

بسم الله الرحمن الرحيم



دانشگاه اصفهان
دانشکده فنی و مهندسی
گروه مهندسی مکانیک

پایان نامه کارشناسی رشته مهندسی مکانیک

طراحی و ساخت دستگاه ایجاد حس نیرو و گشتاور در محیط‌های واقعیت مجازی

استاد راهنما:

دکتر حسن پور

استاد مشاور:

دکتر کریم پور

پژوهشگر:

محمدحسن مختارآبادی

شهریور ماه ۱۳۹۹



دانشگاه اصفهان
دانشکده فنی و مهندسی
گروه مهندسی مکانیک

پایان نامه‌ی کارشناسی رشته‌ی مهندسی مکانیک
آقای محمدحسن مختارآبادی تحت عنوان

طراحی و ساخت دستگاه ایجاد حس نیرو و گشتاور در محیط‌های واقعیت مجازی

در تاریخ توسط هیات داوران زیر بررسی و با نمره به تصویب نهایی رسید.

۱- استاد راهنمای پایان نامه دکتر حسن پور امضاء

۲- استاد مشاور پایان نامه دکتر کریم پور امضاء

۳- استاد داور دکتر امضاء

۴- مدیر گروه دکتر آریایی امضاء

چکیده

امروزه تلاش محققین برای یافتن راهی برای از بین بردن هرچه بیشتر مرز بین واقعیت و فضای مجازی می‌باشد. بیشتر محیط‌های واقعیت مجازی^۱ تجهیزاتی برای اندرکنش تصویری انسان با رایانه هستند، اما گروه محدودی نیز دارای عملگرهای صوتی یا لمسی برای تعامل با کاربر هستند که به تقویت حس حضور در محیط واقعی کمک می‌کنند. در این پژوهش سعی شده است که نمونه‌ای از این عملگرها که با وارد کردن نیرو به دست فرد باعث ایجاد حس بهتر از محیط می‌شوند، طراحی و ساخته شود. به طور کلی این ابزارها را می‌توان به دو دسته متصل به زمین^۲ و غیر متصل به زمین^۳ تقسیم کرد که با توجه به قابلیت آزادی حرکت بیشتر مدل‌های غیر متصل به زمین نمونه ساخته شده در این پژوهش نیز از همین نوع است که نیروی مورد نیاز با دو فن محفظه‌دار^۴ تامین می‌گردد. همچنین محیط شبیه‌سازی به صورت گوی بر روی یک صفحه انتخاب شده است که با تغییر مکان گوی بر روی صفحه با توجه به زاویه صفحه، گشتاورهای متفاوتی به دست فرد وارد می‌شود که این ابزار وظیفه تامین این گشتاورها را دارد.

کلمات کلیدی

واقعیت مجازی، عملگرهای لمسی، نیروی مجازی

¹ Virtual Reality

² Grounded

³ Ungrounded

⁴ Ducted fan

فهرست مطالب

فصل اول مقدمه.....	۱
۱-۱ مقدمه.....	۱
۲-۱ دستگاه AERO-PLANE.....	۲
۳-۱ تعریف پروژه.....	۲
۴-۱ ارزش پروژه.....	۳
۵-۱ مراحل انجام پروژه.....	۳
۶-۱ ساختار پایان نامه.....	۴
فصل دوم طراحی محیط شبیه سازی.....	۵
۱-۲ مقدمه.....	۵
۲-۲ ایجاد محیط شبیه سازی.....	۵
۳-۲ محاسبات مربوط به موقعیت گوی و زاویه صفحه.....	۸
۴-۲ محاسبات مربوط به نیروی هر فن.....	۱۳
۵-۲ جمع بندی.....	۱۶
فصل سوم برنامه نویسی پردازنده.....	۱۷
۱-۳ مقدمه.....	۱۷
۲-۳ برد آردیونو.....	۱۸
۱-۲-۳ توضیح بخش های مختلف کد آردیونو.....	۱۸
۲-۲-۳ تابع محاسبه موقعیت گوی (ballposition()).....	۲۲
۳-۲-۳ تابع محاسبه نیروی هر فن.....	۲۴
۳-۳ سنسور شتاب سنج وژیروسکوپ.....	۲۵
۱-۳-۳ ترکیب داده های سنسور شتاب سنج وژیروسکوپ.....	۲۶
۲-۳-۳ تابع خواندن زاویه از سنسور در آردیونو (sensor()).....	۲۹
۴-۳ مازول بلوتوث.....	۳۰
۵-۳ تابع فرستنده اطلاعات به عملگرهای دستگاه (SENDTODEVICE()).....	۳۵
۶-۳ مازول تغذیه کاهنده.....	۳۶
۷-۳ جمع بندی.....	۳۶

فصل چهارم طراحی و ساخت ۳۷

۱-۴ مقدمه ۳۷

۲-۴ طراحی دستگاه ۳۷

۱-۲-۴ طراحی قطعات دستگاه ۳۷

۲-۲-۴ اتصال قطعات الکترونیکی ۴۴

۳-۴ مونتاژ قطعات دستگاه و انجام آزمایش های مربوطه ۴۵

۴-۴ سنجش کارایی دستگاه ۵۰

۱-۴-۴ نحوه انجام تست ها ۵۱

۲-۴-۴ بررسی نتایج ۵۲

۵-۴ دستورالعمل استفاده از دستگاه ۵۷

۶-۴ جمع بندی ۶۱

فصل پنجم نتیجه گیری و پیشنهادات ۶۲

۱-۵ مقدمه ۶۲

۲-۵ شرح خلاصه ای درمورد پروژه و پایان نامه ۶۲

۳-۵ نوآوری ۶۳

۴-۵ نتیجه گیری ۶۴

۵-۵ پیشنهادات ۶۴

مراجع ۶۶

پیوست الف ۶۹

پیوست ب ۷۳

فهرست اشکال

- شکل ۱-۱ تصویری از دستگاه Aero-plane ۲
- شکل ۱-۲ تصویر محیط قسمت ویرایش نرم افزار 3D animation ۶
- شکل ۲-۲ تصویر نهایی محیط شبیه سازی ۷
- شکل ۳-۲ تنظیمات بلوک VR Sink ۷
- شکل ۴-۲ بلوک VR Sink ۸
- شکل ۵-۲ سیستم مختصات محیط شبیه سازی ۸
- شکل ۶-۲ مشابهت سازی محیط دوبعدی با محیط یک بعدی ۹
- شکل ۷-۲ دیاگرام آزاد گوی ۱۰
- شکل ۸-۲ تصویر از بالای صفحه XZ ۱۲
- شکل ۱-۳ ماژول سنسور MPU9250 ۲۶
- شکل ۲-۳ سیستم مختصات سنسور ژيروسکوپ ۲۷
- شکل ۳-۳ تصویر تمامی بلوک های استفاده شده در سیمولینک ۳۱
- شکل ۴-۳ تنظیمات مربوط به بلوک Query Instrument ۳۴
- شکل ۱-۴ تصویر اسمبلی شده دستگاه ۴۳
- شکل ۲-۴ نقشه اتصال قطعات الکترونیکی به یکدیگر ۴۴
- شکل ۳-۴ تصویر نهایی دستگاه مونتاژ شده ۴۵
- شکل ۴-۴ یک سیگنال PWM، با چرخه کاری 50% [22] ۴۶
- شکل ۵-۴ نمودار نوع سیگنال بر حسب نیروی تولیدی در هر فن ۵۰
- شکل ۶-۴ صفحه ساخته شده به منظور تست دستگاه ۵۱
- شکل ۷-۴ موقعیت های مشخص شده بر روی صفحه ۵۵
- شکل ۸-۴ شارژ شدن باتری دستگاه ۵۸
- شکل ۹-۴ وضعیت غیرفعال کلید فعال سازی فن ها ۵۸
- شکل ۱۰-۴ اتصال سوکت تغذیه دستگاه به باتری ۵۹
- شکل ۱۱-۴ اتصال رایانه به ماژول بلوتوث دستگاه ۵۹
- شکل ۱۲-۴ تعریف متغیر S شامل اطلاعات پورت سریال ۶۰
- شکل ۱۳-۴ اجرای فایل سیمولینک و اتصال دستگاه به محیط شبیه سازی ۶۰
- شکل ۱۴-۴ فعال سازی فن ها و موتور سروو ۶۱

- شکل ۷-۱ شکل کامل دستگاه ۷۰
- شکل ۷-۲ نقشه انفجاری دستگاه ۷۰
- شکل ۷-۳ نقشه دسته دستگاه ۷۱
- شکل ۷-۴ نقشه شافت میانی دستگاه ۷۱
- شکل ۷-۵ نقشه پین دستگاه ۷۲
- شکل ۷-۶ نقشه صفحه طراحی شده در محیط مجازی ۷۲

فهرست جداول

جدول ۱-۳ بردها و ماژول‌های الکترونیکی استفاده شده در دستگاه	۱۷
جدول ۱-۴ فهرست قطعات دسته اول به کار رفته در دستگاه	۳۸
جدول ۲-۴ فهرست قطعات دسته دوم به کار رفته در دستگاه	۴۰
جدول ۳-۴ فهرست قطعات دسته سوم به کار رفته در دستگاه	۴۲
جدول ۴-۴ نیروی وارده توسط دو فن به ازای پالس‌های مختلف ورودی	۴۷
جدول ۵-۴ نیروی ثبت شده برای هر فن برحسب نیوتن برای پالس‌های مختلف	۴۸
جدول ۶-۴ داده‌های مربوط به تست در محیط واقعی	۵۳
جدول ۷-۴ داده‌های مربوط به تست در محیط مجازی (با دستگاه ساخته شده در این پروژه)	۵۴
جدول ۸-۴ درصد حدس‌های درست هر موقعیت در هر دو بخش تست دستگاه	۵۶

فصل اول

مقدمه

۱-۱ مقدمه

عملگرهای لمسی^۱ با افزایش المان های حقیقی می توانند تاثیر مهمی در تجربه فرد حین استفاده از فناوری واقعیت مجازی داشته باشند [1]. و در عین حال با وجود ایجاد تجربه ای مناسب از محیط این وسایل نباید تاثیر زیادی بر حرکت و آزادی عمل فرد داشته باشند که می توان به متصل بودن و یا نبودن این ابزارها به زمین و یا وزن آنها اشاره کرد. یکی از مهمترین المان هایی که می تواند به واقعی تر جلوه دادن محیط واقعیت مجازی کمک کند نیروی عکس العملی است که توسط محیط مجازی به بدن فرد وارد می شود. این نیرو می تواند به شکل های مختلفی ایجاد شود که در دسته بندی ابزارهای متصل به زمین می توان به بازوی رباتیک [2]، ریسمان های متصل به فرد [3] و یا ابزارهایی که به شکل اسکلت بندی بدن انسان ساخته می شوند و به نوعی قابلیت کنترل تمامی حرکات بدن را دارند [4] اشاره کرد که همانطور که اشاره شد نقطه ضعف این سیستم ها محدود کردن آزادی عمل فرد می باشند. از سوی دیگر در دسته بندی ابزارهای غیر متصل به زمین نیز این تنوع وجود دارد که تفاوت عمده آنها در منبع ایجاد نیرو می باشند. به طور مثال این نیرو می تواند به وسیله اثر ژيروسکوپ [5] [6] [7]، محرک الکتریکی ماهیچه ها [8]، ملخ های تولید نیروی پیشران^۲ [1] [9] [10]، تغییر مرکز جرم [11] [12]

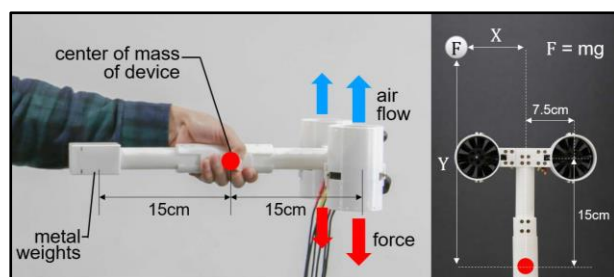
¹ Haptic force-feedback

² Thrust

[13]، صفحات ایجاد نیروی پسا^۱ [14]، حرکت سیال [15] و انواع مختلف دیگر باشد که نیرویی که در این پژوهش ایجاد می‌شود نیروی پیشران می‌باشد. دستگاه ساخته شده در واقع مدل تکامل یافته دستگاه ساخته شده تحت عنوان Aero-plane می‌باشد [1].

۲-۱ دستگاه Aero-plane

همانطور که اشاره شد دستگاه ساخته شده در این پروژه الهام گرفته از دستگاه Aero-plane می‌باشد. دستگاه Aero-plane ابزاری است که به کمک نیروی ایجاد شده در فن‌های تعبیه شده بر روی آن می‌تواند حرکت یک جسم بر روی یک صفحه دوبعدی و همچنین وزن ابزارهای مختلف را شبیه‌سازی نماید. این دستگاه از قرار دادن دو فن بر روی یک دسته که محل در دست گرفتن دستگاه توسط فرد می‌باشد، ساخته شده است، که هر کدام از این فن‌ها قابلیت ایجاد نیرو تا مقدار ۷ نیوتن در زمان ۰/۳ ثانیه می‌باشند و همچنین وزن دستگاه ۱۰۶۹ گرم گزارش شده است. مقاله مربوط به این دستگاه در سال ۲۰۱۹ توسط Seungwoo Je، Myung Jin Kim و نویسندگان دیگر چاپ شده است. سایر اطلاعات مربوط به این دستگاه را می‌توان در مرجع [1] مشاهده نمود. تصویری از این دستگاه را می‌توان در شکل ۱-۱ مشاهده نمود.



شکل ۱-۱ تصویری از دستگاه Aero-plane

۳-۱ تعریف پروژه

دستگاه حاضر شامل دو فن جهت تولید نیروی پیشران متصل به یک دسته که محل در دست گرفتن دستگاه برای فرد است، می‌باشد. با تغییر سرعت هر کدام از فن‌ها می‌توان گشتاورهای مختلفی را به دست فردی که دسته دستگاه را به دست گرفته است وارد نمود. همچنین محیط شبیه‌سازی به عنوان گوی بر صفحه انتخاب

¹ Drag

شده است که فرد با تغییر زاویه دستگاه در دست خود موجب تغییر زاویه صفحه در شبیه ساز شده و بالطبع موقعیت گوی بر روی صفحه تغییر می کند و با توجه به موقعیت مشخص گوی و وزن آن نیرو و گشتاور مشخصی به دست فرد وارد شده و باعث ایجاد حس بهتر از حضور در محیط می شود.

۴-۱ ارزش پروژه

همانطور که ذکر شده این دستگاه در واقع نسخه ارتقا یافته دستگاه Aero-plane می باشد که از مزایای این دستگاه نسبت به دستگاه Aero-plane می توان به موارد زیر اشاره نمود.

۱- فن های تولید نیروی پیشران در دستگاه Aero-plane ثابت می باشند. بدین معنی که این دستگاه توانایی ایجاد نیرو تنها در جهت عمود بر راستای دسته خود و به سمت پایین را دارد، در حالی که در این پروژه با قرار دادن موتور سروو^۱ در انتهای دستگاه و متصل کردن مجموعه فن ها به محور آن این قابلیت به دستگاه داده شده است که در جهات مختلف عمود بر دسته دستگاه به دست فرد نیرو و گشتاور وارد نماید که توضیحات بیشتر مربوط به این ویژگی در فصل های بعد آمده است.

۲- در دستگاه Aero-plane از یک وزنه در انتهای دستگاه به جهت نزدیک کردن مرکز جرم دستگاه به نقطه میانی دسته استفاده شده است و این در حالی است که باتری دستگاه بر روی دستگاه قرار ندارد و با سیم هایی از بیرون به دستگاه متصل شده اند. ولی در این پروژه با قرار دادن باتری در انتهای دستگاه به جای وزنه نه تنها وزن دستگاه افزایش پیدا نکرده است بلکه با استفاده از ارتباط بیسیم^۲ کلیه اتصالات از بیرون حذف گردیده که موجب آزادی حرکت بیشتر فرد شده است.

۵-۱ مراحل انجام پروژه

در مورد مراحل انجام پروژه می توان به موارد زیر اشاره نمود:

- ۱- طراحی محیط شبیه سازی حرکت گوی بر صفحه
- ۲- برنامه نویسی پردازنده دستگاه
- ۳- طراحی قطعات مکانیکی و الکترونیکی دستگاه و نحوه اتصال آن ها به یکدیگر

^۱ Servo motor

^۲ Wireless

۴- خرید و یا پرینت قطعات طراحی شده و مونتاژ

۵- به روزرسانی مشخصات سیستم فرض شده در قسمت طراحی محیط شبیه سازی و برنامه نویسی پردازنده، بر اساس سیستم ساخته شده

۶-۱ ساختار پایان نامه

این پایان نامه شامل پنج فصل می باشد. در فصل دوم به نحوه طراحی محیط شبیه سازی حرکت گوی بر صفحه و محاسبات مربوط به موقعیت گوی و زاویه صفحه پرداخته می شود. فصل سوم به معرفی قطعات الکترونیکی استفاده شده در این پروژه و توضیح برنامه نوشته شده برای پردازنده جهت ایجاد ارتباط مناسب بین این قطعات اختصاص داده شده است. در فصل چهارم نحوه طراحی قطعات سیستم و نحوه مونتاژ آن ها و دستورالعمل استفاده از دستگاه قرار داده شده، در فصل پنجم ابتدا به جمع بندی کل پروژه پرداخته شده و در نهایت پیشنهادهایی برای پروژه های آتی بیان می شود.

فصل دوم

طراحی محیط شبیه‌سازی

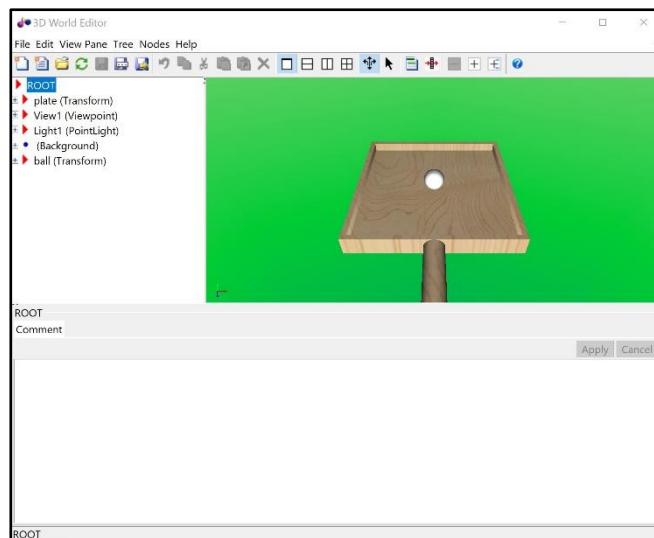
۱-۲ مقدمه

در مورد محیط شبیه‌سازی، می‌توان به هر نوع محیطی که در آن قرار است نیرویی به دست فرد وارد شود اشاره نمود و همانطور که اشاره شد محیط شبیه‌سازی شده در این پروژه به صورت حرکت یک گوی بر روی صفحه با تغییر زاویه آن توسط دست فرد می‌باشد. در این فصل ابتدا به توضیح نحوه ایجاد محیط شبیه‌سازی، سپس به توضیح محاسبات مربوط به موقعیت گوی برحسب زاویه صفحه و در نهایت به توضیح محاسبات مربوط به نیروی هر فن براساس موقعیت گوی پرداخته می‌شود.

۲-۲ ایجاد محیط شبیه‌سازی

منظور از محیط شبیه‌سازی ایجاد فضایی سه بعدی به صورت مجازی و به شکلی است که نمایانگر محیط واقعی می‌باشد. همانطور که در پیشگفتار ذکر شد، فرد مورد نظر با در دست گرفتن دستگاه با این محیط طراحی شده ارتباط برقرار می‌کند. به عنوان مدل پیشرفته‌تر این موضوع می‌توان از عینک‌های واقعیت مجازی برای حس بهتر از حضور در محیط استفاده کرد ولی در این پروژه به استفاده از صفحه نمایش رایانه بسنده شده است. محیط مورد بررسی در این پروژه به صورت یک گوی بر روی یک صفحه تخت چوبی انتخاب شده است که فرد با تغییر زاویه این صفحه می‌تواند موقعیت گوی را در صفحه تغییر دهد. در بررسی‌های انجام شده تصمیم بر آن شد

که از محیط سه بعدی^۱ نرم افزار متلب^۲ استفاده شود. اصول کار با این نرم افزار به این صورت است که می توان با ایجاد یک محیط سه بعدی و به کار بردن اجسام از کتابخانه نرم افزار و یا نرم افزارهای طراحی سه بعدی^۳ و انتخاب رنگ و سایر مشخصات مانند موقعیت، زاویه دید، نور، تصویر پس زمینه^۴ و غیره در درخت طراحی نرم افزار این محیط را با نرم افزار سیمولینک^۵ مرتبط کرد و از طریق این نرم افزار پارامترهایی نظیر موقعیت، زاویه قرار گیری و یا بزرگنمایی^۶ اجسام موجود را تغییر داد. تصویری از محیط ایجاد و ویرایش این نرم افزار در شکل ۱-۲ نشان داده شده است.



شکل ۱-۲ تصویر محیط قسمت ویرایش نرم افزار 3D animation

همانطور که در شکل ۱-۲ مشاهده می شود، اجسام استفاده شده شامل یک صفحه چوبی طراحی شده در نرم افزار سالیدورکس^۷ و یک گوی است. شعاع گوی استفاده شده ۲۰ میلی متر در نظر گرفته شده و ابعاد صفحه نیز در قسمت پیوست به طور کامل قرار داده شده است. تصویر نهایی که فرد در صفحه نمایش مشاهده می کند، در شکل ۲-۲ آورده شده است.

¹ 3D animation

² Matlab

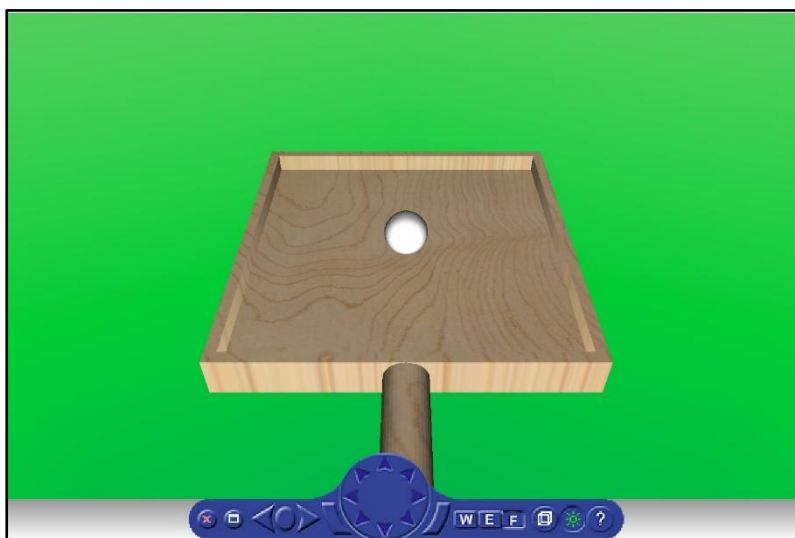
³ Computer-aided design (CAD)

⁴ Background

⁵ Simulink

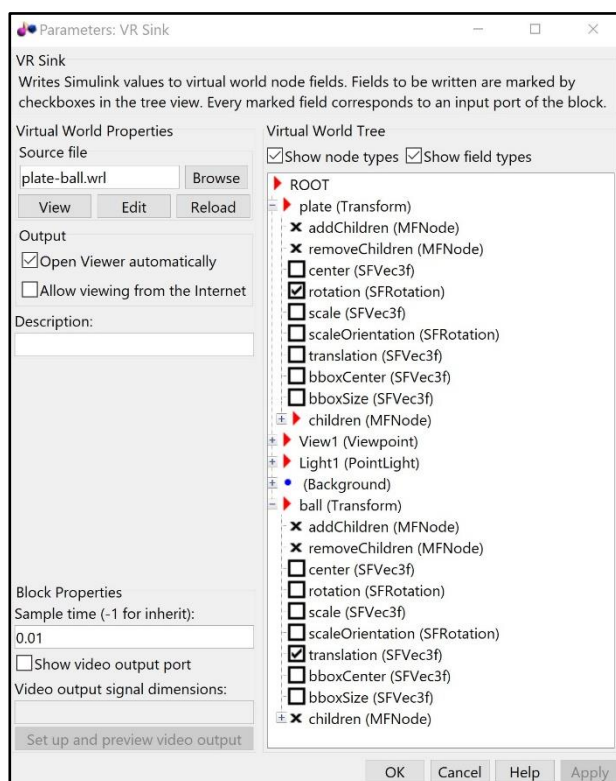
⁶ Scale

⁷ Solidworks



شکل ۲-۲ تصویر نهایی محیط شبیه سازی

همانطور که در بخش قبل اشاره شد، زاویه صفحه و موقعیت گوی را به عنوان پارامترهایی می توان با نرم افزار سیمولینک تغییر داد و کنترل کرد که نکته قابل توجه، قابلیت زنده^۱ بودن این ارتباط است که مورد

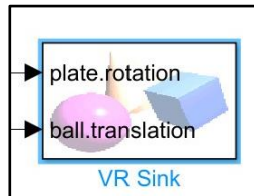


شکل ۲-۳ تنظیمات بلوک VR Sink

استفاده در این پروژه می باشد. برای این کار می توان از بلوک VR Sink در نرم افزار سیمولینک استفاده نمود که تنظیمات مربوط به این بلوک در شکل ۲-۳ مشاهده می شود. در این پنجره ابتدا باید محیطی که در بخش قبل طراحی و در مسیری ذخیره شد، انتخاب گردد. که این کار در کادر Source file از پنجره تنظیمات انجام شده است.

^۱ Realtime

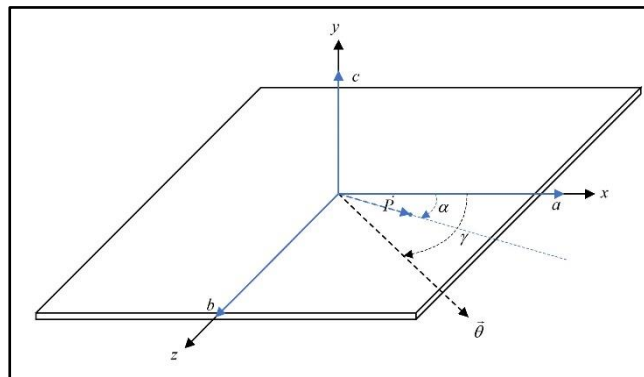
در ادامه باید پارامترهایی که باید در محیط سیمولینک تغییر یابند، انتخاب شوند که همانطور که در شکل ۳-۲ مشاهده می‌شود، از درخت طراحی گوی (Ball) گزینه انتقال (Translation) و از درخت طراحی صفحه (Plate) گزینه چرخش (Rotation) انتخاب گردد. و همانطور که در شکل ۴-۲ مشاهده می‌شوند، با وارد کردن دو سیگنال به ورودی این بلوک می‌توان این دو پارامتر را تغییر داد.



شکل ۴-۲ بلوک VR Sink

۳-۲ محاسبات مربوط به موقعیت گوی و زاویه صفحه

همانطور که در قسمت قبل اشاره شد با ورود سیگنال‌هایی به بلوک VR Sink می‌توان زاویه صفحه و موقعیت گوی را تغییر داد. اصولاً زاویه در این محیط سه‌بعدی به صورت برداری با چهار مولفه تعریف می‌شود که سه مولفه اول بردار یک‌ه‌ی چرخش در فضا و مولفه آخر به معنی میزان چرخش برحسب رادیان می‌باشد. در شکل ۵-۲ سیستم مختصات محیط شبیه‌سازی مشاهده می‌شود.



شکل ۵-۲ سیستم مختصات محیط شبیه‌سازی

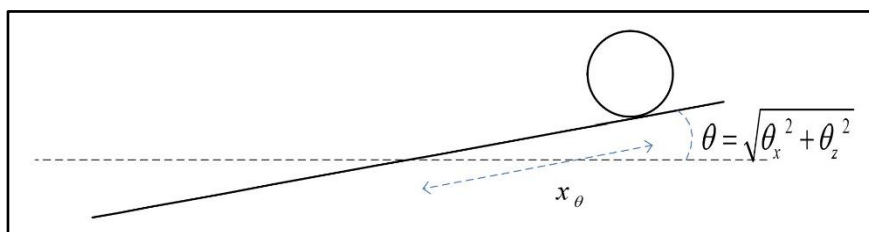
سیستم مختصات مشکی رنگ xyz مشخص شده در شکل ۵-۲، به صورت ثابت می‌باشد و زاویه صفحه برحسب این سیستم مختصات بیان می‌شود. بردار $\vec{\theta}$ مشخص شده همان بردار چرخش صفحه است که در این پروژه مولفه y این بردار صفر در نظر گرفته می‌شود و زاویه صفحه تنها به صورت زوایایی نسبت به دو محور x و z مشخص می‌شود که این زوایا به ترتیب با θ_x و θ_z نشان داده می‌شوند که در واقع این دو زاویه مولفه x و z

بردار $\vec{\theta}$ می‌باشند و اگر γ زاویه این بردار با محور x ها باشد این زاویه طبق رابطه زیر محاسبه می‌گردد:

$$\tan \gamma = \frac{\theta_z}{\theta_x} \quad (1-2)$$

در قسمت مربوط به توضیح کد میکروکنترلر آردینو^۱ به چگونگی تعیین θ_x و θ_z پرداخته خواهد شد. پس با این تفاسیر بردار ورودی به بلوک VR Sink برای زاویه‌ی صفحه به صورت $(\cos \gamma, 0, \sin \gamma, \sqrt{\theta_x^2 + \theta_z^2})$ می‌باشد. که در بخش مربوط به توضیح کد آردینو متغیرهای $\cos \gamma$ ، $\sin \gamma$ و $\sqrt{\theta_x^2 + \theta_z^2}$ به ترتیب با عنوان‌های `angx_vec`، `angz_vec` و `ang_val` مشخص شده‌اند.

برای تعیین موقعیت گوی نسبت به دستگاه کلی (xyz)، دستگاهی متصل به صفحه تحت عنوان *bac* تعریف شده است که با چرخش صفحه این دستگاه نیز می‌چرخد. پس با داشتن مختصات a و b گوی و مقدار زاویه صفحه می‌توان موقعیت گوی در فضا را توصیف کرد. برای نوشتن معادلات مربوط به مقادیر و جهت شتاب گوی با توجه به وضعیت صفحه و محل گوی، می‌توان در هر لحظه از دید عمود بر بردار $\vec{\theta}$ و گذرنده از گوی به سیستم نگاه کرد. با این تفاسیر تصویر مشاهده شده از سیستم مانند آنچه در شکل ۶-۲ مشاهده می‌شود، خواهد بود. در شکل ۶-۲ بردار $\vec{\theta}$ همیشه عمود بر شکل نشان داده شده و به سمت بیرون از صفحه می‌باشد. با توجه به آنکه صفحه حول محور $\vec{\theta}$ می‌چرخد و مولفه‌ی y این بردار در این پروژه صفر در نظر گرفته شده است، در نتیجه صفحه در نمای عمود بر بردار $\vec{\theta}$ مانند آنچه در شکل ۶-۲ نشان داده شده است همیشه به صورت یک خط زاویه‌دار دیده می‌شود که مقدار این زاویه نیز برابر با اندازه بردار $\vec{\theta}$ می‌باشد.



شکل ۶-۲ مشابهت‌سازی محیط دوبعدی با محیط یک‌بعدی

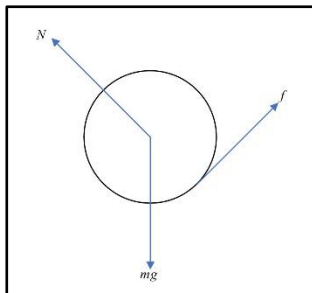
در شکل ۶-۲ مقدار x_θ بیانگر فاصله نقطه محل گوی و بردار چرخش در دستگاه مختصات *bac* می‌باشد و این مقدار در صورت داشتن موقعیت گوی و زاویه بردار چرخش با محور x ها از رابطه‌ی (۲-۲) به دست می‌آید.

¹ Arduino

$$x_{\theta} = \sqrt{a^2 + b^2} \sin(\gamma - \alpha) \quad (2-2)$$

در رابطه‌ی بالا x_{θ} فاصله‌ی گوی از بردار $\vec{\theta}$ می‌باشد و همچنین مقادیر a و b مولفه‌های موقعیت گوی در سیستم مختصات bac می‌باشد که در ادامه نحوه محاسبه این مولفه‌ها شرح داده شده است. زاویه‌ی γ در رابطه‌ی (2-2) زاویه‌ی بردار $\vec{\theta}$ و محور x ها مانند آنچه در شکل 2-5 نشان داده شده، است که از رابطه‌ی (1-2) محاسبه می‌گردد. همچنین زاویه‌ی α در رابطه‌ی (2-2) به اینصورت تعریف می‌شود که، اگر از مبدا مختصات برداری به محل اتصال گوی و صفحه رسم شود (که این بردار با نام بردار \vec{P} در شکل 2-5 نمایش داده شده است)، این زاویه برابر است با زاویه‌ی بین بردار \vec{P} و محور a . همانگونه که در شکل 2-5 نمایش داده شده است. که نحوه محاسبه این زاویه نیز در ادامه آورده شده است.

از آنجا که مقادیر a و b مجهول می‌باشند می‌توان مقدار \ddot{x}_{θ} را بر حسب θ محاسبه و با تصویر آن بر



محورهای a و b مقادیر \ddot{a} و \ddot{b} را بدست آورد. دیاگرام آزاد گوی به تنهایی در شکل 2-7 ترسیم شده است. (در شکل مقابل f نیروی اصطکاک وارد بر گوی است که زاویه‌ی آن با افق θ می‌باشد. همچنین N نیروی عمودی سطح و mg نیروی وزن گوی است.)

شکل 2-7 دیاگرام آزاد گوی

معادلات نیوتن برای گوی نشان داده شده در شکل 2-7 به این صورت نوشته می‌شوند:

$$\begin{cases} \Sigma F_{x_{\theta}} = m a_{x_{\theta}} \Rightarrow f - mg \sin \theta = m (\ddot{x}_{\theta} - x_{\theta} \dot{\theta}^2) \\ \Sigma M_G = I_G \alpha \Rightarrow fR = \frac{2}{5} m R^2 \alpha \end{cases} \xrightarrow{R\alpha = -\ddot{x}_{\theta}} g \sin \theta = -\frac{7}{5} \ddot{x}_{\theta} + x_{\theta} \dot{\theta}^2$$

در روابط بالا α شتاب زاویه‌ای گوی می‌باشد و همچنین I_G لختی دورانی گوی حول مرکز جرم آن

است که برای یک گوی به شعاع R و جرم m از رابطه‌ی $I_G = \frac{2}{5} m R^2$ محاسبه می‌گردد. روابط بالا با این

فرض نوشته شده اند که نیروی اصطکاک همیشه در صفحه نشان داده شده در شکل 2-6، یعنی عمود بر بردار $\vec{\theta}$ باشد در حالی که این نیرو همیشه در راستای حرکت نسبی جسم است و ممکن است الزاما سرعت جسم در صفحه نشان داده شده در شکل 2-6 و یا به عبارت دیگر سرعت جسم بر بردار چرخش عمود نباشد، ولی با یک تقریب می‌توان این فرض را صحیح در نظر گرفت. همچنین می‌توان با در نظر گرفتن یک ثابت و تاثیر آن بر شتاب گرانش که بیانگر نیروی اصطکاکی که از غیر الاستیک بودن دو سطح نتیجه می‌شود، است، می‌توان شتاب گوی

را از رابطه زیر محاسبه نمود.

$$\ddot{x}_\theta = \frac{5}{7} \left(-(C \times g) \sin \theta + x_\theta \dot{\theta}^2 \right) \quad (3-2)$$

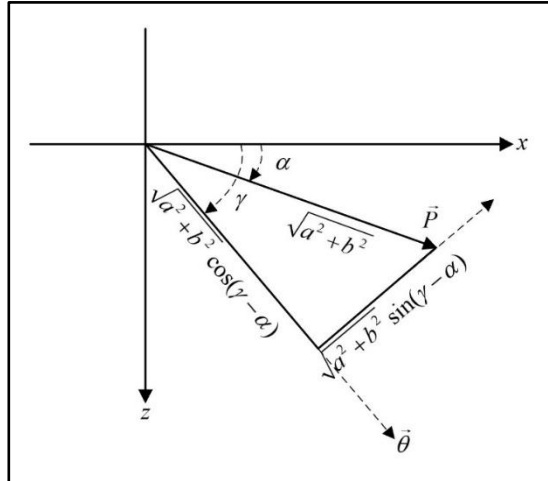
از آنجا که فرض می‌شود که گوی از صفحه جدا نگردد در این قسمت نیازی به نوشتن مولفه عمود بر صفحه‌ی گوی نیست ولی این مولفه در قسمت محاسبه‌ی مربوط به نیروی فن‌ها لحاظ خواهد شد. معادله دیفرانسیل رابطه بالا به صورت عددی توسط برد آردیونو حل می‌شود که در قسمت بررسی کد آردیونو به توضیح آن پرداخته خواهد شد. همانطور که گفته شد جهت این شتاب عمود بر محور چرخش صفحه می‌باشد پس با تصویر کردن این شتاب بر روی دو محور a و b مقادیر شتاب در صفحه‌ی گوی در مختصات bac از روابط زیر بدست می‌آیند:

$$\begin{cases} \ddot{a} = \ddot{x}_\theta \sin \gamma \\ \ddot{b} = -\ddot{x}_\theta \cos \gamma \end{cases} \quad (4-2)$$

با دوبار انتگرال‌گیری از دو رابطه بالا مقادیر a و b که مختصات گوی در دستگاه مختصات bac می‌باشند به دست می‌آیند. نحوه انتگرال‌گیری به صورت عددی می‌باشد که در بخش مربوط به توضیح کد آردیونو به توضیح آن پرداخته خواهد شد. و اکنون کافیت مختصات گوی در دستگاه xyz را به صورت تابعی از a ، b و $\sqrt{\theta_x^2 + \theta_y^2}$ بدست آید. برای مثال طریقه محاسبه مختصه x در ادامه توضیح داده خواهد شد و برای سایر مختصه‌ها به همان شکل قابل محاسبه خواهند بود.

برای نوشتن مختصه‌ی x گوی بر حسب مقادیر a ، b و $\sqrt{\theta_x^2 + \theta_y^2}$ از شکل ۲-۸ که در امتداد خلاف محور y صفحه‌ی xz را نشان می‌دهد، استفاده می‌شود. همانطور که گفته شد، بردار \vec{P} ، بردار رسم شده از مبدا مختصات به محل تماس گوی و صفحه می‌باشد و از آنجایی که مقادیر a و b مولفه‌های این بردار در دستگاه مختصات bac هستند، پس همانطور که در شکل ۲-۸ مشاهده می‌گردد، طول این بردار از رابطه‌ی $|\vec{P}| = \sqrt{a^2 + b^2}$ به دست می‌آید. حال اگر بردار \vec{P} مطابق شکل ۲-۸ بر روی بردار $\vec{\theta}$ و برداری عمود بر بردار $\vec{\theta}$ و گذرنده از محل اتصال گوی و صفحه (انتهای بردار \vec{P}) تصویر شود، با چرخش صفحه حول بردار $\vec{\theta}$ ، تنها بردار با طول $\sqrt{a^2 + b^2} \sin(\gamma - \alpha)$ که در شکل ۲-۸ مشخص شده است حول بردار $\vec{\theta}$ دوران می‌کند. از آنجا که مقدار دوران برابر با $\sqrt{\theta_x^2 + \theta_z^2}$ می‌باشد، اگر بردار دوران یافته را در صفحه‌ی xz تصویر شود، طولی برابر با $\sqrt{a^2 + b^2} \sin(\gamma - \alpha) \cos \sqrt{\theta_x^2 + \theta_z^2}$ خواهد داشت. در نتیجه در صفحه‌ی xz دو بردار با طول‌های $\sqrt{a^2 + b^2} \cos(\gamma - \alpha)$ و $\sqrt{a^2 + b^2} \sin(\gamma - \alpha) \cos \sqrt{\theta_x^2 + \theta_z^2}$ خواهیم داشت که بردار با طول

شکل ۸-۲ همانطور که در شکل ۸-۲ مشاهده می‌گردد در امتداد بردار $\vec{\theta}$ و بردار با طول $\sqrt{a^2 + b^2} \cos(\gamma - \alpha)$ ، تصویر بردار دوران یافته با طول $\sqrt{a^2 + b^2} \sin(\gamma - \alpha) \cos \sqrt{\theta_x^2 + \theta_z^2}$ (که در شکل ۸-۲ مشخص شده است) در صفحه‌ی xz می‌باشد.



شکل ۸-۲ تصویر از بالای صفحه xz

حال با تصویر بردارهای به طول $\sqrt{a^2 + b^2} \cos(\gamma - \alpha)$ و $\sqrt{a^2 + b^2} \sin(\gamma - \alpha) \cos \sqrt{\theta_x^2 + \theta_z^2}$ روی محور x ها، مختصه x مربوط به نقطه تماس گوی و صفحه به دست می‌آید:

$$x = \sqrt{a^2 + b^2} \cos(\gamma - \alpha) \cos \gamma + \sqrt{a^2 + b^2} \sin(\gamma - \alpha) \cos \sqrt{\theta_x^2 + \theta_z^2} \sin \gamma \quad (5-2)$$

توجه شود که α زاویه بین بردار \vec{P} و محور x ها در مختصات bac می‌باشد و از رابطه زیر قابل محاسبه است:

$$\alpha = \tan^{-1} \left(\frac{b}{a} \right) \quad (6-2)$$

این مختصه (مختصه‌ی x گوی در دستگاه مختصات xyz) با این فرض محاسبه شده است شعاع گوی صفر است بدین معنا که محاسبات انجام شده مختصات نقطه تماس گوی و صفحه را می‌دهند. حال با در نظر گرفتن شعاع گوی و نوشتن معادلات به همین ترتیب برای سایر مختصه‌ها می‌توان مختصات مرکز گوی را در دستگاه مختصات xyz از روابط زیر محاسبه نمود:

$$\left\{ \begin{array}{l} x = \sqrt{a^2 + b^2} \cos(\gamma - \alpha) \cos \gamma + \sqrt{a^2 + b^2} \sin(\gamma - \alpha) \cos \sqrt{\theta_x^2 + \theta_z^2} \sin \gamma \\ \quad - R \sin \sqrt{\theta_x^2 + \theta_z^2} \sin \gamma \\ y = \sqrt{a^2 + b^2} \sin(\gamma - \alpha) \sin \sqrt{\theta_x^2 + \theta_z^2} \\ \quad + R \cos \sqrt{\theta_x^2 + \theta_z^2} \\ z = \sqrt{a^2 + b^2} \cos(\gamma - \alpha) \sin \gamma - \sqrt{a^2 + b^2} \sin(\gamma - \alpha) \cos \sqrt{\theta_x^2 + \theta_z^2} \cos \gamma \\ \quad + R \sin \sqrt{\theta_x^2 + \theta_z^2} \cos \gamma \end{array} \right. \quad (7-2)$$

که در روابط بالا R شعاع گوی می‌باشد. و به این صورت بردار ورودی به بلوک VR Sink برای موقعیت گوی به صورت (x, y, z) می‌باشد.

تمامی روابط ذکر شده در صورت در دسترس داشتن θ_x و θ_z قابل محاسبه هستند و در بخش مربوط به استخراج زاویه از سنسور ژيروسکوپ به چگونگی محاسبه این دو زاویه پرداخته می‌شود.

۲-۴ محاسبات مربوط به نیروی هر فن

همانطور که گفته شد برای ایجاد حس بهتر از محیط در فرد هر کدام از فن‌ها باید با توجه به اینکه گوی در چه نقطه‌ای از صفحه قرار دارد و اینکه زاویه صفحه به چه صورت است نیروی لازم را به دست فرد وارد کنند که در ادامه طریقه محاسبه نیروی هر فن توضیح داده خواهد شد.

با توجه به اینکه در این سیستم نیروی گرانش مربوط به وزن صفحه همیشه به سمت پایین می‌باشد در نتیجه می‌توان زاویه‌ی موتور سروو را به گونه‌ای تنظیم نمود که جهت نیروی این فن‌ها همیشه به سمت شتاب گرانش باشد. این امر در راستای محور z قابل تحقق است (با توجه به سیستم مختصات نشان داده شده در شکل ۲-۵) چراکه با توجه به قرار داده شدن یک موتور سروو در انتهای دستگاه می‌توان زاویه فن‌ها را نسبت به دست فرد به گونه‌ای در صفحه xy تغییر داد که در این صفحه زاویه این فن‌ها همیشه در راستای عمودی باشد ولی از آنجایی که امکان چرخش فن‌ها در صفحه yz وجود ندارد این امر را می‌توان خطای سیستم ما در نظر گرفت که در نسخه‌های بعدی آن می‌توان با افزودن این قابلیت به دستگاه از این خطا جلوگیری کرد.

در بخش قبل برای دیاگرام آزاد نشان داده شده در شکل ۷-۲، قانون دوم نیوتن در راستای x_θ که راستای نشان داده شده در شکل ۶-۲ است، نوشته شد. حال برای به دست آوردن نیروی اصطکاک و نیروی عمودی سطح قانون دوم نیوتن در راستای عمود بر x_θ ، که با z_θ نمایش داده می شود، نوشته می شود.

$$\begin{cases} \Sigma F_{z_\theta} = ma_{z_\theta} \Rightarrow N - mg \cos \theta = m(x_\theta \ddot{\theta} + 2\dot{x}_\theta \dot{\theta}) \\ \Sigma M_G = I_G \alpha \Rightarrow fR = \frac{2}{5}mR^2 \alpha \end{cases}$$

که در روابط بالا α شتاب زاویه ای گوی می باشد و از آنجا که $R\alpha = -\ddot{x}_\theta$ می توان نیروی عمودی سطح و اصطکاک را محاسبه نمود.

$$\begin{cases} N = mg \cos \theta + m\ddot{x}_\theta + 2m\dot{x}_\theta \dot{\theta} \\ f = -\frac{2}{5}m\ddot{x}_\theta \end{cases} \quad (۸-۲)$$

و باز هم تاکید می گردد که روابط بالا با این فرض نوشته شده اند که نیروی اصطکاک همیشه بر بردار چرخش (بردار $\vec{\theta}$) عمود باشد.

مولفه های دو نیروی محاسبه شده در بالا در مختصات bac به آسانی قابل محاسبه هستند چراکه نیروی عمودی سطح همیشه بر صفحه عمود است و در نتیجه همیشه در راستای محور c می باشد و همچنین نیروی اصطکاک همیشه در صفحه می باشد در نتیجه همیشه با صفحه ab موازی است. اما برای نوشته مولفه های این نیروها در مختصات xyz ، بهتر است که این دو نیرو در دو راستای افقی و عمودی (F_h و F_v) تصویر شوند. از آنجا که راستای نیروی اصطکاک همیشه در راستای x_θ فرض شد در نتیجه جهت نیروی اصطکاک وارده به صفحه در هر نقطه در جهت \dot{x}_θ در نظر گرفته می شود. در اینصورت دو نیروی عمودی (F_v) و نیروی افقی (F_h) را می توان از روابط زیر محاسبه نمود.

$$\begin{cases} F_v = N \cos \theta - \frac{|\dot{x}_\theta f|}{\dot{x}_\theta} \sin \theta \\ F_h = N \sin \theta + \frac{|\dot{x}_\theta f|}{\dot{x}_\theta} \cos \theta \end{cases} \quad (۹-۲)$$

که در روابط بالا نیروی عمودی به سمت پایین و نیروی افقی در صفحه xz نشان داده شده در شکل ۵-۲ می باشند. دلیل استفاده از قدرمطلق تغییر علامت مولفه مربوط به نیرو اصطکاک با توجه به علامت \dot{x}_θ می باشد.

حال باید گشتاور این نیروها حول دو محور x و z برابر گشتاور دو نیروی فن حول این دو محور و همچنین نیروی عمودی ایجاد شده توسط دو فن و نیروی عمودی محاسبه شده برابر باشد که تحقق این ۳ معادله با ۲ مجهول امکان پذیر نمی باشد در نتیجه در روند محاسبه نیروی دو فن تنها گشتاورهای وارده حول دو محور x و z را در نظر می گیریم و از مقدار نیروی عمودی آن چشم پوشی می شود که این نیز از معایب این سیستم به شمار می آید و راه حل برطرف کردن آن استفاده از دو فن دیگر در انتهای دستگاه می باشد که با توجه به هزینه تمام شده سیستم از به کارگیری دو فن دیگر صرف نظر شده است، هر چند این اختلاف در نیروی عمودی تاثیر زیادی بر نتیجه سیستم نخواهد گذاشت [1].

همچنین همانطور که در فصل ۴ خواهیم دید مرکز جرم دستگاه و مرکز جرم صفحه شبیه سازی شده ممکن است برابر نباشند. در نتیجه می توان گشتاوری که ناشی از این اختلاف مرکز جرم ها می باشد را با $M_{C.M}$ نشان داد که این گشتاور را می توان از رابطه زیر محاسبه نمود.

$$M_{C.M} = Plate_{C.M} \times Plate_{Mass} \times g - Device_{C.M} \times Device_{Mass} \times g \quad (10-2)$$

که در رابطه ی بالا $Plate_{C.M}$ و $Device_{C.M}$ فاصله مرکز جرم های صفحه شبیه سازی شده و دستگاه از مرکز دسته می باشد و همچنین $Plate_{Mass}$ و $Device_{Mass}$ جرم صفحه شبیه سازی شده و دستگاه است.

در نتیجه با نوشتن این معادلات گشتاور در مختصات xyz گذرنده از مرکز دسته دستگاه می توان نیروی فن راست (F_r) و نیروی فن چپ (F_l) را از روابط زیر محاسبه نمود.

$$\begin{cases} F_r = F_v \left(\frac{p_1}{2L_x} - \frac{p_3}{2L_z} \right) + F_h \left(\frac{p_2 \sin \gamma}{2L_x} + \frac{p_2 \cos \gamma}{2L_z} \right) + \frac{M_{C.M}}{2L_z} \\ F_l = F_v \left(-\frac{p_1}{2L_x} - \frac{p_3}{2L_z} \right) + F_h \left(-\frac{p_2 \sin \gamma}{2L_x} + \frac{p_2 \cos \gamma}{2L_z} \right) + \frac{M_{C.M}}{2L_z} \end{cases} \quad (11-2)$$

که در روابط بالا L_x و L_z به ترتیب فاصله در راستای x و z هریک از فن ها تا نقطه وسط دسته دستگاه می باشند و مقادیر p_1 ، p_2 و p_3 مختصات نقطه تماس گوی و صفحه در دستگاه مختصات xyz می باشند که می توان با قرار دادن $R = 0$ این مقادیر را از معادلات (۷-۲) بدست آورد. همچنین F_v و F_h استفاده شده در روابط (۱۱-۲)، همان نیروهای عمودی و افقی محاسبه شده طبق روابط (۹-۲) می باشند.

۵-۲ جمع‌بندی

همانطور که گفته شد محیط شبیه‌سازی شده در این پروژه به صورت حرکت گوی بر روی یک صفحه می‌باشد به‌طوری‌که زاویه صفحه توسط دست فرد قابل کنترل می‌باشد. محیط شبیه‌سازی شده در این پروژه در نرم‌افزار سیمولینک اجرا شده است و محاسبات مربوط به موقعیت گوی بر اساس زاویه صفحه را می‌توان در این فصل مشاهده نمود و همچنین در انتهای فصل نحوه محاسبه نیروی هر فن برای شبیه‌سازی نیروی حرکت گوی بر صفحه با مساوی قرار دادن گشتاورهای ایجاد شده در محیط شبیه‌سازی شده به دست فرد و گشتاورهای ایجاد شده توسط هر فن، بیان شده است.

فصل سوم

برنامه‌نویسی پردازنده

۱-۳ مقدمه

پردازنده مورد استفاده در این پروژه از نوع آردیونو نانو^۱ می‌باشد. همچنین برای قسمت‌های مختلف پروژه از ماژول‌های مختلفی استفاده شده است که برای هر کدام کد آردیونو مربوطه نیز به طور کامل توضیح داده خواهد شد. در جدول ۱-۳ می‌توان لیست کلیه قطعات الکترونیکی استفاده شده را مشاهده نمود.

جدول ۱-۳ بردها و ماژول‌های الکترونیکی استفاده شده در دستگاه

ردیف	نام قطعه (فارسی)	نام قطعه (انگلیسی)
۱	آردیونو نانو با پردازنده Atmega328	Arduino Nano (Atmega328)
۲	ماژول ۹ محوره MPU9250	MPU9250 IMU module
۳	ماژول بلوتوث HC-05	HC-05 Bluetooth module
۴	ماژول تغذیه کاهنده XL4015E	XL4015E Buck converter

در ادامه به توضیح هر کدام از این قطعات می‌پردازیم.

^۱ Arduino Nano

۲-۳ برد آردیونو

آردیونو یک بستر الکترونیکی متن باز برای راحت تر کردن ارتباط بین نرم افزار و سخت افزار می باشد. شیوه کار این برد به این صورت است که ابتدا ورودی هایی را از سنسورهای مختلف دریافت کرده و سپس با انجام عملیات های مختلف بر روی آن ها دستورهای مورد نظر را به عملگرهای خروجی می دهد و چندین سال است که این ابزار در زمینه های بسیار وسیع از جمله علمی و آموزشی، سرگرمی، هنری، برنامه نویسی و غیره مورد استفاده قرار گرفته است [16]. برای کار با این برد باید دستورات نوشته شده به زبان برنامه نویسی C در محیط نرم افزار آردیونو^۱ در حافظه برد یکبار بار گذاری شوند و پس از آن برد با توجه به دستورات نوشته شده عملیات های مختلف را انجام می دهد.

در این پروژه مسیر و روند اجرای عملیات به این شکل می باشد که ابتدا برد آردیونو زاویه دست فرد را از طریق سنسور شتاب سنج^۲ وژیروسکوپ^۳ دریافت کرده و سپس محاسبات مربوط به زاویه صفحه و موقعیت گوی را همانطور که در بخش ۲-۳ توضیح داده شد، انجام می دهد. از این داده ها در برد برای محاسبه سرعت های دو فن و همچنین زاویه موتور سروو استفاده شده و همچنین این داده ها از طرق سنسور بلوتوث به رایانه و نرم افزار متلب جهت به نمایش گذاشتن محیط شبیه سازی، ارسال می شوند و به همین طریق این چرخه ادامه پیدا می کند.

۱-۲-۳ توضیح بخش های مختلف کد آردیونو

در این بخش ابتدا قسمت های مختلف کد آردیونو آورده شده و در مورد هر قسمت توضیح می دهیم.

```
1. #include "MPU9250.h"
2. #include <AltSoftSerial.h>
3. #include <ServoTimer2.h>
4.
5. MPU9250 IMU(Wire, 0x68);
6. int status;
7.
8. AltSoftSerial Bluetooth(8, 9); // RX , TX
9.
10. ServoTimer2 Right_DuctedFan;
11. ServoTimer2 Left_DuctedFan;
12. ServoTimer2 Sweeper;
13.
14. #define a_max 120.16 // mm
15. #define a_min -120.16 // mm
16. #define b_max -104.84 // mm
17. #define b_min -345.16 // mm
18. #define ball_radius 20 // mm
19. #define plate_thickness 0 // mm
```

^۱ IDE

^۲ Accelerometer

^۳ Gyroscope

```

20. #define ball mass 0.08 // Kg
21. #define g 9.81 // m/s^2
22. #define Lx 75 // mm
23. #define Lz 150 // mm
24. #define Device_CM 58.87 // mm
25. #define Plate_CM 208.11 // mm
26. #define Device_Mass 0.986 // Kg
27. #define Plate_Mass 0.38 // Kg
28. #define Wgyro 15
29. #define pulse_efficiency 0.3
30. #define C 0.5
31. #define Angle_Filter_Const 0.6
32.
33. float preRateAxz, preRateAyz, curRateAxz, curRateAyz;
34. float RxEst = 0, RyEst = 0, RzEst = -1;
35. double previousTime, currentTime, elapsedTime;
36. float theta_x, theta_z, theta = 0, dtheta = 0, ddtheta, pretheta, predtheta;
37. float xtheta = 0, dxtheta, ddxtheta, prextheta;
38. float gama, alfa;
39. float a = 0, da = 0;
40. float b = -225, db = 0;
41. float d;
42. float x = 0, y = 20, z = -225;
43. int Right_FanSpeed, Left_FanSpeed, Sweep_Angle = 1500, F_Sweep_Angle = 1500;
44. int i = 1;
45. const int switchpin = 6;

```

در خطوط ۱، ۲ و ۳ کتابخانه‌هایی برای خواندن سنسور شتاب‌سنج، ایجاد ارتباط سریال با ماژول بلوتوث و استفاده از موتور سروو اضافه شده است که کتابخانه AltSoftSerial، ServoTimer2 و MPU9250 از اینترنت قابل دریافت هستند [18] [17] [19]. دلیل استفاده از کتابخانه‌های AltSoftSerial و ServoTimer2 به جای کتابخانه‌های موجود در خود آردیونو (SoftwareSerial و Servo) این است که این دو کتابخانه که در خود نرم‌افزار آردیونو قرار دارند از یک شمارنده^۱ در آردیونو استفاده می‌کنند که موجب اختلال در سیستم می‌شود ولی با استفاده از دو کتابخانه دیگر که ذکر شد، می‌توان برای هر کدام از این کتابخانه‌ها در برد آردیونو شمارنده تعریف کرد تا موجب تداخل نشوند. در خط ۵ یک کلاس تحت عنوان IMU به صورت ارتباط سریال و یا I2C تعریف می‌شود که 0x68 آدرس ماژول برای ارتباط می‌باشد که این آدرس را می‌توان از دیتاشیت ماژول استخراج کرد. در خط ۸ کلاسی تحت عنوان Bluetooth تعریف می‌شود به گونه‌ای که دو پین ۸ و ۹ برد آردیونو را به عنوان RX و TX برای ارسال و دریافت اطلاعات با ماژول بلوتوث، معرفی می‌کند. در خطوط ۱۰، ۱۱ و ۱۲ سه نام برای کلاس ServoTimer2 انتخاب می‌شود که Right_DuctedFan مربوط به موتور سمت راست، Left_DuctedFan مربوط به موتور سمت چپ و Sweeper مربوط به موتور سروو تعبیه شده در عقب دستگاه برای تغییر زاویه دو موتور جلو می‌باشد. از خط ۱۴ تا ۳۱ نیز برخی خصوصیات محیط شبیه‌سازی مانند اندازه صفحه، شعاع گوی، ضریب بازگشت و غیره تعریف می‌شوند. در خطوط ۳۳ تا ۴۵ نیز متغیرهایی که در برنامه استفاده می‌شوند تعریف شده اند.

^۱ Timer

```

1. void setup() {
2.
3.     Serial.begin(9600);
4.     Bluetooth.begin(9600);
5.
6.     while (!Serial) {}
7.
8.     // start communication with IMU
9.     status = IMU.begin();
10.    if (status < 0) {
11.        Serial.println("IMU initialization unsuccessful");
12.        Serial.println("Check IMU wiring or try cycling power");
13.        Serial.print("Status: ");
14.        Serial.println(status);
15.        while (1) {}
16.    }
17.
18.    delay(5000);
19.
20.    Right_DuctedFan.attach(4);
21.    Left_DuctedFan.attach(3);
22.    Sweeper.attach(2);
23.
24.    pinMode(switchpin, INPUT);
25.
26.    // setting the accelerometer full scale range to +/-2G
27.    IMU.setAccelRange(MPU9250::ACCEL_RANGE_2G);
28.    // setting the gyroscope full scale range to +/-500 deg/s
29.    IMU.setGyroRange(MPU9250::GYRO_RANGE_500DPS);
30.
31. }

```

برنامه‌نویسی آردیونو از دو بخش مهم `setup` و `loop` تشکیل می‌شود که در بخش `setup` کدهای نوشته شده تنها یکبار اجرا می‌شوند و بیشتر به تنظیمات سیستم اختصاص داده می‌شود ولی دستورات بخش `loop` تا زمانی که برد روشن باشد دائما تکرار می‌شوند. در کد بالا همانطور که مشاهده می‌شود دستورات در تابع `void setup ()` نوشته شده‌اند. در خطوط ۳ و ۴ سرعت انتقال داده برای هر کدام از پورت‌های `Serial` و `Bluetooth` مشخص شده است. از خط ۶ تا ۱۶ یک حلقه به گونه‌ای قرار داده شده است که وضعیت ارتباط با سنسور شتاب‌سنج را بررسی می‌کند و در صورت نداشتن مشکل اجازه خروج از حلقه و ادامه پیدا کردن روند برنامه داده می‌شود. در خط ۱۸ از دستوری استفاده شده است که اجرای برنامه را در این نقطه به مدت ۵ ثانیه (۵۰۰۰ میلی ثانیه) متوقف می‌کند که دلیل آن این است که با متصل کردن کلیه مدارات به باتری برد آردیونو و موتورهای چرخاننده فن‌ها همزمان کار خود را آغاز می‌کنند و موتورها باید تقریبا به مدت ۷ ثانیه از شروع اتصال به باتری‌شان گذشته باشد تا بتوان آن‌ها را تعریف کرده و مقدار سرعت چرخش‌شان را به آن‌ها وارد کنیم و از آنجایی که شروع به کار کردن برد آردیونو تقریبا مقدار ۴ ثانیه از لحظه متصل کردن برد به باتری می‌گذرد، با در نظر گرفتن یک مقدار بیشتر برای شروع به کار موتورها جهت جلوگیری از اختلال احتمالی می‌توان در این لحظه از برنامه و قبل از تعریف پین‌های موتورها به مدت ۵ ثانیه روند ادامه برنامه را متوقف نمود. در خطوط ۲۰ تا ۲۲ مشخص شده است که هر کدام از موتورها و همچنین سروو موتور به کدام پین‌های آردیونو متصل شده‌اند که

نحوه اتصال کلیه مدارات را می توان در بخش ۴-۲-۲ مشاهده نمود. در خط ۲۴ پینی که به عنوان بررسی وضعیت کلید فعال سازی فن ها مشخص شده است به عنوان ورودی تعریف می شود. در خط ۲۷ محدوده محاسبه شتاب تا ± 2 برابر شتاب جاذبه تعیین شده و در خط ۲۹ نیز محدوده محاسبه سرعت زاویه ای که توسط سنسورژیروسکوپ محاسبه می شود نیز $\pm 500(\text{deg/s})$ تعیین می شود.

```

1. void loop() {
2.
3.   if (i == 1) {
4.     currentTime = millis();
5.     delay(40);
6.     i = 2;
7.   }
8.
9.   previousTime = currentTime;
10.  currentTime = millis();
11.  elapsedTime = (currentTime - previousTime) / 1000;
12.
13.  sensor();
14.
15.  ballposition();
16.
17.  forcecalculator();
18.
19.  sendtoSimulink();
20.
21.  sendtoDevice();
22.
23.  delay(20);
24.
25. }

```

همانطور که گفته شد دستورات مربوط به قسمت loop به صورت نامحدود تکرار می شوند. در روند اجرای برنامه مقداری به عنوان تغییرات زمانی که در بخش های مربوط به انتگرال گیری عددی از آن استفاده می شود محاسبه می شوند به گونه ای که ابتدا مقادیر زمان کنونی (currentTime) در متغیر زمان پیشین (previousTime) ذخیره می شود و سپس زمان کنونی با توجه به دستور millis() استخراج شده و در متغیر زمان کنونی ذخیره می گردد و در نهایت متغیر تغییرات زمان (elapsedTime) از رابطه زیر قابل محاسبه می باشد:

$$elapsedTime = \frac{currentTime - previousTime}{1000} \quad (۱-۳)$$

که دلیل تقسیم شده عبارت بر ۱۰۰۰ این است که زمانی که از دستور millis() استخراج می گردد بر حسب میلی ثانیه می باشد که باید به ثانیه تبدیل گردد. و از آنجایی که از لحظه شروع به کار برد تا شروع حلقه این تابع در حال شمارش زمان می باشد برای جلوگیری از بزرگ شدن ثابت زمانی در لحظه اول نسبت به بقیه زمان ها در خطوط ۳ تا ۷ از یک دستور شرطی به گونه ای استفاده شده است که تنها یکبار و در اولین مرحله اجرای حلقه

اجرا می‌شود و مقدار در لحظه زمان را در متغیر `currentTime` ذخیره می‌کند و پس از متوقف کردن برنامه به مدت ۴۰ میلی‌ثانیه از دستور شرطی خارج شده و در حلقه‌های بعدی مقدار این ثابت زمانی توسط خطوط ۹ تا ۱۱ محاسبه می‌شود.

در ادامه برای جلوگیری از شلوغ شدن کد، هر بخش از وظایف برد آردیونو در تابع مربوطه نوشته و در قسمت `loop` تنها به تابع مورد نظر اشاره شده است.

در خط ۱۳ از تابعی تحت عنوان `sensor()` استفاده شده است که وظیفه آن خواندن زاویه از سنسور شتاب‌سنج می‌باشد. در خط ۱۵ نیز از تابعی برای محاسبه موقعیت گوی بر اساس روابط (۲-۷) استفاده شده است. در خط ۱۷ از تابعی استفاده شده است که وظیفه آن محاسبه سرعت و نیروی تولیدی هر کدام از فن‌های دستگاه می‌باشد. تابع مربوط به خط ۱۹ اطلاعات مربوط به زاویه صفحه و موقعیت گوی را به نرم‌افزار سیمولینک می‌فرستد و تابع مربوط به خط ۲۱ اطلاعات مربوط به سرعت هر فن و موقعیت موتور سروو را به موتور مربوط به خود ارسال می‌کند. هر کدام از این تابع‌ها در بخش مربوط به خود به طول کامل توضیح داده خواهند شد. خط ۲۳ نیز دسترسی به جهت متوقف کردن برد برای مدت ۲۰ میلی‌ثانیه آورده شده است که در بخش مربوط به مازول بلوتوث (بخش ۳-۴) در مورد این میزان مکث بحث خواهد شد.

۲-۲-۳ تابع محاسبه موقعیت گوی (`ballposition()`)

محاسبات مربوط به این تابع به‌طور کامل در بخش ۲-۳ توضیح داده شده است و در این قسمت تنها به توضیح کد نوشته شده در کادر زیر پرداخته می‌شود.

```
1. void ballposition() {
2.
3.     gama = atan2(theta_z , theta_x);
4.
5.     if ((a >= a_max && da > 0) || (a <= a_min && da < 0)) {
6.         da = -1 * pulse_efficiency * da;
7.     }
8.     else {
9.         if (a < a_max && a > a_min) {
10.            ddxtheta = -(C * g) * 1000 * sin(theta) + xtheta * pow(dtheta, 2)) * 5 / 7
11.        ;
12.        }
13.        else {
14.            ddxtheta = 0;
15.        }
16.        float dda = ddxtheta * sin(gama);
17.        da = da + dda * elapsedTime;
18.    }
19.    a = a + da * elapsedTime;
20.    if ((b >= b_max && db >= 0) || (b <= b_min && db <= 0)) {
21.        db = -1 * pulse_efficiency * db;
```



```

22.     }
23.     else {
24.         if (b < b_max && b > b_min) {
25.             ddxtheta = -(C * g) * 1000 * sin(theta) + xtheta * pow(dtheta, 2)) * 5 / 7
26.         };
27.         else {
28.             ddxtheta = 0;
29.         }
30.         float ddb = -1 * ddxtheta * cos(gama);
31.         db = db + ddb * elapsedTime;
32.     }
33.     b = b + db * elapsedTime;
34.
35.     alfa = atan2(b, a);
36.     d = sqrt(pow(a, 2) + pow(b, 2));
37.
38.     prextheta = xtheta;
39.     xtheta = d * sin(gama - alfa);
40.     dxtheta = (xtheta - prextheta) / elapsedTime;
41.
42.     x = d * cos(gama - alfa) * cos(gama) + d * sin(gama - alfa) * cos(theta) *
        sin(gama) - (ball_radius + plate_thickness) * sin(theta) * sin(gama);
43.
44.     y = d * sin(gama - alfa) * sin(theta) + (ball_radius + plate_thickness) *
        cos(theta);
45.
46.     z = d * cos(gama - alfa) * sin(gama) - d * sin(gama - alfa) * cos(theta) *
        cos(gama) + (ball_radius + plate_thickness) * sin(theta) * cos(gama);
47.
48. }

```

تابع تعریف شده برای این محاسبات از نوع void می‌باشد و بدین معناست که در نهایت با اجرای دستورات تابع خروجی خاصی بازگشت داده نمی‌شود و صرفاً هدف از وارد کردن این دستورات در یک تابع منظم‌تر و قابل فهم‌تر کردن کد می‌باشد در غیر اینصورت این دستورات می‌توانستند در تابع loop نوشته شوند.

همانطور که مشاهده می‌شود نام تابع ballposition می‌باشد و متغیری برای ورودی انتخاب نشده است که به این معنی است که می‌توان از متغیرهای کلی^۱ که در نرم‌افزار ایجاد می‌شوند استفاده کنیم. در خط ۳ زاویه γ طبق رابطه (۱-۲) قابل محاسبه است و دلیل استفاده از تابع atan2() به جای تابع atan() این است که با منفی شدن علامت θ_x یا θ_z در صورت استفاده از تابع atan() که به معنی معکوس تابع تانژانت می‌باشد زاویه γ منفی می‌شود، در صورتی که ممکن است این زاویه در ربع دوم و یا سوم صفحه مختصات باشد که این مشکل در تابع atan2() حل می‌شود. از خط ۵ تا ۱۷ از یک دستور شرطی استفاده شده است که دلیل آن تعریف دیواره برای صفحه می‌باشد، بدین صورت که اگر در هر لحظه از محاسبات مقدار a که موقعیت گوی بر حسب صفحه است از محدوده خود خارج شود مقدار da که سرعت گوی بر حسب صفحه و در جهت بردار a می‌باشد به صورت ضریب منفی‌ای از سرعت مرحله قبل گوی که به ضریب بازگشت نامیده می‌شود و در قسمت اولیه کد تعریف شده است، بیان می‌شود که موجب برگشت گوی به داخل محدوده صفحه می‌شود و در غیر اینصورت

^۱ Global

اگر گوی داخل صفحه باشد مقدار شتاب آن (dda) طبق رابطه (۲-۴) و (۲-۳) محاسبه می‌شود و با انتگرال‌گیری خطی همانطور که در خط ۱۶ مشاهده می‌شود مقدار سرعت گوی (da) قابل محاسبه است. و در نهایت با انتگرال‌گیری خطی از da همانطور که در خط ۱۸ مشاهده می‌شود، مقدار a قابل محاسبه است. روند محاسبه b نیز دقیقاً مانند محاسبه a است که در خطوط ۲۰ تا ۳۳ قابل مشاهده است. در خط ۳۵ مقدار زاویه α طبق رابطه (۲-۶) محاسبه می‌شود و در خط ۳۶ برای راحتی مقدار $\sqrt{a^2 + b^2}$ برابر با متغیر d در نظر گرفته شده است. در خط ۳۹ مقدار x_θ طبق رابطه (۲-۲) محاسبه می‌شود و در خط ۴۰ با کم کردن این مقدار از مقدار قبلی خود که در متغیر دیگری ذخیره شده است و تقسیم آن بر تغییرات زمانی مقدار \dot{x}_θ بدست می‌آید. در خطوط ۴۲ تا ۴۶ نیز موقعیت مرکز گوی طبق روابط (۲-۷) محاسبه می‌گردد.

۳-۲-۳ تابع محاسبه نیروی هر فن

محاسبات مربوط به این تابع نیز به‌طور کامل در بخش ۲-۴ توضیح داده شده است و در این بخش تنها به توضیح کدهای نوشته شده در کادر زیر پرداخته می‌شود.

```

1. void forcecalculator() {
2.
3.     float N, f, Fv, Fh, Fr, Fl, p1, p2, p3, Mcm;
4.
5.     p1 = d * cos(gama - alfa) * cos(gama) + d * sin(gama - alfa) * cos(theta) *
        sin(gama);
6.     p2 = d * sin(gama - alfa) * sin(theta);
7.     p3 = d * cos(gama - alfa) * sin(gama) - d * sin(gama - alfa) * cos(theta) *
        cos(gama);
8.
9.     N = ball_mass * g * cos(theta) + ball_mass * xtheta * ddtheta / 1000 + 2 *
        ball_mass * dxtheta * dtheta / 1000;
10.    f = -(2 / 5) * ball_mass * ddxtheta / 1000;
11.
12.    Fv = N * cos(theta) - (abs(dxtheta * f) / dxtheta) * sin(theta);
13.    Fh = N * sin(theta) + (abs(dxtheta * f) / dxtheta) * cos(theta);
14.
15.    Mcm = Plate_CM * Plate_Mass * g - Device_CM * Device_Mass * g;
16.
17.    Fr = Fv * ((p1) / (2 * Lx) - (p3) / (2 * Lz)) + Fh * ((p2 * sin(gama)) / (2 *
        Lx) + (p2 * cos(gama)) / (2 * Lz)) + Mcm / (2 * Lz);
18.    Fl = Fv * ((-1 * p1) / (2 * Lx) - (p3) / (2 * Lz)) + Fh * ((-1 * p2 *
        sin(gama)) / (2 * Lx) + (p2 * cos(gama)) / (2 * Lz)) + Mcm / (2 * Lz);
19.
20.    Right_FanSpeed = 79.875 * Fr + 1176.8;
21.    Left_FanSpeed = 79.875 * Fl + 1176.8;
22.
23.    if (Right_FanSpeed > 1400) {
24.        Right_FanSpeed = 1400;
25.    }
26.    else if (Right_FanSpeed < 1000) {
27.        Right_FanSpeed = 1000;
28.    }
29.    else {
30.        Right_FanSpeed = Right_FanSpeed;
31.    }
32.

```

```

33.  if (Left_FanSpeed > 1400) {
34.      Left_FanSpeed = 1400;
35.  }
36.  else if (Left_FanSpeed < 1000) {
37.      Left_FanSpeed = 1000;
38.  }
39.  else {
40.      Left_FanSpeed = Left_FanSpeed;
41.  }
42.
43. }

```

در خط ۱ این تابع از نوع void و با نام forcecalculator تعریف شده است و در خط ۳ متغیرهای محلی^۱ که در این تابع استفاده می‌شوند از نوع float تعریف می‌شوند. در خطوط ۵، ۶ و ۷ همانطور که در بخش ۲-۴ توضیح داده شد می‌توان مختصات نقطه تماس گوی و صفحه را در دستگاه مختصات xyz از روابط (۲-۷) با قرار دادن $R = 0$ به دست آورد. در خطوط ۹ و ۱۰ نیروهای عمودی سطح و اصطکاک از روابط (۲-۸) محاسبه می‌شوند. در خطوط ۱۲ و ۱۳ نیروهای عمودی و افقی از روابط (۲-۹) محاسبه شده و در خطوط ۱۷ و ۱۸ نیز نیروهای هر یک از فن‌ها از روابط (۲-۱۱) به دست می‌آیند. گشتاور حاصل از اختلاف مرکز جرم صفحه شبیه‌سازی شده و دستگاه نیز در خط ۱۵ و طبق رابطه (۲-۱۰) محاسبه می‌شود. در خطوط ۲۰ تا ۲۱ مقدار پهنای پالس بر حسب نیروی هر فن طبق رابطه (۴-۲) بدست می‌آید و همچنین در خطوط ۲۳ تا ۴۱ از دستورات شرطی به گونه‌ای استفاده شده است که به هر دلیلی پهنای پالسی که قرار است به موتورهای براشلس فرستاده شود از ۱۰۰۰ میکروثانیه کمتر نشده و از ۱۴۰۰ میکروثانیه بیشتر نشود. در این پروژه نیروهایی که قرار است شبیه‌سازی شوند در این بازه قرار دارند.

۳-۳ سنسور شتاب‌سنج و ژيروسکوپ

ماژول استفاده شده در این بخش از ۳ سنسور شتاب‌سنج، ژيروسکوپ و میدان مغناطیس تشکیل شده است که در این پروژه تنها از دو سنسور ژيروسکوپ و شتاب‌سنج استفاده می‌شود. با استفاده از این دو سنسور می‌توان زوایای دست فرد نسبت به محورهای x و z را بدست آورد. برای بدست آوردن زاویه دست فرد نسبت به محور y به تنهایی نمی‌توان از سنسور ژيروسکوپ استفاده نمود چرا که انتگرال‌گیری از سرعت زاویه‌ای نسبت به این محور در طی مدت زمان باعث ایجاد خطا و انحراف از مقدار واقعی می‌شود و باید این سنسور با اطلاعات سنسور میدان مغناطیس ترکیب شود که در این پروژه استفاده نشده است ولی می‌توان برای ارتقای قابلیت‌های سیستم از

¹ Local

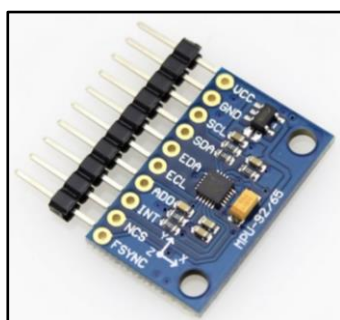
این سنسور نیز بهره برد.

هر کدام از سنسورهای شتاب‌سنج و ژيروسکوپ مزایا و معایبی دارند. بدین صورت که سنسور شتاب‌سنج که شتاب خطی را محاسبه می‌کند به تنهایی می‌توان زاویه دست فرد را از اطلاعات مربوط به شتاب که در حالت ساکن بردار شتاب گرانش می‌باشد، بدست آورد ولی این سنسور با اغتشاش‌هایی همراه است که برای نتیجه سیستم مطلوب نیست و همچنین اگر دست فرد در یک راستا به صورت خطی حرکت کند به دلیل برآیند گرفته شدن این شتاب با شتاب گرانش سیستم به اشتباه تصور می‌کند که زاویه دست فرد تغییر کرده است. از طرفی سنسور ژيروسکوپ که سرعت زاویه‌ای دست فرد حول یک محور را محاسبه می‌کند سنسوری است که تقریباً اغتشاشی در خروجی آن مشاهده نمی‌شود ولی عیب آن این است که اگر صرفاً با انتگرال‌گیری از داده‌های این سنسور زاویه دست فرد را محاسبه کنیم به دلیل خاصیت انتگرال و جمع شدن خطاها در مرور زمان مقدار زاویه از مقدار حقیقی به طور چشمگیری فاصله می‌گیرد [20].

در ادامه سعی شده است که اطلاعات این دو سنسور به گونه‌ای ترکیب شود که هر سنسور عیب سنسور دیگر را پوشانده و خروجی مطلوب به دست می‌آید.

۳-۳-۱ ترکیب داده‌های سنسور شتاب‌سنج و ژيروسکوپ

در مورد ترکیب اطلاعات این دو سنسور می‌توان از فیلترهای متعدد موجود مانند کالمن^۱ و یا پارکس مکانن^۲ و یا دیگر فیلترها استفاده نمود ولی به دلیل پیچیدگی این فیلترها سعی شده است به روش آسان‌تر نتیجه نسبتاً خوب و مناسب این پروژه گرفته شود. کلیه اطلاعات این قسمت مربوط به منبع [21] می‌باشند.



شکل ۳-۱ ماژول سنسور MPU9250

¹ Kalman filters

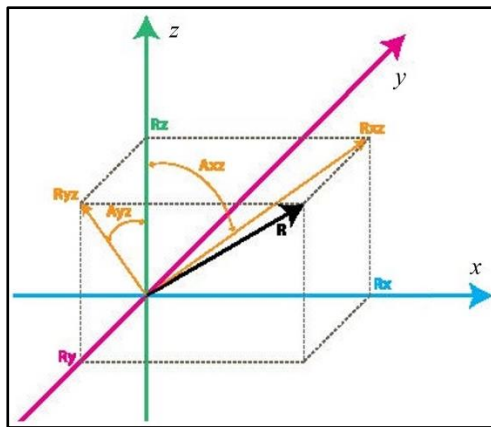
² Parks-McClellan filters

محاسبات این قسمت مربوط به دستگاه مختصات نشان داده شده روی مازول سنسور (شکل ۳-۱) می باشد و در پایان با تغییر علامت ساده می توان نتایج را برای دستگاه مختصات نشان داده شده در شکل ۲-۵ نوشت. روند کلی کار بدین صورت است که ابتدا دو بردار زاویه به وسیله سنسور شتاب سنج وژیروسکوپ ایجاد می شود و در ادامه با میانگین گیری از این دو بردار حاصله برای تعیین زاویه صفحه استفاده می گردد.

جهت قرار گیری سنسور روی دستگاه به گونه ای است که محور y نشان داده شده در شکل ۳-۱ جهت جلوی دستگاه را نشان می دهد. در ابتدا لازم است مقادیر شتاب در راستای ۳ محور را که نام های $RxAcc$ ، $RyAcc$ و $RzAcc$ را بر آن ها گذاشته شده است و همچنین مقدار سرعت زاویه ای در راستای ۲ محور x و y به نام های $RateAxz$ و $RateAyz$ از سنسور استخراج شوند. بردار مربوط به سنسور شتاب سنج به راحتی با توجه به رابطه زیر قابل محاسبه است.

$$RAcc = \frac{(RxAcc, RyAcc, RzAcc)}{\sqrt{RxAcc^2 + RyAcc^2 + RzAcc^2}} \quad (2-3)$$

اما برای محاسبه بردار مربوط به سنسور ژيروسکوپ ابتدا باید با انتگرال گیری از دو سرعت زاویه ای بدست



آمده از سنسور مقدار زاویه مربوط به این بردار یعنی Ayz و Axz محاسبه گردد. که Ayz زاویه بین تصویر بردار مربوط به سنسور ژيروسکوپ بر روی صفحه yz و محور z ها و Axz زاویه بین تصویر بردار مربوط به سنسور ژيروسکوپ بر روی صفحه xz و محور z ها می باشد (شکل ۳-۲).

شکل ۳-۲ سیستم مختصات سنسور ژيروسکوپ

اما نکته ای که در انتگرال گیری باید به آن توجه شود این است که اگر مقدار اولیه همان مقدار زاویه در مرحله قبل باشد موجب افزوده شدن خطاها و انحراف از مقدار حقیقی به مرور زمان می شود پس در هر مرحله از انتگرال گیری باید مقدار اولیه توسط بردار میانگین که با $REst$ نشان داده می شود و در ادامه توضیح داده خواهد شد، به روزرسانی شود تا از افزایش خطا جلوگیری کند. برای این منظور متغیرهایی تحت عنوان Axz^{n-1} و Ayz^{n-1} تعریف می شوند که به معنی مقدار این زاویه در مرحله قبل است و این مقادیر از روابط زیر بدست می آیند.

$$\begin{cases} Axz^{n-1} = \tan^{-1} \left(\frac{RxEst}{RzEst} \right) \\ Ayz^{n-1} = \tan^{-1} \left(\frac{RyEst}{RzEst} \right) \end{cases} \quad (3-3)$$

در نتیجه مقادیر زوایای Axz^n و Ayz^n از روابط زیر قابل محاسبه می‌باشند.

$$\begin{cases} Axz^n = Axz^{n-1} + (\overline{RateAxz}) \times t \\ Ayz^n = Ayz^{n-1} + (\overline{RateAyz}) \times t \end{cases} \quad (4-3)$$

که در روابط بالا t تغییرات زمانی نسبت به مرحله قبل و $\overline{RateAxz}$ و $\overline{RateAyz}$ از روابط زیر بدست می‌آیند.

$$\begin{cases} \overline{RateAxz} = \frac{RateAxz^n + RateAxz^{n-1}}{2} \\ \overline{RateAyz} = \frac{RateAyz^n + RateAyz^{n-1}}{2} \end{cases} \quad (5-3)$$

که دلیل میانگین گرفتن از دو مرحله برای تغییرات زاویه، کمتر کردن میزان اغتشاش در این مقدار است.

حال با توجه به روابط بالا مقدار بردار مربوط به سنسور ژيروسکوپ را بدست می‌آید.

$$RGyro = (RxGyro, RyGyro, RzGyro) \quad (6-3)$$

و اگر فرض کنیم که $|RGyro| = 1$ می‌توان مطابق روند زیر مولفه‌های بردار مربوط به سنسور ژيروسکوپ

را محاسبه نمود:

$$RxGyro = \frac{RxGyro}{\sqrt{RxGyro^2 + RyGyro^2 + RzGyro^2}} = \frac{\frac{RxGyro}{\sqrt{RxGyro^2 + RzGyro^2}}}{\sqrt{1 + \frac{RyGyro^2}{RxGyro^2 + RzGyro^2}}}$$

و در نهایت $RxGyro$ و سایر مولفه‌های $RGyro$ به همین شکل به دست می‌آیند:

$$\begin{cases} RxGyro = \frac{\sin(Axz)}{\sqrt{1 + \cos^2(Axz) \cdot \tan^2(Ayz)}} \\ RyGyro = \frac{\sin(Ayz)}{\sqrt{1 + \cos^2(Ayz) \cdot \tan^2(Axz)}} \\ |RzGyro| = \sqrt{1 - RxGyro^2 - RyGyro^2} \end{cases} \quad (7-3)$$

که علامت $RzGyro$ را می‌توان با علامت $RzEst$ تعیین کرد، به این صورت که اگر $RzEst$ مثبت

باشد $RzGyro$ هم مثبت و اگر $RzEst$ منفی باشد $RzGyro$ نیز منفی است. و در نهایت مقدار بردار نهایی

($RzEst$) را می‌توان به صورت میانگین وزن‌دار از $RAcc$ و $RGyro$ به صورت زیر بدست آورد.

$$REst = \frac{RAcc + W_{Gyro} \times RGyro}{1 + W_{Gyro}} \quad (۸-۳)$$

که W_{Gyro} میزان تاثیر سنسور ژيروسکوپ می‌باشد که معمولاً عددی بین ۵ تا ۲۰ انتخاب می‌شود و در این پروژه این مقدار برابر ۱۵ در نظر گرفته شده است. و در نهایت θ_x و θ_z بحث شده در بخش ۲-۳ را می‌توان از روابط زیر محاسبه نمود.

$$\begin{cases} \theta_x = \sin^{-1} \left(\frac{-RyEst}{\sqrt{RyEst^2 + RzEst^2}} \right) \\ \theta_z = \sin^{-1} \left(\frac{-RxEst}{\sqrt{RxEst^2 + RzEst^2}} \right) \end{cases} \quad (۹-۳)$$

۲-۳-۳ تابع خواندن زاویه از سنسور در آردیونو ($sensor()$)

همانطور که در قسمت ۱-۲-۳ توضیح داده شد از تابعی تحت عنوان $sensor()$ در بخش $loop$ کد آردیونو استفاده شده است که وظیفه آن محاسبه دو زاویه θ_x و θ_z از سنسور شتاب‌سنج و ژيروسکوپ می‌باشد. محاسبات اندازه‌گیری این دو زاویه با توجه به قسمت ۱-۳-۳ نوشته شده است که در ادامه کد مربوط به این تابع مشاهده می‌شود.

```

1. void sensor(){
2.
3.   IMU.readSensor();
4.
5.   float RxAcc = -1 * IMU.getAccelY_mss();
6.   float RyAcc = -1 * IMU.getAccelX_mss();
7.   float RzAcc = IMU.getAccelZ_mss();
8.
9.   float RnAcc = sqrt(pow(RxAcc,2)+pow(RyAcc,2)+pow(RzAcc,2));
10.  RxAcc = RxAcc / RnAcc;
11.  RyAcc = RyAcc / RnAcc;
12.  RzAcc = RzAcc / RnAcc;
13.
14.  preRateAxz = curRateAxz;
15.  preRateAyz = curRateAyz;
16.  curRateAxz = -1 * IMU.getGyroX_rads();
17.  curRateAyz = IMU.getGyroY_rads();
18.  float aveRateAxz = (preRateAxz + curRateAxz) / 2;
19.  float aveRateAyz = (preRateAyz + curRateAyz) / 2;
20.
21.  float Axz = atan2(RxEst,RzEst) + (aveRateAxz) * elapsedTime;
22.  float Ayz = atan2(RyEst,RzEst) + (aveRateAyz) * elapsedTime;
23.
24.  float RxGyro = sin(Axz) / sqrt(1 + pow(cos(Axz),2) * pow(tan(Ayz),2));
25.  float RyGyro = sin(Ayz) / sqrt(1 + pow(cos(Ayz),2) * pow(tan(Axz),2));
26.  float RzGyro = (abs(RzEst)/RzEst) * sqrt(1 - pow(RxGyro, 2) - pow(RyGyro, 2));
27.
28.  RxEst = (RxAcc + WGyro*RxGyro) / (1 + WGyro);

```

```

29. RyEst = (RyAcc + Wgyro*RyGyro) / (1 + Wgyro);
30. RzEst = (RzAcc + Wgyro*RzGyro) / (1 + Wgyro);
31.
32. theta_x = asin((-1 * RyEst) / sqrt(pow(RyEst,2) + pow(RzEst,2)));
33. theta_z = asin((-1 * RxEst) / sqrt(pow(RxEst,2) + pow(RzEst,2)));
34.
35. pretheta = theta;
36. theta = sqrt(pow(theta_x,2) + pow(theta_z,2));
37.
38. predtheta = dtheta;
39. dtheta = (theta - pretheta)/elapsedTime;
40.
41. ddtheta = (dtheta - predtheta)/elapsedTime;
42.
43. }

```

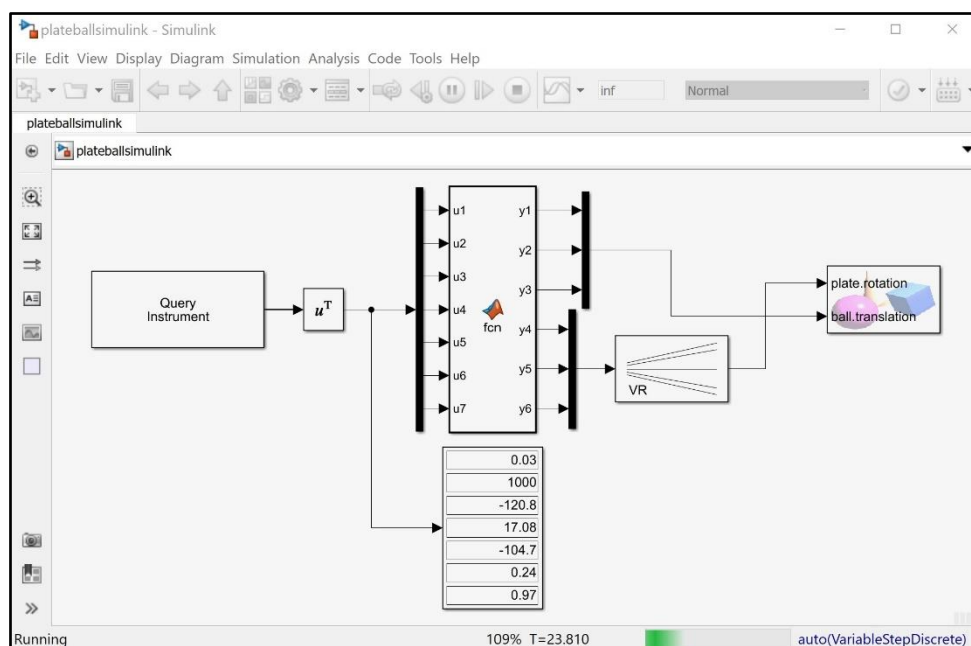
در خط اول نوع تابع از نوع void تعریف شده است و همچنین اسم آن sensor می‌باشد. دستور نوشته شده در خط ۳ به این معنی است که برد آردیونو به ماژول IMU که در قسمت ۳-۲-۱ تعریف شده بود دستور می‌دهد که اطلاعات مربوط به سنسورها را برای آردیونو جهت خواندن آن‌ها ارسال کند. در خطوط ۵ تا ۷ مقادیر خوانده شده از سنسور شتاب‌سنج توسط دستور IMU.getAccel_mss() برای هر محور بر حسب (m/s^2) در متغیرهای مربوط به همان محور ذخیره می‌شوند که دلیل فرق داشتن محور متغیر با محور دستور مشکل برد در فرستادن اطلاعات طبق محور نوشته شده روی آن است که این مشکل با تغییر چند محور به راحتی برطرف می‌شود. از خطوط ۹ تا ۱۲ به جهت یک‌کردن بردار شتاب و بدست آوردن بردار نشان داده شده در رابطه (۳-۲) می‌باشد. خطوط ۱۴ تا ۱۹ مربوط به خواندن اطلاعات مربوط به سنسورژیروسکوپ و همچنین میانگین‌گیری دو مرحله از این اطلاعات مطابق رابطه (۳-۵) می‌باشد. در خطوط ۲۱ و ۲۲ مقادیر زوایای A_{xz} و A_{yz} طبق روابط (۳-۳) و (۳-۴) می‌باشد. در خطوط ۲۴ تا ۲۶ نیز مولفه‌های بردار $RGyro$ طبق روابط (۳-۷) به دست می‌آیند. در خطوط ۲۸ تا ۳۰ نیز مولفه‌های بردار $REst$ مطابق رابطه (۳-۸) به دست می‌آیند و در نهایت در خطوط ۳۲ و ۳۳ مقادیر زوایای θ_x و θ_z طبق روابط (۳-۹) محاسبه و در متغیرهای مربوطه ذخیره می‌گردند. از خطوط ۳۵ تا ۴۱ مقادیر جدید زاویه و سرعت زاویه‌ای از مقادیر پیشین خود کم شده و با تقسیم بر تغییرات زمانی می‌توان سرعت زاویه‌ای و شتاب زاویه‌ای را محاسبه نمود.

۳-۴ ماژول بلوتوث

از آنجایی که اتصال هرگونه سیم به دستگاه موجب محدودیت نسبی در حرکت و همچنین تاثیر بر نیروی وارده به دست فرد می‌شود در این پروژه سعی شده است که هیچگونه اتصالی بین دستگاه و محیط بیرون از طریق سیم نباشد. برای این منظور از ارتباط سریال بلوتوث برای فرستادن اطلاعات مربوط به زاویه صفحه و موقعیت

گوی به نرم افزار متلب جهت نمایش دادن محیط شبیه سازی استفاده شده است.

ماژول بلوتوث استفاده شده در این پروژه از نوع HC-05 می باشد که از انواع معروف این ماژول در پروژه های مختلف می باشد که قابلیت ارسال و دریافت داده را دارد ولی ما از این ماژول تنها برای ارسال اطلاعات از آردیونو به نرم افزار متلب استفاده می کنیم. برای این منظور نیاز به دو پین از آردیونو به نام RX و TX است که به ترتیب به TX و RX ماژول جهت ارسال و دریافت اطلاعات متصل شوند. بر روی برد آردیونو دو پین برای این منظور تعبیه شده است ولی برای اینکه این دو پین که مربوط به پورت سریال خود برد است و برای نمایش اطلاعات در حین تست ممکن است مورد استفاده قرار گیرد، اشغال نشوند با استفاده از کتابخانه AltSoftSerial، همانطور که در قسمت ۳-۲-۱ گفته شد یک پورت سریال دیگر با استفاده از دو پین ۸ و ۹ برد آردیونو با نام Bluetooth ایجاد شده و از این دو پین برای انتقال اطلاعات با ماژول بلوتوث استفاده شده است.



شکل ۳-۳ تصویر تمامی بلوک های استفاده شده در سیمولینک

در نرم افزار سیمولینک همانطور که در شکل ۳-۳ مشاهده می شود از بلوک Query Instrument برای دریافت اطلاعات از ماژول بلوتوث استفاده می شود.

همانطور که در قسمت ۳-۲-۱ توضیح داده شد برای ارسال اطلاعات مربوط به زاویه صفحه و موقعیت گوی از یک تابع تحت عنوان sendtoSimulink() استفاده شد. حال در این قسمت به بررسی قسمت های مختلف این تابع پرداخته می شود.

```

1. void sendtoSimulink() {
2.
3.     float angx_vec, angz_vec, ang_val;
4.
5.     if (theta_x == 0 && theta_z == 0) {
6.         angx_vec = 1;
7.         angz_vec = 1;
8.         ang_val = 0;
9.     }
10.    else {
11.        angx_vec = theta_x / theta;
12.        angz_vec = theta_z / theta;
13.        ang_val = theta;
14.    }
15.
16.    Bluetooth.print(1000);
17.    Bluetooth.print(",");
18.    Bluetooth.print(x);
19.    Bluetooth.print(",");
20.    Bluetooth.print(y);
21.    Bluetooth.print(",");
22.    Bluetooth.print(z);
23.    Bluetooth.print(",");
24.    Bluetooth.print(angx_vec);
25.    Bluetooth.print(",");
26.    Bluetooth.print(angz_vec);
27.    Bluetooth.print(",");
28.    Bluetooth.println(ang_val);
29.
30. }

```

همانطور که در بخش ۲-۳ گفته شد مقدار زاویه صفحه ورودی به بلوک VR Sink باید یک بردار با چهار مولفه باشد که سه مولفه اول مقادیر بردار یکه بردار چرخش و مولفه آخر اندازه چرخش صفحه می باشد. مولفه دوم این بردار همیشه صفر است چون نسبت به بردار y چرخشی در صفحه وجود ندارد ولی مولفه اول و سوم و چهارم این بردار در خطوط ۳ تا ۱۴ قابل محاسبه می باشند و دلیل استفاده از دستور شرطی این است که اگر در موقعیتی هم θ_x و هم θ_z صفر شد با خطای عدد تقسیم بر صفر^۱ در حین اجرای برنامه مواجه نشویم.

در ادامه در خطوط ۱۶ تا ۲۸ اطلاعات مربوط به موقعیت گوی و مولفه اول و سوم بردار چرخش و زاویه چرخش به صورت خط‌هایی از داده به نرم افزار سیمولینک ارسال می شوند. از آنجایی که ممکن است بلوک Query Instrument در نرم افزار سیمولینک داده‌ها را به همان ترتیبی که ارسال شده اند، دریافت نکند (مانند شکل ۳-۳ که داده‌های دریافت شده، همانطور که نمایش داده شده اند به ترتیبی که فرستاده شده‌اند نیستند)، در قسمت مربوط به کد آردیونو اولین داده ارسالی برابر ۱۰۰۰ در نظر گرفته می شود که عددی ثابت است و چون بقیه اطلاعات ارسالی دارای محدوده‌ی کمتری نسبت به این هستند از این عدد استفاده شده است. حال در نرم افزار سیمولینک بین داده‌های دریافت شده و داده‌های مشخص شده برای بلوک VR Sink از یک کد متلب به صورت زیر استفاده شده است تا اولین داده ارسالی را برابر با ۱۰۰۰ در نظر بگیرد و به همان ترتیب سایر داده‌ها را

^۱ Division by zero

در خروجی خود قرار دهد.

```

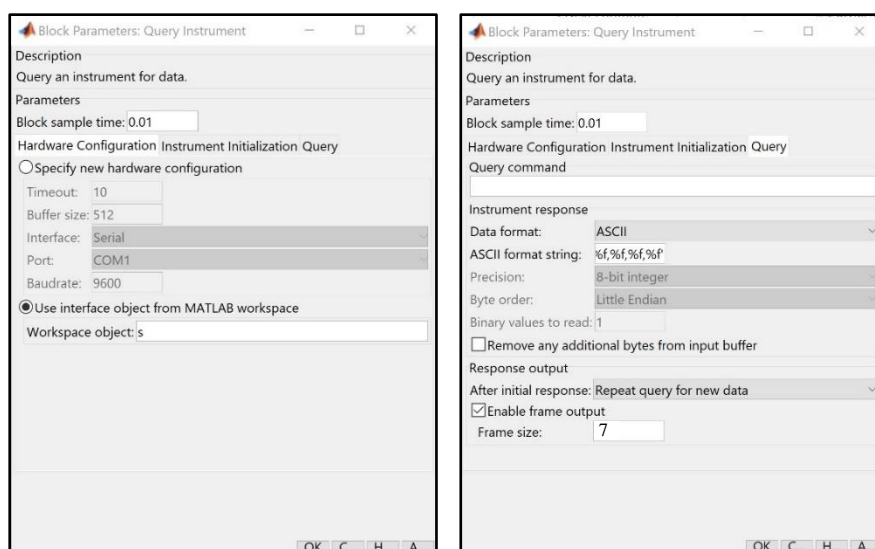
1. function [y1,y2,y3,y4,y5,y6] = fcn(u1,u2,u3,u4,u5,u6,u7)
2.
3. if(u1 == 1000)
4.     y1 = u2;
5.     y2 = u3;
6.     y3 = u4;
7.     y4 = u5;
8.     y5 = u6;
9.     y6 = u7;
10. elseif(u2 == 1000)
11.     y1 = u3;
12.     y2 = u4;
13.     y3 = u5;
14.     y4 = u6;
15.     y5 = u7;
16.     y6 = u1;
17. elseif(u3 == 1000)
18.     y1 = u4;
19.     y2 = u5;
20.     y3 = u6;
21.     y4 = u7;
22.     y5 = u1;
23.     y6 = u2;
24. elseif(u4 == 1000)
25.     y1 = u5;
26.     y2 = u6;
27.     y3 = u7;
28.     y4 = u1;
29.     y5 = u2;
30.     y6 = u3;
31. elseif(u5 == 1000)
32.     y1 = u6;
33.     y2 = u7;
34.     y3 = u1;
35.     y4 = u2;
36.     y5 = u3;
37.     y6 = u4;
38. elseif(u6 == 1000)
39.     y1 = u7;
40.     y2 = u1;
41.     y3 = u2;
42.     y4 = u3;
43.     y5 = u4;
44.     y6 = u5;
45. else
46.     y1 = u1;
47.     y2 = u2;
48.     y3 = u3;
49.     y4 = u4;
50.     y5 = u5;
51.     y6 = u6;
52. end
53.
54. end

```

به طور مثال برای خط ۲۴، اگر ورودی چهارم از داده‌های ارسال شده برابر با ۱۰۰۰ بود در نتیجه داده‌های خروجی از این تابع به ترتیب باید برابر با داده‌های پنجم، ششم، هفتم، اول، دوم و سوم از ورودی این تابع باشند. و از آنجایی که هر کدام از این داده‌ها می‌توانند برابر با مقدار ۱۰۰۰ باشند در نتیجه این دستور شرطی برای هر کدام از پایانه‌های ورودی این تابع نوشته شده است.

در سیمولینک، بلوک Query Instrument اطلاعات ارسال شده از آردیونو را هر بار به صورت یک خط دریافت می‌کند. حال برای اینکه بتوان هر کدام از داده‌ها را در بخش مخصوص به خود استفاده نمود، این سطر از داده با استفاده از بلوک Transpose به ستونی از داده تبدیل می‌گردد و همانطور که مشاهده می‌شود موقعیت گوی به صورت مجزا به بلوک VR Sink و زوایای صفحه برای ایجاد برداری به صورت $(\theta_x, 0, \theta_z, \sqrt{\theta_x^2 + \theta_z^2})$ وارد بلوک VR Signal Expander شده و در نهایت وارد بلوک VR Sink می‌شوند.

تنظیمات مربوط به بلوک Query Instrument نیز در شکل ۳-۴ مشاهده می‌شود.



شکل ۳-۴ تنظیمات مربوط به بلوک Query Instrument

همانطور که در شکل ۳-۴ مشاهده می‌شود، میزان Block sample time برابر با ۰/۰۱ قرار داده شده است که بدین معنا است که این بلوک در هر ثانیه ۱۰۰ داده از مازول بلوتوث می‌خواند که مقدار که مقدار این ثابت زمانی برای بلوک VR Sink نیز باید به همین اندازه باشد. توجه شود که اگر این عدد بزرگتر باشد و داده‌های کمتری در هر ثانیه خوانده شود خروجی مطلوبی به صورت پیوسته بدست نخواهد آمد و اگر این مقدار کوچکتر در نظر گرفته شود به دلیل خواندن داده زیاد موجب کندی سیستم می‌شود که این مقدار انتخاب شده مقدار نسبتاً مناسبی می‌باشد. همچنین میزان مکث قرار داده شده در قسمت loop کد آردیونو که در قسمت ۳-۲-۱ توضیح داده شد نیز باید به گونه‌ای انتخاب شود که متناسب با sample time سیمولینک باشد و مقدار ۲۰ میلی‌ثانیه برای این مکث مقدار مناسبی می‌باشد. همچنین در سربرج Query شکل ۳-۴، در قسمت مربوط به فرمت رشته داده‌ها عبارت '%f,%f,%f,%f,%f,%f,%f,%f' نوشته شده است که بدین معنی است که داده‌ها

به صورت یک خط شامل ۷ متغیر از نوع float باید خوانده شوند. و همانطور که در سر برگ Hardware Configuration مشاهده می شود باید یک متغیر به نام s در قسمت Workspace متلب در ابتدا ایجاد شود که حاوی اطلاعات مربوط به پورت سریال موردنظر می باشد.

۳-۵ تابع فرستنده اطلاعات به عملگرهای دستگاه (sendToDevice())

همانطور که در قسمت loop کد نوشته شده مشاهده می شود این تابع برای فرستادن اطلاعاتی نظیر سرعت فن ها و همچنین زاویه موتور سروو به دستگاه، فراخوانی شده است و در این قسمت به توضیح کد مربوط به این تابع پرداخته می شود.

```

1. void sendToDevice() {
2.
3.     Sweep_Angle = (100 / 9) * ((theta_z * (180 / 3.14) + 90) - 45) + 1000;
4.     F_Sweep_Angle = Angle_Filter_Const * F_Sweep_Angle + (1 - Angle_Filter_Const) *
       Sweep_Angle;
5.
6.     int switchState = digitalRead(switchpin);
7.
8.     if (switchState == HIGH) {
9.         Right_DuctedFan.write(Right_FanSpeed);
10.        Left_DuctedFan.write(Left_FanSpeed);
11.        Sweeper.write(F_Sweep_Angle);
12.    }
13.    else {
14.        Right_DuctedFan.write(1000);
15.        Left_DuctedFan.write(1000);
16.        Sweeper.write(1500);
17.    }
18.
19. }
20.

```

در خط اول از این کد این تابع تحت عنوان sendToDevice و از نوع void تعریف می شود. در خط ۳ زاویه ی محور موتور سروو طبق رابطه (۳-۱) به گونه ای تعیین شده است که موقعیت فن ها همیشه به صورت افقی باشند. برای این منظور به جای متغیر زاویه شافت مشخص شده در رابطه (۳-۱) از زاویه دست فرد که در قسمت ۳-۳-۲ اندازه گیری شده است، استفاده می کنیم و از آنجایی که مثبت این زاویه به صورت پادساعتگرد و مثبت زاویه شافت موتور سروو به صورت ساعتگرد می باشد نیاز به استفاده از علامت منفی در این زاویه نمی باشد. از آنجایی که لرزش جزئی دست فرد بر این زاویه تاثیر گذاشته و موجب لرزش فن ها بر اثر چرخش محور موتور سروو می شود، در خط ۴ با یک نوع میانگین گیری وزن دار از داده های این مرحله و مرحله قبل می توان این لرزش ها را تا حدی از بین برد. ثابت این میانگین گیری نیز به صورت آزمون خطا برابر ۰/۶ فرض شده است. در خط ۶ متغیری برای دریافت اطلاعات کلید تعریف می شود و در خطوط ۸ تا ۱۷ از یک دستور شرطی به گونه ای

استفاده شده است که در صورتی که کلید غیرفعال باشد فن‌ها ثابت و شافت موتور سروو در وضعیت ۹۰ درجه قرار گیرد و در صورتی که کلید فعال باشد سرعت فن‌ها برابر با مقدار مشخص شده در قسمت ۳-۲-۳ و زاویه فن برابر مقدار مشخص شده در خط ۴ باشد.

۳-۶ مازول تغذیه کاهنده

به عنوان تغذیه دستگاه در این پروژه از یک باتری لیتیوم پلیمر ۱۱.۱ ولت ۲۳۰۰ میلی آمپر- ساعت استفاده شده است ولی برای قسمت‌های مربوط به برد آردیونو و قطعات الکترونیکی ولتاژ باید در محدوده ۵ تا ۶ ولت باشد. برای این منظور از یک مازول تغذیه کاهنده^۱ XL4015 استفاده شده است. طریقه کار با این مازول به این صورت است که با متصل کردن قسمت ورودی به باتری ذکر شده و خروجی به مدار مربوطه می‌توان با استفاده از پیچ تعبیه شده بر روی این مازول ولتاژ خروجی را در یک مقدار ثابت تنظیم کرد. در مورد مزیت این مازول می‌توان به این نکته اشاره کرد که با تغییر ولتاژ ورودی مازول، ولتاژ خروجی در همان سطح تنظیم شده باقی می‌ماند و تغییر نمی‌کند. شیوه اتصال تمامی این بردهای الکترونیکی در قسمت ۴-۲-۲ توضیح داده شده است.

۳-۷ جمع‌بندی

پردازنده استفاده شده در این پروژه برد آردیونو می‌باشد و همچنین برای هر قسمت از پروژه بستگی به کاربرد و نیاز از مازول‌هایی مانند مازول سنسور ژيروسکوپ و شتاب‌سنج، مازول بلوتوث و مازول کاهنده ولتاژ استفاده شده است. روند اجرای برنامه‌های نوشته شده در برد آردیونو نیز بدین صورت است که ابتدا زاویه دست فرد توسط سنسور ژيروسکوپ و شتاب‌سنج سنجیده شده و برد آردیونو با توجه به روابط مربوطه موقعیت گوی را محاسبه کرده و توسط سنسور بلوتوث به نرم‌افزار سیمولینک منتقل می‌کند. همچنین برد آردیونو با محاسبه نیروی هر فن از طریق روابط اشاره شده، سرعت هر فن برای ایجاد این نیرو و زاویه‌ی مناسب برای موتور سروو را مشخص کرده و به هر موتور ارسال می‌کند.

^۱ Buck convertor

فصل چهارم

طراحی و ساخت

۴-۱ مقدمه

مطالب این فصل را می‌توان به طور کلی به سه قسمت تقسیم نمود. در قسمت اول فصل به طراحی کلیه قطعات مکانیک و الکترونیکی و نحوه اتصال آن‌ها به یکدیگر پرداخته می‌شود. در قسمت دوم این فصل ابتدا به نحوه مونتاژ قطعات و ساخت مجموعه پرداخته و در نهایت آزمایش‌هایی برای مشاهده رفتار سیستم و بدست آوردن ویژگی‌های آن و همچنین تستی برای بررسی عملکرد دستگاه انجام داده می‌شود. و در آخر این فصل دستورالعمل استفاده از دستگاه به طور مفصل آورده شده است.

۴-۲ طراحی دستگاه

در این قسمت ابتدا به بررسی قطعات مکانیکی طراحی شده و در نهایت به طریقه به هم متصل کردن قطعات الکترونیکی پرداخته می‌شود.


۴-۲-۱ طراحی قطعات دستگاه

در این پروژه سعی شده است تمامی قطعات مورد استفاده به صورت فایل CAD طراحی شوند تا بتوان





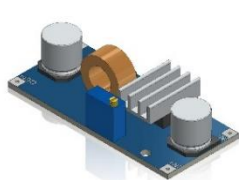

قبل از شروع فرایند ساخت دید جامعی نسبت به طرح به دست آید. قطعات مورد استفاده در این پروژه را می توان به ۳ دسته کلی تقسیم نمود، دسته اول مربوط به قطعات آماده ای است که می توان آن ها را از فروشگاه های مربوطه خریداری نمود از جمله این قطعات می توان به بردهای الکترونیکی، داکت فن ها، باتری و غیره که در ادامه لیست تمامی این قطعات و توضیحات کامل، در جدولی ارائه خواهد شد. دسته دوم مربوط به قطعات جانبی ای است که مختص این پروژه طراحی شده اند و امکان خریداری آن ها وجود ندارد. این قطعات توسط فرایند پرینت سه بعدی^۱ ساخته می شوند و در دسته سوم قطعاتی هستند که به نسبت قطعات دسته دوم از پیچیدگی کمتری برخوردارند و این قطعات را می توان از روی نقشه کشیده شده از روی طرح، ساخت. به طور مثال دسته دستگاه یا شافت میانی دستگاه از این دسته می باشند.

تمامی قطعات استفاده شده در این پروژه را به همراه توضیحات و خلاصه ای از کاربرد آن ها را می توان در جداول جدول ۱-۴، جدول ۲-۴ و جدول ۳-۴ مشاهده نمود.

جدول ۱-۴ فهرست قطعات دسته اول به کار رفته در دستگاه

ردیف	نام قطعه	کاربرد قطعه	وزن (gr)	تصویر
۱	Flexible Coupling 5-5mm	اتصال سر موتور سروو به شافت میانی	16.1	
۲	MG-996R Servo	چرخاندن فن ها حول محور گذرنده از دست فرد	56.8	
۳	Servo Head	اتصال شافت موتور سروو به کوپلینگ انعطاف پذیر	1.1	

^۱ 3D Printing

	134.8	ایجاد نیروی مجازی از طریق چرخش پره‌ها	3800KV Brushless Motor with 12 Blades, 64mm QX- MOTOR Ducted Fan	۴
	44.1	کنترل سرعت فن‌ها از طریق پالس‌های برد آردیونو	AT-40A-4S T- MOTOR ESC	۵
	176.7	منبع تغذیه دستگاه	2300mAh, 45C, 3Cells Li-Po Battery	۶
	5.6	پردازنده مرکزی دستگاه	Arduino Nano Atmega328	۷
	15.9	تبدیل ولتاژ باتری به ولتاژ مناسب برای بردهای الکترونیکی	XL4015 Buck Converter Module	۸
	2.2	اندازه‌گیری زاویه دست فرد	MPU-9250 9-Axis MEMS	۹




	3.2	ارسال اطلاعات مربوط به زاویه دست فرد و موقعیت گوی به سیمولینک	HC-05 Bluetooth Module	۱۰
	4.5	قرار دادن شافت در میانه دسته	625ZZ Ball Bearing	۱۱
	1.1	کلید فعال سازی فن ها	Toggle Switch	۱۲

جدول ۴-۲ فهرست قطعات دسته دوم به کار رفته در دستگاه

ردیف	نام قطعه	کاربرد قطعه	وزن (gr)	تصویر
۱	Servo Holder	نگه دارنده ی موتور سرو	10.8	
۲	Board Holder	محفظه ی بردهای الکترونیکی	33.9	

	12.4	درپوش محفظه بردهای الکترونیکی	Cap	۳
	67.6	نگهدارنده‌ی فن‌ها	Fan Holder	۴
	19.3	محکم کننده‌ی فن سمت چپ در جای خود	Left Holder	۵
	19.3	محکم کننده‌ی فن سمت راست در جای خود	Right Holder	۶
	11.9	نگهدارنده‌ی باتری	Battery Holder	۷
	0.75	نگهدارنده‌ی سنسور شتاب‌سنج	MPU Holder	۸

جدول ۴-۳ فهرست قطعات دسته سوم به کار رفته در دستگاه

ردیف	نام قطعه	کاربرد قطعه	وزن (gr)	تصویر
۱	Handle	محل به دست گرفتن دستگاه توسط فرد	125.5	
۲	Shaft	انتقال قدرت از کوپلینگ انعطاف پذیر به نگه دارنده ی فن ها	35.1	
۳	Pin	اتصال شافت میانی به نگه دارنده ی فن ها	1	

همانطور که گفته شد قطعات مشخص شده در دسته اول را می توان به صورت آماده از فروشگاه های مربوطه با توجه به نام ذکر شده در جدول ۴-۳ خریداری نمود.

در مورد قطعات دسته دوم همانطور که اشاره شد این قطعات توسط فرایند پرینت سه بعدی ساخته خواهند شد و با توجه به پیچیدگی این قطعات و غیرقابل فهم بودن نقشه های آن ها از کشیدن نقشه از این قطعات صرف نظر شده است و به جای آن تمامی این قطعات با پسوند مورد استفاده در دستگاه های پرینتر سه بعدی (stl) در سایت GitHub بارگذاری شده اند که با کلیک بر روی [لینک](#)^۱ به راحتی قابل دسترسی می باشند. این قطعات به صورت آنلاین و از طریق سایت [صنعت بازار](#)^۲ پرینت شده اند و تمامی قطعات از جنس فیلامنت پلی لاکتیک اسید^۳، با ضخامت لایه ۰/۲ میلی متر، تراکم ۳۰ درصد و رنگ خاکستری می باشند.

همچنین در مورد قطعات مربوط به دسته سوم، از آنجایی که این قطعات از پیچیدگی کمتری نسبت به

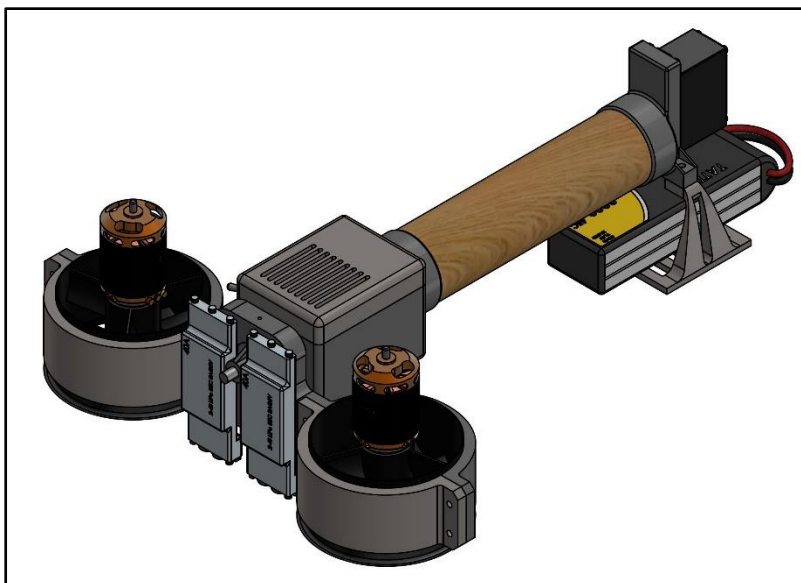
^۱ <https://github.com/mohamadhasan-m/Haptic-Device>

^۲ <https://www.sanatbazar.com/services/print3d>

^۳ PLA

دسته دوم برخوردارند و متفاوت بودن جنس آن‌ها می‌توان این قطعات را از روی نقشه‌های ارائه شده برای هر کدام در پیوست الف، ساخت. همچنین خاطرنشان می‌شود که دلیل استفاده از جنس چوب برای دسته دستگاه ایجاد حس لامسه بهتر در فرد هنگام در دست گرفتن دستگاه از آنجایی که جنس صفحه مجازی نیز چوب است، می‌باشد.

و در نهایت تصویر کامل و اسمبلی شده دستگاه را می‌توان در شکل ۴-۱ مشاهده نمود.



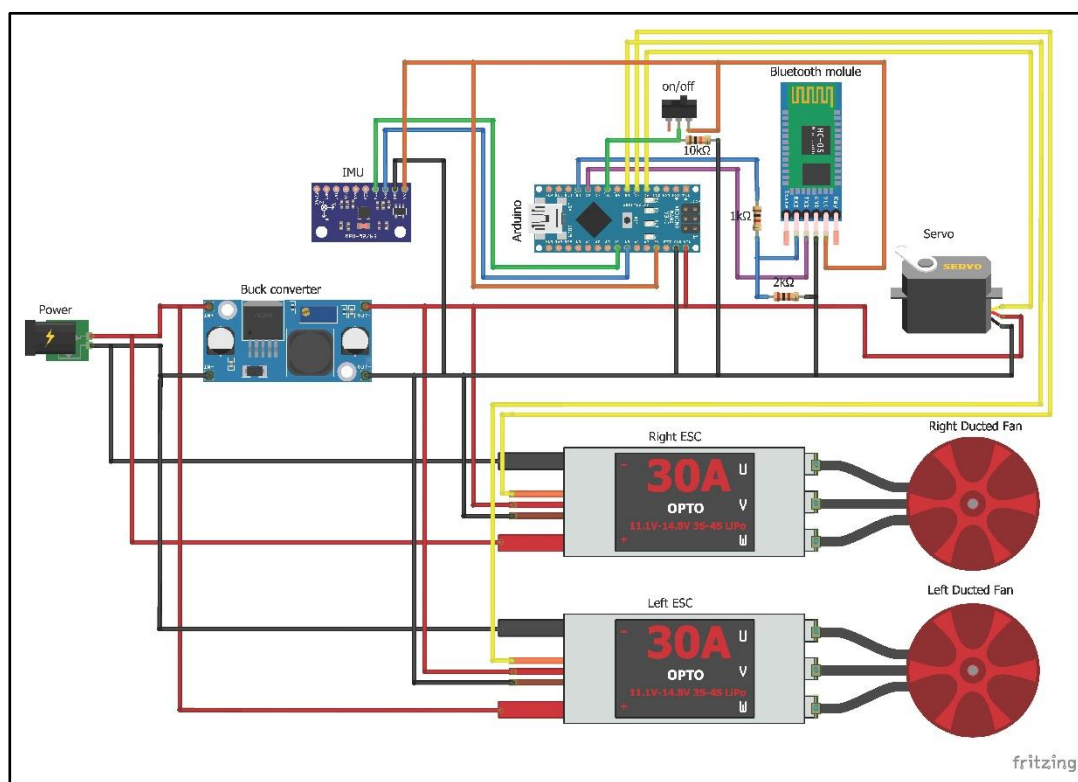
شکل ۴-۱ تصویر اسمبلی شده دستگاه

همچنین همانطور که در جداول ارائه شده‌ی مربوط به لیست قطعات (جدول ۴-۱، جدول ۴-۲ و جدول ۴-۳) مشاهده می‌شود، برای هر قطعه جرم آن نیز مشخص شده است که با جمع کردن این جرم‌ها، جرم ۹۷۵.۶ گرم برای کل دستگاه بدون در نظر گرفتن جرم قطعات جانبی از جمله پیچ‌ها، سیم‌ها و غیره به دست می‌آید. در قسمت ساخت نیز جرم تمام شده دستگاه مشخص شده است. همچنین با توجه به خروجی‌های نرم‌افزار سالی‌دورکس مشاهده می‌شود که مختصات مرکز جرم نسبت به نقطه میانی دسته دستگاه به صورت (58.84 , -24.00 , 0.07) میلی‌متر می‌باشد که محور z در راستای دسته دستگاه و به سمت فن‌ها و همچنین محور y در راستای عمود بر محور دسته دستگاه و به سمت بالا می‌باشد. همچنین با استفاده از خروجی‌های نرم‌افزار سالی‌دورکس از روی صفحه مجازی طراحی شده در این نرم‌افزار مشاهده می‌شود که مختصات مرکز جرم این صفحه مجازی نسبت به نقطه میانی دسته آن به طوری که محور z در راستای دسته صفحه و به سمت صفحه است و همچنین محور y در راستای عمود بر دسته صفحه و به سمت بالا می‌باشد، به صورت (208.11 , -6.25 , 0.00) میلی‌متر می‌باشد که از مختصه z این دو نقطه می‌توان در بخش مربوط به کد

آردیونو برای محاسبه گشتاور مربوط به اختلاف این دو نقطه استفاده نمود. همچنین جرم صفحه مجازی نیز در این پروژه برابر با ۳۸۰ گرم فرض شده است. این جرم با توجه به ظرفیت نیروی تولیدی در هر فن انتخاب شده است.

۲-۲-۴ اتصال قطعات الکترونیکی

تمامی قطعات الکترونیکی در محفظه نگهداری قطعات الکترونیکی که در جدول ۲-۴ مشخص شده است، قرار داده شده اند و هر کدام توسط سیم‌هایی به یکدیگر متصل می‌شوند. طریقه به یکدیگر متصل کردن قطعات را می‌توان به صورت شماتیک در شکل ۲-۴ مشاهده نمود.



شکل ۲-۴ نقشه اتصال قطعات الکترونیکی به یکدیگر

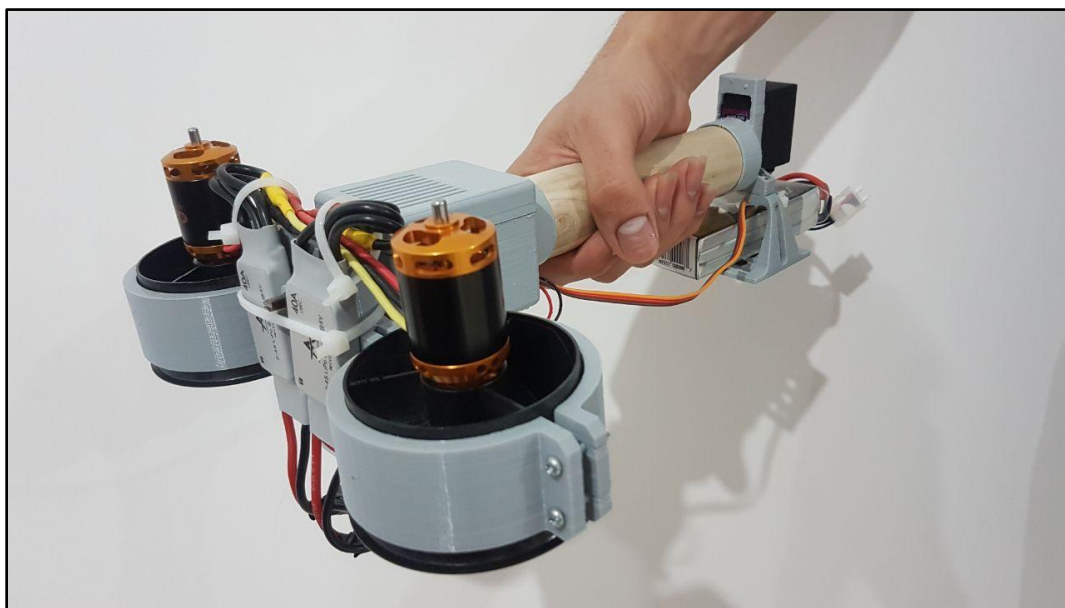
نکته مورد توجه در مورد اتصال قطعات الکترونیکی به یکدیگر این است که تمامی پایه‌های زمین^۱ این قطعات جهت مشخص شدن مرجع مدار باید به یکدیگر متصل شوند. همچنین پایه مثبت ESC ها، سروو موتور و برد آردیونو (Vin) باید به پایه مثبت مبدل ولتاژ متصل شود که بدین معناست که این قطعات جریان را به صورت مستقیم از مبدل ولتاژ دریافت می‌کنند و پایه‌های مثبت مازول بلوتوث و همچنین مازول سنسور شتاب‌سنج و ژيروسکوپ به پایه مثبت آردیونو (+5V) متصل می‌شود که بدین معناست که این مازول‌ها جریان

^۱ GND

مورد نیاز خورد را از برد آردیو نو دریافت می کنند.

۳-۴ مونتاژ قطعات دستگاه و انجام آزمایش های مربوطه

پس از تهیه کلیه قطعات آماده از فروشگاه های مربوطه و همچنین اتمام پرینت قطعاتی که به منظور پرینت سه بعدی طراحی شده بودند، می توان کار مونتاژ قطعات را مطابق نقشه انفجاری ای که در پیوست الف مشاهده می شود، شروع کرد. در قسمت طراحی قطعات مکانیکی به این نکته توجه شده است که تا حد امکان از اتصالات غیر دائم مانند پیچ و پین، به جای اتصالات دائم مانند چسب استفاده شود تا در صورت از کار افتادن یک قطعه به آسانی بتوان عملیات دموونتاژ را انجام داده و قطعه معیوب را جایگزین نماییم. همچنین در مرحله اتصال مدارات الکترونیکی، ماژول ها اغلب به صورت اتصالات پین دار به یکدیگر متصل شده اند و از لحیم کردن پایه های ماژول ها جلوگیری شده است تا به آسانی بتوان ماژول را به هر دلیلی جدا کرده و یا متصل کنیم. در نهایت تصویر نهایی دستگاه مونتاژ شده را می توان در شکل ۳-۴ مشاهده نمود.

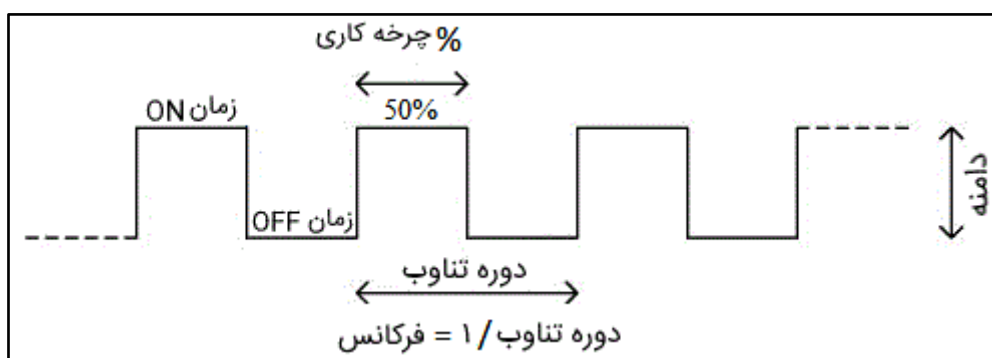


شکل ۳-۴ تصویر نهایی دستگاه مونتاژ شده

همانطور که در قسمت ۴-۲-۱ ذکر شد، مجموع جرم کلیه قطعات طراحی شده در این پروژه برابر با ۹۷۵.۶ گرم می باشد. حال پس از اتمام کار مونتاژ دستگاه با احتساب وزن قطعات جانبی مانند سیم ها، پیچ ها و غیره مشاهده می شود که وزن تمام شده دستگاه برابر با ۹۸۶ گرم می باشد و با مقایسه با وزن دستگاه Aero-plane که برابر با ۱۰۶۹ گرم می باشد [1]، مشاهده می شود که با وجود اینکه دستگاه ساخته شده در این

پروژه هیچ ارتباط فیزیکی ای با محیط خارجی ندارد و باتری بر روی دستگاه قرار داده شده است وزن کلی دستگاه نسبت به نمونه ساخته شده کمتر می باشد که از زودتر خسته شدن دست فرد جلوگیری می کند.

برای کنترل موتور سروو و همچنین دو فن تعبیه شده در دستگاه باید از مدولاسیون پهنای باند یا به اختصار PWM استفاده نمود. در این روش فرکانس پالس های ارسال شده ثابت می ماند ولی مدت زمان High بودن پالس و یا چرخه کاری^۱ تغییر می کند [22]. به طور مثال شکل ۴-۴ یک سیگنال PWM، با چرخه کاری 50% را نشان می دهد.



شکل ۴-۴ یک سیگنال PWM، با چرخه کاری 50% [22]

فرکانس پالس های ایجاد شده توسط آردیونو و کتابخانه ServoTimer2 به صورت ۵۰ هرتز می باشد که این بدین معناست که فاصله زمانی دو پالس ارسال شده برابر با ۲۰ میلی ثانیه می باشد. زمان High بودن پالس برای موتور سروو و همچنین در موتورهای براشلس برای چرخاندن فن ها مقداری بین ۱ تا ۲ میلی ثانیه می باشد. که در مورد موتور سروو این زمان موجب قرار گیری موقعیت شافت موتور در محلی خاص و در مورد موتورهای براشلس این زمان موجب چرخیدن شافت موتور با سرعتی خاص می باشد.

با فرستادن سیگنال های مختلف PWM مشاهده می شود که موتور سروو در پالس با زمان High، ۱۵۰۰ میکروثانیه در وضعیت ۹۰ درجه قرار می گیرد و وضعیت موتورهای براشلس و فن ها در حالت افقی قرار دارند و همچنین در پالس با زمان High، ۱۰۰۰ میکروثانیه موتور سروو در وضعیت ۴۵ درجه قرار گرفته و فن ها به اندازه ۴۵ درجه به صورت پادساعتگرد و در پالس با زمان High، ۲۰۰۰ میکروثانیه موتور سروو در وضعیت ۱۳۵ درجه قرار گرفته و فن ها به اندازه ۴۵ درجه در جهت ساعتگرد می چرخند. در نتیجه می توان پهنای پالس ورودی به موتور سروو را به صورت تابعی از موقعیت موتور سروو طبق رابطه زیر محاسبه نمود.

^۱ Duty cycle

$$PulseWidth = \frac{100}{9} (ShaftAngle - 45) + 1000 \quad (۱-۴)$$

که در رابطه بالا پهنای پالس برحسب میکروثانیه و زاویه شافت برحسب درجه می‌باشد. از این رابطه در بخش مربوط به برنامه‌نویسی پردازنده برای مشخص کردن موقعیت موتور سروو و در نتیجه وضعیت فن‌ها می‌توان استفاده نمود.

همچنین در یک آزمایش دیگر با قرار دادن دستگاه بر روی یک دستگاه سنجش وزن و فرستادن سیگنال‌ها با چرخه کاری مختلف و ثبت وزن مربوط به هر سیگنال در جدولی مانند جدول ۴-۴ می‌توان رابطه بین زمان High بودن پالس و نیروی تولیدی در هر فن را بدست آورد.

این آزمایش طی ۳ مرحله انجام شده است و در هر مرحله با فرستادن سیگنال‌هایی با چرخه کاری مختلف از زمان High بودن ۱۲۰۰ میکروثانیه تا ۱۴۰۰ میکروثانیه، نیروی وارد شده ثبت گردیده است و نتایج در جدول ۴-۴ وارد شده‌اند.

جدول ۴-۴: نیروی وارده توسط دو فن به ازای پالس‌های مختلف ورودی

ردیف	زمان High بودن پالس (μs)	نیروی ثبت شده در مرحله اول آزمایش (gr)	نیروی ثبت شده در مرحله دوم آزمایش (gr)	نیروی ثبت شده در مرحله سوم آزمایش (gr)
۱	1200	53	50	46
۲	1210	81	76	71
۳	1220	110	103	96
۴	1230	139	129	121
۵	1240	170	158	148
۶	1250	201	188	179
۷	1260	230	215	206
۸	1270	257	242	230
۹	1280	287	267	255
۱۰	1290	314	295	282
۱۱	1300	340	320	305

337	345	370	1310	۱۲
356	373	394	1320	۱۳
381	397	422	1330	۱۴
402	420	445	1340	۱۵
426	443	473	1350	۱۶
445	464	497	1360	۱۷
468	489	518	1370	۱۸
482	509	547	1380	۱۹
506	530	557	1390	۲۰
523	542	579	1400	۲۱

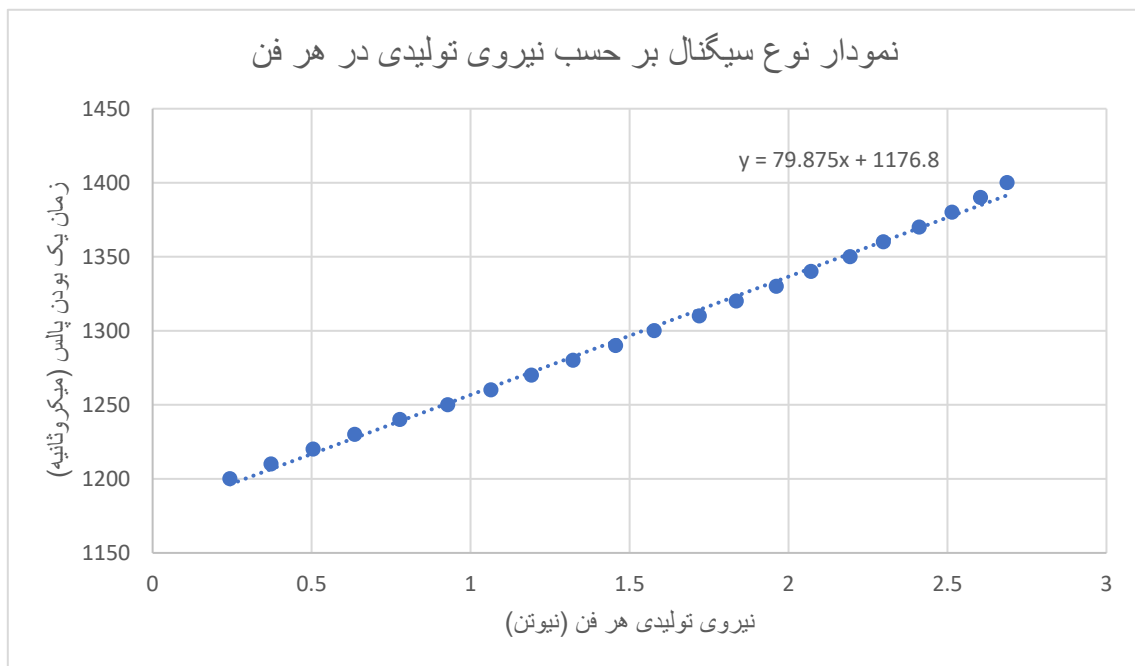
با توجه به محدودیت در مهار کردن دستگاه برای اندازه گیری نیروها و لرزش دستگاه در اثر افزایش بیشتر سرعت آن‌ها در ثبت نیروها به همین مقادیر بسنده شده است. هرچند که با توجه به کاربرد تعریف شده برای این دستگاه نیاز به نیروهای بیشتر نخواهد بود. نیروهای ثبت شده در جدول ۴-۴ مربوط به حالتی است که هر دو فن با هم کار کنند پس نیروی هر فن در هر پالس برحسب نیوتن را می‌توان در جدول ۴-۵ مشاهده نمود.

جدول ۴-۵ نیروی ثبت شده برای هر فن برحسب نیوتن برای پالس‌های مختلف

ردیف	زمان High بودن پالس (μs)	نیروی محاسبه شده در مرحله اول آزمایش (N)	نیروی محاسبه شده در مرحله دوم آزمایش (N)	نیروی محاسبه شده در مرحله سوم آزمایش (N)	میانگین نیروهای محاسبه شده (N)
۱	1200	0.259965	0.24525	0.22563	0.243615
۲	1210	0.397305	0.37278	0.348255	0.37278
۳	1220	0.53955	0.505215	0.47088	0.505215
۴	1230	0.681795	0.632745	0.593505	0.636015
۵	1240	0.83385	0.77499	0.72594	0.77826

0.92868	0.877995	0.92214	0.985905	1250	۶
1.064385	1.01043	1.054575	1.12815	1260	۷
1.191915	1.12815	1.18701	1.260585	1270	۸
1.322715	1.250775	1.309635	1.407735	1280	۹
1.456785	1.38321	1.446975	1.54017	1290	۱۰
1.577775	1.496025	1.5696	1.6677	1300	۱۱
1.72002	1.652985	1.692225	1.81485	1310	۱۲
1.836105	1.74618	1.829565	1.93257	1320	۱۳
1.962	1.868805	1.947285	2.06991	1330	۱۴
2.071545	1.97181	2.0601	2.182725	1340	۱۵
2.19417	2.08953	2.172915	2.320065	1350	۱۶
2.29881	2.182725	2.27592	2.437785	1360	۱۷
2.411625	2.29554	2.398545	2.54079	1370	۱۸
2.51463	2.36421	2.496645	2.683035	1380	۱۹
2.604555	2.48193	2.59965	2.732085	1390	۲۰
2.68794	2.565315	2.65851	2.839995	1400	۲۱

همانطور که مشاهده می‌شود در طی انجام آزمایش در هر مرحله به دلیل کم شدن شارژ باتری تعبیه شده بر روی دستگاه نیروهای تولید شده در هر فن به ازای چرخه کاری ثابت، کاهش می‌یابد. این اتفاق را می‌توان از ضعف‌های سیستم در نظر گرفت چرا که نیروی وارده توسط فن‌ها علاوه بر چرخه کاری سیگنال وارده به آن، به شارژ باتری نیز بستگی خواهد داشت. ولی با توجه به اختلاف ناچیز این نیروها می‌توان از داده‌های موجود برای هر چرخه کاری سیگنال میانگین گرفت و به عنوان نیروی تولیدی فن در آن سیگنال در نظر گرفت. که این میانگین‌ها نیز در جدول ۴-۵ وارد شده‌اند. حال با رسم نمودار زمان High بودن پالس بر حسب میانگین نیروهای محاسبه شده و گذراندن خط از این داده‌ها می‌توان زمان High بودن پالس را به صورت تابعی از نیروی تولید شده در هر فن را بدست آورد.



شکل ۴-۵ نمودار نوع سیگنال بر حسب نیروی تولیدی در هر فن

همانطور که در نمودار بالا مشاهده می‌شود پهنای بر حسب نیروی تولیدی هر فن را می‌توان از رابطه زیر محاسبه نمود.

$$PulseWidth = 79.875 \times Force + 1176.8 \quad (۴-۲)$$

که در رابطه بالا پهنای پالس بر حسب میکروثانیه و نیروی تولیدی بر حسب نیوتن می‌باشد. از این رابطه می‌توان در قسمت مربوط به برنامه‌نویسی پردازنده برای مشخص کردن پهنای پالس مورد نیاز برای تولید هر نیرو استفاده نمود.

۴-۴ سنجش کارایی دستگاه

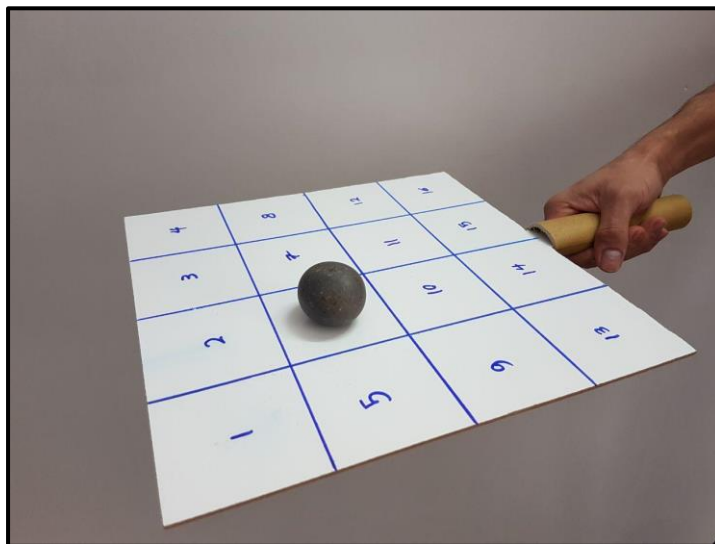
در این قسمت از پایان‌نامه به جهت ارزیابی و سنجش کارایی دستگاه، بر روی چند شرکت کننده مطالعاتی صورت گرفته است. اساس طراحی این تست به منظور مقایسه عملکرد دستگاه ساخته شده و نمونه واقعی احساس گوی بر صفحه می‌باشد. بدین منظور از ۴ شرکت کننده با سن‌های مختلف استفاده شده است که مشخصات هر کدام از آن‌ها و داده‌های جمع‌آوری شده از آنها را می‌توان در پیوست ب مشاهده نمود.

انجام این تست را می‌توان به دو بخش تقسیم نمود. در بخش اول مطالعه بر روی شرکت کنندگان با استفاده از نمونه واقعی صفحه‌ای است که وزنه‌ای بر روی آن قرار داده می‌شود و در بخش دوم برای این منظور از دستگاه

ساخته شده در این پروژه استفاده می شود. در ادامه بر بررسی هر کدام از این تست ها و نتایج حاصل شده پرداخته می شود.

۴-۴-۱ نحوه انجام تست ها

برای مطالعه بخش اول تست ابتدا صفحه ای با ابعاد و شکل تقریبی صفحه طراحی شده در محیط شبیه سازی، ساخته شده است. همانطور که در شکل ۴-۶ مشاهده می شود، صفحه موجود به قسمت های مختلف تقسیم شده و برای هر قسمت عددی در نظر گرفته شده است.



شکل ۴-۶ صفحه ساخته شده به منظور تست دستگاه

نحوه انجام بخش اول تست به این صورت است که ابتدا از شرکت کننده درخواست می شود صفحه را مطابق شکل ۴-۶ در دست گیرد و سپس با قرار دادن وزنه ی ۱۰۰ گرمی در موقعیت های ۱، ۴، ۱۳ و ۱۶ که چهار گوشه ی صفحه می باشند، از او خواسته می شود که به گشتاورهای وارده به دست خود دقت کند تا با قرار دادن وزنه در سایر موقعیت ها در مرحله بعد، شرکت کننده بتواند موقعیت وزنه را به نسبت این چهار موقعیت ذکر شده تشخیص دهد. در حین تست با استفاده از هدفون مجهز به فناوری حذف صداهای محیط بیرون^۱، برای شرکت کننده آهنگی ملایم پخش می شود تا صداهای خارجی ممکن تاثیری در نتیجه مطالعه نداشته باشد. هر مرحله از آزمایش شامل مدت زمان ۱۰ دقیقه می باشد و در این زمان از شرکت کننده خواسته می شود موقعیت گویی که توسط مسئول برگزارکننده ی تست به صورت تصادفی^۲ تغییر می کند را بدون نگاه کردن به صفحه،

^۱ Noise Canseling

^۲ Random

تشخیص داده و اعلام کند. اعداد تصادفی به گونه‌ای تهیه شده اند که در آخر هر شرکت کننده برای هر موقعیت مشخص شده بر روی صفحه ۴ عدد را تشخیص می‌دهد که این اعداد در فرم‌هایی که در پیوست آورده شده اند، وارد می‌شوند. در پایان هر ۱۰ دقیقه به شرکت کننده اجازه داده می‌شود که با حرکات کششی مناسب به دست خود استراحت دهد و برای ادامه تست آماده شود.

نحوه انجام بخش دوم تست مشابه بخش اول می‌باشد، با این تفاوت که در این بخش شرکت کننده به جای صفحه نشان داده شده در شکل ۴-۶ دستگاه ساخته شده در این پروژه را در دست می‌گیرد. پس از شبیه‌سازی نقاط گوشه ذکر شده توسط دستگاه، موقعیت‌های مختلف گوی توسط دستگاه برای فرد ایجاد شده و از او می‌خواهیم که موقعیت حس شده را اعلام کند. در قسمت مربوط به محاسبه نیروی فن‌ها به جهت شبیه‌سازی دقیق از محیط، وزن گوی و صفحه و همچنین موقعیت مرکز جرم صفحه بر اساس صفحه ساخته شده و وزنه قرار داده شده بر روی صفحه به‌روزرسانی شده اند. بدین صورت که وزن گوی ۱۰۰ گرم، وزن صفحه ۲۳۳ گرم و فاصله مرکز جرم صفحه از محل در دست گرفتن آن برابر با ۲۴۴ میلی‌متر می‌باشد.

همانطور که گفته شد، در این تست از ۴ شرکت کننده استفاده شده است و همچنین برای هر موقعیت ۴ عدد حدس زده شده توسط شرکت کنندگان ثبت می‌شود و با توجه به اینکه صفحه دارای ۱۶ موقعیت می‌باشد در هر بخش به طور کلی ۲۵۶ داده جمع‌آوری می‌شود و با توجه به اینکه این تست شامل ۲ بخش است، در ادامه به بررسی این ۵۱۲ داده‌ی ثبت شده پرداخته می‌شود.

۴-۴-۲ بررسی نتایج

داده‌های ثبت شده برای هر شرکت کننده را می‌توان در پیوست ب مشاهده نمود. در این بخش داده‌ها را در جدولی وارد شده و سپس با معرفی روشی، برای هر موقعیت مشخص شده بر روی صفحه درصدی را که بیانگر صحت تشخیص آن موقعیت توسط فرد می‌باشد، محاسبه می‌گردد.

داده‌های تمامی شرکت کننده ها برای بخش اول تست که مربوط به محیط واقعی گوی بر روی صفحه است در جدول ۴-۶ آورده شده اند. داده‌های وارد شده در این جدول به این صورت می‌باشند که اعداد ۱ تا ۱۶ نوشته شده در سمت راست جدول مشخص کننده موقعیت وزنه بر روی صفحه می‌باشند و اعداد ۱ تا ۱۶ نوشته شده در بالای جدول بیانگر عدد حدس زده شده توسط شرکت کننده است. به طور مثال برای موقعیت ۱۴، ۱۲ حدس مربوط به موقعیت ۱۴، ۱ حدس مربوط به موقعیت ۱۳، ۱ حدس مربوط به موقعیت ۱۵ و ۲ حدس مربوط به

موقعیت ۱۰ می باشد.

جدول ۴-۶ داده های مربوط به تست در محیط واقعی

موقعیت های حدس زده شده																
16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	
										3	2			1	10	1
			1			2				2	2	1	2	6		2
				1	2				6			3	4			3
1				2				6				7				4
			2			1	8				4				1	5
		1			1	3	1		1	5	1		1	2		6
	1			4	3	1			3			1	3			7
4	1			6				3				2				8
		1	4				4				7					9
			1		3	7			1	3	1					10
2	3				7				4							11
4	2			6	1			2	1							12
			8			1	4				2				1	13
	1	12	1			2										14
2	8	1			5											15
8	2			5				1								16

۱۶ موقعیت مشخص شده بر روی صفحه

در جدول ۴-۶ اعداد مشخص شده با رنگ آبی پررنگ مربوط به تعداد دفعات حدس های درست مربوط

به هر موقعیت و اعداد با رنگ آبی کم رنگ مربوط به حدس هایی غیر از عدد موقعیت موجود می باشند.

داده های مربوط به بخش دوم تست نیز به همین صورت جمع آوری شده اند که در مشاهده می شوند.

جدول ۴-۷ داده‌های مربوط به تست در محیط مجازی (با دستگاه ساخته شده در این پروژه)

موقعیت‌های حدس زده شده																
16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	
											1			1	14	1
							1			2	3			5	5	2
								3	1			4	7	1		3
				1				3	1			11				4
							4			1	8			2	1	5
			2			3	2			3	3			3		6
1			1	1	1			4	6				2			7
				2				5				9				8
						1	3			1	9				2	9
		3	4			3	2			3	1					10
2	5	1		4	1	1			1	1						11
4			1	3				7				1				12
			5			2	4			2	3					13
		11	3			1					1					14
3	13															15
7				4	2			1	2							16

۱۶ موقعیت مشخص شده بر روی صفحه

نحوه وارد کردن داده‌های مربوط به این بخش کاملاً مشابه بخش اول یعنی تست در محیط واقعی می‌باشند و اعداد آبی پررنگ بیانگر موقعیت‌های درست حدس زده شده و اعداد آبی کم‌رنگ مربوط به حدس‌هایی غیر از موقعیت موردنظر است.

همانطور که گفته شد برای هر موقعیت در هر بخش از تست ۱۶ داده موجود می‌باشد. درصدی از این داده‌ها مربوط به حدس درست و موقعیت موردنظر می‌باشند، درصد دیگری از این داده‌ها مربوط به حدس موقعیت‌های مجاور موقعیت موردنظر است و در آخر درصدی نیز مربوط به سایر موقعیت‌های صفحه می‌باشند. در این پژوهش برای محاسبه‌ی درصد حدس صحیح برای هر موقعیت از ۱۶ موقعیت حدس زده شده موقعیت‌های درست را با ضریب ۱، موقعیت‌های مجاور موقعیت موردنظر را با ضریب ۰/۲۵ و سایر موقعیت‌ها را با ضریب صفر با یکدیگر جمع می‌کنیم.

به طور مثال درصد حدس صحیح در مورد موقعیت ۴ از بخش دوم تست به این صورت محاسبه می گردد. همانطور که در جدول ۷-۴ مشاهده می شود حدس های مربوط به این موقعیت به اینصورت است که ۱۱ حدس مربوط به موقعیت ۴، ۱ حدس مربوط به موقعیت ۷، ۳ حدس مربوط به موقعیت ۸ و ۱ حدس مربوط به موقعیت ۱۲ می باشد.

همانطور که در شکل ۷-۴ مشاهده می شود، موقعیت های ۳، ۷ و ۸، موقعیت های مجاور موقعیت ۴ محسوب می گردند و حدس های مربوط به سایر موقعیت ها به غیر موقعیت های ۴ و ۳، ۷ و ۸ در محاسبه به حساب نمی آیند چرا که از موقعیت موردنظر دور بوده و داده اشتباه به حساب می آیند.

۱	۲	۳	۴
۵	۶	۷	۸
۹	۱۰	۱۱	۱۲
۱۳	۱۴	۱۵	۱۶

شکل ۷-۴ موقعیت های مشخص شده بر روی صفحه

همانطور که گفته شد برای موقعیت ۴، ۱۱ حدس مربوط به این موقعیت است و ضریب ۱ را به خود می گیرند، ۱ حدس مربوط به موقعیت ۷ و ۳ حدس مربوط به موقعیت ۸ هستند که ضریب ۰/۲۵ را به خود می گیرند و ۱ حدس مربوط به عدد ۱۲ است که در شمارش به حساب نمی آید. بدین ترتیب درصد حدس صحیح برای این موقعیت و در مورد بخش دوم تست از رابطه زیر محاسبه می گردد.

$$CorrectGuess(\%)_{Position\ 4} = \frac{1 \times 11 + 0.25 \times (1 + 3) + 0 \times 1}{16} \times 100 = 75\%$$

دلیل استفاده از ضریب ۰/۲۵ برای موقعیت های مجاور این است که، در بدترین شرایط حداقل تعداد حدسی که باید از موقعیت های مجاور زده شود تا به موقعیت موردنظر پی برده شود، ۴ می باشد.

به همین ترتیب برای سایر موقعیت‌ها و در هر بخش می‌توان این درصد را محاسبه نمود. درصد‌های محاسبه شده برای هر موقعیت و بخش در جدول ۴-۸ آورده شده‌اند.

جدول ۴-۸ درصد حدس‌های درست هر موقعیت در هر دو بخش تست دستگاه

شماره موقعیت	درصد حدس‌های درست در محیط واقعی	درصد حدس‌های درست در محیط مجازی
1	71.875	90.625
2	46.875	46.875
3	39.0625	57.8125
4	53.125	75
5	40.625	62.5
6	46.875	35.9375
7	37.5	50
8	31.25	48.4375
9	43.75	35.9375
10	57.8125	39.0625
11	57.8125	29.6875
12	53.125	35.9375
13	57.8125	40.625
14	81.25	75
15	57.8125	85.9375
16	60.9375	53.125
میانگین	52.34375	53.90625

همانطور که در جدول ۴-۸ مشاهده می‌شود دستگاه ساخته شده در مورد نیمه بالایی صفحه که از موقعیت ۱ تا ۸ می‌باشد عملکرد خوبی نسبت به محیط واقعی از خود نشان می‌دهد و در اکثر موقعیت‌های ناحیه بالایی صفحه درصد‌های حدس درست در محیط مجازی از محیط واقعی بیشتر است. اما در ناحیه پایین صفحه که شامل اعداد ۹ تا ۱۶ می‌باشد دستگاه به نسبت محیط واقعی درصد‌های کمتری از حدس‌های صحیح را به خود اختصاص داده است و دلیل آن را می‌توان افزایش نیروی فن برای ایجاد گشتاور بیشتر و در نتیجه خطا در حس ایجاد شده دانست. به طور مثال دستگاه برای ایجاد حس موقعیت ۱۳ نسبت به موقعیت ۱۴ به اینصورت عمل می‌کند که فن سمت چپ را با سرعت بیشتری به چرخش در می‌آورد تا گشتاور مورد نیاز حول محور دسته دستگاه را ایجاد

کند که این امر موجب ایجاد گشتاوری بیشتر حول محور عمود بر دسته دستگاه نیز می شود که مطلوب ما نیست و همانطور که در جدول ۴-۷ مشاهده می شود برای موقعیت ۱۳، تعداد زیادی از حدس ها به موقعیت های ۹ و ۵ که گشتاور بیشتری حول محور عمود بر دسته دستگاه را می طلبند، اختصاص یافته است. به همین ترتیب برای سایر موقعیت های نیمه پایین صفحه نیز می توان این نکته را یادآور شد. در فصل آینده به ارائه پیشنهادی برای برطرف نمودن این مشکل پرداخته می شود.

همچنین همانطور که از درصدهای ارائه شده مشخص است به طور تقریبی درصد حدس های صحیح نیمه سمت چپ صفحه، چه در حالت واقعی و چه مجازی از درصدهای سمت راست بیشتر است. که دلیل این امر را می توان به استفاده از دست چپ و یا راست فرد دانست. در این تست تمامی شرکت کنندگان دستگاه و صفحه را توسط دست راست خود نگه داشته بودند و از آنجایی که چرخش دست راست در جهت پادساعتگرد آسان تر می باشد، در نتیجه موقعیت های سمت چپ که دست را به همین صورت می چرخانند بهتر حدس زده شده اند. و در آخر از میانگین کلیه درصدها مشاهده می شود که درصد حدس صحیح در محیط واقعی عدد ۵۲/۳۴ و در محیط مجازی و با استفاده از دستگاه عدد ۵۳/۹۱ می باشد که نزدیک بودن حس ایجاد شده در فرد توسط دستگاه، به محیط واقعی را نشان می دهد.

۵-۴ دستورالعمل استفاده از دستگاه

مراحل به کار انداختن و استفاده از دستگاه را می توان به طور کلی به دو بخش راه اندازی دستگاه و اتصال به نرم افزار سیمولینک تقسیم کرد. در ادامه تمامی مراحل راه اندازی و اتصال دستگاه به سیمولینک با شرح داده شده است.

۱- قبل شروع کار با دستگاه ابتدا باید از شارژ بودن باتری دستگاه اطمینان حاصل نمود. شکل ۴-۸ نشان می‌دهد که با سبز شدن هر سه چراغ تعبیه شده بر روی شارژر هر سه سلول باتری دستگاه به طور کامل شارژ شده‌اند.



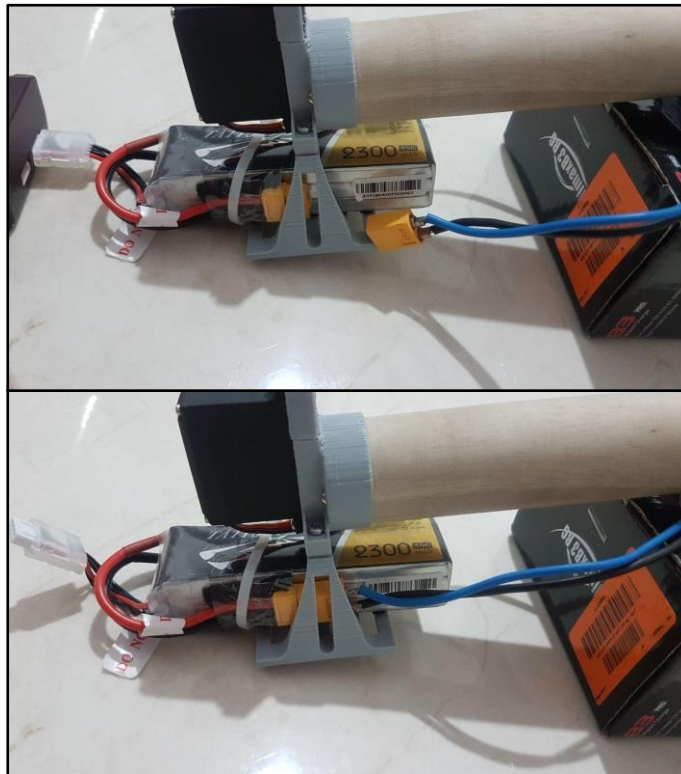
شکل ۴-۸ شارژ شدن باتری دستگاه

۲- همچنین قبل از روشن کردن دستگاه باید از غیرفعال بودن کلید تعبیه شده بر روی دستگاه برای فعال کردن فن‌ها، مطمئن باشیم. مطابق شکل ۴-۹ وضعیت غیرفعال این کلید به سمت دسته دستگاه می‌باشد.



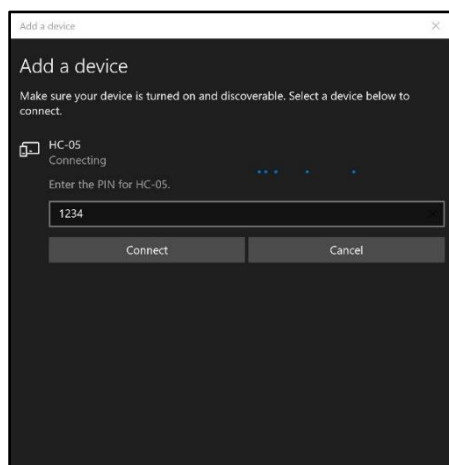
شکل ۴-۹ وضعیت غیرفعال کلید فعال‌سازی فن‌ها

۳- برای روشن کردن دستگاه باید سوکت تغذیه دستگاه را مطابق شکل ۴-۱۰ به باتری متصل کرد. با متصل کردن دستگاه به باتری صدایی از موتورهای براشلس دستگاه شنیده می شود. با شنیدن صدای ممتد پس از حدود ۱۰ ثانیه از اتصال دستگاه به باتری، مرحله راه اندازی دستگاه به پایان می رسد.



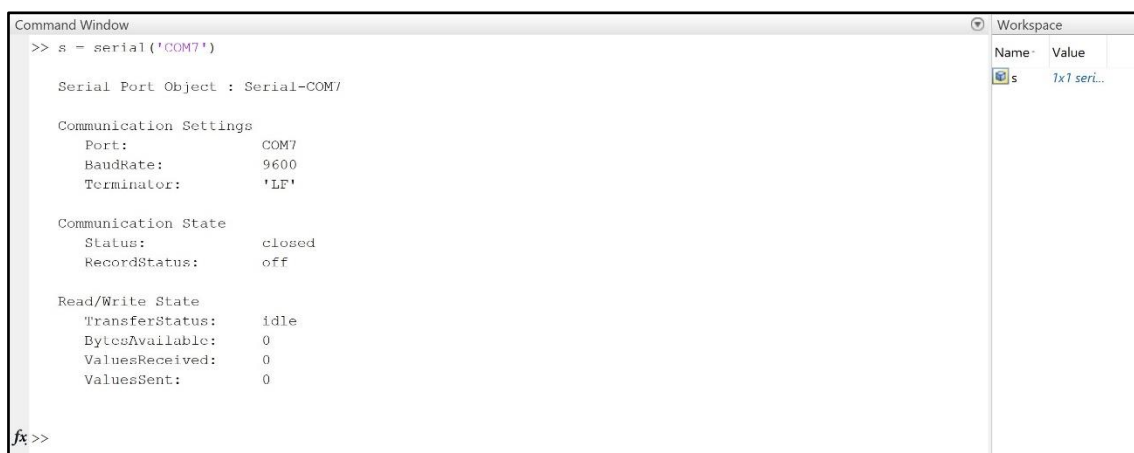
شکل ۴-۱۰ اتصال سوکت تغذیه دستگاه به باتری

۴- حال به مرحله اتصال دستگاه به نرم افزار سیمولینک می رسیم. ابتدا درگاه بلوتوث رایانه را روشن کرده و با جستجو در دستگاه های بلوتوث موجود در اطراف رایانه به دستگاه بلوتوث با نام HC-05 متصل می شویم. رمز این دستگاه عبارت ۱۲۳۴ می باشد.



شکل ۴-۱۱ اتصال رایانه به ماژول بلوتوث دستگاه

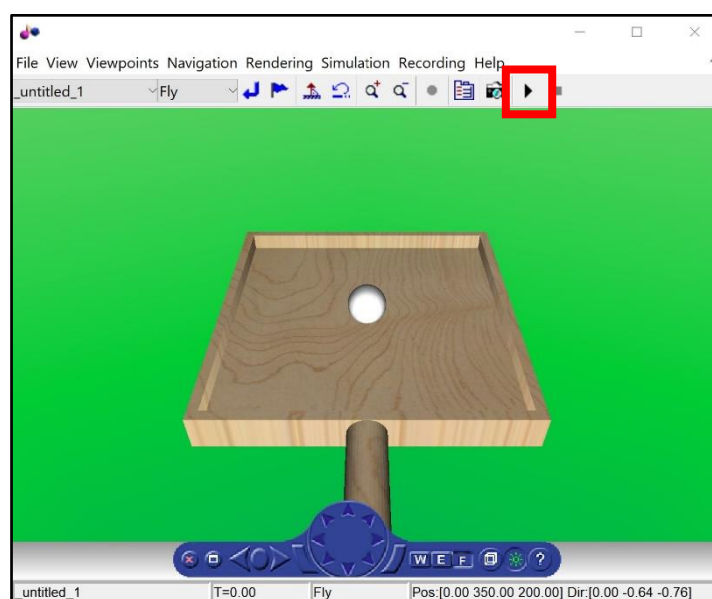
۵- برای اتصال دستگاه به محیط شبیه‌سازی ابتدا باید در نرم‌افزار متلب یک متغیر با نام S که حاوی اطلاعات مربوط به پورت سریال برای انتقال داده است، تعریف شود برای این منظور عبارت $s = \text{serial}('COM7')$ را در قسمت Command window نرم‌افزار متلب تایپ کرده و سپس کلید Enter را می‌زنیم.



شکل ۴-۱۲ تعریف متغیر S شامل اطلاعات پورت سریال

همانطور که در شکل ۴-۱۲ مشاهده می‌شود، یک متغیر با مشخصات نوشته شده در قسمت Workspace متلب تعریف شده است.

۶- حال فایل سیمولینک ساخته شده را باز کرده و با اجرا کردن آن مشاهده می‌شود که صفحه مجازی با حرکت دستگاه، حرکت می‌کند.



شکل ۴-۱۳ اجرای فایل سیمولینک و اتصال دستگاه به محیط شبیه‌سازی

۷- در این مرحله می‌توان با حرکت دادن دستگاه، صفحه شبیه‌سازی شده و در نتیجه گوی را حرکت داد ولی فن‌ها و موتور سروو در حالت خاموش خود می‌باشند. حال در صورتی که دست فرد در وضعیت مناسبی قرار دارد و فرد پس از مدتی با محیط شبیه‌سازی آشنا شده است، می‌توان کلید فعال‌سازی فن‌ها را در حالت فعال یعنی رو به طرف فن‌ها قرار داد و مشاهده می‌شود که فن‌ها و موتور سروو شروع به کار می‌کنند.



شکل ۴-۱۴ فعال‌سازی فن‌ها و موتور سروو

۶-۴ جمع‌بندی

مطالب این فصل را می‌توان به سه بخش کلی طراحی، مونتاژ و بررسی سیستم و دستورالعمل استفاده از دستگاه تقسیم کرد. در قسمت طراحی به بررسی قطعات مکانیکی طراحی شده و همچنین نحوه اتصال مدارات الکترونیکی پرداخته شد. در بخش مربوط به مونتاژ و بررسی سیستم به نحوه مونتاژ قطعات و اتصال آن‌ها به یکدیگر و در آخر به بررسی وزن دستگاه و چگونگی کنترل موتور سروو و فن‌ها، نحوه بدست آوردن رابطه بین پهنای پالس ورودی به موتور سروو و فن‌ها با موقعیت شافت موتور سروو و نیروی تولیدی توسط فن‌ها و همچنین به انجام تست‌هایی برای سنجش کارایی دستگاه، پرداخته شد. و در آخر دستورالعمل کامل نحوه راه‌اندازی دستگاه با تصاویری مربوط به هر مرحله بیان گردید.

فصل پنجم

نتیجه‌گیری و پیشنهادها

۱-۵ مقدمه

همانگونه که امروزه کاربرد فناوری واقعیت مجازی را می‌توان در زمینه‌های مختلف اعم از آموزشی، بازی و سرگرمی، نظامی، پزشکی و غیره مشاهده نمود، در نتیجه نزدیک کردن هرچه بیشتر این محیط مجازی به محیط واقعی می‌تواند تاثیر به سزایی در ارتباط برقرار کردن فرد با محیط داشته باشد. برای نزدیک شدن بیشتر این فضای مجازی به واقعیت می‌توان علاوه بر المان‌های دیداری و شنوایی از المان‌های تحریک حس لامسه بهره برد. عملگرهای مختلفی برای ایجاد تحریک در حس لامسه وجود دارند که عملگرهای ایجاد نیروی مجازی را می‌توان در دسترس‌ترین این مجموعه دانست. منشا این نیرو می‌تواند مختلف باشد ولی در نهایت چیزی که اهمیت پیدا می‌کند ایجاد هرچه بهتر حس واقعیت در فرد می‌باشد.

۲-۵ شرح خلاصه‌ای درمورد پروژه و پایان‌نامه

همانطور که در بخش مقدمه این فصل گفته شد تحریک حس لامسه با وارد کردن نیرو به فرد را می‌توان در عین در دسترس بودن، راه موثری برای نزدیک کردن محیط مجازی به واقعیت دانست. در این پروژه سعی شده است که این نیرو به وسیله چرخش دو فن بر روی یک دستگاه ایجاد شود. از آنجا که در این دستگاه از دو فن

استفاده شده است می توان نیروها و گشتاورهایی با اندازه و جهت مختلف را ایجاد و به دست فرد موردنظر وارد نمود. محیط شبیه سازی در این پروژه به گونه ای انتخاب شده است که فرد با حرکت دادن دستگاه موجب حرکت صفحه در محیط شبیه سازی می شود و به موجب آن گویی بر روی این صفحه حرکت کرده و در موقعیت های مختلفی قرار می گیرد. با حرکت گوی در موقعیت های مختلف بر روی صفحه گشتاورهای مختلفی به دست فرد باید وارد شود که فن های تعبیه شده بر روی دستگاه وظیفه تامین این گشتاورها را بر عهده دارند.

در فصل دوم به بیان نحوه ایجاد محیط شبیه سازی و همچنین محاسبات مربوط به موقعیت گوی بر اساس وضعیت صفحه و در نهایت محاسبه میزان نیروی موردنیاز هر فن بر اساس موقعیت گوی پرداخته شد.

در فصل سوم نیز برنامه نویسی پردازنده جهت رسیدن به هدف پروژه بیان شد که شامل نحوه خواندن اطلاعات از سنسور شتاب سنج و ژيروسکوپ، محاسبات مربوط به موقعیت گوی و نیروی موردنیاز هر فن و چگونگی ارسال اطلاعات به رایانه و محیط شبیه سازی می باشد.

همچنین در فصل چهارم ابتدا به نحوه طراحی قطعات مختلف سیستم و چگونگی اتصال و ارتباط آنها به یکدیگر پرداخته شد و سپس آزمایش هایی جهت مشاهده رفتار سیستم و تست های جهت ارزیابی عملکرد دستگاه انجام شد و در نهایت دستورالعمل استفاده از دستگاه ساخته شده، شرح داده شد.

۳-۵ نوآوری

اساسا در ساخت این دستگاه از یک مدل ساخته شده تحت عنوان Aero-plane الهام گرفته شده است [1]، و می توان دستگاه ساخته شده در این پروژه را مدل ارتقا یافته شده دستگاه Aero-plane دانست. از جمله امکانات اضافه شده در این دستگاه می توان به موارد زیر اشاره نمود.

۱- فن های تولید نیروی پیشران در دستگاه Aero-plane ثابت می باشند. بدین معنی که این دستگاه توانایی ایجاد نیرو تنها در جهت عمود بر راستای دسته خود و به سمت پایین را دارد، در حالی که در این پروژه با قرار دادن موتور سروو^۱ در انتهای دستگاه و متصل کردن مجموعه فن ها به محور آن این قابلیت به دستگاه داده شده است که در جهات مختلف عمود بر دسته دستگاه به دست فرد نیرو و گشتاور وارد نماید که توضیحات بیشتر مربوط به این ویژگی در فصل های بعد آمده است.

۲- در دستگاه Aero-plane از یک وزنه در انتهای دستگاه به جهت نزدیک کردن مرکز جرم دستگاه

¹ Servo motor

به نقطه میانی دسته استفاده شده است و این درحالی است که باتری دستگاه بر روی دستگاه قرار ندارد و با سیم‌هایی از بیرون به دستگاه متصل شده‌اند. ولی در این پروژه با قرار دادن باتری در انتهای دستگاه به جای وزنه نه تنها وزن دستگاه افزایش پیدا نکرده است بلکه با استفاده از ارتباط بیسیم^۱ کلیه اتصالات از بیرون حذف گردیده که موجب آزادی حرکت بیشتر فرد شده‌است.

۵-۴ نتیجه‌گیری

با امتحان دستگاه در دو حالت فعال و غیرفعال بودن فن‌ها و مقایسه دو حالت با یکدیگر نتیجه می‌شود که در حالت فعال بودن فن‌ها فرد حس بهتری از کنترل گوی بر روی یک صفحه دارد. همچنین برای اطمینان از موثر بخش بودن دستگاه مشاهده می‌شود که در صورت متصل نبودن دستگاه به محیط شبیه‌سازی و تنها از روی نیروی فن‌ها، فرد به طور تقریبی می‌تواند موقعیت گوی را حدس بزند که بیانگر اضافه شدن المان دیگر برای احساس حضور بهتر فرد در محیط شده است.

۵-۵ پیشنهادها

در این قسمت از فصل به بررسی مشکلات و معایب سیستم پرداخته و راه‌حل را به عنوان پیشنهاد ارائه می‌کنیم.

همانطور که گفته شد در این پروژه از دو فن برای ایجاد گشتاور به دست فرد استفاده شده است. در نتیجه با منتقل کردن نیروی فن به دست فرد مشاهده می‌شود که به دست فرد یک نیرو و همچنین یک گشتاور وارد می‌شود. اساس کار سرعت فن‌ها و نیروی تولید شده در این دستگاه براساس گشتاورهای تولیدی آن‌ها می‌باشند و به طور مثال برای ایجاد گشتاور بزرگتر باید سرعت فن موردنظر افزایش یابد که موجب افزایش نیروی وارده به دست فرد نیز می‌شود که ممکن است مطلوب سیستم نباشد. راه‌حل رفع این مشکل می‌تواند استفاده از دو فن دیگر در انتهای دستگاه باشد، به طوری که فن‌های موجود در انتهای دستگاه نیرویی در خلاف جهت فن‌های تعبیه شده در جلوی دستگاه وارد کنند که موجب ایجاد گشتاور به هر مقدار می‌شود، در صورتی که نیروی وارد شده به دست فرد نیز تغییر نمی‌کند و حتی می‌تواند صفر باشد.

^۱ Wireless

همانطور که در فصل‌های پیشین ذکر شد، نیروی تولیدی در هر فن علاوه بر پهنای پالس فرستاده شده برای موتورها به میزان شارژ باتری نیز بستگی دارد و با تغییر میزان شارژ باتری ممکن است دقیقاً نیروی مورد نظر ما تولید نشود. که به عنوان راه‌حل می‌توان با سنجش میزان شارژ باتری در هر لحظه و انجام آزمایش‌هایی در مورد میزان نیروی وارده در هر میزان شارژ باتری و هر پهنای پاس، تابعی دو متغیره برای نیروی هر فن بر حسب پهنای پالس وارد شده به موتور و میزان شارژ باتری دستگاه را بدست آورد.

از جمله معایب این سیستم می‌توان به میزان صدای تولیدی در هر فن اشاره نمود که ممکن است باعث ایجاد حس ناخوشایند در فرد شود. برای رفع چنین مشکلی نیز می‌توان با در اختیار قرار دادن هدفون‌هایی به فرد و پخش موزیک و یا صداهایی از محیط شبیه‌سازی مانند صدای حرکت گوی بر روی صفحه تا حدودی تاثیر صدای نامطلوب فن‌ها را کاهش داد.

همانطور که گفته شد در این پروژه با قرار دادن موتور سروو در انتهای دستگاه سعی شده است که وضعیت فن‌ها برای شبیه‌سازی نیروی وزن صفحه به صورت افقی باشند. که این افقی بودن تنها در راستای محور گذرنده از دسته دستگاه امکان‌پذیر می‌باشد و فن‌ها امکان چرخش حول محور عمود بر دسته دستگاه را ندارند که در نسخه‌های بعدی این دستگاه می‌توان این قابلیت را نیز به سیستم افزود.

مراجع

- [1] S. Je, M. J. Kim, W. Lee, B. Lee, X.-D. Yang, P. Lopes and A. Bianchi, "Aero-plane: A Handheld Force-Feedback Device that Renders Weight Motion Illusion on a Virtual 2D Plane," in *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*, 2019.
- [2] M. Sagardia, B. Weber, T. Hulin, G. Hirzinger and C. Preusche, "Evaluation of visual and force feedback in virtual assembly verifications," in *2012 IEEE Virtual Reality Workshops (VRW)*, 2012.
- [3] S. Jeong, N. Hashimoto and S. Makoto, "A novel interaction system with force feedback between real-and virtual human: an entertainment system:" virtual catch ball"," in *Proceedings of the 2004 ACM SIGCHI International Conference on Advances in computer entertainment technology*, 2004.
- [4] C. Carignan, J. Tang and S. Roderick, "Development of an exoskeleton haptic interface for virtual task training," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009.
- [5] J. M. Walker, H. Culbertson, M. Raitor and A. M. Okamura, "Haptic orientation guidance using two parallel double-gimbal control moment gyroscopes," *IEEE transactions on haptics*, vol. 11, no. 2, pp.

267-278, 2017.

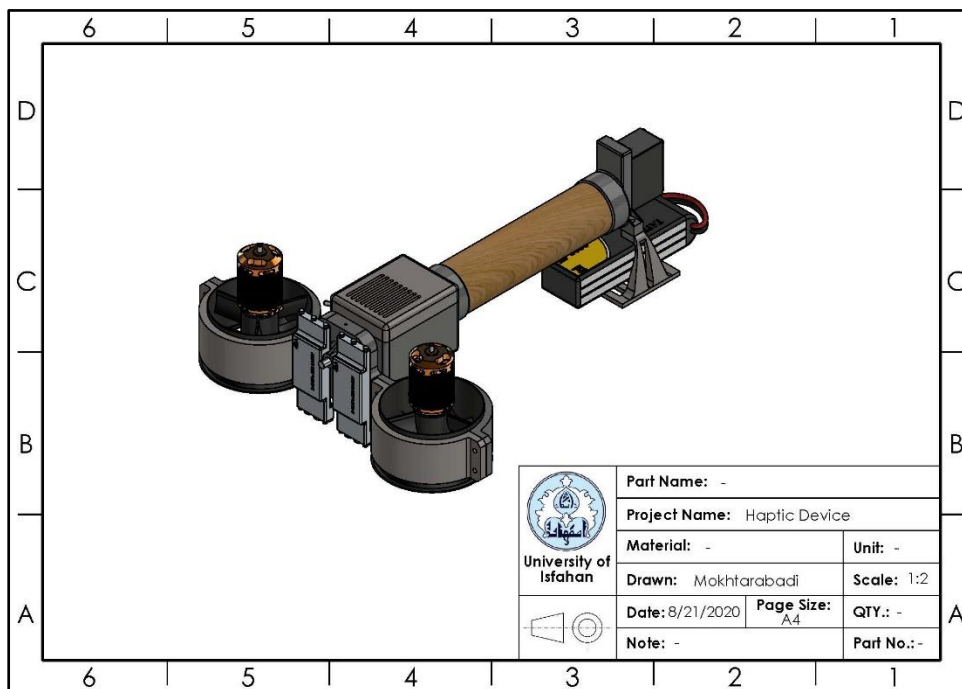
- [6] M. Antolini, M. Bordegoni and U. Cugini, "A haptic direction indicator using the gyro effect," in *2011 IEEE World Haptics Conference*, 2011.
- [7] K. N. Winfree, J. Gewirtz, T. Mather, J. Fiene and K. J. Kuchenbecker, "A high fidelity ungrounded torque feedback device: The iTorqU 2.0," in *World Haptics 2009-Third Joint EuroHaptics conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, 2009.
- [8] P. Lopes, S. You, L.-P. Cheng, S. Marwecki and P. Baudisch, "Providing haptics to walls & heavy objects in virtual reality by means of electrical muscle stimulation," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, 2017.
- [9] S. Heo, C. Chung, G. Lee and D. Wigdor, "Thor's hammer: An ungrounded force feedback device utilizing propeller-induced propulsive force," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, 2018.
- [10] S. Je, H. Lee, M. J. Kim and A. Bianchi, "Wind-blaster: a wearable propeller-based prototype that provides ungrounded force-feedback," in *ACM SIGGRAPH 2018 Emerging Technologies*, 2018, pp. 1-2.
- [11] J. Shigeyama, T. Hashimoto, S. Yoshida, T. Aoki, T. Narumi, T. Tanikawa and M. Hirose, "Transcalibur: weight moving VR controller for dynamic rendering of 2D shape using haptic shape illusion," in *ACM SIGGRAPH 2018 Emerging Technologies*, 2018, pp. 1-2.
- [12] C. Swindells, A. Unden and T. Sang, "TorqueBAR: an ungrounded haptic feedback device," in *Proceedings of the 5th international conference on Multimodal interfaces*, 2003.
- [13] A. Zenner and A. Krüger, "Shifty: A weight-shifting dynamic passive haptic proxy to enhance object perception in virtual reality," *IEEE transactions on visualization and computer graphics*, vol. 23, no. 4, pp. 1285-1294, 2017.
- [14] A. Zenner and A. Krüger, "Drag: on: A virtual reality controller providing haptic feedback based on drag and weight shift," in *Proceedings of the 2019 CHI Conference on Human Factors in*

Computing Systems, 2019.

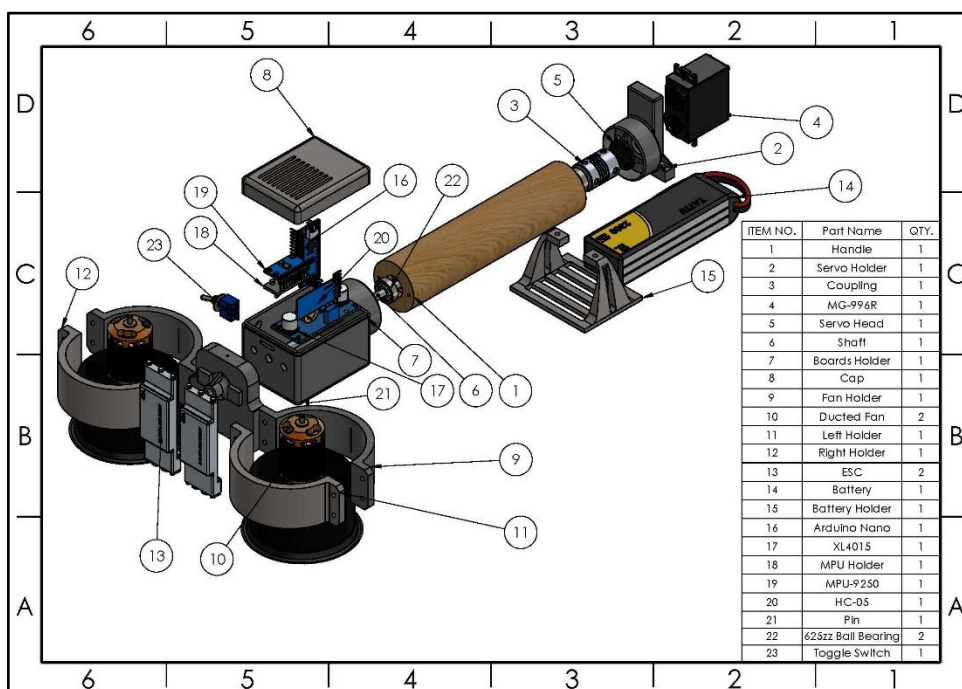
- [15] R. Niiyama, L. Yao and H. Ishii, "Weight and volume changing device with liquid metal transfer," in *Proceedings of the 8th International Conference on Tangible, Embedded and Embodied Interaction*, 2014.
 - [16] "What is Arduino?," Arduino, 5 February 2018. [Online]. Available: <https://www.arduino.cc/en/Guide/Introduction>. [Accessed 2020].
 - [17] P. Stoffregen, "AltSoftSerial," GitHub, 6 December 2019. [Online]. Available: <https://github.com/PaulStoffregen/AltSoftSerial>. [Accessed 2020].
 - [18] N. Bontrager, "ServoTimer2," GitHub, 5 June 2017. [Online]. Available: <https://github.com/nabontra/ServoTimer2>. [Accessed 2020].
 - [19] Bolder Flight Systems, "MPU9250," GitHub, 02 May 2020. [Online]. Available: <https://github.com/bolderflight/MPU9250>. [Accessed 2020].
 - [20] F. Abyarjoo, A. Barreto, J. Cofino and F. R. Ortega, "Implementing a sensor fusion algorithm for 3D orientation detection with inertial/magnetic sensors," in *Innovations and advances in computing, informatics, systems sciences, networking and engineering*, Springer, 2015, pp. 305-310.
 - [21] Starlino, "A Guide To using IMU (Accelerometer and Gyroscope Devices) in Embedded Applications," Starlino electronics, 29 December 2009. [Online]. Available: http://www.starlino.com/imu_guide.html. [Accessed 2020].
- [۲۲] س. س. حمیدی, "مدولاسیون پهنای پالس," فرادرس, ۲۰۱۸. December 13. [Online]. Available: <https://blog.faradars.org/pwm/>. [Accessed 2020].

پیوست الف

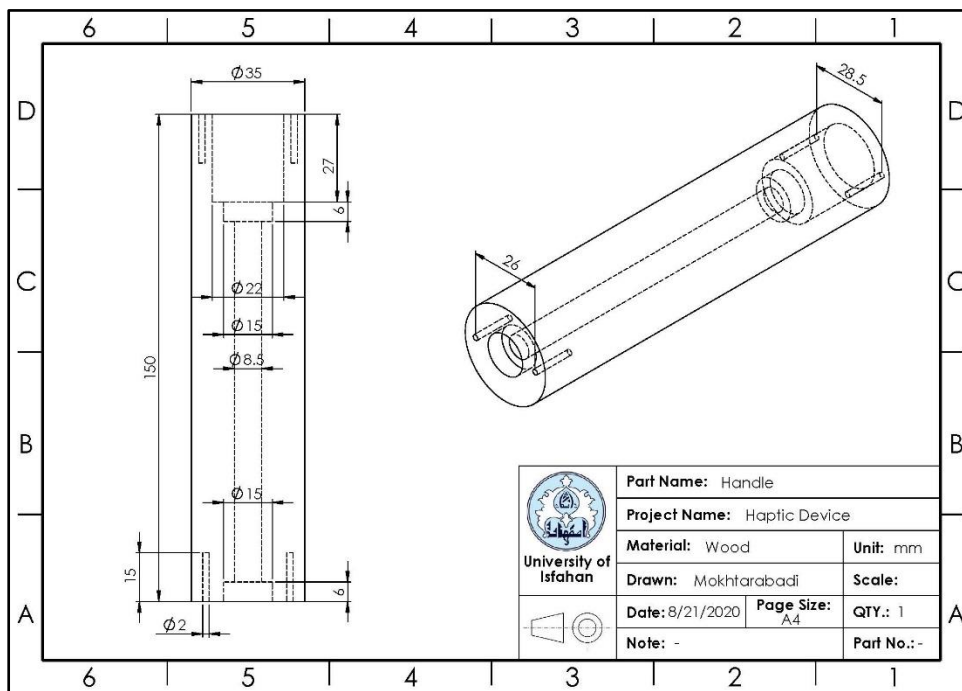
همانطور که در قسمت طراحی دستگاه گفته شد، دسته‌ای از قطعات مکانیکی دستگاه بایستی از روی نقشه‌های خروجی از روی قطعه طراحی شده در نرم‌افزار سالدورکس ساخته شوند. در این قسمت از پایان‌نامه می‌توان نقشه‌های ساخت این قطعات را به طور کامل مشاهده نمود و همچنین شکل کلی دستگاه و نقشه انفجاری دستگاه نیز در این قسمت به جهت راهنمایی برای مونتاژ قطعات دستگاه قرار داده شده‌است.



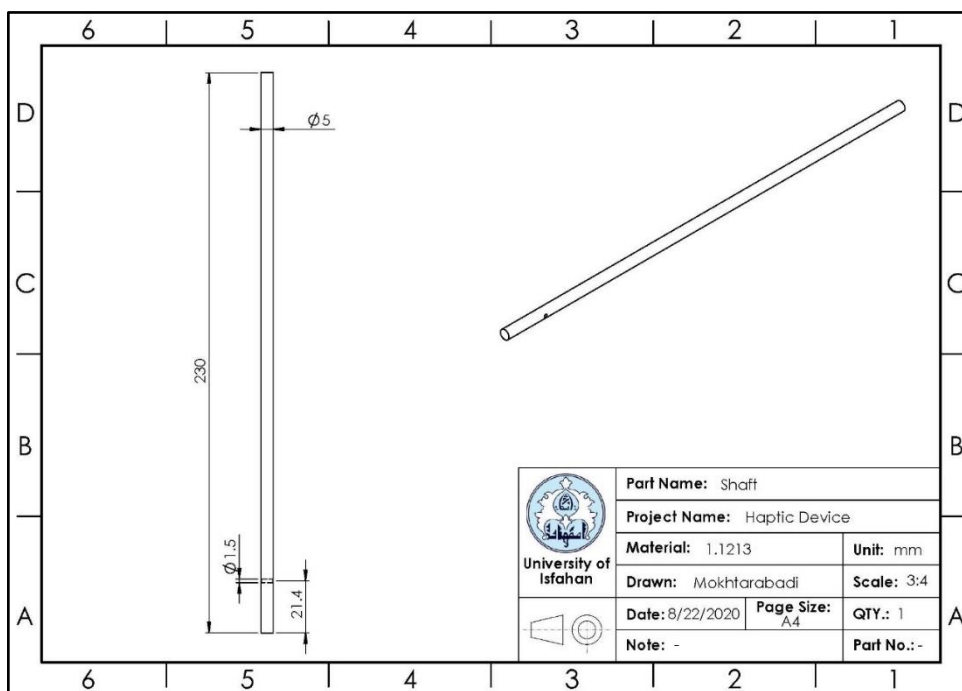
شکل ۷-۱ شکل کامل دستگاه



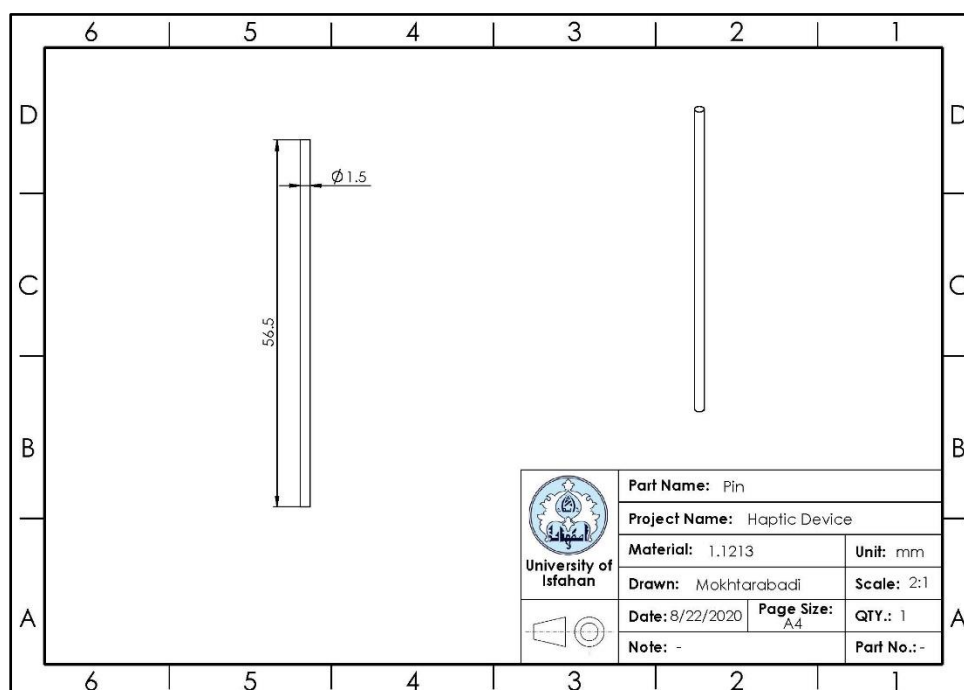
شکل ۷-۲ نقشه انفجاری دستگاه



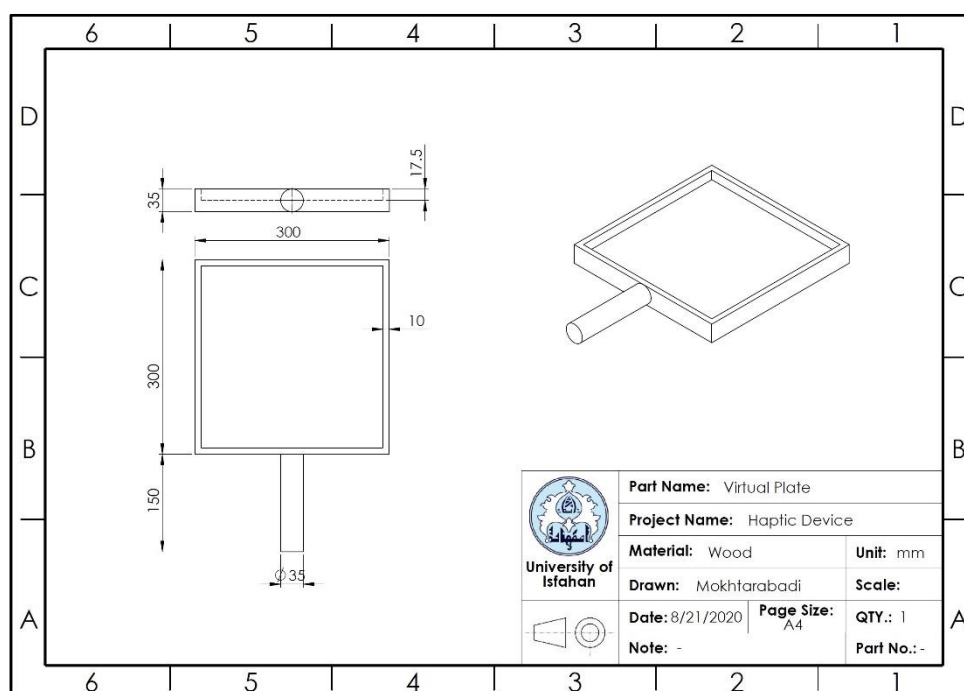
شکل ۷-۳ نقشه دسته دستگاه



شکل ۷-۴ نقشه شافت میانی دستگاه



شکل ۷-۵ نقشه پین دستگاه



شکل ۷-۶ نقشه صفحه طراحی شده در محیط مجازی

پیوست ب

در این قسمت فرم‌های مربوط به داده‌های تست معرفی شده در بخش ۴-۴ آورده شده است. در هر یک از این فرم‌ها اطلاعات شرکت کننده مانند نام و نام خانوادگی، سن و جنسیت آورده شده است. داده‌های وارد شده در این فرم به صورت ۴ عدد در جلوی هر موقعیت صفحه چه در حالت واقعی و چه در حالت مجازی می‌باشد که بدین معناست که هر شرکت کننده برای یک موقعیت ۴ داده در حالت واقعی و ۴ داده در حالت مجازی حدس زده است.

VR-Handle Performance Evaluation Form									
Participant Information:									
Name: Fatemeh Mokhtarabdi					Age: 42		Male <input type="checkbox"/> / Female <input checked="" type="checkbox"/>		
Reality					Virtual				
Position No.	Guessed Positions				Position No.	Guessed Positions			
1	1	6	1	6	1	5	1	1	1
2	3	13	2	2	2	5	1	6	1
3	7	12	7	7	3	4	2	3	3
4	4	16	4	8	4	8	4	4	4
5	5	9	9	5	5	5	9	5	9
6	3	11	6	10	6	2	6	5	13
7	12	11	3	7	7	7	7	3	7
8	4	8	8	12	8	8	8	4	4
9	13	13	9	9	9	6	5	5	9
10	10	6	10	11	10	10	10	5	6
11	7	7	11	15	11	10	14	6	12
12	8	12	8	16	12	16	8	8	16
13	10	9	13	5	13	9	13	5	10
14	14	14	14	15	14	10	14	14	5
15	11	16	11	15	15	15	15	15	15
16	12	16	16	12	16	16	7	16	8

VR-Handle Performance Evaluation Form									
Participant Information:									
Name: Ali Mokhtarabadi					Age: 50		Male <input checked="" type="checkbox"/> / Female <input type="checkbox"/>		
Reality					Virtual				
Position No.	Guessed Positions				Position No.	Guessed Positions			
1	6	1	5	1	1	1	1	1	1
2	6	5	2	2	2	6	5	2	9
3	3	11	11	7	3	3	8	3	3
4	8	4	8	8	4	4	4	8	8
5	9	9	9	9	5	9	5	5	5
6	10	5	6	6	6	6	9	13	5
7	11	3	3	7	7	7	7	7	8
8	12	4	12	8	8	8	8	4	4
9	5	5	5	5	9	10	5	1	5
10	5	10	10	6	10	13	9	6	6
11	11	7	11	11	11	12	12	12	7
12	12	15	12	7	12	16	8	8	8
13	13	1	5	9	13	6	6	5	9
14	14	14	10	10	14	14	14	13	13
15	15	14	16	11	15	15	15	15	16
16	8	16	12	12	16	11	12	12	7

VR-Handle Performance Evaluation Form									
Participant Information:									
Name: Aseman Mokhtarabadi					Age: 16		Male <input type="checkbox"/> / Female <input checked="" type="checkbox"/>		
Reality					Virtual				
Position No.	Guessed Positions				Position No.	Guessed Positions			
1	2	1	1	1	1	1	1	1	1
2	3	10	10	5	2	5	1	1	1
3	4	4	7	7	3	8	8	4	4
4	4	12	4	8	4	12	7	4	4
5	9	10	5	13	5	9	5	6	1
6	10	9	14	7	6	10	5	10	9
7	12	10	15	11	7	13	11	8	16
8	16	16	15	12	8	8	4	4	12
9	9	13	13	14	9	5	5	9	9
10	11	11	10	13	10	14	13	9	13
11	11	16	11	15	11	15	16	15	15
12	11	16	12	15	12	12	12	16	8
13	13	13	9	13	13	13	13	5	13
14	14	14	14	13	14	14	14	14	13
15	15	15	11	15	15	15	15	16	16
16	16	16	15	16	16	16	12	16	16

VR-Handle Performance Evaluation Form									
Participant Information:									
Name: Ehsan Mokhtarabadi					Age: 18		Male <input checked="" type="checkbox"/> / Female <input type="checkbox"/>		
Reality					Virtual				
Position No.	Guessed Positions				Position No.	Guessed Positions			
1	1	5	1	1	1	1	1	1	2
2	6	2	2	4	2	2	2	2	2
3	3	3	3	4	3	7	4	3	3
4	12	8	4	4	4	4	4	4	4
5	9	13	1	5	5	5	5	1	1
6	6	9	2	2	6	2	10	6	2
7	4	7	12	12	7	8	8	12	3
8	12	12	16	16	8	4	4	4	12
9	5	5	5	9	9	9	5	5	5
10	7	10	10	6	10	14	13	10	14
11	7	11	15	16	11	16	11	15	15
12	16	12	12	16	12	13	12	8	4
13	13	13	9	13	13	9	10	13	9
14	14	14	14	14	14	14	14	14	14
15	15	15	15	11	15	15	15	15	15
16	12	16	15	16	16	12	11	16	16

Abstract

Today, researchers are trying to bring the virtual world as close as possible to reality. Most of the virtual reality environments are technologies for visual interaction between computers and humans. But a limited group also has the voice or touch actuators to interact with the user that help enhance a sense of presence in the real world. In this research, an attempt has been made to design and build an example of these actuators that create a better sense of the environment by applying force to the hand. In general, these tools can be divided into grounded and ungrounded categories. In this project, we present an ungrounded haptic device that can render the illusion of a weight dynamically moving on a virtual plane. The forces and torques that should be applied to the user's hands are generated by two jet propeller.

Keywords

Virtual Reality, Force-Feedback Device, Haptic



**University of Isfahan
Faculty of Engineering
Department of Mechanical Engineering**

B.Sc Thesis

**Design and manufacture of a device to create a sense of force and
torque in virtual reality environments**

**Supervisor:
Dr. Hassanpour**

**Advisor:
Dr. Karimpour**

**By:
Mohammad Hassan Mokhtarabadi**

September 2020

