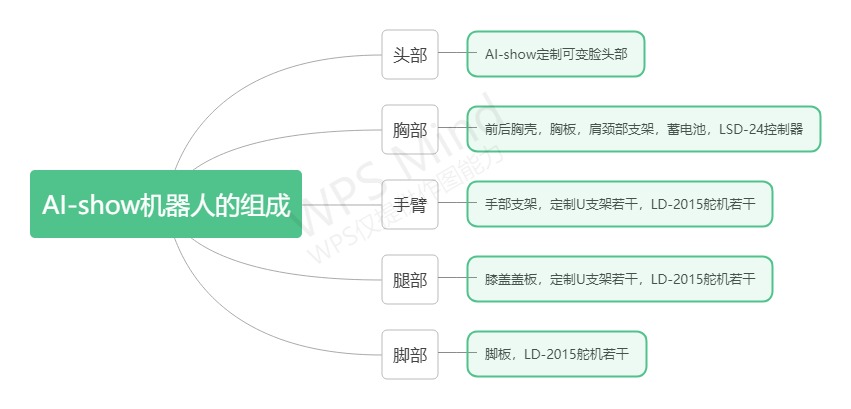
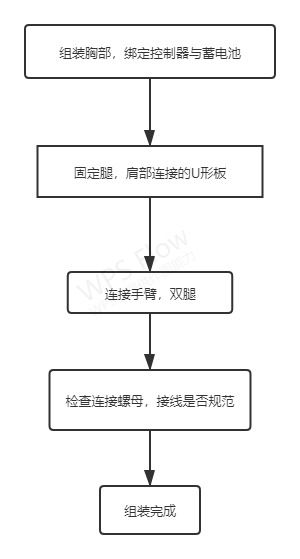
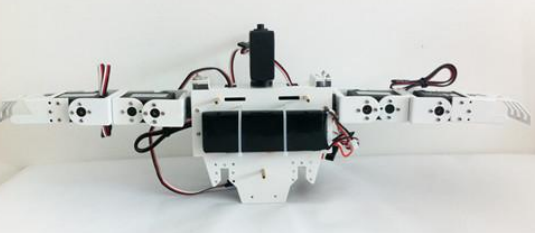
本组机器人的结构设计分为三个板块，分别是外壳支架，LSC-24路舵机控制器，LD-2015舵机。

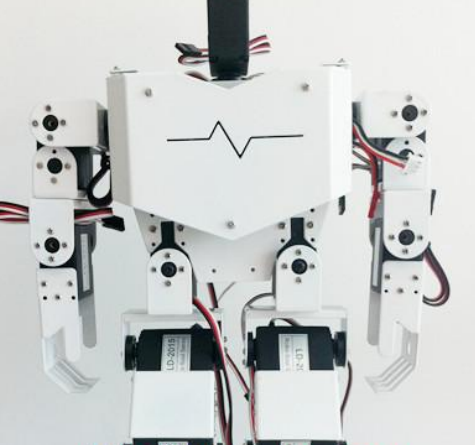
下面将采用流程图和图片的方式描述：

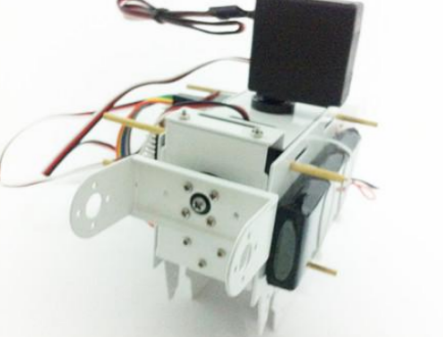


以下展示胸部与手臂，腿部的连接流程：









在机器设计及制作中，该款机器人设计在健全性，安全性，功能性，可玩性都有良好的展现。同时也能胜任本组参赛任务，多舵机使得机器的能动性提高，在原有机器基础上安装定制头部，在机器人表演时，可随舞蹈动作的变化触发变脸行为，提高了可玩性和观赏性。

在调整舵机协同运行方面，本组用c++语言设计并实现了多组舵机并行运行：

|  |
| --- |
| //以15毫秒一步的速度转动  #include <Servo.h>  Servo servo1; // 建立Servo实例  Servo servo2;  const int servo1Pin = 8;  // servo1 接 Pin 8  const int servo2Pin = 9;  // servo2 接 Pin 9  const int every1 = 15;  const int every2 = 15;  int servo1Target = 0;  int servo2Target = 0;  int servo1Dir = 10;  int servo2Dir = 1;  int posOfServo1 = 90;  int posOfServo2 = 90;  void setup() {    servo1.attach(servo1Pin, 500, 2500); // pin8,    servo2.attach(servo2Pin, 500, 2500); // pin9    Serial.begin(9600);    servo1.write(posOfServo1);    servo2.write(posOfServo2);    Serial.println("2 Servo test");    delay(500);  }  void loop( ) {    checkServo1();    checkServo2();    if (posOfServo1 > 179)      servo1Target = 0;    if (posOfServo1 < 1 )      servo1Target = 180;    if (posOfServo2 > 179)      servo2Target = 0;    if (posOfServo2 < 1 )      servo2Target = 180;  }    void checkServo1( ) {    static uint32\_t Timer;    if (Timer > millis())      return;    Timer = millis() + every1;    if (servo1Target > posOfServo1) {      posOfServo1 += servo1Dir;    } else if (servo1Target == posOfServo1) {      return;    } else {      posOfServo1 += -servo1Dir;    }    servo1.write(posOfServo1);  }  void checkServo2( ) {    static uint32\_t Timer;    if (Timer > millis())      return;    Timer = millis() + every2;    if (servo2Target > posOfServo2) {      posOfServo2 += servo2Dir;    } else if (servo2Target == posOfServo2) {      return;    } else {      posOfServo2 += -servo2Dir;    }    servo2.write(posOfServo2);  }  void checkServo3(){    //依Servo1，servo2，以同样方法添加更多舵机  } |

**机器各部分材料选用说明**

包含但不限于说明常用材料选用以及评估是否合理。

机器的支架结构采用航空铝材6061，该材料有一定延展性，同时在机器人整体重量分布设计中表现优异，减轻了舵机负担和动能损耗，同时也有一定的经济性。

舵机，在机器人中，常用到的驱动装置有直流电机、步进电机、舵机。其中直流电机的优点主要体现在大启动转矩与高调速性方面，直流电动机可以在重负载条件下，实现均匀、平滑的[无级调速](https://baike.baidu.com/item/%E6%97%A0%E7%BA%A7%E8%B0%83%E9%80%9F)，而且调速范围较宽。步进电机工作原理为：每接收到一个脉冲，电机会向前转动一个固定距离。但步进电机在体积重量方面没有优势，能源利用率低，超过负载时会破坏同步，高速工作时会发出振动和噪声，并不适合机器人使用。舵机位置控制精确，稳定性好，结构紧凑，接口简单，质量轻，体积小，便于安装，成本低廉，但产生扭力较小，调速性能较弱。本次制作的机器人为小型机器人，需要一定的控制精确度，但对调速性能并无特殊要求。经过上述分析，使用舵机作为机器人的驱动方式。舵机最早出现在航天与航海方面，经过长时间的发展，已广泛应用于生产生活当中。舵机的体积很小，作用大，主要用于角度控制，通常使用于航模、智能小车、机器人，在工业上也有应用，比如船舶，导弹姿态变换的俯仰、偏航、滚转运动都是靠舵机相互配合完成的。舵机在许多工程上都有应用，不仅限于船舶，在线制造，制药等工业控制。舵机属于一种位置伺服的驱动器，主要由可调电位计、控制板、小型马达、变速齿轮组、输出轴组成，如图2.3。电机通过齿轮连接到控制轮上。当电机旋转时，电位器的电阻会发生变化，因此控制电路可以精确地调节电机的运动量和方向。 当电动机的轴是在所希望的位置时，电源供应到电动机被停止。如果不是，则电动机以适当的方向转动。通过信号线通过电脉冲发送所需位置。电机的速度与其实际位置和所需位置之间的差值成比例。因此，如果电机接近所需位置，它将缓慢转动，否则它将快速转动。所以，舵机使用到的是比例控制。



图2.3 舵机结构

舵机的控制通过控制线发送可变宽度或脉冲宽度调制（PWM）的电脉冲来实现，PWM周期为20毫秒，脉冲的宽度决定着舵机的旋转角度，通常舵机的旋转角度为-90度到90度。当接收到PWM信号后，舵机会移动到对应位置，如果在舵机保持位置时外力推压伺服机构，舵机将不会移出该位置。伺服可以施加的最大力量称为伺服的扭矩额定值。

其他机械零件：金属板、90度支架、双折板等，其材质为铝镁合金，是一种广泛应用于航空器制造的材料。特点是重量轻、硬度高、延展性好、可用于制作承力结构。非金属零件包括：输出头，橡胶垫，增加螺丝与金属板之间的摩擦力，电池为锂电池，可支持机器人分钟的正常运行。金属与非金属零件共同组成了机器人硬件系统。

为了实现舞蹈动作，增加灵活性，不断地提高机器人的自由度。但高的自由度可能会导致控制系统的复杂化和机器人不稳定性的提高，在本组机器人设计中，额外的自由度会使机器人重心偏高，重量增加，脚踝关节舵机负载增大。所以有必要对自由度即运动学冗余度做出规划。由于机器人具有对称性，可将机器人分解为独立的腿与独立的机械手进行分析。冗余度被定义为“具有比刚好能完成给定任务所需自由度更多的自由度时，就具有了冗余自由度”。也就是说自由度是否冗余，是根据所需要完成的任务来确定的。如图为机器人及示意图，机器人可实现双足行走等功能。双腿之上的两个自由度操作机器人上方的质量补偿模块，解决行走之时的倾斜即ZMP补偿问题。对于机器人的分析，可以得出以下结论：机器人行走或运动，在不增加脚底面积的情况下，需要有类似“补偿质量”的模块存在。同时，为了实现机器人舞蹈时双腿的开合功能，需要增加垂直于双腿6自由度的自由度。经过进一步改进，机器人示意图如图所示。改进后，ZMP的控制由髋关节和踝关节共同承担。再进一步提高自由度，在机器人脚掌前加装1个脚趾自由度，在髋关节以上加装1个偏摆（yaw）自由度，可以更精确调节ZMP、提高运动时双脚支撑时间，从而提高稳定性。但增加脚趾需要对脚趾的刚性与重量做出严格要求，增加偏转关节会升高机器人重心，从现有的条件来看，弊大于利。经过以上材料的分析，机器人双腿自由度规划如下：髋关节设置滚动（roll）、俯仰（pitch）2个自由度，膝关节设置1个俯仰自由度，踝关节设置滚动、俯仰2个自由度，双腿共计10自由度。机器人上肢可以看作是一个机械臂，需要完成的“任务”是配合双腿及音乐，做出简单的摆动动作，根据从简原则做出规划：腕关节1个滚动自由度，肘关节1个滚动自由度，肩关节1个滚动自由度。

