慧科
只为赋能人才

基于Yanshee的二次开发
灯光、动作篇

——— 张洪涛

编程语言

- C语言：可移植性好，可直接控制硬件，效率高

慧科
只为赋能人才

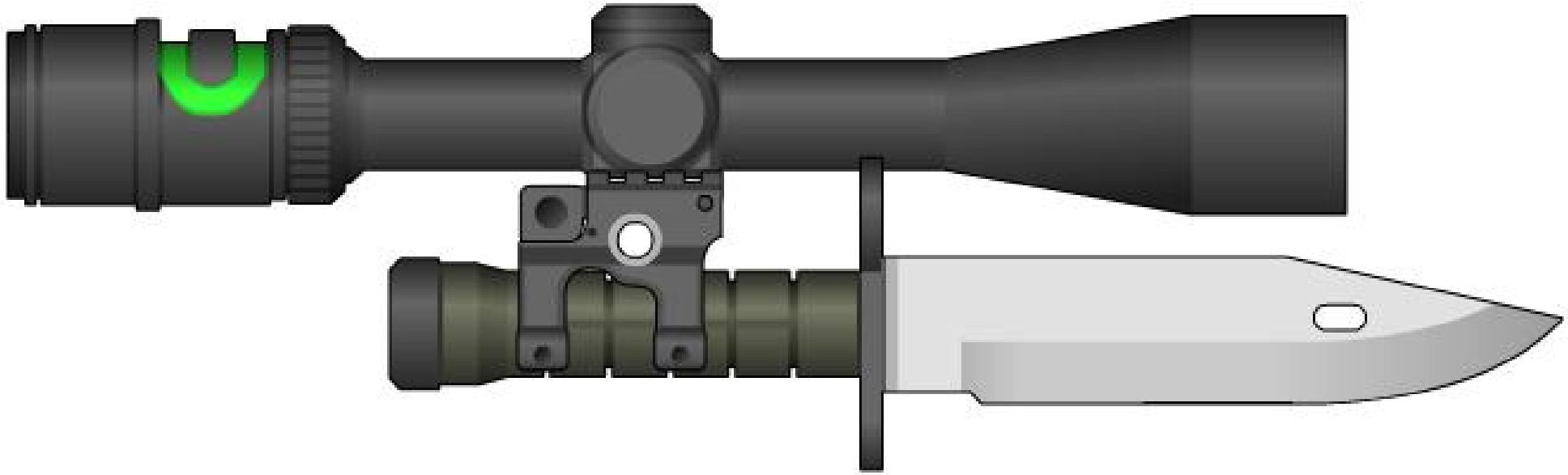
编程语言

- JAVA： 面向对象。跨平台，与平台无关。

慧科
只为赋能人才

编程语言

- 汇编：晦涩难懂，直接控制底层硬件
- 效率最高



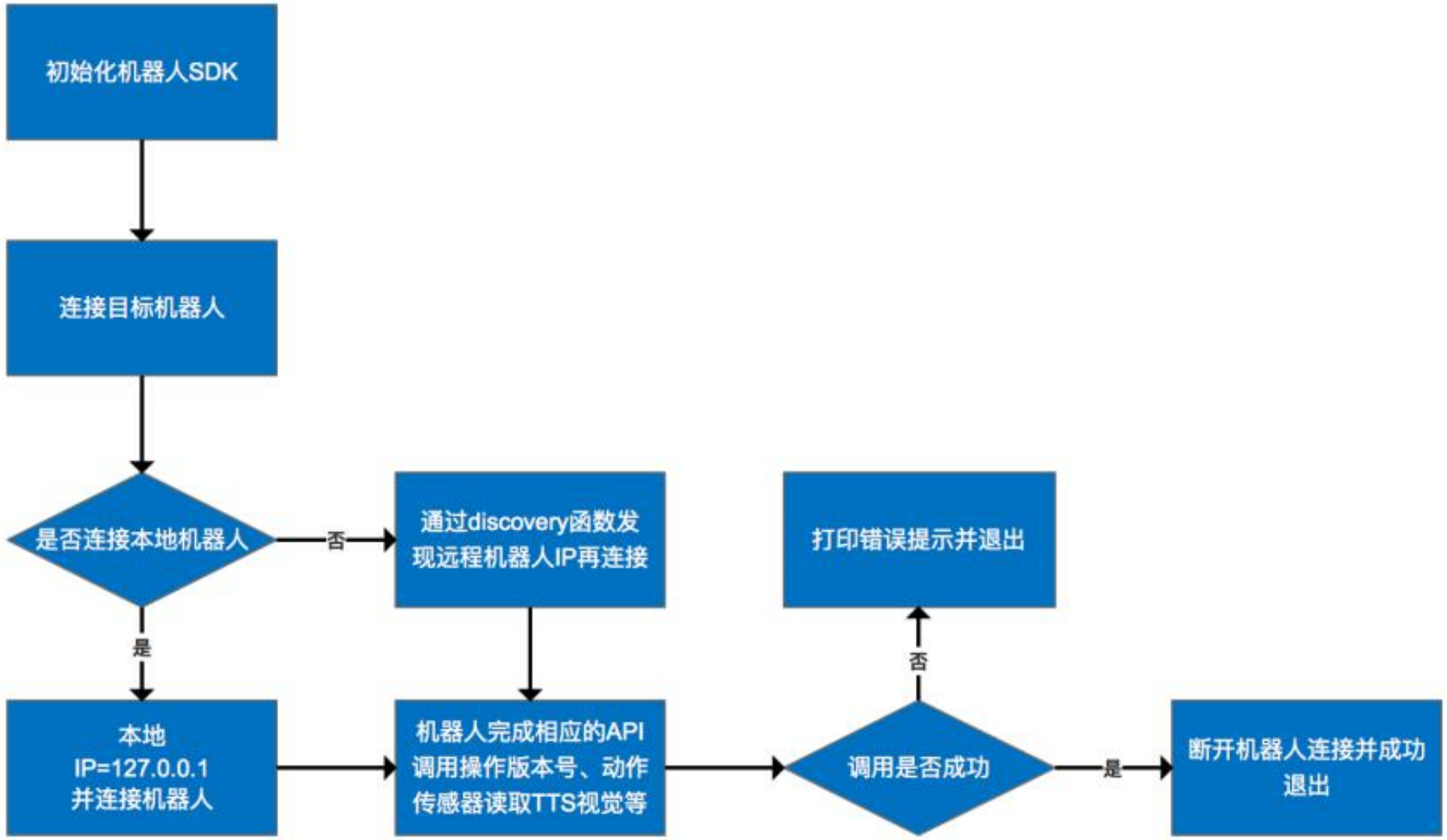
慧科
只为赋能人才

编程语言

- python：胶水语言，可轻松将不同语言制作的模块连接在一起
- 语法简单，易学
- 库多，功能强

慧科
只为赋能人才

Yanshee SDK代码编辑流程概述



慧科
只为赋能人才

程序流程

```
#!/usr/bin/python
# _*_ coding: utf-8 -*-
import time
import RobotApi

RobotApi.ubtRobotInitialize()
#------------------------------Connect-------------------------
gIPAddr=""
robotinfo=RobotApi.UBTEDU_ROBOTINFO_T()
#The robot name you want to connect
robotinfo.acName="Yanshee_438F"
ret=RobotApi.ubtRobotDiscovery("SDK", 15, robotinfo)
if (0!= ret):
    print ("Return value: %d"% ret)
    exit(1)
gIPAddr=robotinfo.acIPAddr
ret=RobotApi.ubtRobotConnect("SDK", "1", gIPAddr)
if (0!= ret):
    print ("Can not connect to robot %s"%robotinfo.acName)
    exit(1)
```

```
#----------------------------Disconnect--------------------
RobotApi.ubtRobotDisconnect("SDK","1",gIPAddr)

RobotApi.ubtRobotDeinitialize()
```

慧科
只为赋能人才

灯光控制接口

名称	bool ubtSetRobotLED(char *Type, char *Color, char *Mode)
描述	设置 机器人LED灯
参数	【Type】button camera mic 【color】灯颜色 white-白 red-红 green-绿 blue-蓝 yellow-黄 purple-紫 cyan-青绿； 【mode】off-关on-常亮 flicker-频闪 breath-呼吸 alternation 轮流
返回值	-1-- 操作失败；0 -- 操作成功
调用方式	同步

慧科
只为赋能人才

调用方式

- 同步调用：阻塞式调用
- 异步调用：非阻塞式调用
- 回调：回调是解决异步函数执行结果的处理方法。在异步调用的时候，如果希望将执行的结果返回并进行处理时，我们可以通过回调的方法进行解决。

慧科
只为赋能人才

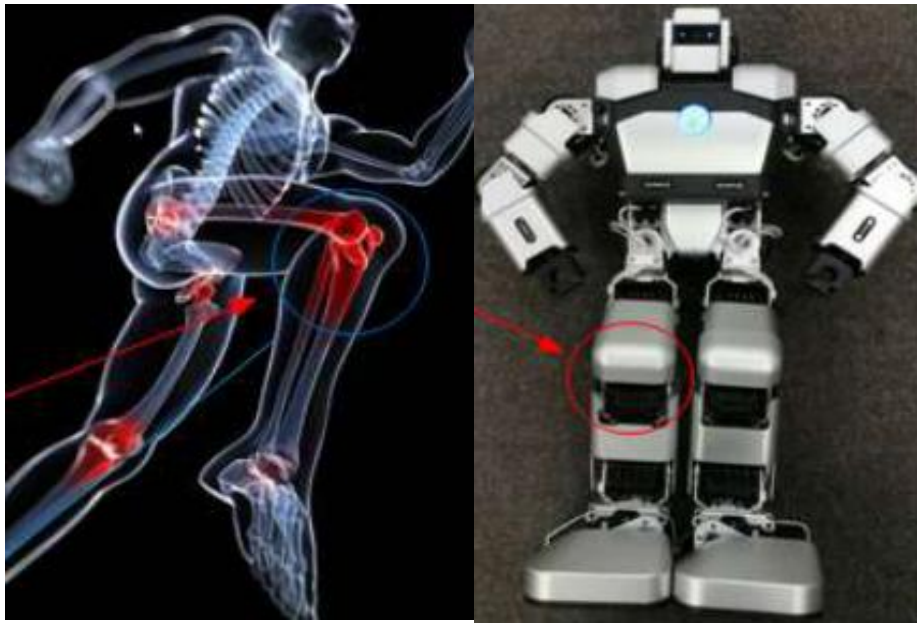
练习

- 熟悉机器人SDK控制流程
- 调用接口实现闪烁灯、呼吸灯

慧科
只为赋能人才

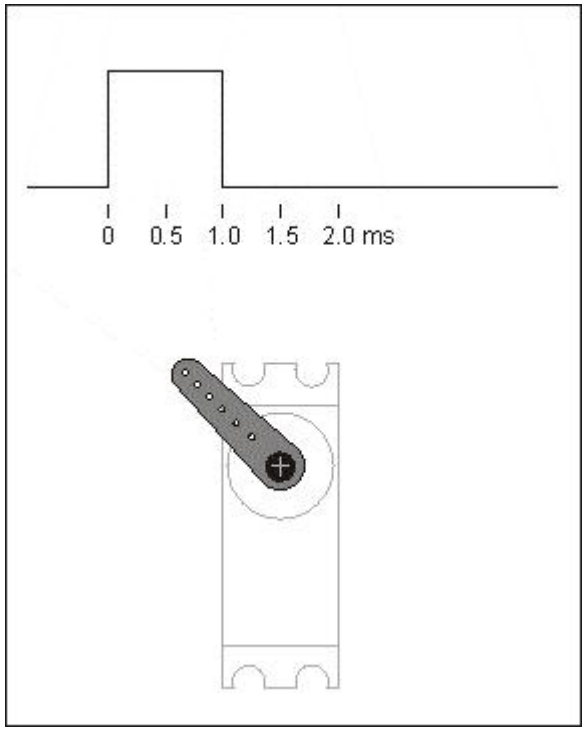
机器人运动

- 大自然中，从动物界到人类，它们的肢体活动都是靠关节的连接和肢体的互动完成的。而我们机器人想模仿人类的活动也必须有相应的关节才能活动，于是就有了舵机的出现，舵机一般意义上可以直接理解为机器人的关节，我们通过控制他们的角度和结构件连接之后完成机器人组合动作，例如：当人要迈出向前走一步的动作的时候，我们的胯关节提起、膝关节弯曲向前、踝关节跟着朝上伸出，完成向前走一步的动作。由此可见，没有关节就没有运动可言，通过关节的配合，动物们完成了他们肢体的第一次转动。

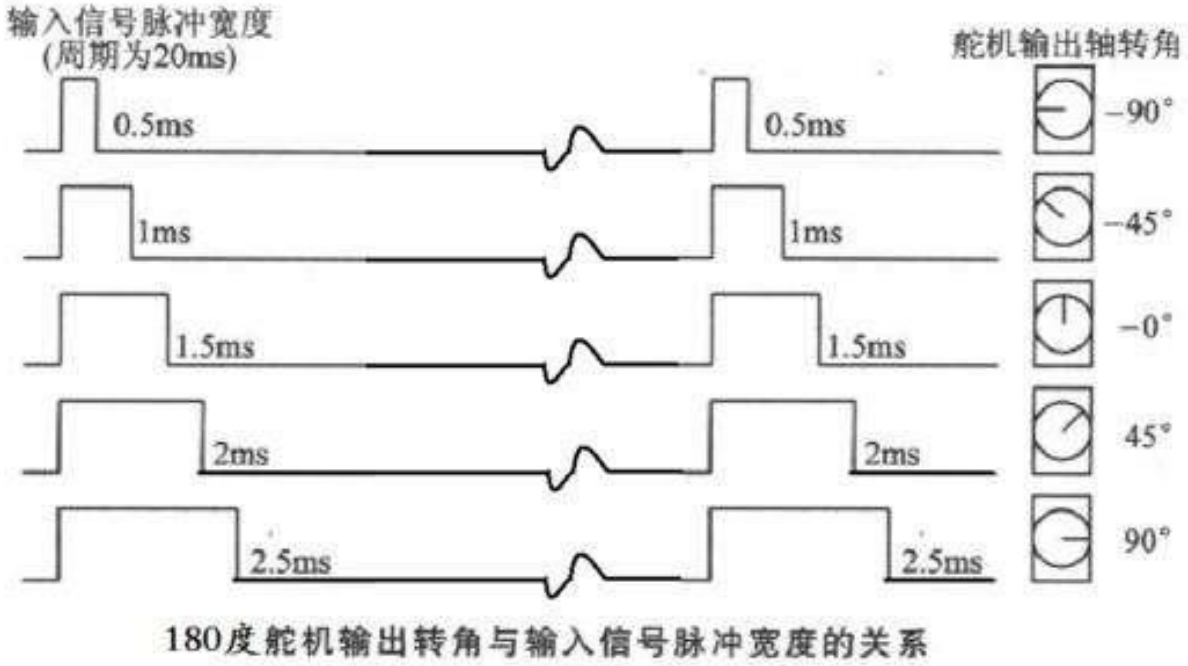


机器人动作

- 舵机它是一种带有输出轴的小装置。当我们向伺服器发送一个控制信号时，输出轴就可以转到特定的位置。只要控制信号持续不变，伺服机构就会保持轴的角度位置不改变。如果控制信号发生变化，输出轴的位置也会相应发生变化。



PWM信号



180度舵机输出转角与输入信号脉冲宽度的关系

慧科
只为赋能人才

舵机的分类

- **模拟舵机**运行时，以一定频率（一般为 50Hz）不断从控制端接收 PWM 信号。每个信号所指示的角度都与当前的角度相差不多。每次旋转一点角度，最后转到目的位置。

- **数字舵机**也要用PWM 信号驱动电机，但它接收的控制信号与模拟舵机不同。通常来说，数字舵机只需要接收一个脉冲信号，获取到需要转到的角度。舵机内部的处理器对这个角度分析，自动产生连续的控制脉冲，以很高的频率（300Hz）发送给舵机，驱使舵机旋转。这样不但让控制变得更简单，提高舵机接收的信号频率，也能让转动更加流畅。

慧科
只为赋能人才

串行总线及舵机ID

- 一般来说，舵机在运行时，从控制端接收控制信号。如果每个舵机都单独拉一根线，不但控制比较复杂，而且对布线也限制很大。

- 与 PWM 控制不同，总线上传输的是编码过的二进制消息。舵机接收到消息后，按照协议解析，判断消息的目的地（舵机 ID）。当消息目的地的 ID 与自己一致时，就依照消息内容，做出相应的反应。这样一根总线就能控制多个舵机。不仅如此，舵机也可以传输信号给控制端。

慧科
只为赋能人才

舵机控制

名称	int ubtSetRobotServo(UBTEDU_ROBOTSERVO_T *servoAngle, int iTime);
描述	控制 单个、多个、所有 舵机的运动
参数	【servoAngle】包含了1到17号舵机的0 – 180度对应角度。 【Time】舵机运行时间，单位为20ms（最小大于10）
返回值	-1为失败， 0为成功
调用方式	同步

慧科
只为赋能人才

舵机控制

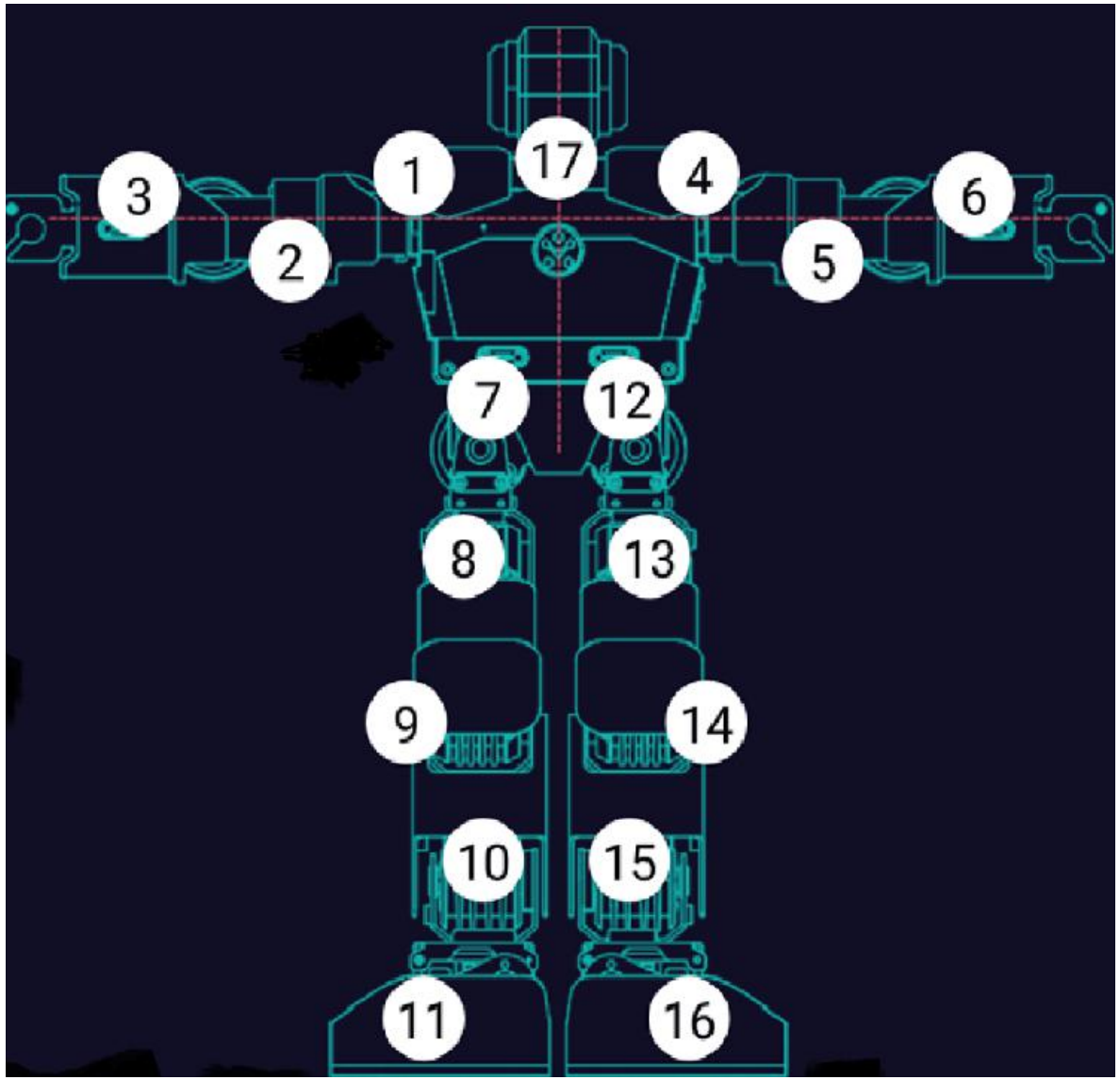
- 通过调用 RobotApi.ubtSetRobotServo(UBTEDU_ROBOTSERVO_T *servoAngle, int iTime);函数给舵机设置角度值，其中前一个参数是表示 Yanshee 全身各个舵机角度的结构体，后一个参数是舵机运动的时间。通过对 servoinfo. SERVO17_ANGLE 赋值，我们可以改变 Yanshee 头部舵机的角度（SERVO17_ANGLE 是头部舵机的序号，关于舵机序号，将在下一页讲解）。

- 时间参数以 20ms 为单位，即当参数为 1 时，表示运动时间为 20ms。那么，当函数RobotApi.ubtSetRobotServo(servoinfo, 20)执行到时，表示头部将花费 400ms，转动到 60 度的位置。

慧科
只为赋能人才

舵机编号

- Yanshee机器人正面舵机分布说明图



慧科
只为赋能人才

舵机回读

- 总线的结构使舵机向控制端回传数据成为了可能。Yanshee 的舵机就有这样的功能。
- 当控制端发送特定指令给舵机时，舵机可以从内部的传感器读取当前的位置，并组装一条包含位置信息的消息传回控制端，控制端可以将这个位置信息记录下来以供以后使用。这个过程我们叫做回读。

慧科
只为赋能人才

舵机掉电

- 如果驱动电路不给舵机发送 PWM 信号，舵机就会失去动力。表现出来就是舵机“变软”，人可以轻易用手掰动舵机转动。这种状态就是舵机的掉电状态。相对的，舵机“变硬”，难以掰动的状态称作上电状态。

慧科
只为赋能人才

练习

- 使用APP中的回读编程，使机器人完成一个动作
- 注意机器人平衡问题

慧科
只为赋能人才

舵机回读

	void ubtGetRobotServo(UBTEDU_ROBOTSERVO_T *servoAngle)
描述	回读 单个、多个、所有舵机的角度，直接返回全部舵机角度值，通过类似下面结构来读取： servoinfo =RobotApi.UBTEDU_ROBOTSERVO_T() servoinfo.SERVO1_ANGLE servoinfo.SERVO2_ANGLE ..... servoinfo.SERVO17_ANGLE
参数	servoAngle 0-180 对应舵机角度，值为255表示舵机无应答
返回值	-1-- 操作失败；0 -- 操作成功
调用方式	同步

慧科
只为赋能人才

局部动作控制

名称	bool ubtSetRobotMotion(char* Type, char* direct, int speed, int repeat)
描述	控制机器人做局部动作，例如：蹲下、举手、
参数	【Type】：crouch蹲下、raise 举手、 stretch 伸手、come on 招手、wave挥手、bend 弯曲、walk 走路、turn around 转身、head 转动头部、bow鞠躬； 【direct】：left 左、right右、both双手、front前、back后; 【speed】：1/2/3/4/5（五个速度等级，依次更快），默认为3； 【Repeat】重复次数；
返回值	-1-- 操作失败；0 -- 操作成功
调用方式	同步

慧科
只为赋能人才

播放动作文件

	bool　ubtStartRobotAction (char* Name, int repeat)
描述	执行动作文件
参数	【Name】动作文件名，例如：push up（俯卧撑）、bow（鞠躬）等(/mnt/1xrobot/res/hts)
返回值	-1-- 操作失败；0 -- 操作成功
调用方式	异步

慧科
只为赋能人才

动作暂停

名称	bool　ubtStopRobotAction (void)
描述	停止执行动作文件
参数	空
返回值	-1-- 操作失败；0 -- 操作成功
调用方式	同步

慧科
只为赋能人才

按键事件检测

名称	int ubtEventDetect(char *pcEventType, char *pcValue, int iTimeout)
描述	执行动作文件
参数	【pcEventType】要检测事件类型，目前默认只支持button类型； 【pcValue】事件重复次数，0：没有发生 其它是实际发生次数； 【iTimeout】设置事件超时时间，最小10s,最大10min。
返回值	-1-- 操作失败；0 -- 操作成功
调用方式	异步

慧科
只为赋能人才

练习

- 通过SDK提供的方法，测试机器人的多个动作
- 通过直接控制每个舵机角度的方法让机器人执行一个复杂的动作，比如：俯卧撑
- 注意机器人的平衡问题
- 可添加简单的逻辑，比如：通过按键控制机器人的动作

慧科
只为赋能人才

练习

- 1、金鸡独立：
  - 保存机器人初始的舵机状态
  - 控制舵机的方式实现金鸡独立（腿抬高，手臂动作）
  - 站稳
  - 恢复舵机的状态

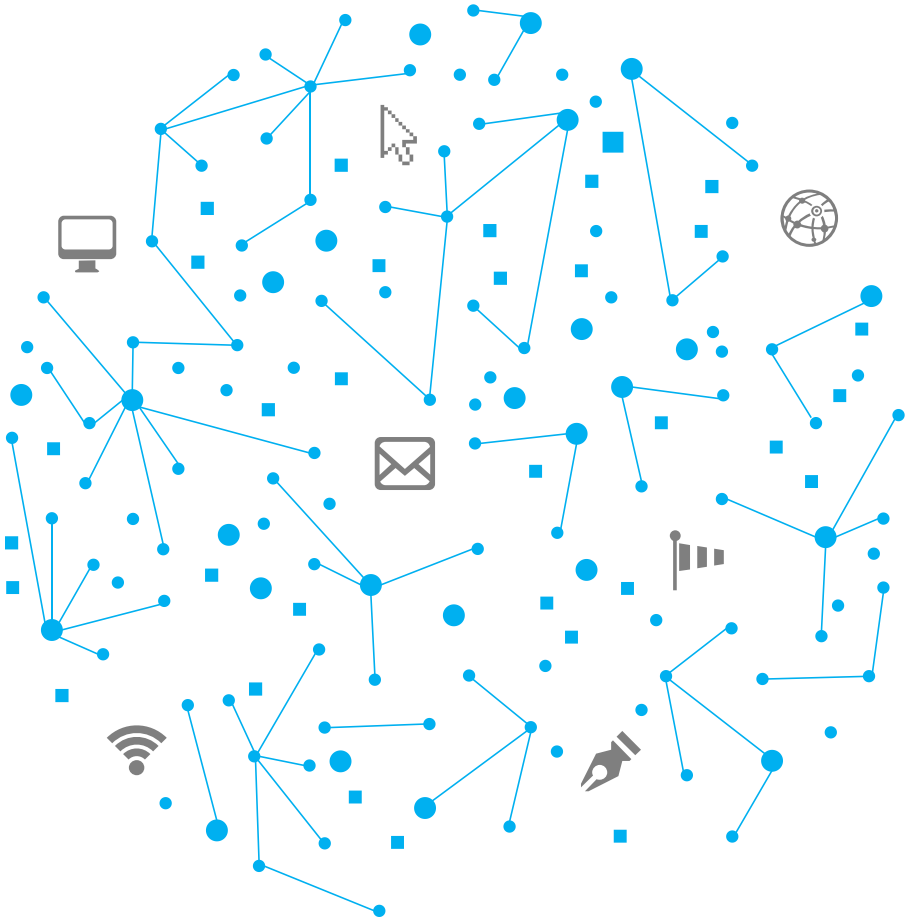
  2、使用回读编程，完成一个动作序列（可以加音乐）

  保存到机器人

  通过ubtRobotStartAction接口调用自己编排的动作

慧科

只为赋能人才



THANK YOU

谢谢观看!