

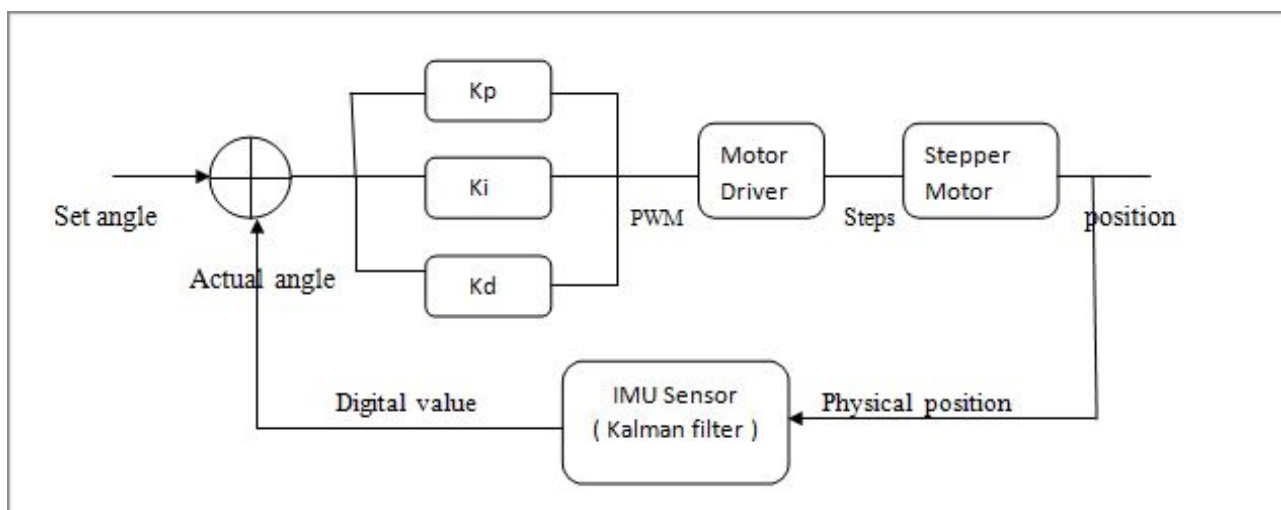
به نام خدا

طراحی مفهومی اولیه

میلااد حکیمی - صادق حایری - سحر رجبی

کارکرد:

به طور کلی سیستم از چند سنسور یک پردازنده و ۲ موتور تشکلی شده است، دیتا از دو سنسور ژيروسکوپ و شتابسنج توسط پردازنده گرفته می‌شود، به وسیله یک فیلتر مناسب داده‌های این دو سنسور به داده مناسب برای اندازه گیری زاویه‌ی انحراف ربات از سطح افق تبدیل می‌شود. زاویه‌ی لحظه‌ای ربات به ورودی الگوریتم کنترلی ما می‌رود، حال با استفاده از یکی از الگوریتم‌های مناسب برای این نوع سیستم جهت و میزان سرعت حرکت به طرفین به موتورها داده می‌شود. حال ربات می‌تواند در صورت فاصله گرفتن از حالت عمودی میزان انحراف را محاسبه کرده و با سرعت مناسب به همان سمت حرکت کند و خود را به حالت اولیه باز گرداند. (رفرنس اول)



همچنین در صورت موفقیت در اجرای فاز اول پروژه در فاز دوم کنترل ربات از راه دور اضافه می‌شود و در فاز سوم با استفاده از سنسورهای فاصله‌سنج از برخورد ربات با موانع جلوگیری خواهد شد. دقت شود که هدف اصلی پروژه تعادل ربات می‌باشد و دو فاز بعد صرفاً برای ایجاد جذابیت بیشتر پروژه انجام خواهند شد!

ورودی‌ها:

در فاز اول، سیستم به دو ورودی شتاب و سرعت زاویه‌ای برای رسیدن به یک مدل از وضعیت تعادل ربات نیاز دارد. برای این کار از سنسور MPU-۶۰۵۰ استفاده می‌شود که که ویژگی‌های مربوط به زاویه‌سنج و شتاب‌سنج آن در زیر آورده شده است:

زاویه‌سنج:

سه خروجی دیجیتال برای سرعت زاویه‌ای حول محورهای X، Y و Z، با بازه‌های $250^{\circ}+$ ، $500^{\circ}+$ ، $1000^{\circ}+$ و $2000^{\circ}+$ درجه بر ثانیه ارائه می‌دهد که ما از بازه‌ی $250^{\circ}+$ برای رسیدن به دقت بیشتر استفاده خواهیم کرد.

دارای یک فیلتر پایین‌گذر با قابلیت تنظیم دیجیتال که از آن برای حذف نویز استفاده می‌کنیم.

شتاب‌سنج:

سه خروجی دیجیتال برای شتاب برای سه محور اصلی در بازه‌های 2^+ ، 4^+ ، 8^+ و 16^+ که ما از بازه‌ی 4^+ استفاده خواهیم کرد.

همچنین در فاز دوم برای کنترل ربات از راه دور با استفاده از ماژول بلوتوث دستورات کنترلی را به برد آردوینو ارسال می‌کنیم.

همچنین در فاز دوم برای کنترل ربات از راه دور با استفاده از ماژول بلوتوث hc-04 دستورات کنترلی را به برد آردوینو ارسال می‌کنیم. (فرنس دوم)

این ماژول از پروتکل USART برای انتقال داده استفاده می‌کند.
قدرت سیگنال این ماژول -8۰dbm می‌باشد و برد بین ۱۰ تا ۳۵ متر دارد.
دارای ۲ مد master - slave است.
از baud rate های ۹۶۰۰, ۱۹۲۰۰, ۳۸۴۰۰, ۵۷۶۰۰, ۱۱۵۲۰۰, ۲۳۰۴۰۰, ۴۰۸۰۰ پشتیبانی می‌کند.

در فاز سوم پروژه برای تشخیص مانع از دو سنسور آلتراسونیک Srf-04 استفاده خواهیم کرد. (فرنس سوم)
دارای دقت اندازه‌گیری ۳ سانتی‌متر می‌باشد.
بازه‌ی قابل اندازه‌گیری بین ۳ سانتی‌متر تا ۳ متر است.
فرکانس نمونه‌برداری ۴ بار در ثانیه است.

پردازش:

ما در این پروژه با توجه به پردازش محدود و حافظه مورد نیاز کم، و عدم نیاز به پردازش موازی از یک برد آردوینو مدل UNO استفاده خواهیم کرد. (فرنس چهارم)
همچنین در فاز سوم از یک کامپیوتر شخصی برای ارسال دستورات کنترلی استفاده می‌کنیم.

شبکه و انتقال دیتا:

در این فاز تمام محاسبات و پردازش‌ها با استفاده از برد آردوینو انجام می‌شود. این برد ورودی و خروجی دیجیتال، I2C و سریال را پشتیبانی می‌کند.
همانطور که گفته شد؛ برای انتقال دستورات کنترلی از طریق ماژول بلوتوث به یک کامپیوتر شخصی متصل می‌شویم و دستورات را دریافت و سپس برای برد آردوینو با استفاده از پروتکل USART ارسال می‌کنیم.

خروجی‌ها:

خروجی سیستم تنها دو فاکتور سرعت و جهت چرخش برای هر یک از دو موتور است که در فاز اول تنها به وسیله‌ی الگوریتم‌های پیاده‌سازی شده برای حفظ تعادل ربات، و در فازهای بعدی به صورت ترکیبی از دستور حرکتی ارسال شده توسط ریموت کنترل و الگوریتم‌های گفته شده است که نهایتاً با حرکت موتورهای ربات را در حال تعادل به مکان مناسب هدایت خواهد کرد.

ملاحظات محیط فیزیکی:

مدلسازی این پروژه ممکن است در شرایط محیطی خاصی به خوبی عمل نکند. برای مثال، برای عکس‌العمل در مقابل ضربه‌های شدید لازم است که با سرعت بسیار زیاد پردازش لازم انجام شده، و عکس‌العمل توسط موتورهای صورت گیرد. خطا در اندازه‌گیری توسط سنسور، و یا نویزهای محیطی هم می‌تواند باعث بی‌ثباتی ربات شود.
از طرفی به خاطر اینکه کنترل ربات با استفاده از دو چرخ صورت می‌گیرد، تغییر اصطکاک باعث تغییر مدل خواهد شد. یک راه این مشکل می‌تواند محدود کردن پروژه به حرکت بر روی یک سطح به خصوص باشد.

حجم داده‌ی مورد تعامل:

در این پروژه برای تعیین وضعیت ربات فقط به سرعت زاویه‌ای حول محور عمود بر محور تعادل ربات و شتاب کلی حرکت صفحه‌ی ربات نیاز داریم بنابراین در هر مرحله ۳ داده‌ی ۱۶ بیتی از سنسورها گرفته می‌شود.

سرعت نمونه برداری سنسورها 1kHz است که در این صورت حجم داده ها ۴۸kb/s می باشد.

الگوریتم های مورد استفاده:

فیلترها: (رفرنس پنجم)

ماشین ما با حرکت کردن باید هنگامی که به آن نیرویی وارد میشود و از حالت عمودی منحرف میشود دوباره به حالت عمودی (که حالت ایده آل ماست) برسد. برای انجام این کار باید ابتدا درجهی انحراف را پیدا کنیم سپس سرعت موتور ها را برای حرکت ماشین به جلو یا عقب تنظیم کنیم تا ربات به حالت عمود برگردد.

برای اندازه گیری زاویه همانطور که گفته شد دو سنسور داریم: شتاب سنج و زاویه سنج شتاب سنج و زاویه سنج هردو مزایا و معایبی دارند.

برای مثال شتاب سنج میتواند نیروی گرانش بر ماشین را اندازه گیری کند که بر اساس آن میتون زاویه ی انحراف را به دست آورد ولی علاوه بر آن نیروهای دیگر را نیز اندازه میگیرد پس مقداری نویز و خطا دارد. زاویه سنج سرعت زاویه ای را اندازه میگیرد و با انتگرال گیری از آن میتوان زاویه ی ماشین را محاسبه کرد. مشکل این است که این روش محاسبه کامل نیست و انتگرال مقداری انحراف دارد. در نتیجه برای مدت کوتاهی مقدار محاسبه شده دقیق است ولی با گذشت زمان مقدار انحراف محاسبه شده از مقدار اصلی فاصله میگیرد.

این مشکلات میتوانند با ترکیب کردن این دو سنسور حل شوند و روش های مختلفی برای ترکیب کردن آنها وجود دارد.

چون برای انجام محاسبات از آردوینو استفاده میکنیم دو روشی که در ادامه ذکر میشوند برای ما مناسب ترند:

Kalman filter: که به عنوان تخمین خطی مرتبه دوم نیز از آن یاد می شود، الگوریتمی است که حالت یک سیستم پویا را با استفاده از مجموعه ای از اندازه گیری های شامل خطا در طول زمان برآورد می کند. این فیلتر معمولاً تخمین دقیق تری را نسبت به تخمین بر مبنای یک اندازه گیری واحد را بر مبنای استنباط بیزی و تخمین توزیع احتمال مشترکی از یک متغیر تصادفی در یک مقطع زمانی ارائه می کند.

این الگوریتم در دو گام اجرا می شود. در گام پیش بینی، فیلتر کالمان تخمینی از وضعیت فعلی متغیرها را در شرایط عدم قطعیت ارائه می کند. زمانی که نتیجه اندازه گیری بعدی بدست آید، تخمین قبلی با میانگین وزن دار آپدیت می شود. به این ترتیب که وزن اطلاعاتی که دارای قطعیت بیشتری هستند، بیشتر خواهد بود. الگوریتم بازگشتی می باشد و با استفاده از ورودی های جدید و حالات محاسبه شده قبلی به صورت بی درنگ اجرا می شود. در آردوینو کتابخانه ای برای پیاده سازی این روش وجود دارد.

Complementary filter: که ترکیبی از چند فیلتر است و اطلاعاتی را که از ورودی های مختلف دریافت میکند با هم ترکیب کرده و بهترین نتیجه را به عنوان خروجی میدهد. که روش استفاده از آن بسیار آسان است.

سیستم کنترلی:

برای پیاده سازی feedback control system مورد نیاز برای تصحیح وضعیت ربات، می توانیم از دو نوع رایج کنترلر ها، یعنی PID و LQR استفاده کنیم. هریک از این دو روش به اختصار در ادامه توضیح داده خواهند شد:

PID Controller: در این روش، در هر مرحله میزان خطای مقادیر فعلی، از وضعیت مورد- انتظار سیستم بدست آمده و با توجه به مقدار این خطا، سیستم با استفاده از سه قسمت تناسبی، انتگرال گیر و مشتق گیر با بررسی سیر تغییرات مقدار خطا در گذشته و پیش بینی روند تغییر آن، سعی در کاهش خطا و رساندن ربات به حالت ایده آل دارد.

LQR Controller: این کنترلر سعی در رساندن یک تابع به حالت بهینه دارد. به این صورت که پارامترهای موثر زمانی و محیطی را در هر لحظه دریافت میکند و خروجی را در جهت نزدیک کردن مقدار کنونی تابع به حالت ایده آل تولید میکند.

به نظر می رسد در مورد ربات هایی که مشابه پاندول معکوس عمل میکنند، کنترلر PID به خاطر وابستگی به خطا در لحظات مختلف، واکنش نرم تری به تغییرات نشان میدهد و با تغییرات جزئی سعی در بهبود خطا دارد، اما کنترلر LQR با توجه به ماهیت بهینه سازی تابعی که دارد، در مجموع زمان کمتری برای رسیدن به حالت تعادل پایدار مصرف میکند اما واکنش های شدیدتری دارد. از طرفی هر دو کنترلر در صورت انتخاب مناسب فیلترها میتوانند ربات را به خوبی کنترل کنند.

مقالات و مراجع مورد استفاده:

- 1- http://socialledge.com/sjsu/index.php/S14:_Androbot
- 2- <http://www.electronicaestudio.com/docs/istd016A.pdf>
- 3- <http://inside.mines.edu/~whoff/courses/EEENG383/lab/SRF04%20Technical%20Documentation.pdf>
- 4- <https://www.farnell.com/datasheets/1682209.pdf>
- 5- <https://sites.google.com/site/myimuestimationexperience/filters/complementary-filter>

تجهیزات و ابزارهای مورد نیاز:

برد آردوئینو - موتور به همراه گیربکس - درایورها - ماژول بلوتوث - فاصله سنج اولتراسونیک - بدنه ی پیلکسی - بردبرد و ...

هزینه:

برآورد

نام	تعداد	قیمت کل (تومان)
Arduino Uno Board	1	25000
Gear-Motor-120	2	11000
L298 Motor Driver	1	10000
Bluetooth Module	1	16000
MPU6050 Module	1	20000
Ultrasonic Sensor	2	10000
Total		92000

+ برای بدنه از پوسته پلکسی استفاده می‌کنیم، همچنین برای باتری از پاوربانک استفاده خواهیم کرد.