

淡江大學電機工程學系  
系統組  
專題實驗報告

人形機器人籃球投籃競賽  
策略之設計

指導教授： 劉智誠 博士  
研究生： 周承緯  
專題生： 洪致中、李長鑫、任庭宇

中華民國 111 年 10 月

# 目錄

目錄 .....	ii
圖目錄 .....	iv
表目錄 .....	vi
第 1 章、序論 .....	1
1.1、研究背景與動機 .....	1
1.2、本專題之架構 .....	1
第 2 章、人形機器人平台 .....	2
2.1、人形機器人之規格 .....	2
2.1.1、機構 .....	2
2.1.2、機構強化 .....	5
2.1.2、伺服機 .....	8
2.2、人形機器人之運動控制板 .....	10
2.2.1、IPC .....	10
2.2.2、FPGA .....	11
2.2.3、轉接電路板 .....	13
2.2.4、腳電開關 .....	16
1. 2.2.5、感測器 .....	19
第 3 章、Gazebo 模擬平台 .....	21
3.1、Gazebo 模擬平台介紹 .....	21
3.1.1、Gazebo 模擬平台優點 .....	21
3.2、物理模擬引擎 .....	21
3.2.1、動力學模擬 .....	21
3.2.2、背景環境擬真 .....	21
3.2.3、3D 圖形 .....	22
3.2.3、機器人模型 .....	23
3.2.4、感測器插件 .....	23
3.2.5、可使用命令行工具執行程式 .....	23
3.3、物件模型設計介紹 .....	23
3.3.1、SolidWorks .....	24
3.3.3、urdf 檔案格式 .....	25
3.3.4、DAE 檔案格式 .....	25
第 4 章、人形機器人之系統架構 .....	26
4.1、人形機器人作業系統之簡介 .....	26
第 5 章、人形機器人之人機介面 .....	28
5.1、影像介面 .....	28
5.2、步態介面 .....	29

5.3、動作介面.....	30
第 6 章、人形機器人之比賽行為決策.....	31
6.1、籃球競賽之比賽規則 .....	31
6.2、籃球競賽之策略介紹 .....	32
6.2.1、籃球競賽之投籃策略介紹 .....	32
6.3、籃球競賽之上籃策略介紹 .....	40
6.4.0、籃球競賽之夾球動作介紹 .....	42
第 7 章、結論 .....	46

## 圖目錄

圖 2.1、頭部機構正視圖 .....	2
圖 2.2、頭部機構側視圖 .....	3
圖 2.3、手部機構正視圖 .....	3
圖 2.4、手部機構側視圖 .....	3
圖 2.5、身體正視圖 .....	4
圖 2.6、身體後視圖 .....	4
圖 2.7、身體側視圖 .....	4
圖 2.8、腳部機構正視圖 .....	4
圖 2.9、腳部機構側視圖 .....	4
圖 2.10、卡頭片 .....	5
圖 2.11、鏡頭片支撐 .....	5
圖 2.12、更改後肩部 .....	5
圖 2.13、肩部實裝圖 .....	5
圖 2.14、箱扣 .....	6
圖 2.15、滾珠軸承套件 .....	6
圖 2.16、腰部 .....	6
圖 2.17、IMU 固定座正視圖 .....	6
圖 2.18、IMU 固定座側視圖 .....	7
圖 2.19、IMU 固定座後視圖 .....	7
圖 2.20、雙腿機構實裝 .....	7
圖 2.21、腳踝口(同大腿口).....	7
圖 2.22、十一代人形機器人之伺服機種類分布圖 .....	8
圖 2.23、轉接電路板(a)正面與(b)反面 .....	13
圖 2.24、手、腳伺服機控制訊號圖 .....	14
圖 2.25、手、腳伺服機回授訊號圖 .....	14
圖 2.26、頭部伺服機訊號圖 .....	15
圖 2.27、IMU 訊號圖 .....	15
圖 2.28、腳電開關:(a)舊版與(b)新版 .....	16
圖 2.29、插座位置:(a)舊版插座位置(正面)與(b)新版插座位置(背面) .....	16
圖 2.30、電路圖:(a)舊版電路圖與(b)新版電路圖 .....	17
圖 2.31、二極體(D1)、(D2) .....	17
圖 2.32、電容(C13)、(C14).....	18
圖 2.33、電源與電容並聯電路 .....	18
圖 2.34、大(紅)、小(藍)指撥以及洩力開關(黃).....	18
圖 3.1、Gazebo 模型匯入介面 .....	22
圖 3.2、於 Gazebo 執行機器人模擬 .....	24

圖 4.1、ROS 的系統架構.....	27
圖 4.2、ROS 的傳輸方式.....	27
圖 5.1、影像介面說明圖 .....	28
圖 5.2、步態介面說明圖 .....	29
圖 5.3、動作介面說明圖 .....	30
圖 6.1、籃球競賽場地圖 .....	31
圖 6.2、人形機器人視野搜尋方向圖 .....	32
圖 6.3、頭部伺服馬達之轉動圖(a)(b).....	32
圖 6.4、人形機器人之遠近示意圖 .....	33
圖 6.5、預備夾球動作(a)(b)(c)(d) .....	34
圖 6.6、夾球動作(a)(b)(c) .....	35
圖 6.7、開始找框 .....	35
圖 6.8、追蹤到框 .....	36
圖 6.9、遠近對比圖 .....	37
圖 6.10、大小對比圖 .....	37
圖 6.11、上籃減速線示意圖 .....	40
圖 6.12、上籃策略追蹤籃框流程圖 .....	41
圖 6.13、蹲下與舉手 .....	42
圖 6.14、手下降、夾球、轉手與站立持球 .....	42
圖 6.15、投籃舉手動作 .....	43
圖 6.16、投籃(a)(b)(c).....	43
圖 6.17、上籃舉手動作(a)(b)(c)(d).....	44
圖 6.18、上籃動作(a)(b) .....	45

## 表目錄

表 2.1、伺服器規格表 .....	9
表 2.1、IPC 規格表 .....	10
表 2.1、FPGA 規格表 .....	12
表 2.1、攝影機規格表 .....	19
表 2.1、IMU 規格表 .....	20

# 第1章、序論

## 1.1、研究背景與動機

本專題主要討論「國際智慧機器運動大賽:人形機器人組(HuroCup)」籃球競賽之策略流程，並介紹第十一代人形機器人的設計以及操作。如何使人形機器人在比賽場地中知道球與籃框之位置且可以將球投進籃框，使用攝影機來擷取影像，再藉由顏色特徵分辨出球與籃框，讓人形機器人對環境進行感知，最後利用面積測距法以及權重計算投籃力道，使機器人可以自我定位並將球順利投進籃框。

## 1.2、本專題之架構

本專題共分為六個章節，第十一代人形機器人是以視覺影像作為判斷，進行自主運動的機器人。第一章作為序論。說明本專題文件之架構與內容;第二章為介紹人形機器人機構設計與運動控制板之應用;第三章為講解人形機器人之 Gazebo 平台應用;第四章為人形機器人的系統架構，介紹人形機器人使用的 ROS 系統;第五章為人形機器人之人機介面的設計，詳細講解人機介面的功能以及其使用方式;第六章為「國際智慧機器運動大賽:人形機器人組(HuroCup)」的相關說明，從比賽規則及流程到為人形機器人編寫的程式策略和動作設計等都會在此章節說明;第七章為結論。

# 第2章、人形機器人平台

## 2.1、人形機器人之規格

### 2.1.1、機構

本專題在機構方面是以人形機器人執行動作穩定性、輕量化以及發揮鋁合金件最大剛性為目標，並且模仿了人體的骨架結構，在伺服機之間透過自行設計的機構將彼此固定住，而伺服機轉動處則設為人形機器人之可動關節，人形機器人的核心電路裝置在人形軀幹的部分，裝上外殼後可以增加對核心電路的保護性。以下將人形機器人之機構設計分成四部分詳細介紹：(1)頭部機構、(2)手部機構、(3)身體機構以及(4)腳部機構。

#### (1)頭部機構

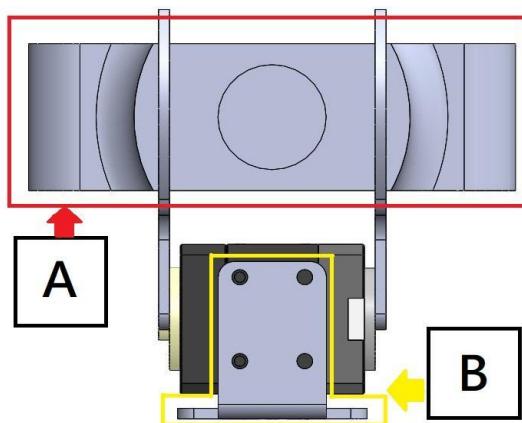


圖 2.1、頭部機構正視圖

(A)攝影機、(B)攝影機座

頭部機構是由 3D printer 所建構完成，如圖 2.1 所示，將攝影機架在設計好的機構上，來使頭部可以做 x、y 軸之旋轉控制。

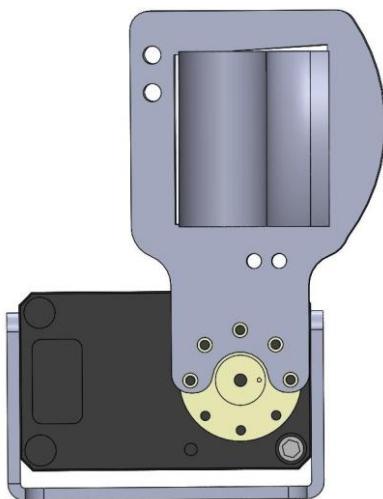


圖 2.2、頭部機構側視圖

## (2)手部機構

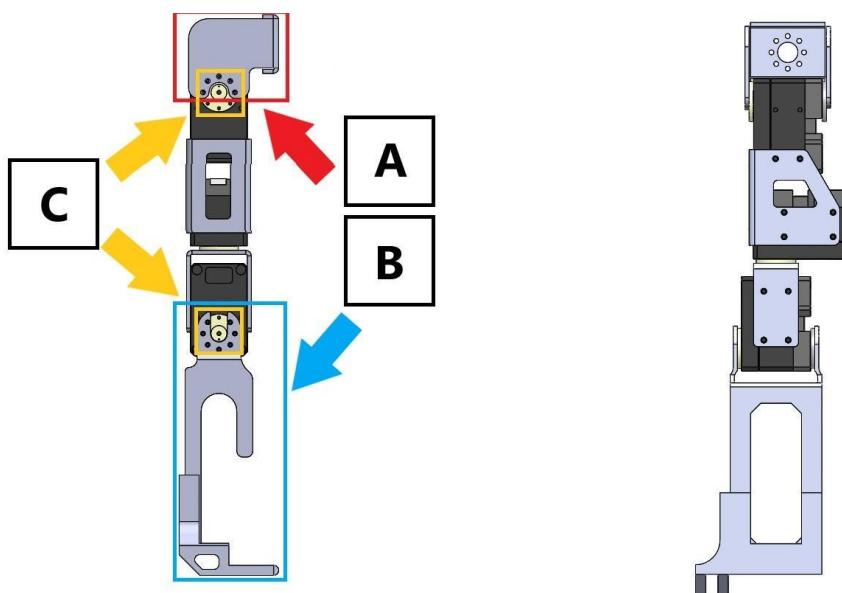


圖 2.3、手部機構正視圖

圖 2.4、手部機構側視圖

(A)肩膀關節、(B)手爪、(C)轉動關節

### (3)身體機構

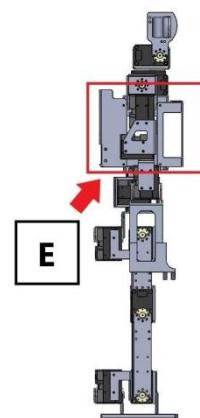
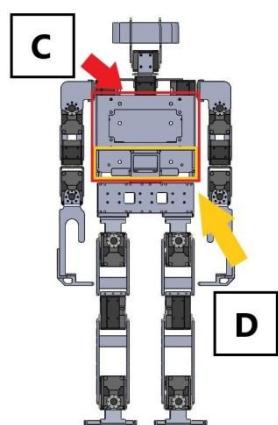
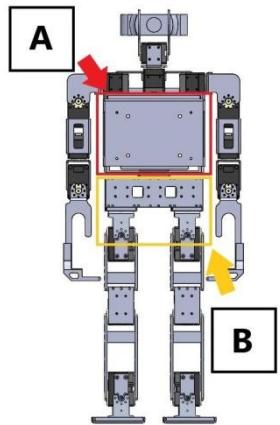


圖 2.5、身體正視圖

圖 2.6、身體後視圖

圖 2.7、身體側視圖

(A)前蓋(前外殼)、(B)髋關節及腰部、(C)後蓋(後外殼)

(D)腳電池放置處、(E)電路板放置處及走線空間

### (4)腳部機構

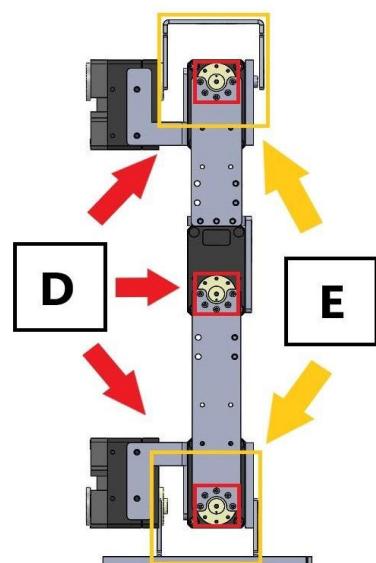
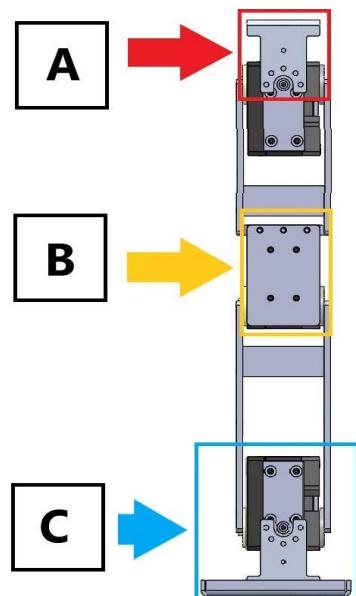


圖 2.8、腳部機構正視圖

圖 2.9、腳部機構側視圖

(A)髋關節、(B)膝關節、(C)踝關節、(D)轉動關節、(E)預留轉動空間

## 2.1.2、機構強化

### (1)頭部機構強化

藉由分析歷代人形機器人機構上的優缺點，第十一代人形機器人做了一些改良。首先在頭部的部分更改了卡頭片的挖洞尺寸，更符合鏡頭外觀，避免有間隙搖晃。以及鏡頭片支撐(卡頭片中間支撐)，之前的是板金折件，公差大，現在修改成一體成形，較無誤差疑慮。

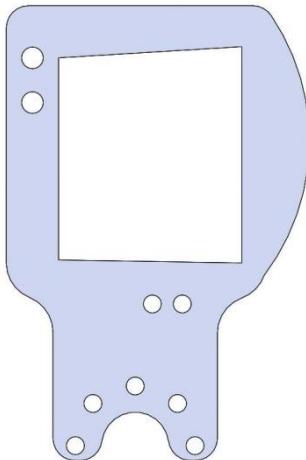


圖 2.10、卡頭片

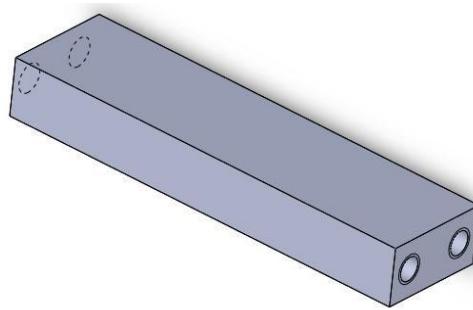


圖 2.11、鏡頭片支撐

### (2)手部機構強化

由於修改後的 IMU 固定座較先前寬，因此將肩部機構加寬(肩部、左肩、右肩)。

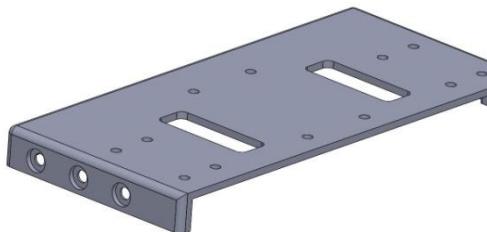


圖 2.12、更改後肩部

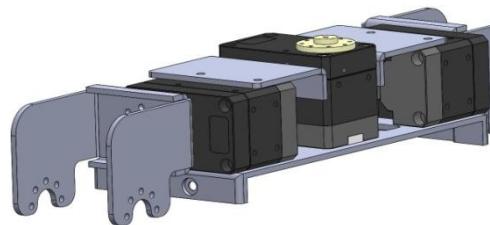


圖 2.13、肩部實裝圖

### (3)身體機構強化

首先是電池擺放固定機構改成箱扣，舊的是用兩顆螺絲跟機構，機構跟螺絲容易不見。再來是軸承套件，這樣的設計可以避免伺服機承受旋轉以外的受力，延長伺服機齒隙出現時間，以及腰部結構改成四片組合，用以增加結構強度。

IMU 固定座則是在圖 2.17(A)處墊高 IMU，避免其因下方焊點凸起影響，且於(B)處增厚，鎖轉接電路板時較不易滑牙。以及於 (C)處擋電池的地方加厚，避免變形以及斷裂。最後是圖 2.19(D)處凹槽配合身體下蓋凸起四支支撐，可以有效防止 PLA 因螺絲鎖緊而斷裂。

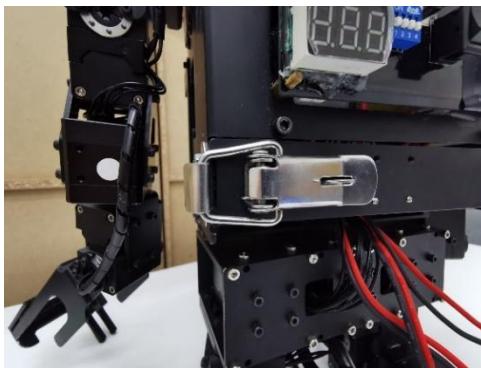


圖 2.14、箱扣

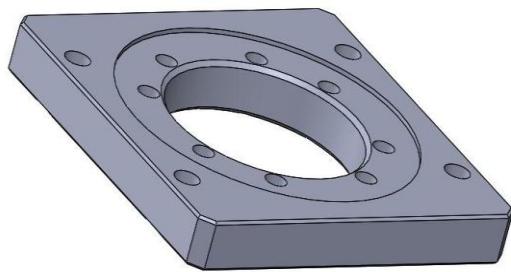


圖 2.15、滾珠軸承套件

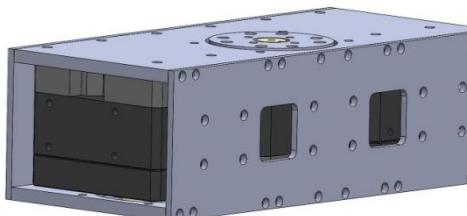


圖 2.16、腰部

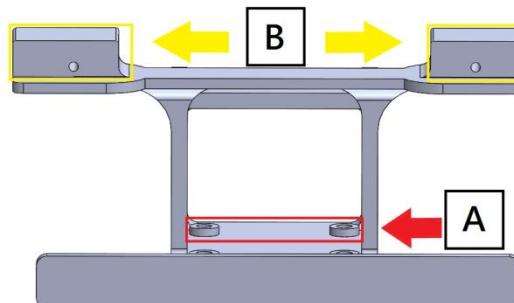


圖 2.17、IMU 固定座正視圖

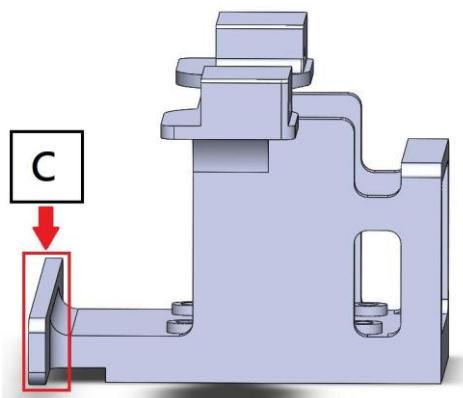


圖 2.18、IMU 固定座

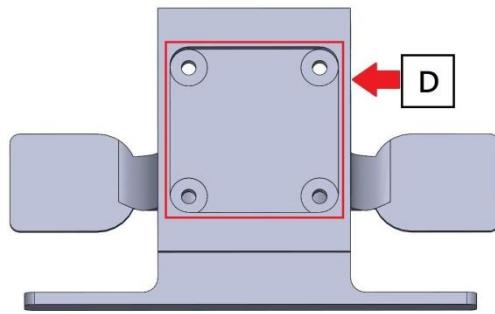


圖 2.19、IMU 固定座

#### (4)腳部機構改良

首先是原先腿部機構採用板金折件，現今改良 CNC 組合件，可達到精度、鋼性較佳的結果，且搭配螺絲膠使用，可減少因零件鬆動而造成需要調整人形機器人的機會。並且於大腿ㄇ字機構，取消兩邊凸起處，使腳部的可動範圍更廣，上方十字開孔為維修用。

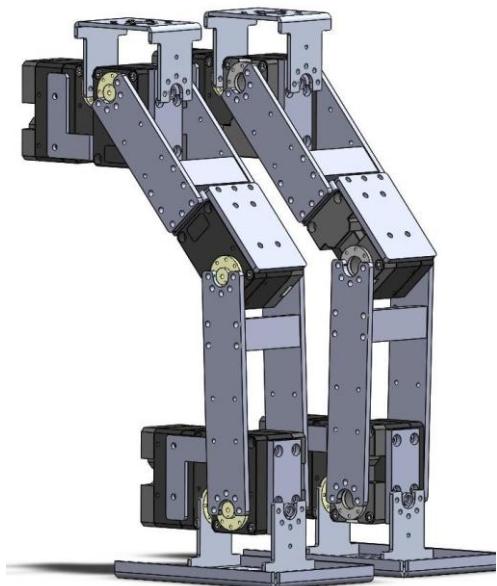


圖 2.20、雙腿機構實裝

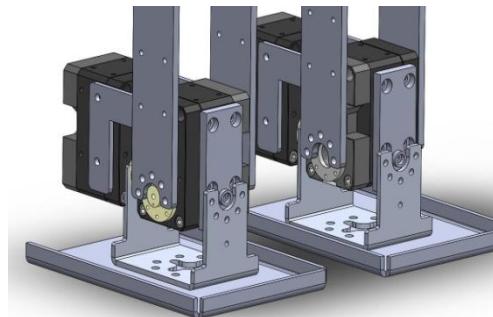


圖 2.21、腳踝ㄇ(同大腿ㄇ)

## 2.1.2、伺服機

本專題使用之第十一代人形機器人，主要結構可以分為頭部、手部以及腳部等三個部位。在伺服機之配置上，腳部使用之伺服機為(1)XH540-W270，如圖 2.22 黃色框線所示，頭部及部分手部使用之伺服機為(2)XM430-W350，如圖 2.22 紅色框線所示，部分手部使用之伺服機為(3)XM430-W210，如圖 2.22 藍色框線所示。

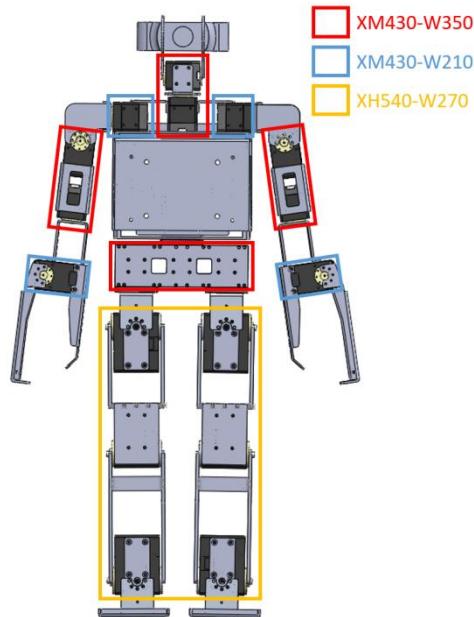


圖 2.22、十一代人形機器人之伺服機種類分布圖

- (1) XH540-W270 則為扭矩大、轉速較慢，因為它必須要支撐起整隻人形機器人，所以選用大扭矩的伺服機，速度相對慢但應用在走路上也已足夠，詳細規格如表2.1 所示。
- (2) XM430-W350 則為扭矩、轉速較適中，應用於上半身受力更高之關節，應用在拉弓等不需要速度的動作同時，速度適中的特性使他足以應付競賽需求。
- (3) XM430-W210 有扭矩小、轉速高的特性，雖然扭矩相對較小，但已足夠應付爬梯及趴地爬起等動作，速度快則可因應在投籃的動作。

表 2.1、伺服器規格表

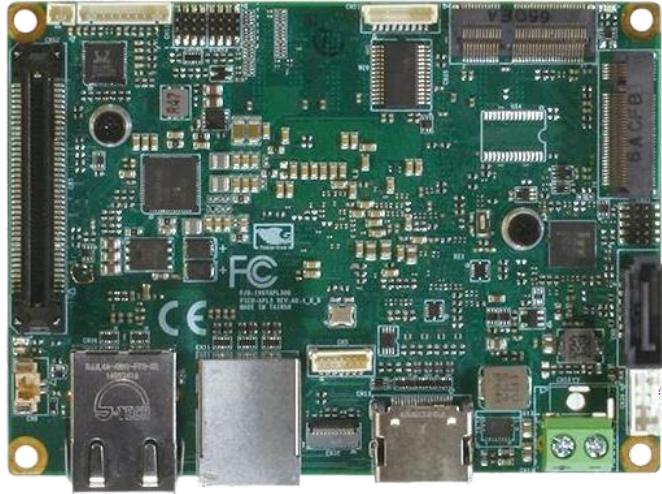
	(a)XM430-W350	(b)XM430-W210	(c)XH540-W270
實體圖			
外型尺寸	28.5mm x 46.5mm x 34mm	28.5mm x 46.5mm x 34mm	33.5mm x 58.5mm x 44mm
扭力	3.8Nm(11.1V) 4.1Nm(12.0V) 4.8Nm(14.8V)	2.7Nm(11.1V) 3.0Nm(12.0V) 3.7Nm(14.8V)	9.2Nm(11.1V) 9.9Nm(12.0V) 11.7Nm(14.8V)
齒輪比	353.5 : 1	212.6 : 1	272.5 : 1
速度	43~57rpm	70~95rpm	36~46rpm
重量	82g	82g	165g
工作電壓	10~14.8V	10~14.8V	10~14.8V
可動角度	360°	360°	360°
使用部分	頭、腰、手	手	腳

## 2.2、人形機器人之運動控制板

### 2.2.1、IPC

本專題之人形機器人之核心板為 PICO-APL3 工業電腦(Industrial PC, IPC)，詳細規格如表 2.2 所示。其中包含了處理器、記憶體、USB2.0、USB3.0 及 RS-232 port 等基本裝置，主要是用來進行影像處理與運動學(Kinematics)末端點傳送。

表 2.2、IPC 規格表

	PICO-APL3 工業電腦
實體圖	
CPU	Intel® Pentium® N4200(2.4GHz)
System Memory	Onboard DDR3L 4GB
Com Port	COM1 : RS-232、COM2 : RS-232
Power Requirement	Single 12V DC Power Input
Size	3.94" x 2.84"(100mm x 72mm)

## 2.2.2、FPGA

本專題之人形機器人使用 FPGA DE10-Nano Kit 當作運動控制板，詳細規格如表2.3所示。其中 FPGA 應用包含了處理器、記憶體、序列配置器件以及 GPIO 等基本裝置，並透過 GPIO 腳位與伺服機及 IMU 進行訊號傳輸，進而控制伺服機及得到感測器資訊。除了處理周邊硬體外，也負責人形機器人之雙腳逆運動學(Inverse kinematics)計算。

人形機器人頭部、手部以及腳部伺服機訊號格式為 RS-485 訊號，而 FPGA 輸出訊號格式為 TTL，因此控制伺服機之訊號會透過轉接電路板做訊號轉換。

IMU 所使用的訊號格式為 I2C，與 FPGA 直接進行訊號傳輸，再將 IMU 測量之角速度與加速度經過卡爾曼濾波(Kalman filter)後，得到人形機器人之姿態資訊。

IPC 將人形機器人之行走軌跡點經由轉接電路板連接至 FPGA 之 GPIO 腳位，直接將行走軌跡點傳送給 FPGA，由 FPGA 進行加速運算，用以分擔 IPC 運算量，且經由逆運動學(Inverse kinematics)方式推算出每顆伺服機需轉動之刻度，進而控制人形機器人之伺服機。

表 2.3、FPGA 開發板規格表

	FPGA DE10-Nano Kit
實體圖	
CPU	Intel Cyclone® V SE 5CSEBA6U23I7NDK(110K LEs)
HPS	Dual-core ARM Cortex-A9 (HPS)
SDRAM	1GB DDR3 SDRAM (32-bit data)
Logic Gates	110KLE
Serial Configuration Devices	EPCS128
GPIO	2x40-pin
Power Requirement	5V DC power adapter
Size	68.6x107mm

### 2.2.3、轉接電路板

為了供應 IPC 和伺服機所需的電源以及處理 IPC 與伺服機之間訊號轉換問題，因而設計此款轉接電路板。此轉接電路包含了12V轉5V的降壓模組和轉換 IPC 與伺服機所需訊號格式的電路設計，轉接電路板實體圖如圖2.23 所示。



圖 2.23、轉接電路板(a)正面與(b)反面

其中伺服機所使用的訊號格式為 RS-485 訊號，IPC 所輸出的為 RS-232 及 USB 訊號，FPGA 所輸出的訊號格式為 TTL 訊號，IMU 使用 I2C 訊號。以下將以從 IPC 所輸出的四種訊號介紹：(1)手、腳伺服機控制訊號、(2)手、腳伺服機回授訊號、(3)頭部伺服機訊號以及(4)IMU 訊號，分別說明如下：

#### (1)手、腳伺服機控制訊號

轉接電路板先將 IPC 的 USB 訊號經由 FT232RL 晶片轉換為 TTL 訊號，再經由 LVC245 晶片的 TTL 訊號給 FPGA 使用，而 FPGA 輸出後的 TTL 訊號會經由 SP-485 晶片的 RS-485 訊號以供應伺服機所使用，轉換過程如圖 2.27 所示。

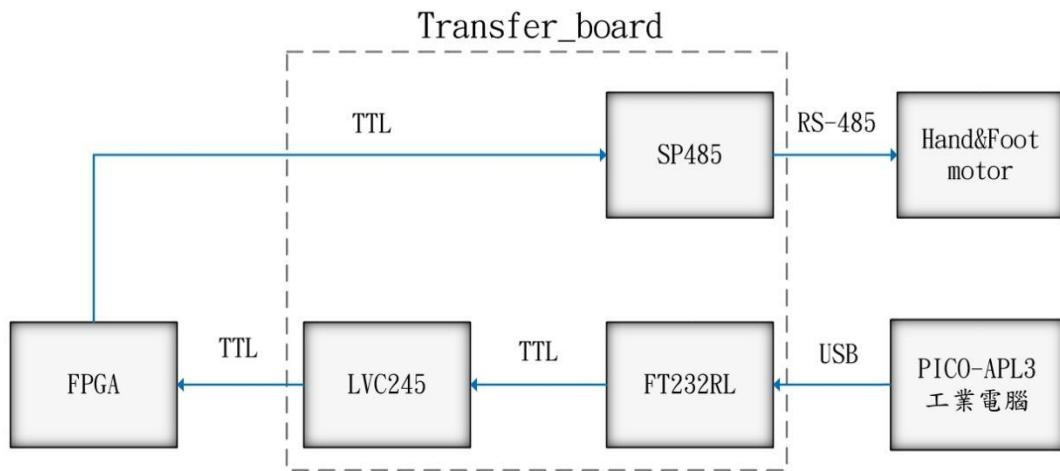


圖 2.24、手、腳伺服機控制訊號圖

### (2)手、腳伺服機回授訊號

轉接電路板會先將伺服機的 RS-485 訊號經由 SP-485 晶片轉換為 TTL 訊號，再經由 FPGA 的 TTL 訊號傳回 LVC245 晶片，而 LVC245 晶片輸出後的 TTL 訊號會經由 FT232RL 轉換成 USB 訊號傳回 IPC，轉換過程如圖 2.28 所示。

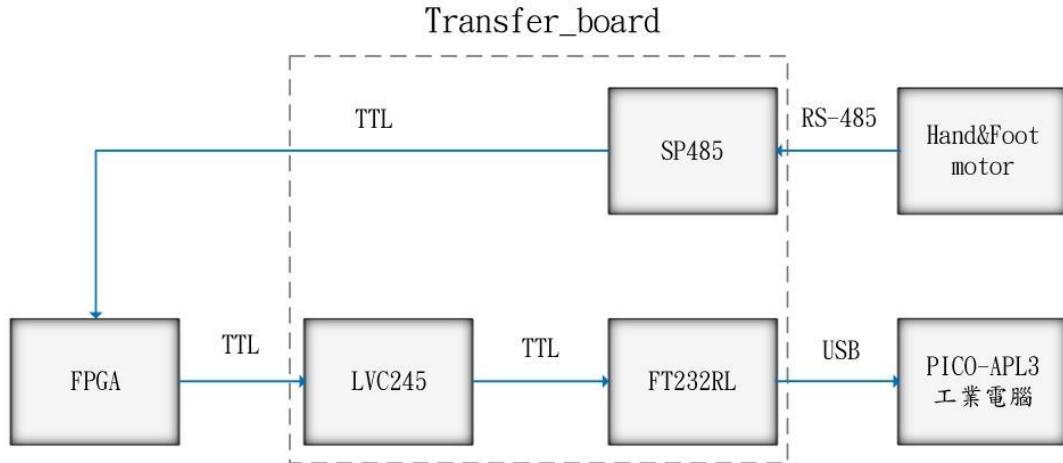


圖 2.25、手、腳伺服機回授訊號圖

### (3)頭部伺服機訊號

轉接電路板會先將 IPC 的 RS-232 訊號經由 SP232 晶片轉換成 TTL 訊號，再經由 LVC245 輸出後的 TTL 訊號給 FPGA 使用，而 FPGA 輸出後的 TTL 訊號會經由 SP-485 晶片的 RS-485 訊號以供應頭部伺服機所使用，轉換過程如圖 2.29 所示。

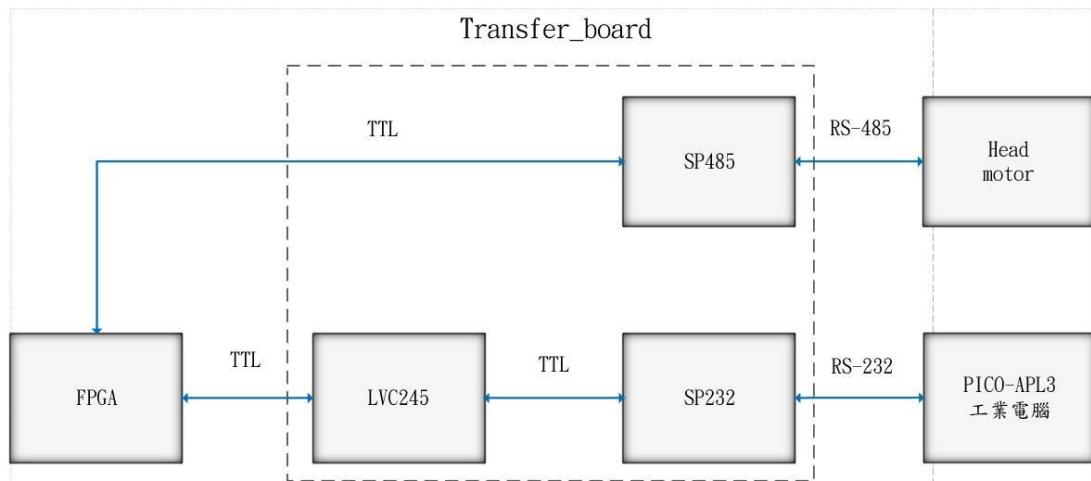


圖 2.26、頭部伺服機訊號圖

#### (4)IMU 訊號

轉接電路板會先將 IPC 的 RS-232 訊號經由 SP232 晶片轉換成 TTL 訊號，再經由 LVC245 輸出後的 TTL 訊號給 FPGA 使用，最後再經由 FPGA 所輸出的 I2C 訊號到 IMU 上，其中 IPC 的 RS-232 訊號經 SP232 轉 TTL 訊號用在 IMU 資訊和頭部伺服機的資訊，轉換過程如圖 2.30 所示。

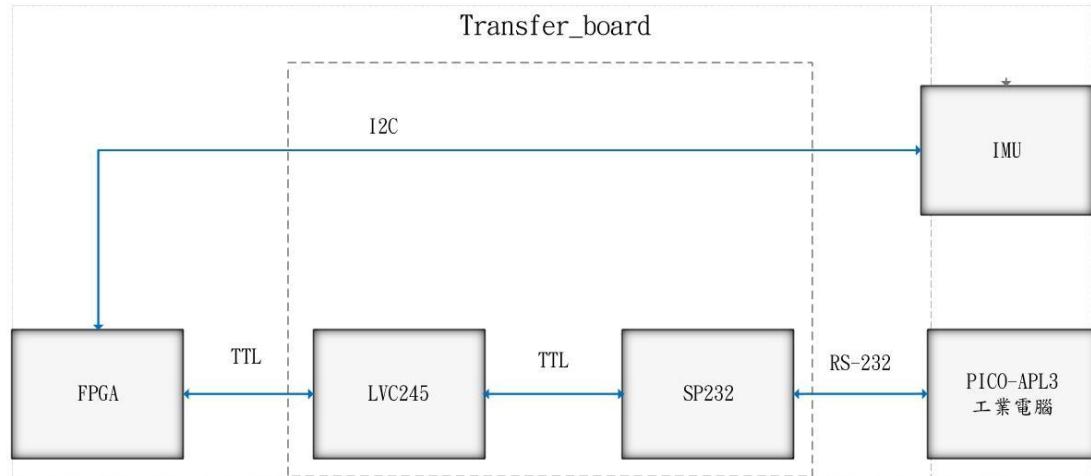


圖 2.27、IMU 訊號圖

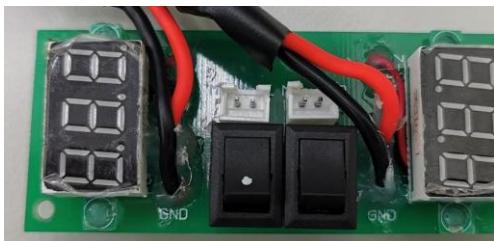
為提升轉接電路板的效率，降低能源的損耗，以下為新版轉接電路板之改良如下：(1)將電源合併模組移至腳電開關、(2)取消 Arduino 電路，分別說明如下：

(1) 轉接電路板上之電源合併模組，新版將其移至腳電開關，達到節省走線的效果。

(2) 可使用新的 IMU 上 Arduino 進行計算，因此可以取消轉接電路板的電路，降低了轉接電路板的耗電量以及減少其耗電量。

## 2.2.4、腳電開關

腳電開關部件主要用於人形機器人伺服機供電開關做使用，配合七段顯示器，方便使用者快速控制伺服機電源，以下為新版腳電開關之改良：(1)插座移至後方、(2)雙電源合併電路、(3)增加穩壓電路以及(4)合併洩力開關，分別說明如下：



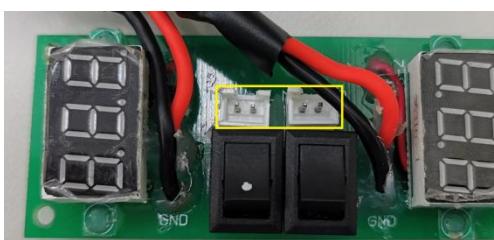
(a)舊版



(b)新版

圖 2.28、腳電開關(a)舊版與(b)新版

### (1) 插座移至後方



(a)舊版插座位置(正面)

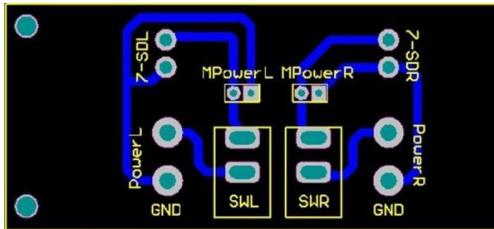


(b)新版插座位置(背面)

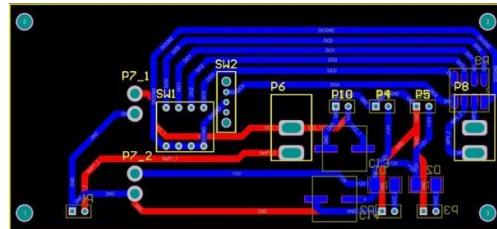
圖 2.29、插座位置(a)舊版插座位置(正面)與(b)新版插座位置(背面)

## (2) 雙電源合併電路

舊版腳電開關之電源電路位於轉接電路板上，新版將其整合，達到節省走線的效果，如圖 2.35 所示。



(a)舊版電路圖



(b)新版電路圖

圖 2.30、電路圖(a)舊版電路圖與(b)新版電路圖

由於合併雙電源，為避免兩顆電池互相影響，加入二極體(D1)、(D2)以防止兩顆電池互相充電，如圖 2.36 所示。

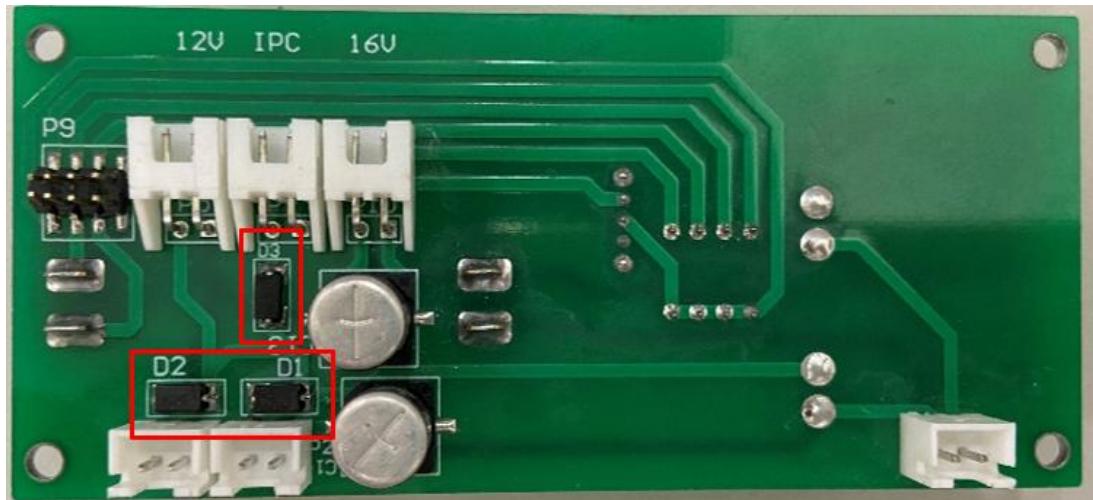


圖 2.31、二極體(D1)、(D2)

### (3) 增加穩壓電路

新版電路上不僅保留原有的穩壓電容，更新增了伺服機電源的穩壓電容，讓電路電壓不穩定時可以達到保持電壓穩定的效果，如圖 2.37、2.38 所示。

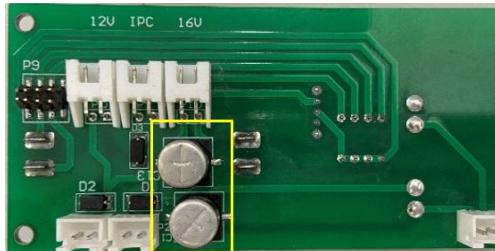


圖 2.32、電容(C13)、(C14)

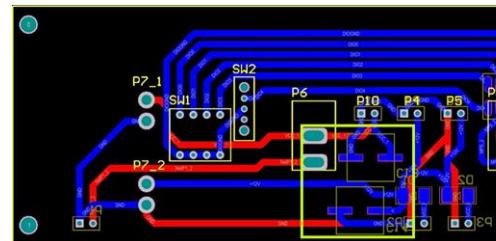


圖 2.33、電源與電容並聯電路

### (4) 合併洩力開關

新版腳電開關將洩力開關以及大小指撥一併整合在一起，使背蓋電路板剩一塊，在維修更換上或是操作者使用起來方便許多，如圖 2.39 所示。



圖 2.34、大(紅)、小(藍)指撥以及洩力開關(黃)

## 1. 2.2.5、感測器

為了使人形機器人更符合人類設計，因此增加感測器來使人形機器人與人類更相像，在第十一代人形機器人中，使用之感測器有攝影機與慣性量測單元( Inertial Measurement Uni, IMU)。

人形機器人在執行策略時需要瞄準目標物或避開障礙物等動作，因此使用幀數較高的C930E高清網路攝影機，詳細規格如表2.4 所示。能夠讓人形機器人在影像處理上更清晰、準確，此攝影機亦具有90°的廣角，使人形機器人的視野更寬廣，清楚掌握前方的情況。

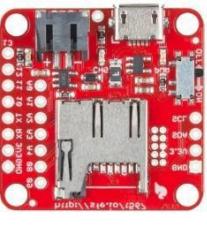
表 2.4、攝影機規格表

C930E 高清網路攝影機	
實體圖	
外型尺寸	43 * 94 * 71 mm <sup>3</sup>
最大解析埠	1080p/30fps- 720p/60fps
對焦類型	自動對焦
視野	90°

IMU 是使用 SparkFun 9DoF Razor 九軸多功能 IMU，詳細規格如表 2.5 所示。它在單個 IC 中安裝了一個三軸加速度計，三軸陀螺儀和三軸磁力計。LSM9DS1 配備了數字接口，使用上可以非常靈活。它支持 I2C 和 SPI，因此很難找到不兼容的微控制器。而 IMU 在人形機器人身上，可藉由加速計回傳值進行修正控制，也使人形機器人在行走間有不穩時，可藉由 IMU 進行感測，並給予適當的修正量。

此 IMU 與之前的 GY-87 差異為通信方式更多元、靈敏度更高，以及基準點更準確，較不會因移動後有所誤差，比較如表2.5 所示。

表 2.5、IMU 規格表

	10DOF 多功能感測器模組	SparkFun 9DoF Razor
實體圖	 <p>GY-87</p>	
外型尺寸	2.2*1.7cm	3.2*3.2 cm
電源供應	3-5V	3.7-4.2 V
通信方式	I2C	SPI/I2C
使用晶片	MPU6050、HMC5883L、BMP180	SAMD21、MPU-9250

# 第3章、 Gazebo 模擬平台

## 3.1、 Gazebo 模擬平台介紹

Gazebo 是一個開源機器人模擬平台，具有模擬三維物理實驗環境的功能，藉由添加物件建構現實世界的各種場景；除了能夠使用傳感器模型庫，也能夠自行創建新的傳感器；擁有仿真模擬引擎，可為機器人增加重力、摩擦力，實現更為逼真的模擬環境。我們將利用它模擬各個競賽項目的環境，藉此有效地模擬機器人會遇到的實際環境。

### 3.1.1、 Gazebo 模擬平台優點

Gazebo 其中一個優點是與機器人作業系統(Robot Poerating System, ROS)皆由開源機器人組織 (Open Source Robotics Foundation, OSRF) 維護，在操作上能夠良好地配合。它支援多個高性能物理引擎，其中最為典型的 ODE 可模擬機器人的運動學、動力學、雷射雷達、攝像頭、IMU 等感測器，也使用OGRE 呈現 3D 畫面，透過高質量的照明、陰影和紋理，讓環境被渲染得更為逼真，且環境建構迅速，易於更動。

## 3.2、 物理模擬引擎

Gazebo 模擬引擎具有許多特徵，可分為以下六部分：(a)動力學模擬、(b)背景環境擬真、(c)障礙物模型、(d)機器人模型、(e)感測器插件與(f)可使用命令行工具執行程式。其分別描述如下：

### 3.2.1、 動力學模擬

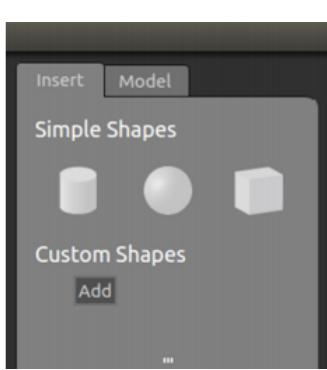
Gazebo 中支援多個高效能的物理引擎，包含 ODE(Open Dynamics Engine)、Bullet、Simbody、DART(Dynamic Animation and Robotics Toolkit)，開發者使用 C++ 語言即可進行三維機器人動力學模擬。

### 3.2.2、 背景環境擬真

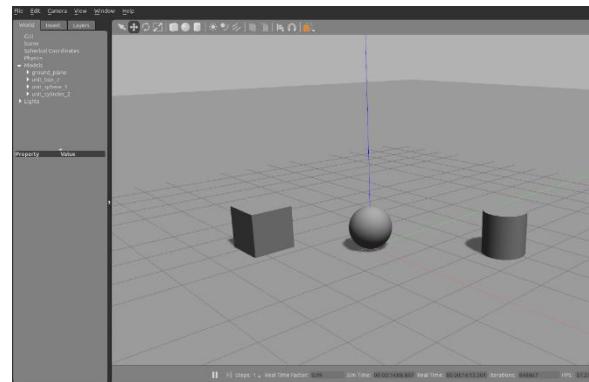
Gazebo 的空間裡可以加入多種環境物件，如光線、重力等基本物理量，匯入的物件(如機器人、競賽項目道具、邊線等場地規範)也可在此環境中做擺放，並自行調整其位置、大小和角度，盡可能地模仿於實際場地時測試之所有物理環境。

### 3.2.3、3D 圖形

Gazebo 中內建三種幾何體，如圖3.1 所示，而由 CAD、Blender 等各種2D、3D 設計軟體所繪製之自行設計環境、道具，匯入後將顯示於圖3.1(a)的 Custom Shapes 中，使環境模擬更趨於真實。



(a) Gazebo 內建幾何體



(b) 放置 Gazebo 內建幾何體

圖 3.1、Gazebo模型匯入介面

### **3.2.3、 機器人模型**

Gazebo 中提供多種機器人模型，包括 PR2、Pioneer2 DX 等，也可使用 SDF 建構自定義機器人。而本專題的第十一代小人形機器人使用之 URDF 檔，經設定後亦可轉成 SDF 檔匯入 Gazebo 內。

### **3.2.4、 感測器插件**

在 Gazebo 裡可自由加入插件，並放置於任何位置。本專題加入攝影機元件及 IMU 元件。其中，攝影機元件模擬人形機器人頭部的攝影機，並將攝影機所拍攝的畫面呈現於人機介面，建立人形機器人策略所需之色模。IMU 元件則模擬機器人三軸之傾斜角度，亦使用人機介面回傳資訊，透過此回授訊號對人形機器人進行控制。

### **3.2.5、 可使用命令行工具執行程式**

本專題透過 ROS 的通信方式，使用其命令行工具，開啟各元件功能、接收運動控制所傳輸的資料及人機介面的資訊，而在 Gazebo 中也可使用命令行工具，於虛擬世界中開啟元件模擬及執行策略，亦可透過指令控制物體、修改環境參數等，在使用上非常便利。

## **3.3、 物件模型設計介紹**

於 Gazebo 中所用的道具物件模型、人形機器人、場地圖，是分別由(1) SolidWorks 以及(2) 3ds Max 所製成。另外，SolidWorks 所使用的表達、描述人形機器人的模型檔案格式(3) urdf 檔，還有 3ds Max 用來呈現三維空間的(4)dae 檔也會一併介紹。

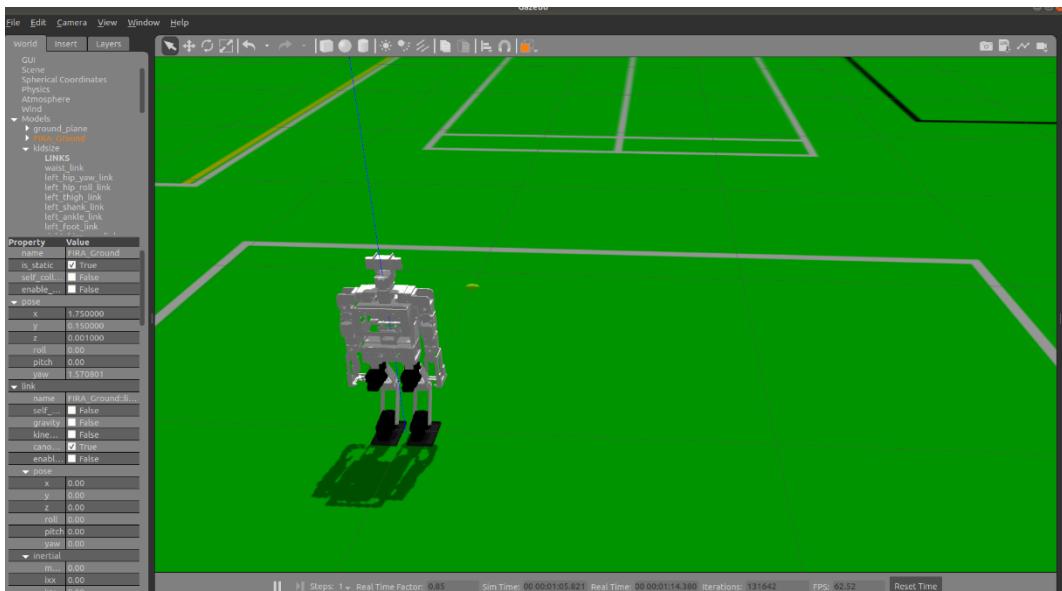


圖 3.2、於 Gazebo 執行機器人模擬

### 3.3.1、SolidWorks

SolidWorks 是採參數化、特徵為基礎的方式來建立模型。「參數化」是指通過計算距離、半徑或是斜率等參數的絕對與相對關係後形成模型。「特徵」也是構成形狀的基礎，即是在物件特徵上添增、消除形狀等來建構模型。SolidWorks 便於製作 3D 精密模型。除了一般的物件製作，完成的零件也可以直接進行組裝並進行模擬執行，並藉此確認零件間是否有設計不周或是瑕疵的部分。所以我們利用 SolidWorks 來製作人形機器人的機身，包含機身零件模型建構、零件組裝都是在此完成。另外，競賽項目當中有較為複雜的道具也可以在 SolidWorks 進行製作。

### 3.3.2、3ds Max

3ds Max 的功用比起 SolidWorks 的物件製作更偏向建構場景。3ds Max 可以將各種模型、物件丟入設計好的「空間」當中，因此建構出來的資料檔是以描述三維空間的檔案格式存在。3ds Max 製作的物件模型，雖然無法達到 SolidWorks 的程度，不過其優秀的場景、動畫製作能力是 SolidWorks 所沒有的。所以比起製作物件、模型，更常被用來製作場景跟動畫。以 Gazebo 的模擬測試而言，我們用以製作模擬競賽的場地空間或簡易的大型物件。

### **3.3.3、 urdf 檔案格式**

urdf 屬於 xml 的文件格式，是一種表達並描述機器人的文件格式，被大量使用於ROS 當中。urdf 在描述機器人時，運用了 link 跟 joint 的連結來建構機器人，其中 link 就像骨頭，而 joint 則是關節。爾後再將每一個部件連結到 link 並上標上相對位置、旋轉角度等的空間定義，最後再將部件的形狀加以描述，這是一個完整的urdf 檔描述機器人的方式。

### **3.3.4、 DAE 檔案格式**

DAE(Digital Asset Exchange)與 urdf 一樣屬於 xml 文本格式。DAE 格式的儲存數據具有相當高的自由度，其原因是 DAE 是將文件的數據分割並分類成好幾塊，數據紀錄包含骨架、節點、3D 模型等，甚至是動畫數據都有。也因為如此，DAE 格式內容相較於其他三維空間格式會更加複雜。在開啟文本時需使用 COLLADA(COLLAborative Design Activity)插件，此插件專門給 3D 互動程式使用的，而儲存的檔案格式即為 DAE 。

# 第4章、人形機器人之系統架構

## 4.1、人形機器人作業系統之簡介

在機器人作業系統(Robot Operating System, ROS)上，是由美國史丹佛大學(Stanford University)之人工智慧實驗室(Stanford Artificial Intelligence Laboratory)與美國機器人技術公司Willow Garage 於 2007 年所共同合作開發完成的，並且由Willow Garage 公司發表，目前 ROS 主要由開放原始碼機器人基金會(Open Source Robotics Foundation)開發及維護，是以 GNU/Linux 作業系統為基礎所開發，專門用於機器人之軟體系統開發與架設的系統框架，其優點為可藉由程式模組化方式以及點對點(Node to Node)設計來達到跨語言的溝通與分散式架構。在 ROS 的系統架構中，包含了眾多不同的套件(Package)，一個套件內包含了各種程序稱為節點(Node)，所以主要有四項:(a)套件(Package)(b)節點 (Node)(c)議題(Topic)(d)服務(Service)，每個節點之間藉由議題(Topic)或服務 (Service)進行通訊，如圖4.1 與圖4.2 所示。

### (a) 套件(Package)

整個ROS 是由各種不同的套件所組成，如：影像視覺套件、策略任務套件以及動作規劃套件...等，而每個套件內容由一個或多個節點所組成。

### (b) 節點(Node)

套件包含一個或多個的節點，其為實際程式執行的部份，每個節點即代表一個獨立執行的程式，各個節點的協同工作帶動了整個機器人的運作，而節點之間的通訊是藉由議題與服務來進行的資訊傳輸。

### (c) 議題(Topic)

議題為 ROS 進行通訊的方法之一，其為單向傳輸。當發送端(Publisher)將資訊傳送至議題時，議題會自動廣播將資料送至相關的訂閱端(Subscriber)中。

### (d) 服務(Service)

服務為 ROS 進行通訊的方法之一，其為雙向傳輸。服務端(Service)會提供請求(Request)及回覆(Response)的資料格式，由客戶端(Client)發送請求給服務端，再由服務端做適當處理後送出回覆資料給客戶端。

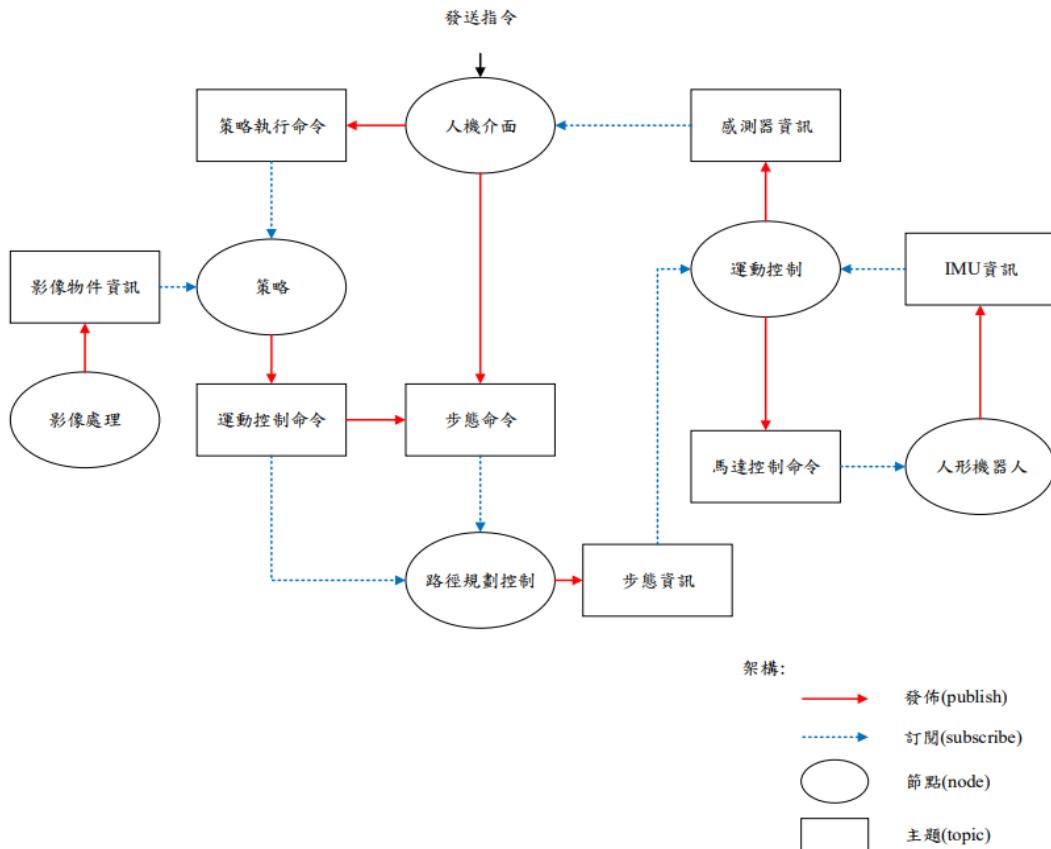


圖 4.1、ROS 的系統架構

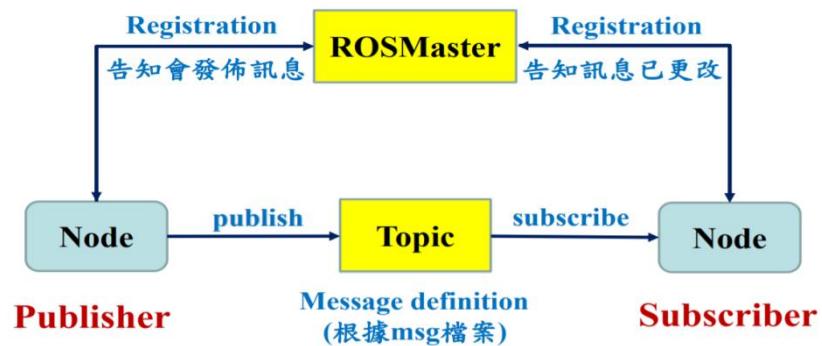


圖 4.2、ROS 的傳輸方式

# 第5章、人形機器人之人機介面

人形機器人主要設計來參與 FIRA 國際機器人比賽，為了方便使用者操作，因此設計影像介面、步態介面以及動作介面，本章將分成三小節介紹影像、步態以及動作介面。

## 5.1、影像介面

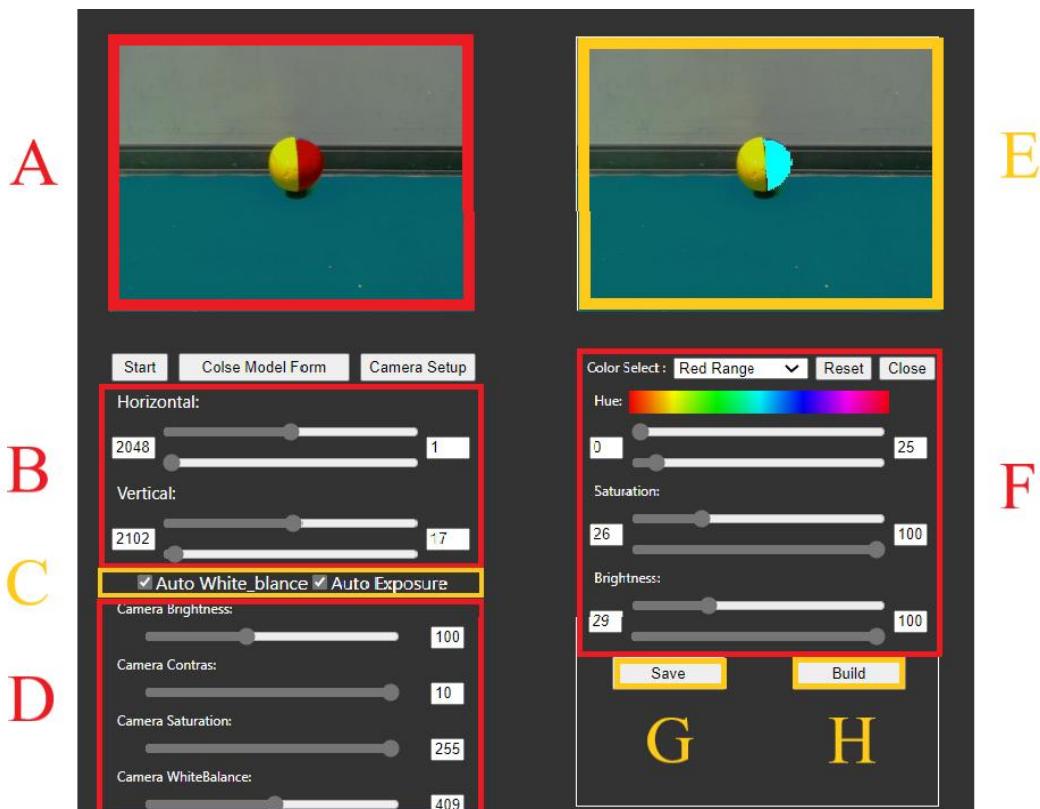


圖 5.1、影像介面說明圖

- A. 攝影機之影像
- B. 調整頭部伺服馬達角度
- C. 自動調整白平衡、曝光
- D. 調整攝影機參數
- E. 處理過後之影像
- F. 調整色相參數
- G. 儲存色相參數之數值
- H. 建立影像模組

## 5.2、步態介面

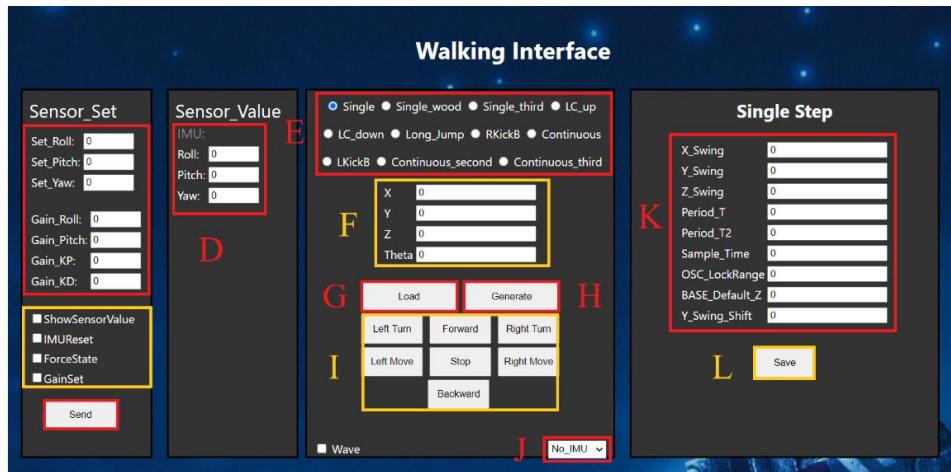


圖 5.2、步態介面說明圖

- |                 |           |
|-----------------|-----------|
| A. 選取需要送出的項目    | B. 送出平衡參數 |
| C. 目前機器人的 IMU 值 | D. 步態模式切換 |
| E. 移動參數調整       | F. 讀取參數檔  |
| G. 執行步態         | H. 控制按鍵   |
| I. 步態平衡的 IMU 設定 | J. 步態參數調整 |
| K. 儲存步態參數       |           |

## 5.3、動作介面

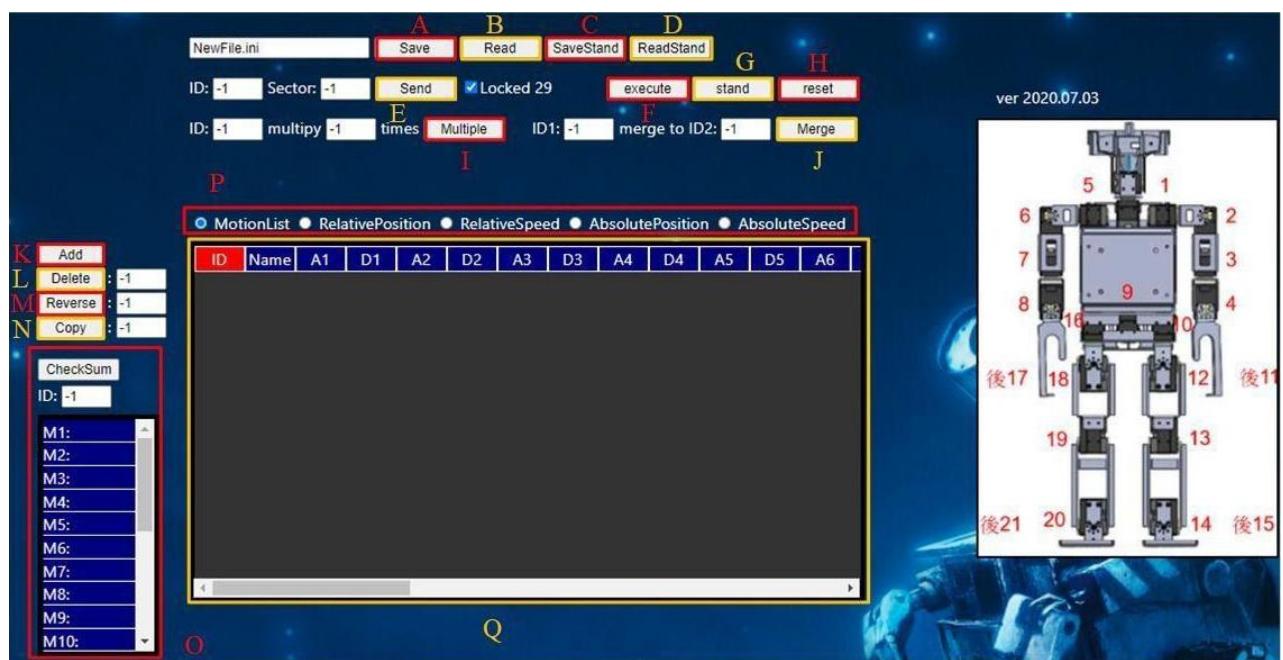


圖 5.3 動作介面說明圖

- |            |         |           |
|------------|---------|-----------|
| A. 存檔      | B. 讀檔   | C. 存取站姿   |
| D. 讀取站姿    | E. 燒錄動作 | F. 執行動作   |
| G. 回歸站立    | H. 按鈕重置 | I. 角度乘倍數  |
| J. 角度相加    | K. 新增動作 | L. 刪除動作   |
| M. 角度方向相反  | N. 複製角度 | O. 總馬達轉動量 |
| P. 選取速度/角度 | Q. 動作數值 |           |

## 第6章、人形機器人之比賽行為決策

### 6.1、籃球競賽之比賽規則

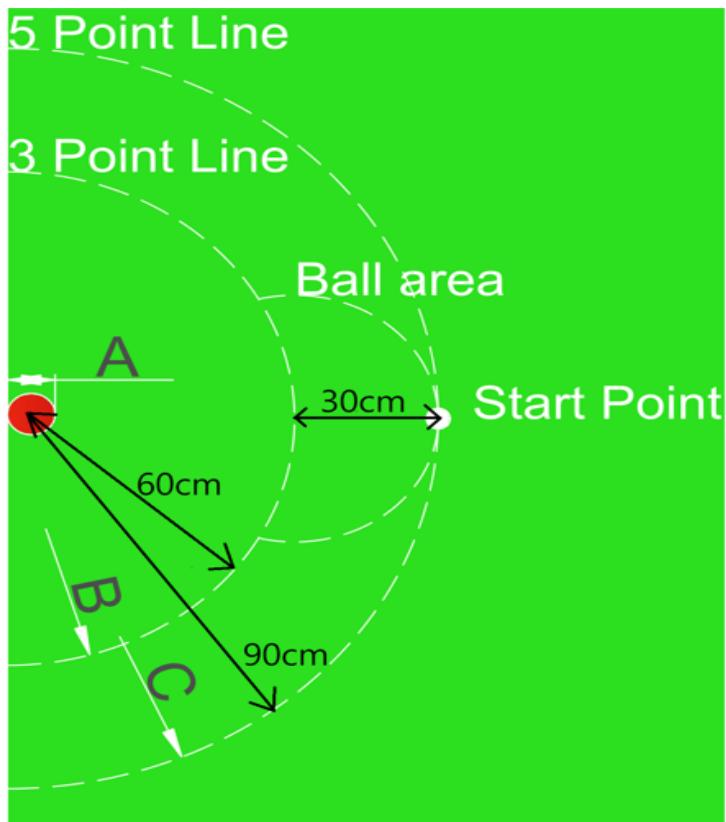


圖 6.1、籃球競賽場地圖

籃球競賽(Basketball)總共有五輪，每回合有兩分鐘之時間限制，場地上畫有三分線、五分線以及放置球的區域界線。每一輪競賽開始前，人形機器人需面向籃框並放置於場地起點(Start Point)，放置人形機器人後，由裁判隨機決定球在球區(Ball area)的位置，這一輪的球區位置不會變更，如圖 6.1 所示。裁判吹哨後，人形機器人需前往球所在的位置將球拿起，若球成功投進且人形機器人未碰到籃框架，則依照人形機器人所在的位置決定分數：站在 90 公分的五分線外時得 5 分；站在 90 公分的五分線內以及 60 公分的三分線外時得 3 分；站在 60 公分的三分線內時得 2 分，否則得 0 分。將五輪得分加總後分數由高到低依次排名，如遇到總分相同則以投進花費時間由少到多依次排名。

## 6.2、籃球競賽之策略介紹

在專題中，籃球策略結合伺服馬達追蹤控制、影像誤差計算、面積測量以及計算投籃力道為架構，為了能在比賽中獲得高分，設計投籃與上籃策略來達到此目的，以下會分成投籃策略以及上籃策略兩部分作介紹。

### 6.2.1、籃球競賽之投籃策略介紹

本小節將投籃策略區分為五個環節做介紹，分別為 **A.搜尋場上的球** **B.追蹤球** **C.夾取球** **D.尋找籃框** **E.追蹤籃框**，以下將依序介紹每個環節的設計。

#### A. 搜尋場上的球

為了能最大限度搜尋球區中的球，人形機器人的視野將會以方形的方式移動，如圖 6.2 所示。在進行此流程時將會先設定好左右邊界，再讓水平頭部伺服馬達以每 10 刻度向左轉，直到轉至左邊界後，再右轉回去到右邊界，接著垂直頭部伺服馬達轉向離人形機器人較遠處，如圖 6.3 (a) 所示，再以水平頭部伺服馬達以每 10 刻度向左轉，直到轉至左邊界，最後垂直頭部伺服馬達轉向離人形機器人較近處，如圖 6.3 (a)所示，。



圖 6.2、人形機器人視野搜尋方向圖

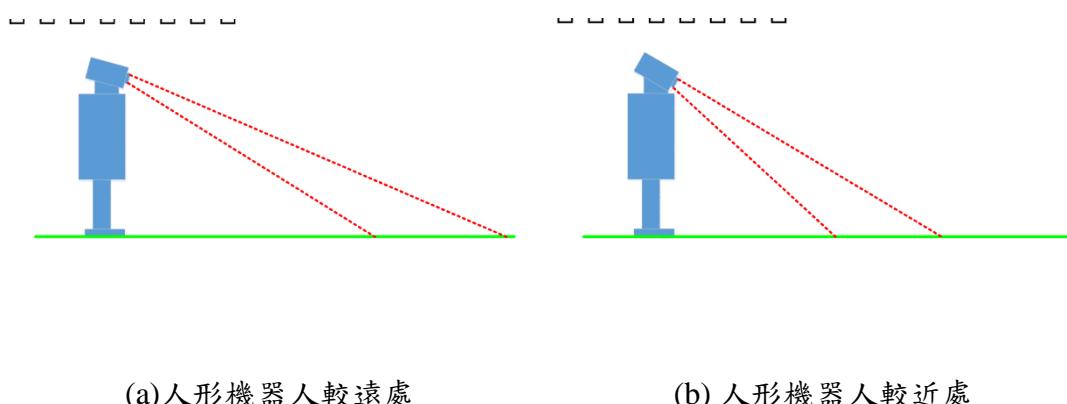


圖 6.3、頭部伺服馬達之轉動圖(a)(b)

當在搜尋場上的球流程中找到球後，會根據頭部伺服馬達的刻度計算出球的位置，這時會有三種情況

- (1)可以直接夾球：這時不會開啟步態，而是會進入預備直接夾球姿勢，並開啟直接夾球旗標
- (2)球太近：開啟後退旗標
- (3)其他：直接進入下一階段

## B. 追蹤球

為了使追蹤球的過程同時具備精確度與速度，人形機器人所執行的步態會依據球的距離遠近去更改，當人形機器人移動過程中，會持續轉動頭部伺服馬達將球固定在影像的特定位置，此時頭部伺服馬達的刻度為人形機器人判斷行走方向之依據，當垂直頭部伺服馬達刻度小於特定的數值時，代表球與人形機器人的距離相對較近，因此會切換到較慢的行走速度，藉此提升此流程的精準度，反之，大於特定的數值時，則切換到較快的行走速度，以縮短整體策略執行所使用的時間，如圖 6.4 所示。同時會去判斷水平頭部伺服馬達是否超出設定的刻度，來決定是否需要執行步態的左右、前後修正，最後當頭部伺服馬達的刻度皆符合各項條件時就會停止人形機器人。

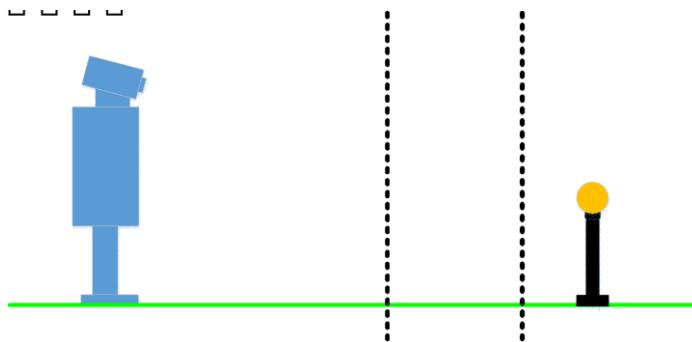


圖 6.4、人形機器人之遠近示意圖

### C. 夾取球

此流程中會先判斷是否有開啟直接夾球旗標，

(1)沒有開啟時：人形機器人在完成預備夾球姿勢後，會接著執行腰部夾球修正，未修正如圖 6.5(a)所示，完成修正後如圖 6.5(b)所示，使得球位於畫面中心小於誤差值，即執行夾球動作如圖 6.6 所示。

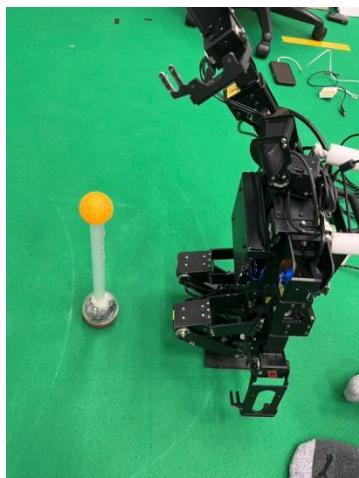


圖 6.5 (a)

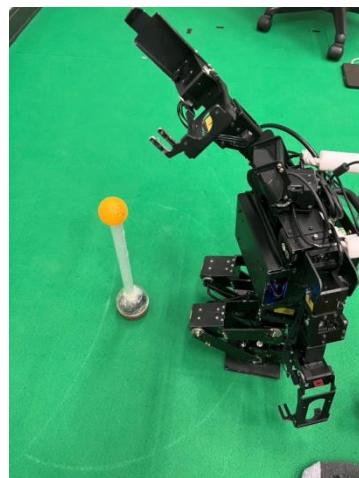


圖 6.5 (b)

(2)有開啟時：由於在這之前已完成預備直接夾球姿勢，會直接執行腰部夾球修正，未修正如圖 6.5(c)所示，完成修正後如圖 6.5(d)所示，使得球位於畫面中心小於誤差值，即執行夾球動作，如圖 6.6 所示。



圖 6.5(c)



圖 6.5(d)

圖 6.5 預備夾球動作(a)(b)(c)(d)

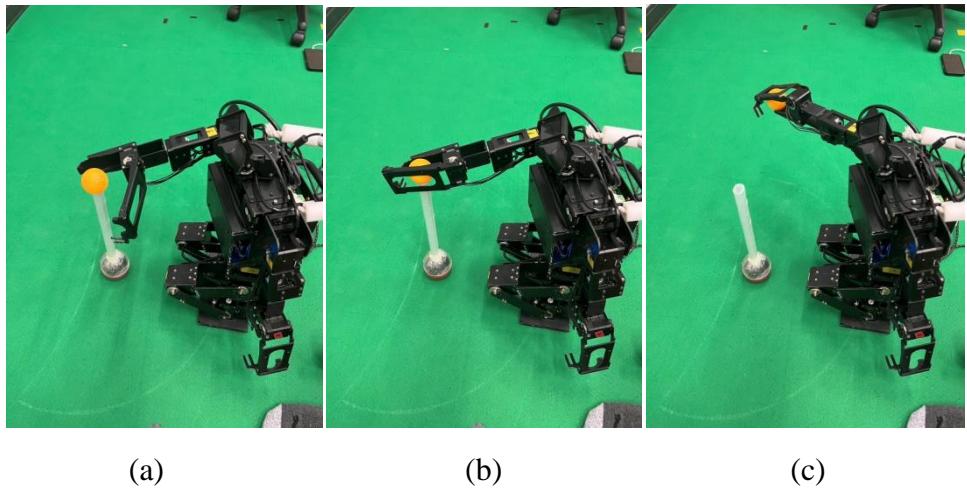


圖 6.6 夾球動作(a)(b)(c)

#### D. 尋找籃框

在夾球結束後人形機器人執行站立持球動作，此時人形機器人將開始轉動頭部攝像頭，尋找籃框位置，如圖 6.7 所示，之後鎖定籃框，防止在進入下一階段 E.追蹤籃框時丟失目標。

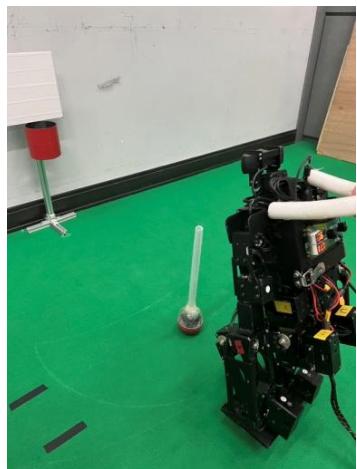


圖 6.7 開始找框

## E. 追蹤籃框

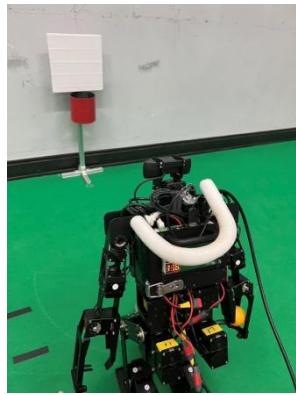


圖 6.8 追蹤到框

追蹤籃框前如圖 6.8，這時會分為五分線投籃策略與不走路投籃策略兩種。

### 五分線投籃：

首先利用籃框面積大小判斷人形機器人與籃框的相對距離，使人形機器人向後倒退至設定的範圍內，由於人形機器人找到籃框的時候並非完全面對籃框，利用水平頭部伺服馬達刻度判斷來進行左右旋轉，使人形機器人完全面對籃框，此時人形機器人會以面積測距來計算距離，接著計算籃框與垂直基準線的誤差後，執行轉腰動作使右手臂對準籃框中心，再以距離為判斷依據，計算出投籃力道後執行投籃。

### 不走路投籃：

與五分線策略不同的地方是，在直接夾球起身之後並找到框後，不會開啟步態，而是之接近入預備投球姿勢，再利用頭部的水平位置刻度判斷執行轉腰動作，當頭部與腰部刻度小於定值時，機器人再以面積測距來計算距離，接著計算籃框與垂直基準線的誤差後，執行細部轉腰使右手臂對準籃框中心，再以距離為判斷依據，計算出投籃力道後執行投籃。

## ● 面積測距

人形機器人的攝影機其影像解析度為 320\*240 像素。面積測距的原理為將已知物件透過影像擷取到的特徵點，固定於視野中的特定位置。此測距的特徵點為籃框水平中心點，人形機器人會先轉動頭部伺服馬達，將視野的中央對準其特徵點，再根據人形機器人位於 60cm 與 90cm 測量之籃框面積，推算出目標物件與攝影機的相對距離。

此測距的用法在於先對準籃框的水平中心線後，利用機器人所測得距離籃框 60cm( $L_{60cm}$ )及 90cm( $L_{90cm}$ )的面積  $A_{60}$  及  $A_{90}$  進行計算推斷目前機器人位於  $L_{now}$ ，如圖 6.9 所示：

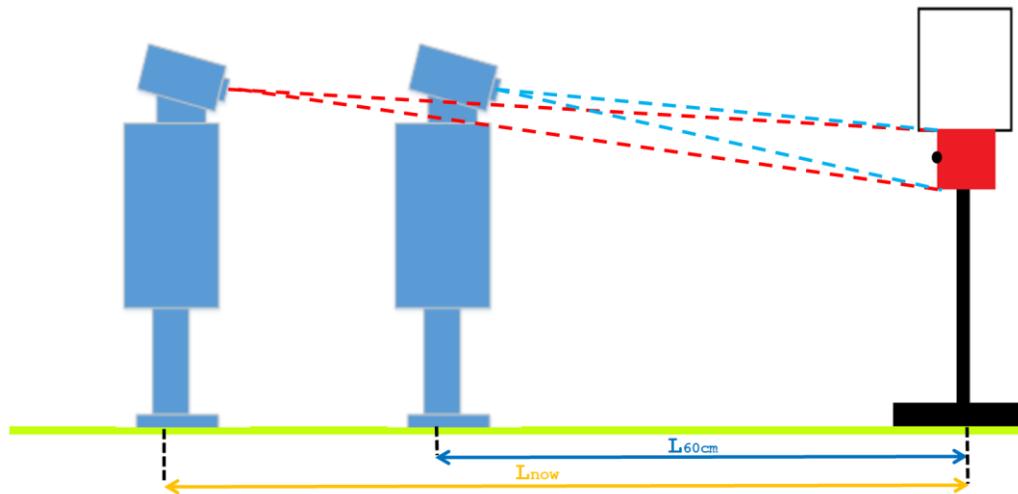


圖 6.9 遠近對比圖

機器人移動所造成的視角誤差不大，將其誤差忽略不計可將畫面簡化，如圖 6.10 所示：

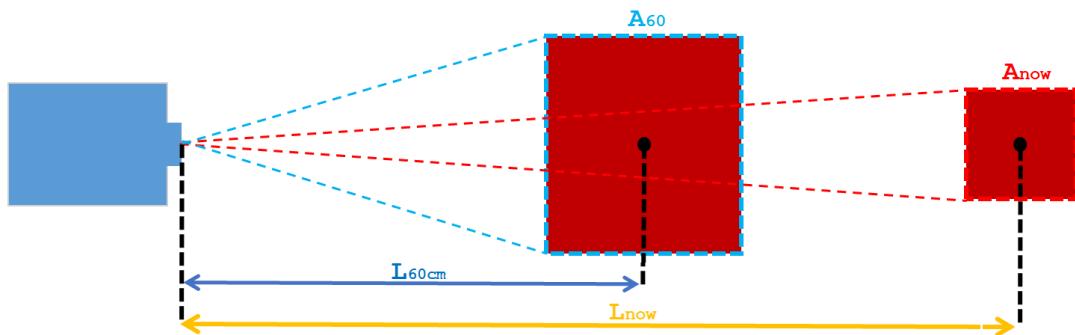


圖 6.10 大小對比圖

根據面積與距離平方成反比可得：

$$A_{60} : A_{now} = (L_{now})^2 : (L_{60cm})^2$$

經移項可得整理後，令以 60 為基準點所得的  $L_{now}$  為  $D_{60}$  可得以下公式：

$$D_{60} = \sqrt{\frac{(60\text{cm})^2 * 60\text{cm 所測得之籃框面積}}{\text{當前位置所測得之籃框面積}}}$$

以 90cm 為基準點，同理可得以下公式：

$$D_{90} = \sqrt{\frac{(90\text{cm})^2 * 90\text{cm 所測得之籃框面積}}{\text{當前位置所測得之籃框面積}}}$$

根據測量之籃框面積採用較近的基準點，又因為忽略機器人移動所造成的視角誤差加入誤差值 E 進行修正，即為機器人所測得與籃框的距離  $L_{now}$ ，如式()所示：

$$L_{now} = \frac{D_{60} + D_{90}}{2} + E$$

## ● 計算投籃力道

在投籃力道方面，根據距離  $D$  分別計算出權重  $W_{50}$ 、 $W_{60}$ 、 $W_{70}$  和  $W_{80}$ ，如式(6.1)、(6.2)、(6.3) 及(6.4)示：

$$W_{50} = \begin{cases} 1 & , 0 < D \leq 50 \\ \frac{(60-D)}{10} & , 50 < D \leq 60 \\ 0 & , Otherwise \end{cases} \quad (6.1)$$

$$W_{60} = \begin{cases} \frac{(D-50)}{10} & , 50 < D \leq 60 \\ \frac{(70-D)}{10} & , 60 < D \leq 70 \\ 0 & , Otherwise \end{cases} \quad (6.2)$$

$$W_{70} = \begin{cases} \frac{(D-60)}{10} & , 60 < D \leq 70 \\ \frac{(80-D)}{10} & , 70 < D \leq 80 \\ 0 & , Otherwise \end{cases} \quad (6.3)$$

$$W_{80} = \begin{cases} \frac{(D-70)}{10} & , 70 < D \leq 80 \\ 1 & , 80 < D \\ 0 & , Otherwise \end{cases} \quad (6.4)$$

固定測量特定距離所需的投籃力道，採用權重計算的方式，將特定距離所需的力道  $S_{50}$ 、 $S_{60}$ 、 $S_{70}$  和  $S_{80}$  作為變數輸入，最後輸出人形機器人投籃所需的力道  $S_{throw}$ ，如式(6.5)所示：

$$S_{throw} = W_{50} \times S_{50} + W_{60} \times S_{60} + W_{70} \times S_{70} + W_{80} \times S_{80} \quad (6.5)$$

## 6.3、籃球競賽之上籃策略介紹

本小節將上籃策略區分為六環節做介紹，分別為 A.搜尋場上的球 B.追蹤球 C.夾取球 D.尋找籃框 E.追蹤籃框 F.瞄準籃框，以下將針對與投籃策略有所差異的 E.追蹤籃框與 F.瞄準籃框作介紹。

### A. 搜尋場上的球

此流程與 6.2.1 的 A 相同。

### B. 追蹤球

此流程與 6.2.1 的 B 相同。

### C. 夾取球

此流程與 6.2.1 的 C 相同。

### D. 尋找籃框

此流程與 6.2.1 的 D 相同。

### E. 追蹤籃框

尋找籃框後，人形機器人和投籃一樣並非完全面對籃框，與投籃策略相異之處在於，首先利用人形機器人的垂直頭部伺服馬達刻度判斷籃框的距離，若距離大於減速線時，利用水平頭部伺服馬達刻度判斷，得到人形機器人本身與籃框的相對位置來執行快速的左前進、右前進或前進，當距離小於減速線後，將前進速度放慢使得機器人與上籃的停止線距離誤差縮小。此方法可以減少完整上籃策略的完成時間，當人形機器人在上籃範圍內時，進入瞄準籃框流程。示意圖如圖 6.11 所示：

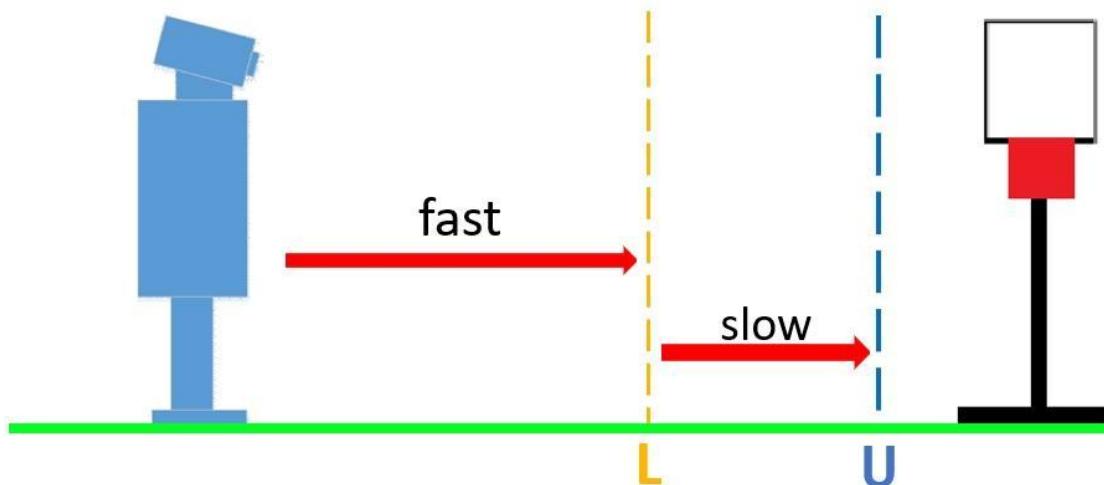


圖 6.11、上籃減速線示意圖

L. 減速線 U. 上籃停止線

上籃策略追蹤籃框流程圖如圖 6.12 所示。

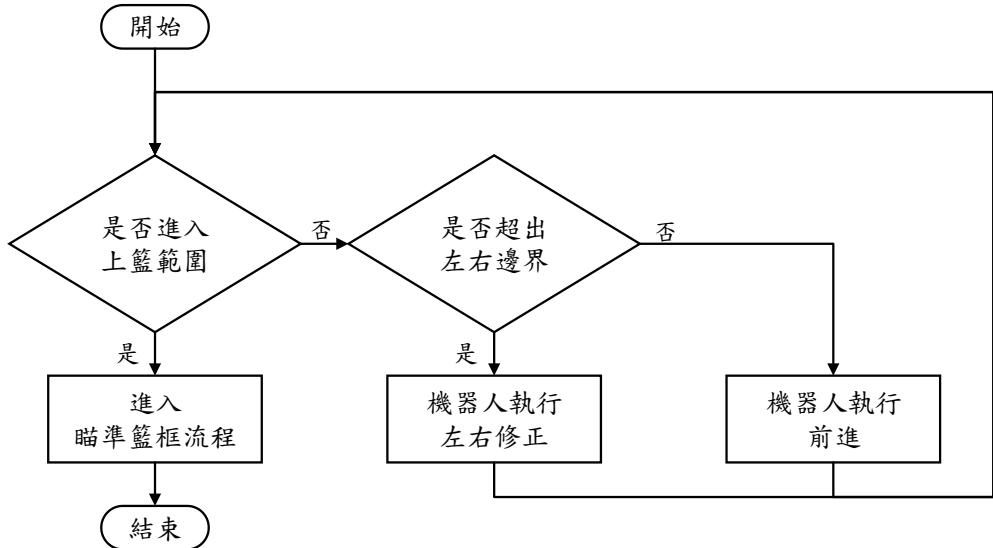


圖 6.12、上籃策略追蹤籃框流程圖

## F. 瞄準籃框

人形機器人進入上籃範圍時停止前進，此時雖然人形機器人大致對準籃框，但還是有些微差距會影響上籃穩定性，若人形機器人未完全面對籃框，將會造成水平頭部伺服馬達向左轉或向右轉，因此利用人形機器人最後的水平頭部伺服馬達刻度與上籃基準線的刻度，可以計算出目前人形機器人對於籃框有多少水平角度誤差值，再利用誤差值讓腰部伺服馬達進行修正，使人形機器人右手臂對準籃框，最後執行上籃動作。

## 6.4.0、籃球競賽之夾球動作介紹

為完成籃球項目，本專題設計以下動作，分別為

- (A)蹲下與舉手、(B)手下降夾球、轉手與站立持球、(C)投球舉手動作、(D)投籃、  
(E)上籃舉手動作、(F)上籃。以下將依序介紹每個動作的設計。

(A) 蹲下與舉手

為了將位於一定高度的球夾起，因而設計蹲下的姿勢，如圖 6.13 所示。右手會先往右舉起，以避免碰觸到球，接著再把手舉高，方便下一步的夾球動作。

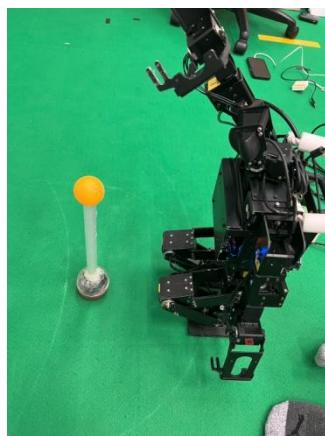
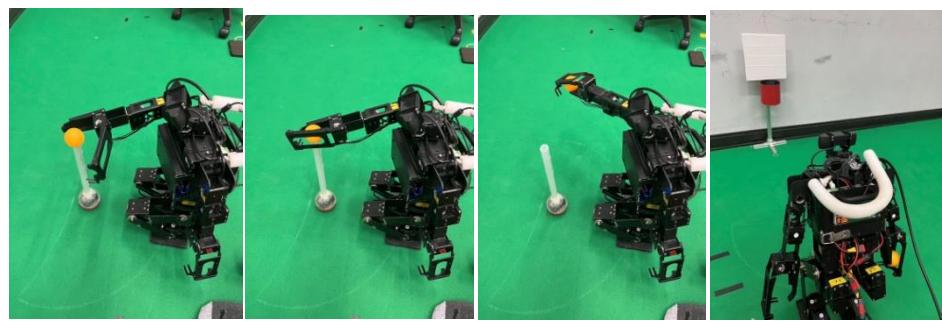


圖 6.13、蹲下與舉手

### (B) 手下降夾球、轉手與站立持球

右手會先下降到可以夾到球的位置，如圖 6.14(a)所示，的手爪會將球夾入右手的手爪中，以完成夾球動作，如圖 6.14(b)所示。為了防止手收回時會卡到球柱，執行轉手動作，如圖 6.14(c)所示。執行完以上動作後，將執行站立持球的動作，如圖 6.14(d)所示。



(a)手下降 (b)夾球 (c)轉手 (d)站立持球

圖 6.14 手下降、夾球、轉手與站立持球

### (C) 投籃舉手動作

在執行投籃舉手動作的過程中，右手臂會向上舉起，將球保持在右手爪上，如圖 6.15 所示。接著腰部會進行前後傾的修正使上半身垂直於地面，以確保在執行 E、追蹤籃框流程中的轉腰修正後，可保持同樣地的出手角度。



圖 6.15、投籃舉手動作

### (D) 投籃

執行投籃動作時，將會先打開右手爪，再微調頭部刻度後微調腰部刻度，讓投球路線對準球框，如圖 6.16(a)所示。投籃時會配合 E、追蹤籃框中計算投籃力道功能來改變揮動速度，以達到各種力道的投籃軌跡，如圖 6.16(b)(c)所示。

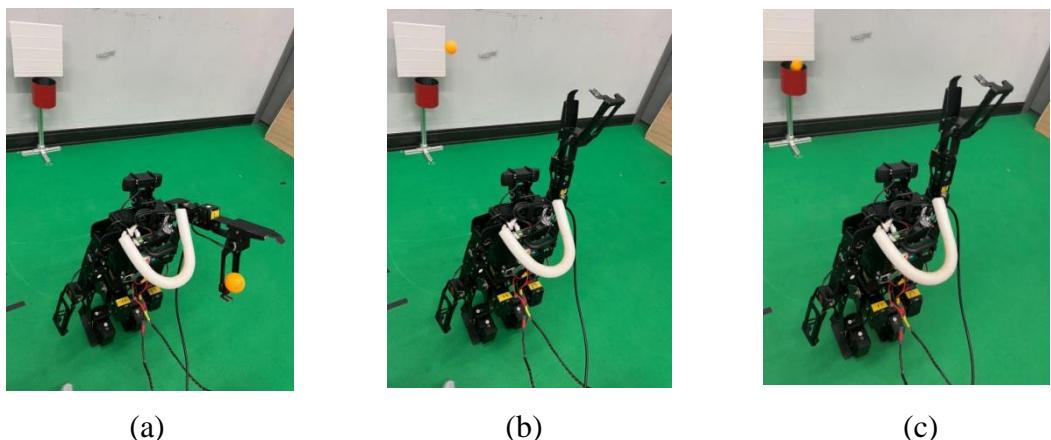


圖 6.16、投籃(a)(b)(c)

## (E) 上籃舉手動作

在執行上籃舉手動作時，最開始為站立在籃框前，如圖 6.16(a)所示，之後腰部會進行前後傾的修正使上半身垂直於地面，以確保在執行 **F、瞄準籃框** 流程中的轉腰之後，可保持同樣地的出手角度，人形機器人與籃框的距離較近，為了避免觸碰到籃框的情況下將手舉起，會先將右手爪向外轉動，如圖 6.16(b)所示，再將右手臂向上舉起，如圖 6.16(c)所示，最後再執行 **F、瞄準籃框** 的角度修正，使手臂可以正對籃框，如圖 6.16(d)所示。



(a) 站立在籃框前



(b) 右手臂向外轉動



(c) 右手臂向上舉起

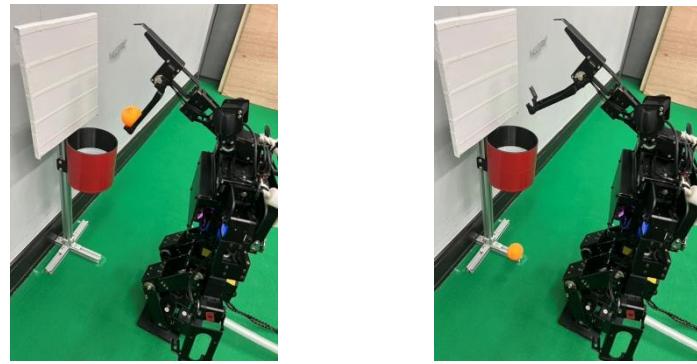


(d) 角度修正

圖 6.17、上籃舉手動作(a)(b)(c)(d)

## (F) 上籃

為了確保球可以穩定落下，人形機器人的右手臂在上一步角度修正後，右手爪再慢慢打開，如圖 6.17(a)所示，最後右手爪再快速的向上轉動，把球放進球框中，如圖 6.17(b)所示。



(a)右手爪打開

(b)上籃

圖 6.18、上籃動作(a)(b)

## 第7章、結論

本專題透過機器人作業系統 ROS(Robot Operating System)，整合機械機構、電子電路、感測器運用以及馬達控制，實現透過視覺判斷達到自主運動的人形機器人，其未來發展將擴展到機器人比賽之外，透過人形機器人，可以代替人類進行許多具有高風險的行為或是輔助人類達成更高效率的作業。

人形機器人的策略藉由影像辨識讀取資訊，針對不同的運動控制，結合多種演算法來實現目的。在計算距離方面，運用「面積測距」計算人形機器人與籃框的距離；在夾球方面，利用機器人能夠轉腰的優勢，再站定點後能夠針對水平誤差進行修正，以提高夾球穩定性；在投籃方面，為了縮短完成比賽的時間，新增了不走路投籃的策略，同樣利用轉腰的優勢，達成對正籃框的目的。人形機器人整合以上功能所實現的專題成果。

2022 年 8 月參加於國立高雄科技大學舉辦的「2022 國際智慧機器運動大賽：人形機器人組(HuroCup)」，在整個實驗室的努力下，以本專題成果參與並完成競賽，且在競賽中拿下冠軍的榮譽。藉由比賽過程與各校交流發現可以如何優化及對於策略更多的可能性，未來本實驗室將會持續對機器人的機構、電路及影像判斷等方面進行改良，使機器人的操作以及判斷更為智慧且順暢，最終以對本專題開發之人形機器人進行產品化為目標。