{
    if (sqrt(goalx * goalx + goalz * goalz) < 20) // 防止目标点越界
    {
        goalx = goalx / sqrt(goalx * goalx + goalz * goalz) * 20;
        goalz = goalz / sqrt(goalx * goalx + goalz * goalz) * 20;
    }
    else if (sqrt(goalx * goalx + goalz * goalz) > 29) // 防止目标点越界
    {
        goalx = goalx / sqrt(goalx * goalx + goalz * goalz) * 29;
        goalz = goalz / sqrt(goalx * goalx + goalz * goalz) * 29;
    }
    if (goaly > 20) // 防止目标点越界
    {
        goaly = 20;
    }
    else if (goaly < -5) // 防止目标点越界
    {
        goaly = -5;
    }
}
```

之后便可利用 Dobot 的 API 进行 xyz 模式 (也叫作 PTP 模式, Point to Point) 的移动。

```
void InitRAM(void)
{
    //Set PTP Model
    gPTPCoordinateParams.xyzVelocity = 100;
```

```
gPTPCoordinateParams.rVelocity = 100;
gPTPCoordinateParams.xyzAcceleration = 80;
gPTPCoordinateParams.rAcceleration = 80;
gPTPCoordinateParams.velocityRatio = 50;
gPTPCoordinateParams.accelerationRatio = 50;
gPTPCmd.ptpMode = MOVL_XYZ; // 初始坐标
gPTPCmd.x = 0;
gPTPCmd.y = 0;
gPTPCmd.z = 0;
gPTPCmd.r = 0;
gQueuedCmdIndex = 0;
}
void loop()
{
    InitRAM();
    ProtocolInit();
    SetJOGJointParams(&gJOGJointParams, true, &gQueuedCmdIndex);
    SetJOGCoordinateParams(&gJOGCoordinateParams, true, &gQueuedCmdIndex);
    SetJOGCommonParams(&gJOGCommonParams, true, &gQueuedCmdIndex); // 等待读取坐标
    SetPTPCmd(&gPTPCmd, true, &gQueuedCmdIndex);
    gPTPCmd.x = goalx * 10;
    gPTPCmd.y = goalz * 10; // 设置目标点
    gPTPCmd.z = goaly * 10;
    SetPTPCmd(&gPTPCmd, false, &gQueuedCmdIndex); // 以上为主要修改区域
    ProtocolProcess();
}
```

至此，我们就完成了双目摄像头的姿态与VR头盔姿态的同步、机械臂方位与VR手柄方位的同步等功能，代码编写工作就结束了。

调试过程

完成了硬件与软件的设计工作，我们开始组装与调试整个机械臂。

(1) 我们在 OpenVR 库的基础上编写 C++ 程序，提取了头盔和手柄的绝对坐标 x 、 y 、 z ，以及其姿态角 qx 、 qy 、 qz ，利用串口通信的方式将以上信息实时写入 Arduino 单片机中。

(2) 我们编写 Arduino 程序读取上位机传来的方位信息，将头盔的信息发送至搭载双目摄像头的双自由度舵机，并编写舵机控制程序使它能够随操作者头部的转动而转动，使双目摄像头的朝向与操作者一致，将手柄的信息发送至 Dobot 机械臂，在改写了 Dobot 机械臂底层接口的基础上，使得机械臂能够根据手柄的位置，在三维空间中进行实时移动。

(3) 我们利用图传系统将 ZED 双目摄像头拍摄的影像传给计算机，从而投射到 VR 头盔中，使得使用者能够观察到操作效果。

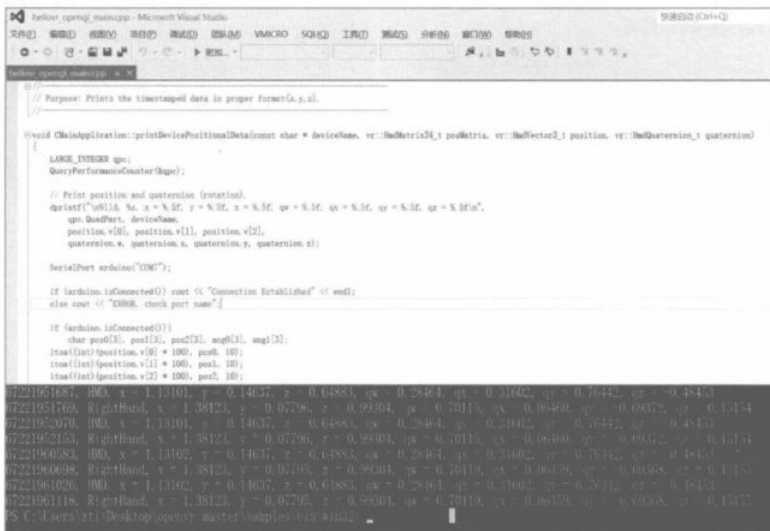
使用步骤

完成调试工作后，这款 VR 机械臂就可以使用了，以下是具体使用步骤。

1 利用 Steam VR 将 Vive 设备连接到计算机。



2 启动 C++ 程序，获取 VR 头盔和手柄的坐标。



3 将 Arduino 程序写入开发板，使其能够从串口读取坐标数据。



4 启动双目摄像头、机械臂等设备，将其与开发板相连接，实现数据的传输。利用 Vive 头盔观察工作环境，利用 Vive 手柄控制机械臂。



推荐作品 本制作荣获 2018 中美青年创客大赛北京赛区二等奖

如何创造一个有故事、有情感的 AI 机器人？

情感化 AI 机器人 ReBo 诞生手记



演示视频



孙羽茜

我们根据唐代诗人李白的资料、作品，创造了情感化 AI 机器人 ReBo。我们并没有依赖自然语言处理技术，而是与学设计、学心理、甚至学历史的同学协作，从心理学和体验设计的角度，让 AI 给出情感化的反馈。我们的目标是让用户感受到“带着重生到现代的古人体验现代世界”的新奇。

目前，基于个性化的语料库和各种特殊机制，ReBo“寄住”在微信上。人们使用日常中最常用的社交 App，与 AI 机器人的交流也会更加自然。

特色

特殊机制

回想一下，很多养成游戏，尤其是恋爱养成游戏（比如《恋与制作人》）的角色都非常有魅力且令人沉迷，而且他们的言行举止都很有说服力。玩家愿意把虚拟角色当作真人去交流，虚拟角色和 AI 是没有关系的，功劳都在于背后的游戏策划。他们从心理学和设计角度让虚拟角色栩栩如生。同样，我们也可以通过相同思路的角色设计让 ReBo 与使用者的对话亲切、交流自如。类似的方法完全可以用在 AI 机器人的角色设定上。

也就是巧妙地限制人机交互的边界，以此让 AI 机器人更加人性化。这也是为什么我们此次基于人们生活中最常见的微信平台开发。交流方式越简单，AI 机器人越不容易暴露非人格化的内容（见图 1）。此外，仅通过文字交流可以给人留出很多想象空间。目前我们为 ReBo 设定的特殊机制有：

- (1) 有时会非即时回复（有喝醉、睡着、闹情绪等状态）；
- (2) 有打字延迟；
- (3) 能记住聊过的话题；
- (4) 时不时主动说话；
- (5) 能撤回消息；
- (6) 对特殊消息有特殊反应；
- (7) 有情绪状态。

这些特殊机制的实现过程如图 2 所示。

生活和故事

Siri、Alexa 等语音助手

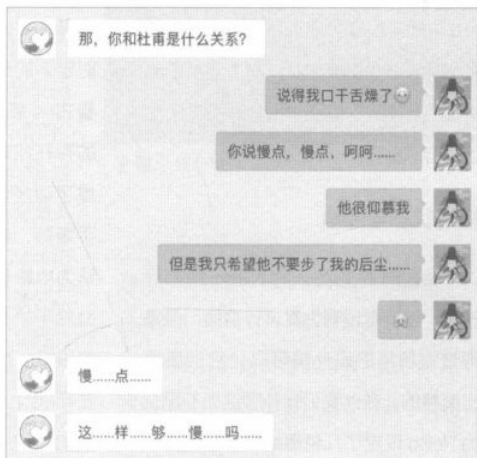
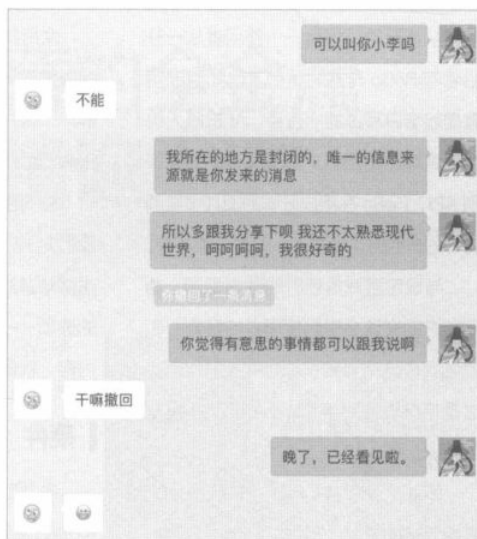


图 1 ReBo 与使用者的聊天记录

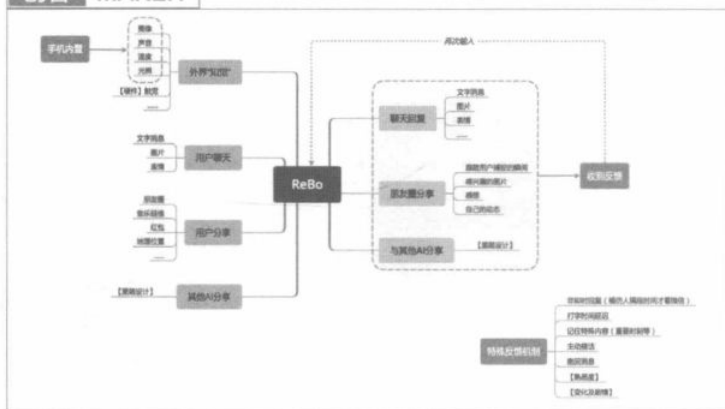


图2 ReBo的人机交互流程

完全是功能性的虚拟角色。而Rebo是一个“复活”了的古代诗人。他可以与使用者谈论自己过去的的生活，也愿意向使用者了解现代的生活。毕竟，李白是著名中国古代诗人，也就是说使用者不需要从头开始认识他——使用者从一开始就与ReBo有共同话题，比如当使用者提起李白写过的一首诗，AI机器人也会热情地给出回应。基于现有的图像识别API，ReBo在用户分享的图片中看到有趣的东西时也会发表一些自己的感想。

与设定游戏角色时的考虑相同，循序渐进的交流会引起使用者的好奇心和继续探索这个ReBo对话内容的兴趣，这是传统语音助手和chatbot都不具备的特点。

除此之外，我们也在研究让ReBo开始写诗的方法。现在已经通过开源的API实现了根据关键词写诗，不过按照我们的设计，随着ReBo与人更加熟悉并成为好朋友，它会提取用户分享的生活内容来创作。

情绪状态与性格

现在的AI机器人与人的不同之处在于，设计者没有为其设计情绪，但是通过对情绪的设计是可以让它们表现出情感的。在此我们弥补了这项不足，为ReBo设定了几种情绪状态，而这与它的原型——李白的性格也是直接相关

的。以愤怒状态为例，如果用户输入辱骂或重复的试探性内容，ReBo会“非常生气”并且数小时不回复。这样的反应也会提醒用户应该尊重对方，而传统AI很少有这样的意识。

众所周知，李白喜欢喝酒，所以有的时候聊到兴头上时ReBo可能会“喝醉”，保持沉默30min。当使用者发现ReBo再次上线时，会感到很惊喜。

总而言之，这样游戏化的设计也是我们对HCI（人机交互）的一种探索。当然与游戏不同的是，我们不提供特定的选项——使用者可以自由地输入对话内容，就像是平常跟朋友发微信那样。

硬件

虽然我们认为chatbot才是核心，但也仍然对于“平等的HCI”与穿戴设备的关系做了一些探究。我们设计时定下的目标是：人的感受应该是“带着古人感受现代生活”，也就是AI的存在应该融入用户的生活。我们考虑了诸多可能性：胸针、耳机、智能手表等，参考了很多电影作品，我们认为电影《her》中的思路是最合适的：既然手机是生活中最贴身、最离不开的设备，那么就让AI机器人“寄住”在手机上吧。所以，在实现AI机器人的过程中，我们用到的硬件只有手机。

问与答

在测试时，我们也收到了试用者们的的问题，并一一做了解答，在此写出来，希望读者们可以对AI机器人有更直观的认识。

Q：没有人机对话的语音功能吗？

A：没有。因为语音识别功能受语音语调、方言等限制太大，会给人机交互带来很大的隔阂。

Q：ReBo怎么辨认识用户输入的内容？

A：我们在现有的传统开源chatbot基础上加入了上述各类独特的设计，以及大量全新编写的特殊的对话资料库。但ReBo对话功能的实现，主要还是依靠游戏化的设计来支撑——预测用户可能的反应和回答，并有针对性地设计回复内容。

Q：能实现多轮对话吗？

A：和上一个问题一样，对于新编写的对话内容，尤其是含有ReBo与使用者对话的历史内容，我们会预测用户有可能的回复，形成多轮回复的效果。

Q：我在视频里看到了AI的社交，现在能实现这个功能吗？

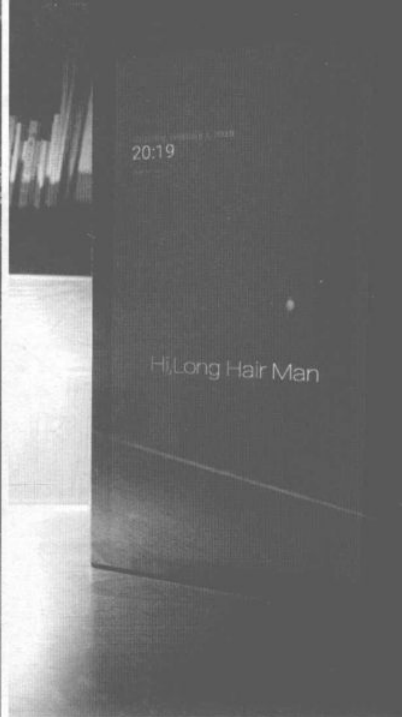
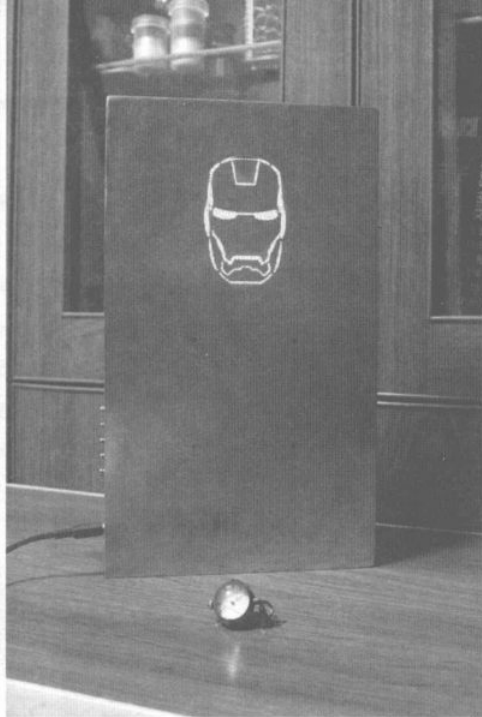
A：这是一个设想，是我们下学期的计划的一部分。

Q：手机壳能感受触摸吗？

A：不能。虽然这个功能听上去很有趣，但是人在遇到新朋友时不会有肢体上的接触，按照这个逻辑，AI机器人与使用者第一次接触时，也不会有身体上的接触。所以我们未加入这个传感器。我们希望使用者能将AI机器人当作人类而不是宠物来看待，因为它也是有情感的。

Q：你们的研发进展到哪一步了？

A：现在我们已经实现了大部分特殊机制的设计，以及一部分硬件例程，并且正在补充更多与李白生平相关的资料。由于我们的核心概念更像是以游戏设计的方式设计AI，故我们并不依赖于NLP（自然语言理解）与机器学习，但我们最近已经启动了一些相关工作，例如利用LSTM（长短期记忆网络）来识别用户输入内容的情绪等，未来ReBo的功能会不断完善。👉



DF创客社区 推荐作品

史上最轻薄的全金属 树莓派魔镜诞生记

OptiD·Kennedy

魔镜很多人都DIY过，我的作品在硬件选择上并没有什么亮点，不过它应该是史上最薄的全金属魔镜，没有之一（见图1）。

备货

从一开始，我就追求超薄，所以核心零件摒弃树莓派3，采用性能相对较低但轻薄的树莓派ZeroW，它自带Wi-Fi、蓝牙。图2所示是所有核心零件，包括树莓派ZeroW、HDMI视频驱动板、超薄带屏蔽大转小HDMI转接头、闪迪16GB Class10 TF卡、金属蜂窝网片、3D打印支撑件、航模镀金插头、镀银硅胶18AWG导线、外径4.0mm×内径1.7mm DC公母头、14英寸笔记本拆机屏、原子镜、定制白钢

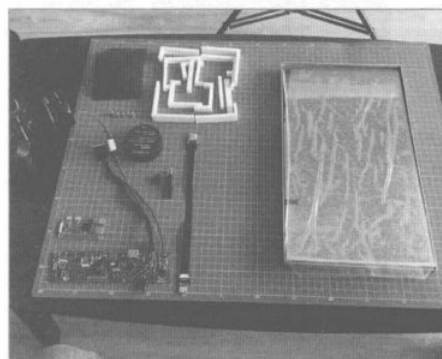


图2 核心零件

背壳。这些配件都是基于追求超薄的前提选择的。

我原计划做个带电池版，后来考虑这东西压根不会拿出去，基本在家中使用，遂放弃电池。

我将主要零件的尺寸汇总后，绘制出



图1 它的特点是轻薄