

# 六足仿生机器人的设计与实现

樊启润, 唐彪, 孙开鑫, 万美琳, 卢仕

(湖北大学 物理与电子科学学院, 武汉 430062)

**摘要:** 所述六足仿生机器人基于 SoC FPGA 平台实现, 结合了机械结构设计、六足步态控制、蓝牙传输技术、弯曲传感器、OpenCL 图像处理加速、VR 显示等诸多技术; ARM 部分作为主控, 存储摄像头视频图像, 并调用 FPGA 模块对图像处理加速, 通过路由器架设的局域网向 VR 眼镜输出视频流信息; FPGA 部分用于接收蓝牙信号, 驱动机器人手臂运动, 摄像头拍摄角度切换以及六足行进; 实际操作时操作者需佩戴自制的数字手套和 VR 眼镜; 操作数据手套上的方向按键可控制机器人移动; 数据手套的每个手指上安装有弯曲传感器, 用于控制机械手臂跟随人手实时运动; VR 眼镜中放置一个智能手机作为显示终端, 实时显示机器人摄像头获取的画面; 经过多次实际测试, 操作者佩戴 VR 眼镜及数据手套均可远程操控机器人抓取置于复杂地形中的水瓶。

**关键词:** 六足仿生机器人; SoC FPGA; OpenCL 图像处理加速; VR 显示

## Design and Implementation of Hexapod Bionic Robot

Fan Qiqun, Tang Biao, Sun Kaixin, Wan Meilin, Lu Shi

(Faculty of Physics and Electronic Science, Wuhan 430062, China)

**Abstract:** The hexapod bionic robot is based on the Soc FPGA platform, combining with the mechanical structure design, controlling of gait, technology of bluetooth transmission, bending sensor, acceleration of image processing in OpenCL, VR display, etc. ARM part is a master to store camera video images, to call FPGA module accelerating image processing, and to output video information to VR glasses by WLAN. The functions of FPGA part are to receive the signal of bluetooth, to drive robot arm, to control the servos of camera, and to drive servos of hexapod base. Operator needs to wear a data glove and VR glasses when operating the robot. The direction keys on the data glove can control the robot to move. Each finger of the data glove is equipped with a bending sensor, which is used to control the mechanical arm to move in real time. A smart phone is placed in the VR glasses as a display terminal to display the real-time images which captured by the robot camera. After many practical tests, the operator can remotely control the robot to grab bottles placed in complex terrain.

**Keywords:** hexapod bionic robot; SoC FPGA; openCL image processing acceleration; VR display

## 0 引言

近年来, 用机器人替代人类进入条件恶劣或者高度危险的区域工作已经成为一种趋势。例如机器人进行水下作业、无人机进行侦查勘测、机器人在废墟中探测生命迹象等。然而许多在路面作业的轮式或履带式机器人在遇到复杂地形时, 因地形原因将无法正常工作。其次大多数的机械臂无法像人手一般灵活, 在特殊环境下增加了控制者的操作难度为解决以上问题, 本项目基于 SOC-FPGA 平台设计实现了一种可在复杂地形行动, 机械臂可跟随人手实时作业, 可运用 VR 眼镜实时观察行径中的全景图像的六足仿生机器人。

## 1 系统框架设计

系统框图如图 1 所示, 主要由视频成像设备、仿生机

器人主体、远程控制设备三部分构成。其中视频成像设备是戴在操作者头部的 VR 眼镜, 远程控制设备是穿戴在操作者手上的数据采集手套。

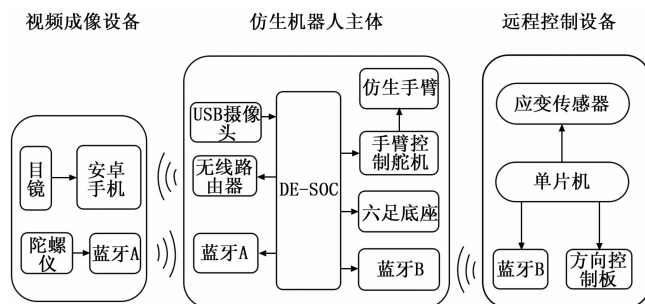


图 1 系统整体框架

### 1.1 机器人主体设计

#### 1.1.1 外部结构设计

机器人主体实物如图 2 所示, 由六足底座、DE1-SOC 开发板、仿生手臂、USB 摄像头、无线路由器及两个蓝牙模块组成。DE1-SOC 开发平台作为机器人的信息处理中心, 是基于台湾友晶公司设计生产的 Intel SoC FPGA 硬件设计平台, 具备强大的双核 Cortex-A9 处理器和行业领先

收稿日期: 2019-02-20; 修回日期: 2019-03-06。

基金项目: 国家自然科学基金(61704050)。

作者简介: 樊启润(1997-), 男, 湖北武汉人, 主要从事 FPGA 应用方向的研究。

通讯作者: 卢仕(1990-), 男, 硕士, 主要从事 FPGA 应用、数字 IC 设计方向的研究。

的可编程逻辑器件<sup>[1]</sup>。高清 USB 摄像头作为机器人的视频输入设备，同时配备两个舵机控制其拍摄角度。无线路由器用于构建局域网，实现机器人到视频成像设备的视频实时传输。蓝牙 A、蓝牙 B 两个蓝牙传输模块，分别用于操作者头部运动信息和远程控制设备信息的接收。

如图 3 所示，机器人的运动部件为对称的六脚蜘蛛结构的底座，六足支撑状态站立时，底座尺寸为长：48 cm；宽：43.5 cm；净高：9 cm；包含开发平台及摄像头主体高度：48.7 cm。

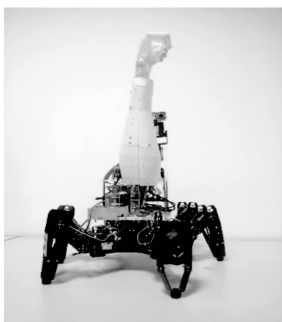


图 2 机器人主体实物图

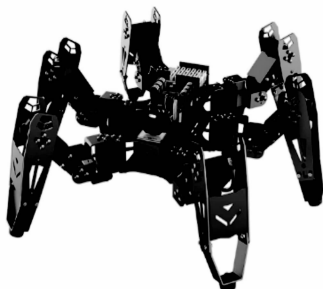


图 3 六足底座模型

机器人单足尺寸为：宽 5.5 cm；伸直状态下，足长：23 cm。且第一关节长：14 cm；第二关节长：9 cm；第三关节长：5 cm。每只脚均装配有三个舵机，如图 4 所示，实现了三个自由度运动的控制。1 号舵机控制“足”在 XY 平面的旋转；2、3 号舵机控制“足”的两个不同关节在垂直于 XY 平面方向的伸缩。舵机主要参数如表 1 所示。

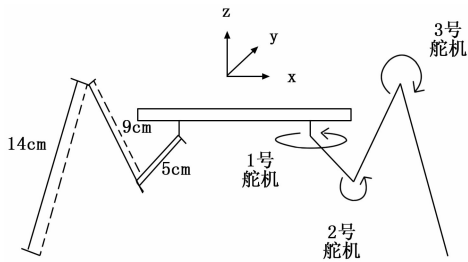


图 4 足部舵机分布图

表 1 六足底座舵机参数

舵机型号	重量	尺寸
LD-2015	60g	40 * 20 * 40.5mm
速度	堵转扭矩	工作电压
0.16sec/60°	15—17kg/cm	6—7.4V

如图 5 所示，仿生手臂及手指采用 3D 打印制作，手臂截面直径最大为 10.5 cm，手掌张开手臂全长为 42.5 cm，内部安装有 5 个 MG995 型号舵机，具体参数如表 2 所示。用 5 条高强度编织绳将其与手指连接。通过舵机来牵扯编织绳，实现对手指弯曲程度的控制。

### 1.1.2 控制逻辑设计

控制逻辑分为三个部分，分别是陀螺仪的数据收集与



图 5 仿生手臂内部结构

表 2 手臂内部舵机参数

舵机型号	重量	尺寸
MG995	62g	40 * 20 * 36.5mm
速度	堵转扭矩	工作电压
0.13—0.17sec/60°	13kg/cm	3—7.2V

处理；弯曲传感器信号接收与处理；方向控制指令的接收与处理。

陀螺仪数据收集与处理模块，细分为如下子模块：蓝牙连接指令存储模块，串口数据发送模块，串口数据接收模块，数据处理模块，算法模块，舵机控制模块。蓝牙信号传输工作波特率数值设置为 1 200。其工作逻辑为，串口数据发送模块发送连接指令，连接成功后串口接收模块接收数据。数据经数据处理模块提取出有效数据，如陀螺仪的滚转角、俯仰角、偏航角。有效数据输出至算法模块，转换为摄像头舵机的控制信号。最后将控制信息发送至舵机控制模块，实现对摄像头拍摄角度的控制。

在弯曲传感器信号接收与处理模块中，其蓝牙信号传输部分波特率数值设置为 9 600，原理与陀螺仪信号接收相同。在数据接收后传入数据处理模块进行解码操作，输出预置的控制信号给舵机控制模块，实现对机械臂的控制。

方向控制指令的接收与处理模块包含指令接收，指令发送以及解码模块。蓝牙传输波特率数值设置为 115 200，经指令发送模块发送握手信号，连接成功后，接收模块接收蓝牙数据。经解码模块将其与预置数据进行比对。得出相应的控制信号，将其发送至舵机控制模块，从而驱动底座运动。

## 1.2 远程控制设备设计

远程控制设备为一个自制的数字手套，如图 6 所示，由六片应变传感器、方向控制板以及蓝牙模块组成。应变传感器获取操作者五个手指和手臂的弯曲程度<sup>[2]</sup>，转换成数据信号后连同控制板输出的方向信号一起通过蓝牙模块发送至仿生机器人主体，实现对六足底座和仿生手臂的控制。

## 1.3 视频成像端设计

视频成像端实物如图 7 所示，主体为一个 VR 头盔，内部放置一部安卓手机，陀螺仪，锂电池以及蓝牙模块。陀螺仪采集操作者头部运动信息，通过蓝牙模块发送至仿生机器人主体，信号经处理后用于控制机器人摄像头的拍摄



机器人的空间位置，其目的是改变仿生手臂的方向以及机器人前进方向。如图 11 所示，旋转过程中仿生机器的步态设计中同样分为六足支撑阶段和单足摆动阶段。其中第一阶段与第五阶段为六足支撑阶段，用于平衡仿生机器人重心。单足摆动阶段，采用相邻足连续工作的方式，即 1、2 足协调运动，3、4 足协调运动，5、6 足协调运动。运动顺序依次为：1 号足、2 号足、3 号足、4 号足、5 号足、6 号足。在单足摆动过程中，运动分为垂直上升阶段、水平摆动阶段、垂直下降阶段。设置每足单次抬起高度为 15 cm，单次旋转角度为 15°，1 号足至 6 号足依次执行步态完毕一个旋转周期结束。

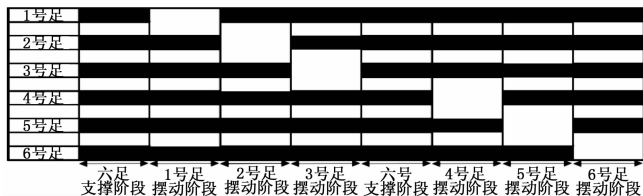


图 11 旋转步态

## 2.2 远程控制设备设计

如图 12 所示，远程控制设备系统包括控制仿生手臂运动的数据采集手套、捆绑于操作者手臂上的单片机、蓝牙模块以及方向控制面板。单片机对来自数据采集手套和来自控制面板的这两类信号进行采集、编码等操作，并通过蓝牙模块将处理结果发送至仿生机器人主体，以实现远程遥控。

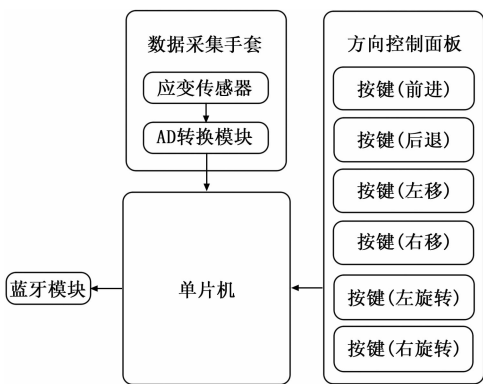


图 12 远程控制设备系统框图

手部动作采集手套上对应手指的部分有五条应变式弯曲传感器，手肘位置也贴有一条弯曲传感器。由公式  $R = \rho L / S$  可知金属阻值与材料、长度以及横截面积有关，取其微分形式可得  $dR = \rho / S dL + L / S d\rho - \rho L / S^2 dS$ 。故弯曲传感器在手指运动作用下产生的机械形变会导致其电阻阻值的变化，从而间接地影响流过金属的电流大小。应变式弯曲传感器实物如图 13 所示。

当测得弯曲传感器电阻值变化量为  $\Delta R$  时，经过测量电路获取电压变化值  $\Delta V$ ，将  $\Delta V$  的值与预先设置的参考值进行对比，来判断手指的弯曲程度。 $\Delta V$  经 AD 转换后传入单片机进行编码，最后通过蓝牙模块将编码信息发送至 DE1-SoC 部分的蓝牙接收端。



图 13 应变传感器实物图

## 2.3 实时 VR 成像实现方法

系统实时 VR 成像的实现框图如图 14 所示<sup>[6]</sup>，USB 摄像头获取图像信息传输至 DE1-SoC 开发平台的 HPS 部分，存入 DDR 内存。FPGA 部分通过 AXI bridge 访问 DDR 内存，调用图像数据进行 VR 成像处理，并将处理结果写回 DDR 内存，ARM 再从内存中读取处理完后得数据，发送到同一局域网内的安卓手机。为加快图像处理的速度，编写 OpenCL 加速逻辑进行图像处理操作。VR 头套内部安装有陀螺仪用于探测操作者头部的动作变化，头部的当前状态信息通过蓝牙模块传输至 DE1-SoC 开发平台，用于控制摄像头的转动。

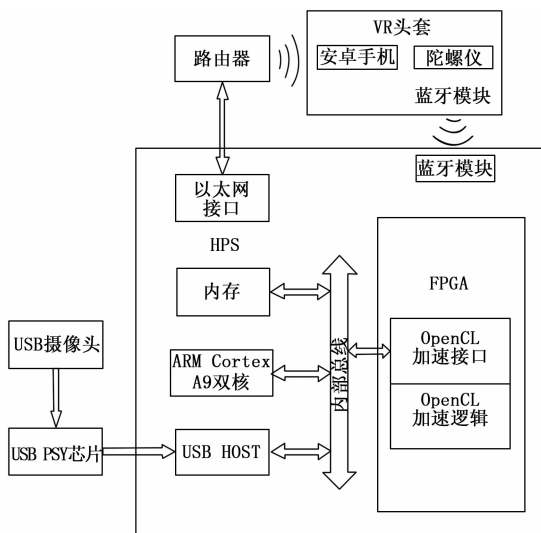


图 14 实时 VR 成像系统图

### 2.3.1 视频数据采集及处理

加速系统的软件设计主要是为加速内核填充图像数据，并进行双目化处理。通过对 ARM 的 Linux 系统内核重新编译，加入 USB 摄像头的驱动，然后编写应用程序，调用 USB 摄像头，获取实时图像，每帧图像的分辨率为 640 × 480，将此图像通过 AXI bridge 传输给 FPGA，在 FPGA 中运行 OpenCL 加速子系统，对图像进行处理。

在 OpenCL 加速子系统中，OpenCL 编译器将图像双目化处理内核算法编译为 OpenCL 加速逻辑连接到 OpenCL 接口上，并记录在二进制 OpenCL 内核文件中<sup>[7-8]</sup>。在加速处理图像数据时，调用 OpenCL 二进制文件更新 FPGA 逻辑。处理之后的图像通过 AXI bridge 存入 ARM 端的 DDR 内存

指定区间。ARM 处理器再从此内存区间内读取图像数据, 图像处理结果如图 15 所示。

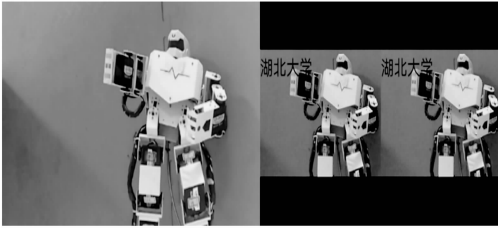


图 15 原图与处理后图像对比

### 2.3.2 VR 头套设计

本设计利用 I<sup>2</sup>C 总线协议驱动的 MPU-6050 陀螺仪可以通过测量加速度或者角速度来测量  $x$ 、 $y$ 、 $z$  轴三个方向的旋转角度。如图 16 所示, 以  $x$  轴正方向作为人眼的观察方向, 三个旋转角度分别为绕  $x$  轴旋转的滚转角 Roll、绕  $y$  轴旋转的俯仰角 Pitch 以及绕  $z$  轴旋转的偏航角 Yaw<sup>[9]</sup>。

测量 Roll 角、Pitch 角、Yaw 角时, 首先从相应寄存器地址读取绕三个方向旋转角速度 GYT\_X、GYT\_Y、GYR\_Z, 均以“度/秒”作为单位。以初始位置作为基准, 分别对时间进行积分, 从而求出在三个方向上与前一位置的相对旋转角度。陀螺仪滚转角、俯仰角、偏航角计算方法如下:

$$Roll = ((RollH \ll 8) | RollL) / 32768 * 180^\circ \quad (1)$$

$$Pitch = ((PitchH \ll 8) | PitchL) / 32768 * 180^\circ \quad (2)$$

$$Yaw = ((YawH \ll 8) | YawL) / 32768 * 180^\circ \quad (3)$$

式中, RollH, RollL, PitchH, PitchL, YawH, YawL 均可从相应寄存器读取。

如图 16 所示, 已知初始位置与相对旋转角度可以唯一确定当前人眼视线方向  $\vec{a}$  ( $x, y, z$ ), 结合三角公式可求得  $\vec{a}$  与  $xoy$  平面夹角为  $\alpha$ , 其在  $xoy$  平面内的投影 OH 与  $x$  轴夹角为  $\beta$ 。 $\alpha$ 、 $\beta$  的值即为控制 USB 摄像头的两个舵机分别对应的偏移角度<sup>[10]</sup>。根据其值的大小向舵机输出不同占空比的 PWM 脉冲信号驱动舵机转动。

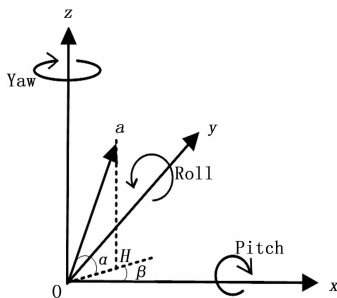


图 16 俯仰角、滚转角、偏航角分布图

DE1-SoC 的 ARM 部分通过有线网口连接到路由器, 安卓手机则通过无线热点连接路由器, 使两者同处一个局域网。ARM 端获取到安卓手机的 IP 后, 将实时获取到的经 OpenCL 加速系统处理之后的摄像头数据发送至此 IP。同时安卓手机端需安装自行开发用于接收 DE1-SoC 数据

的 APP, 此 APP 的功能即是将通过局域获取的图像信号通过安卓手机对其进行数字图像处理, 生成两幅相同的图像, 并呈现在同一副画面中。当人眼通过目镜同时对两幅图片进行观察时, 会产生立体视差角, 从而使人眼感受到立体的效果, 实现 VR 图像的成像。

## 3 实验结果与分析

### 3.1 六足底座步态测试

机器人步态的稳定性测试内容分为前进步态测试和旋转步态测试, 在机器人六足底座表面固定一个玻璃烧杯, 在其中加入 250 mL 自来水, 前进步态测试中控制机器人在平地上行走 10 米距离, 旋转步态测试中控制机器人原地旋转 10 圈。完成规定操作, 记录烧杯中液体的剩余量。液体剩余量越多, 表征该步态稳定性越高。如表 3、4 所示, 共有六种前进步态, 六种旋转步态, 每种步态进行 10 次测试取平均值, 记录测试结果如表 5 所示。

表 3 前进步态方案

	前进步态顺序			
	第一阶段	第二阶段	第三阶段	第四阶段
方案一	6 足支撑	6 号足摆动	4 号足摆动	2 号足摆动
方案二	空闲	6 号足摆动	4 号足摆动	2 号足摆动
方案三	空闲	6 号足摆动	4 号足摆动	2 号足摆动
方案四	6 足支撑	1 号足摆动	2 号足摆动	3 号足摆动
方案五	空闲	1 号足摆动	2 号足摆动	3 号足摆动
方案六	空闲	1 号足摆动	2 号足摆动	3 号足摆动
	前进步态顺序			
	第五阶段	第六阶段	第七阶段	第八阶段
方案一	6 足支撑	5 号足摆动	3 号足摆动	1 号足摆动
方案二	6 足支撑	5 号足摆动	3 号足摆动	1 号足摆动
方案三	空闲	5 号足摆动	3 号足摆动	1 号足摆动
方案四	6 足支撑	4 号足摆动	5 号足摆动	6 号足摆动
方案五	6 足支撑	4 号足摆动	5 号足摆动	6 号足摆动
方案六	空闲	4 号足摆动	5 号足摆动	6 号足摆动

表 4 旋转步态方案

	旋转步态顺序			
	第一阶段	第二阶段	第三阶段	第四阶段
方案一	6 足支撑	1 号足摆动	2 号足摆动	3 号足摆动
方案二	空闲	1 号足摆动	2 号足摆动	3 号足摆动
方案三	空闲	1 号足摆动	2 号足摆动	3 号足摆动
方案四	6 足支撑	1 号足摆动	3 号足摆动	5 号足摆动
方案五	空闲	1 号足摆动	3 号足摆动	5 号足摆动
方案六	空闲	1 号足摆动	3 号足摆动	5 号足摆动
	旋转步态顺序			
	第五阶段	第六阶段	第七阶段	第八阶段
方案一	6 足支撑	4 号足摆动	5 号足摆动	6 号足摆动
方案二	6 足支撑	4 号足摆动	5 号足摆动	6 号足摆动
方案三	空闲	4 号足摆动	5 号足摆动	6 号足摆动
方案四	6 足支撑	2 号足摆动	4 号足摆动	6 号足摆动
方案五	6 足支撑	2 号足摆动	4 号足摆动	6 号足摆动
方案六	空闲	2 号足摆动	4 号足摆动	6 号足摆动

表 5 机器人步态测试结果

方案编号	平均剩余量/(mL)	
	前进步态	旋转步态
方案一	226	235
方案二	197	189
方案三	154	180
方案四	201	198
方案五	169	167
方案六	132	179

测试结果可得出, 前进步态的方案一与旋转步态的方案一稳定性均相对其他步态稳定性有所提高。故在实际设计中前进与旋转的步态均采用方案一, 来对步态进行规划。

### 3.2 机械臂控制功能测试

为测试机械手臂的控制功能, 操作者穿戴远程控制手套, 在 10 秒钟内连续做出 10 个不同的表示数字的手势。观察机械手臂是否可以同步跟随操作者的手势。进行的 10 次测试结果如表 6 所示。

表 6 机械臂控制测试结果

编号	机械臂跟随测试指标	
	成功跟随次数	成功跟随比例
1	10	100%
2	9	90%
3	10	100%
4	9	90%
5	10	100%
6	8	80%
7	10	100%
8	10	100%
9	9	90%
10	10	100%
平均值	9.5	95%

由测试数据可以得出, 机械手臂跟随操作者的成功比例高达 95%, 实现了对机械臂的基本控制与操作。

### 3.3 视频传输功能测试

机器人视频传输测试方案为, 在测试房间的地板以及墙面上粘贴 10 个醒目的数字标识。操作者佩戴视频数据采集头套, 站立于房间中心位置。操作者需从数字 1 开始依次寻找目标, 当操作者通过视频采集头套观察到标记时, 取下头套观察实际情景是否与 VR 成像相同。若基本一致记录成功, 若两者差距较大则视为失败。记录观察结果后, 开始寻找下一个数字标记。每 10 个数字标记为一轮测试, 共进行 10 轮测试, 测试结果如表 7 所示。

由测试结果可知, 在视频传输测试中 VR 成像的平均正确率高达 96%, 基本实现机器人的视频传输功能。

### 3.4 复杂地形作业测试与分析

为判定机器人在复杂地形中是否可以稳定行走并准确抓取物品, 选取室内楼梯作为测试场地。楼梯标准为, 踏步高度: 150 mm, 每级台阶宽度: 250 mm, 级数: 10 级。

表 7 视频传输测试结果

编号	视频传输测试指标	
	成像正确次数	成像准确率
1	9	90%
2	10	100%
3	10	100%
4	9	90%
5	10	100%
6	10	100%
7	9	90%
8	9	90%
9	10	100%
10	10	100%
平均值	9.6	96%

测试内容设置为, 操作者佩戴 VR 头套与远程控制设备, 从开始操纵六足仿生机器人开始计时。机器人从楼梯底部出发, 攀爬过程中, 需准确抓取置于楼梯中部的水瓶, 并将其运送至楼梯顶部平稳放置, 期间若水瓶滑落则视为测试失败。进行 50 次重复试验, 并记录完成时间。测试结果如图 17 所示。

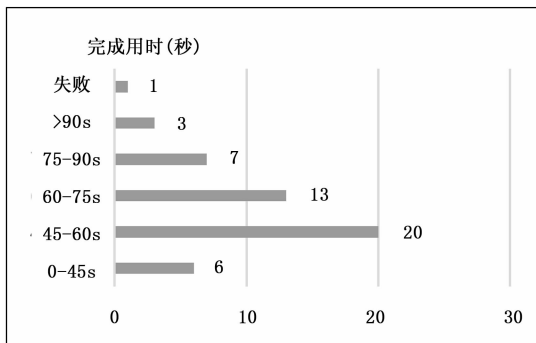


图 17 楼梯任务完成时间分布图

从测试结果可得出, 机器人 50 次完成任务的用时绝大多数分布在 45~60 s 区间, 从 45 s 开始, 时间越长所分布的次数越少。在第 26 次试验过程中, 因机器人重心不稳, 导致主体大幅度晃动, 致使水瓶摔落。故仍需优化机器人在攀爬高倾斜度地形时的步态设计。综上所述, 测试结果已满足项目所设计的测试指标。机器人具备了在复杂地形中作业的能力与较高的操作稳定性。

## 4 结束语

本文充分利用 FPGA 与 ARM 各自的优势, 基于 SoC FPGA 平台实现了一套集实时图像传输、行进路线遥控、仿生手臂跟随人手实时运动等多功能于一体的机器人。参考蜘蛛模型构造六足底座, 并对其行驶过程中的步态进行仿生设计, 能够在大多数复杂地形中畅通行驶。同时, 参照人类手臂与手掌构造机械手模型, 可以灵活抓取物品。将方向控制面板与应变式传感器进行结合, 制作出一个可感知手部动作和发送行进指令的操作手套, 实现了对机器人

(下转第 270 页)