

1. hay algunas referencias duplicadas, pero por falta de tiempo no sé cuales son. si lo saben avisen :)

UNIVERSIDAD CENTROCCIDENTAL

“LISANDRO ALVARADO”

ROBOT INTELIGENTE PARA JUGAR TRES EN LÍNEA

JUAN RADA VILELA

Cabudare, 2005.

UNIVERSIDAD CENTROCCIDENTAL “LISANDRO ALVARADO”
DECANATO DE CIENCIAS Y TECNOLOGÍA
POSTGRADO DE CIENCIAS DE LA COMPUTACIÓN

ROBOT INTELIGENTE PARA JUGAR TRES EN LÍNEA

Trabajo presentado para optar al grado de
Magister Scientiarum

Por: JUAN RADA VILELA

Cabudare, 2005.

ROBOT INTELIGENTE PARA JUGAR TRES EN LÍNEA

Por: JUAN RADA VILELA

Trabajo de grado aprobado

Douglas Domínguez

Tutor

Jesús Contreras

Luis Alvarado

Cabudare, 03 de Mayo de 2005.

DEDICATORIA

A mis princesitas: Alejandra, Evilus, Nathalia y Paula.

A mi Madre y a mis Padres.

A mis abuelos: Nona y Remy, Lucía y Enrique.

A la memoria de Tío Perucho.

A Eduardo Molina y a Oswaldo “J” Hernández.

AGRADECIMIENTOS

A mi Madre por su apoyo y ayuda incondicional en todo momento.

A mis Padres por su apoyo incondicional y sus invaluables colaboraciones: Juan P. Rada con la Teoría de Juegos, Raúl E. Encinoza con el Robot.

A mis abuelos: Nona y Remy, Lucía y Enrique; por su ayuda y apoyo.

Al Ing. Douglas Domínguez por su excelente orientación y por presentarme el fascinante mundo de la teoría de juegos.

A FUNDACITE Lara por su excelente labor como fundación.

A la Universidad Fermín Toro por su sobresaliente desempeño en la formación de los estudiantes.

A todos los profesores con quienes tuve el honor de recibir clases, en especial a: Luis Alvarado, Douglas Domínguez, Edecio Freitas, Samary Paez y Jorge Rodríguez.

A mis amigos, en especial a Oswaldo “J” Hernández por su tremenda ayuda para llegar a este punto, a Eduardo Molina por su constante apoyo y gran interés en el Equilibrio Nash!

A las chicas que me acompañaron durante esta aventura de 5 años (y un poquito más!), cronológicamente: Olga Pardi, Mardia Ramos, Florangel Rivas, Maritza Pasquarelli y Estefania Alfaro.

Y por supuesto, a todas las personas que colaboraron de una forma u otra en el desarrollo de este trabajo de grado.

Muchas Gracias.

Juan C. Rada.

ÍNDICE GENERAL

	PÁG.
DEDICATORIA	iii
AGRADECIMIENTOS	iv
ÍNDICE GENERAL	v
ÍNDICE DE CUADROS	viii
ÍNDICE DE GRÁFICOS	ix
ÍNDICE DE ILUSTRACIONES	x
ÍNDICE DE PENDIENTES	xi
ÍNDICE DE NOTAS	xii
RESUMEN	xiii
ABSTRACT	xiv
INTRODUCCIÓN	1
CAPÍTULO	
I EL PROBLEMA	3
Planteamiento del Problema	3
Objetivos	5
Objetivos Generales	5
Objetivos Específicos	5
Justificación e Importancia	5
Alcances y Limitaciones	7
Alcances	7
Limitaciones	7
II MARCO TEÓRICO	8
Antecedentes de la Investigación	8
Bases Teóricas	10
Mecánica	10

	Leyes de Newton	10
	Equilibrio	10
	Modelo y análisis matemático para el diseño estructural del brazo	11
	Engranajes	14
	Electrónica	14
	Microcontrolador 8951	14
	Puerto P0	15
	Puerto P1	15
	Puerto P2	15
	Puerto P3	16
	Otros pines	16
	Memoria RAM Interna	17
	Comunicación Serial	18
	Otros Componentes	20
	Latch	20
	Motores Paso a Paso	20
	Teoría de Juegos	22
	Bases Legales	25
	Sistema de Hipótesis	25
	Operacionalización de las Variables	25
III	MARCO METODOLÓGICO	31
	Tipo de Investigación	31
	Población y Muestra	31
	Diseño de la Investigación o Procedimiento	32
	Técnicas e Instrumentos de Recolección de Datos	32
	Técnicas de Procesamiento y Análisis de los Datos	32
IV	RESULTADOS	33
V	CONCLUSIONES	34

VI	RECOMENDACIONES	35
	REFERENCIAS BIBLIOGRÁFICAS	36
	ANEXOS	36
A	Curriculum Vitae	37
B	Instrumento de Recolección de Datos	38

ÍNDICE DE CUADROS

	PÁG.
1. Secuencias Paso a Paso	22
2. Título del cuadro	22
3. Titulo del cuadro	24
4. Título del cuadro	31

ÍNDICE DE GRÁFICOS

	PÁG.
1. Esto debería ser un gráfico (e.g. barras)	29
2. Esto debería ser un gráfico (e.g. de líneas)	30

ÍNDICE DE ILUSTRACIONES

	PÁG.
1. Modelo Matemático del Brazo Mecánico	12
2. Brazo Mecánico con Motores Paso a Paso	26
3. Diagrama de Cuerpo Libre – Eslabón B	27
4. Diagrama de Cuerpo Libre – Eslabón A	28
5. Piñón – Engranaje	28
6. Microcontrolador MCS-51	29
7. Diagrama – 74LS175	29

ÍNDICE DE PENDIENTES

	PÁG.
1. Revisar redaccion	3

ÍNDICE DE NOTAS

	PÁG.
1. hay algunas referencias duplicadas, pero por falta de tiempo no sé cuales son. si lo saben avisen :)	

UNIVERSIDAD CENTROCCIDENTAL “LISANDRO ALVARADO”

DECANATO DE CIENCIAS Y TECNOLOGÍA

POSTGRADO DE CIENCIAS DE LA COMPUTACIÓN

ROBOT INTELIGENTE PARA JUGAR TRES EN LÍNEA

Autor: Juan Rada Vilela

Tutor: Douglas Domínguez

RESUMEN

Este trabajo de grado utiliza la Teoría de Juegos para crear Inteligencia Artificial y así lograr que un robot tome óptimas decisiones en situaciones finitas, no cooperativas y de información perfecta. El campo de aplicación está enfocado hacia el juego Tres en Línea, donde el robot participa como jugador y utiliza la estrategia definida por el Equilibrio Nash para tomar óptimas decisiones en el juego. Aunque Tres en Línea es sólo un ejemplo de situación, se explica detalladamente el procedimiento para conseguir un Equilibrio Nash en cualquier situación finita de información perfecta; el cual determina una n -tupla de estrategias tal que, cuando los n jugadores utilizan su estrategia respectiva S_n , todos obtendrán el mejor beneficio posible. En otras palabras, ningún jugador puede obtener mejor beneficio que el obtenido con la estrategia S_n . Las estrategias que conforman el Equilibrio Nash para este juego son no perdedoras, es decir, el robot jamás perderá un juego. Ahora, para la construcción del robot se utiliza el equilibrio estático de cuerpos rígidos; lo que permite siempre un equilibrio en la estructura y así los actuadores ejecutan un trabajo mínimo. El robot consiste en tres articulaciones y cuatro grados de libertad para posicionar la muñeca en cualquier punto del campo de juego. Los grados de libertad son transversal vertical, transversal radial, transversal rotacional y torsión. Su efector final es un marcador que a través de un sistema mecánico es capaz de dibujar el símbolo O (correspondiente a la jugada del robot).

Palabras Clave: Robótica, Inteligencia Artificial, Teoría de Juegos, Equilibrio Nash, Algoritmo de Zermelo, Microcontrolador 8951.

UNIVERSIDAD CENTROCCIDENTAL “LISANDRO ALVARADO”

DECANATO DE CIENCIAS Y TECNOLOGÍA

POSTGRADO DE CIENCIAS DE LA COMPUTACIÓN

INTELLIGENT ROBOT FOR PLAYING TIC-TAC-TOE

Author: Juan Rada Vilela

Supervisor: Douglas Domínguez

ABSTRACT

This is the abstract

Keywords: These are the keywords.

INTRODUCCIÓN

El desarrollo de un sector de la robótica está dirigido hacia la creación de robots cada vez más semejantes al ser humano, tanto en términos corporales como en términos de inteligencia. Corporalmente, implica un diseño mecánico para que la estructura siempre permanezca en equilibrio. Este equilibrio será estable si la estructura mantiene el equilibrio para cualquier sistema de cargas aplicadas, ó inestable si se mantiene sólo para un conjunto particular.

En términos de inteligencia es necesario modelar el raciocinio que tienen los seres vivos para reaccionar ante cualquier situación, lo cual siempre conlleva hacia una toma de decisiones y, la calidad de éstas, dependerá netamente de la herramienta utilizada para crear Inteligencia Artificial.

Existen diversas herramientas para tomar decisiones a nivel de inteligencia artificial, por ejemplo: las redes neuronales emplean algoritmos de aprendizaje iterativo, los sistemas inteligentes utilizan inferencias lógicas, los algoritmos genéticos aplican la teoría de la evolución darwiniana, entre otros.

La teoría de juegos también permite tomar decisiones. Aunque su principal campo de acción se ubica en la economía, ésta describe matemáticamente cualquier situación en la que dos o más individuos toman decisiones en búsqueda de un resultado que genere bienestar. Dicho bienestar puede ser individual o grupal, y dependerá netamente de la estrategia que utilice cada participante.

Un concepto fundamental en la teoría de juegos es el Equilibrio Nash. Este define una estrategia para cada individuo que participa en el “juego” y, cuando éstas son utilizadas, todos los participantes obtendrán el mejor bienestar posible. En otras palabras, se habrán tomado las decisiones óptimas.

Este trabajo de grado se basa en la construcción de un robot inteligente capaz de tomar decisiones óptimas en situaciones finitas de competencia que, a modo de ejemplo, están enfocadas al juego Tres en Línea.

Este trabajo de grado se estructura como sigue: el Capítulo I presenta la descripción

del problema, justificación de la investigación, objetivos, alcances y limitaciones de la misma. El Capítulo II presenta los antecedentes de la investigación junto a las bases teóricas del trabajo, las cuales son divididas en secciones: Mecánica, Electrónica y Teoría de Juegos.

El Capítulo III describe el marco metodológico donde se encuentra la naturaleza del trabajo y las tres fases para el desarrollo de la investigación: Diagnóstico, Factibilidad y Diseño. Además, explica brevemente el diseño del sistema.

El Capítulo IV presenta los resultados obtenidos y se explica detalladamente los módulos que integran al sistema; en el Capítulo V se presentan las conclusiones y recomendaciones.

CAPÍTULO I

EL PROBLEMA

Planteamiento del Problema

T. Kanade (2004), en entrevista con Popular Mechanics en Español opina que “la mejor definición de robot es aquella que habla de un sistema que registra el mundo real exterior para interpretarlo y tomar decisiones útiles e inteligentes que tengan un impacto en el mundo real”. A lo que Popular Mechanics en Español (2004) considera que “Es, básicamente, a lo que nos dedicamos los humanos. A decidir.” **1. Revisar redaccion**

Muchas situaciones de la vida cotidiana se basa en la toma de decisiones: una pareja decide si ir al cine o a un concierto, una empresa decide si entrar ó no a un mercado. Estas son decisiones que se toman de acuerdo a intereses basándose en instintos, experiencias o incluso en sentido común. En nada difieren de los juegos: en ajedrez, dos personas deciden sobre cómo jugar para lograr un jaque mate; en Tres en Línea dos personas deciden sobre donde colocar los símbolos X y O para lograr tres en línea.

La toma de decisiones en robótica requiere de inteligencia artificial, la cual posee diversas herramientas para su modelaje. Tales herramientas son: las redes neuronales y sus algoritmos de aprendizaje iterativo; los sistemas inteligentes y sus inferencias lógicas sobre la base de conocimientos; los algoritmos genéticos y su evolución por la teoría darwiniana; entre otras.

Aunque con estas herramientas cualquier robot puede llegar a tomar las mejores decisiones, el problema radica en extensas situaciones finitas de información perfecta con interdependencia estratégica (e.g. Tres en Línea) donde no es fácil demostrar que el resultado obtenido es el óptimo.

Por ejemplo, se construye un robot capaz de tomar decisiones en el juego Tres

en Línea y su inteligencia artificial está definida por un sistema inteligente. De esta forma, el robot jugará utilizando la experiencia previa y con el tiempo jugará mejor. El resultado es empírico, aunque tenga mucha experiencia surge la pregunta: se han tomado las mejores decisiones?

Utilizando el mismo ejemplo, pero en lugar de utilizar un sistema inteligente su inteligencia artificial está definida por una red neuronal. Sin entrar en detalles respecto a su diseño e independientemente si son de aprendizaje supervisado o no, el robot aprenderá a tomar decisiones. Si toma una decisión errónea, se modifican los pesos de las neuronas y, de esta forma, convergerá hacia una buena toma de decisiones; pero aunque el error sea mínimo, surge nuevamente la pregunta: se han tomado las mejores decisiones?

La misma pregunta surge utilizando algoritmos genéticos.

Una forma de responder a esta pregunta es analizando todas las posibles combinaciones estratégicas del juego para determinar si en realidad se tomaron las mejores decisiones, lo cual es poco práctico debido al tamaño del árbol de juego.

Asimismo, esta inquietud no sólo tiene lugar en el juego Tres en Línea, existirá siempre que se utilicen estas herramientas para la toma de decisiones en situaciones similares.

Por el problema expuesto anteriormente y en búsqueda de una herramienta que permita tomar decisiones óptimas y justificadas, se plantea el uso de la teoría de juegos para crear Inteligencia Artificial y así solventar (de la mejor manera posible) cualquier problema de decisión en estas situaciones. Luego, para su aplicación en el campo de la robótica, se construirá un robot que utilice esta herramienta para tomar decisiones en el juego Tres en Línea.

Objetivos

Objetivos Generales

Construir un robot inteligente capaz de jugar Tres en Línea. Por el problema expuesto anteriormente y en búsqueda de una herramienta que permita tomar decisiones óptimas y justificadas, se plantea el uso de la teoría de juegos para crear Inteligencia Artificial y así solventar (de la mejor manera posible) cualquier problema de decisión en estas situaciones. Luego, para su aplicación en el campo de la robótica, se construirá un robot que utilice esta herramienta para tomar decisiones en el juego Tres en Línea.

Objetivos Específicos

1. Diagnosticar la utilidad de la teoría de juegos en problemas para la toma de decisiones en el campo de la robótica.
2. Estudiar la factibilidad técnica, económica y operativa del sistema propuesto.
3. Construir un robot inteligente capaz de jugar Tres en Línea.

Justificación e Importancia

Es claro que la toma de decisiones es importante para cualquier ser vivo, pero más importante aún es tomar una óptima decisión. Este resultado se logra utilizando la teoría de juegos que, con fundamentos matemáticos, demuestra que las decisiones tomadas son óptimas.

Con este trabajo de investigación se pretende introducir la teoría de juegos en el campo de la robótica con la construcción de un robot inteligente capaz de tomar decisiones óptimas en el juego Tres en Línea. Aunque este juego es sólo un ejemplo de situación donde se toman decisiones, se explicará detalladamente la teoría para su

aplicación a cualquier situación finita de información perfecta.

Vale destacar que este trabajo es de carácter innovador dentro de la Universidad Fermín Toro, ya que hasta la fecha (Abril de 2005) no se han realizado trabajos de grado donde se utiliza la teoría de juegos como herramienta para la toma de decisiones a nivel de inteligencia artificial.

Una opinión sobre el uso de la teoría de juegos en esta área es presentada por J. Roach (2004): “Los objetivos de estas dos áreas son tan similares que es solamente natural, para la teoría de juegos e inteligencia artificial, crear una sinergia para formular novedosos enfoques en la solución de diversos problemas”.

Por otro lado, el robot construido será adquirido por la Fundación para la Ciencia y Tecnología del Estado Lara (FUNDACITE Lara) para fomentar investigaciones en el campo de la robótica. Por este motivo, se creará un robot flexible para que pueda ser utilizado en una mayor gama de aplicaciones.

Este trabajo de grado está orientado al beneficio de la comunidad por las siguientes razones

1. Se explica detalladamente una herramienta que permite tomar decisiones óptimas, la cual podrá ser utilizada para desarrollar desde sistemas hasta robots inteligentes que no sólo tomarán buenas decisiones sino que tomarán las mejores.
2. Se explica el diseño del robot desde un punto de vista mecánico, lo cual permitirá servir de apoyo bibliográfico para las consideraciones que se deben tomar en cuenta al momento de la construcción de cualquier robot.
3. El código del software desarrollado en alto y bajo nivel están comentados y bajo el esquema de software libre, lo cual dará una mejor visión sobre cómo crear Inteligencia Artificial utilizando la teoría de juegos además de cómo controlar un robot; de igual manera se podrá disfrutar de los beneficios que ofrece el software libre.

Por lo antes mencionado, este trabajo de grado se ubica en el Polo 2 (Hombre, Ciudad y Territorio) que al intersectarlo con el eje correspondiente, diseño y mantenimiento de sistemas inteligentes, ubica la línea de investigación Inteligencia Artificial y

Robótica.

Alcances y Limitaciones

Alcances

1. Construir un brazo robótico capaz de tomar decisiones óptimas en el juego Tres en Línea.
2. Construir un efector final capaz de dibujar la jugada del robot en el campo de juego.
3. Conseguir una estrategia óptima para cada jugador.
4. Interpretar el campo de juego por visión artificial.
5. Facilitar la interacción humano – robot mediante un tablero de juego.
6. Integrar el software y el hardware mediante un protocolo de comunicación bidireccional.

Limitaciones

1. El robot sólo dibujará su jugada respectiva utilizando el símbolo O.
2. Las jugadas realizadas en el campo de juego deberán realizarse dentro de los márgenes establecidos y de un tamaño no menor a 4 cm².
3. Los algoritmos de visión artificial están diseñados para reconocer los símbolos O, cualquier otro símbolo será interpretado como una jugada del adversario.
4. La posición de la cámara digital no podrá alterarse luego de reiniciar el microcontrolador.
5. La iluminación del escenario no podrá alterarse.
6. El campo de juego deberá permanecer en la posición establecida.
7. El robot no evita obstáculos.

CAPÍTULO II

MARCO TEÓRICO

Antecedentes de la Investigación

G. Hochmuth (2003), publicó un artículo en la Universidad de Stanford donde hace uso de algoritmos genéticos para desarrollar un programa que evolucione hacia una estrategia perfecta en el juego Tres en Línea. El autor redujo todos los posibles juegos a sólo 827 aprovechando la simetría del campo de juego y tomando sólo en cuenta las posibles jugadas que lleven por lo menos hacia un empate; aclara que este número de posibles jugadas fue encontrado empíricamente.

Este método le permitió desarrollar una estrategia perfecta después de 373 generaciones donde el programa no perdió ni un solo juego. Se debe resaltar que el autor consideró como perfecta a una estrategia en la que el programa no pierde. Este artículo lleva por nombre *On the Genetic Evolution of a Perfect Tic-Tac-Toe Strategy*.

B. Blum (2002), realizó un proyecto en la Universidad de Virginia donde entrena a una red neuronal para jugar Tres en Línea. El autor utilizó dos redes neuronales: una para aprender las reglas del juego y otra para aprender una estrategia competente en el juego. La primera es el jugador en sí, el cual aprende a ejecutar jugadas legales; la segunda le indica a la primera lo que sería bueno jugar.

El jugador consiste en una red neuronal backpropagation con nueve neuronas de entrada, nueve de salida y un número variable en la capa oculta, cuando realiza una jugada ilegal se debilitan los pesos y cuando es legal se incrementan. La segunda red neuronal tiene el mismo diseño y se entrena para aprender una estrategia. Para entrenarlas, creó un campeonato entre 10 jugadores y elige el mejor para ser probado compitiendo con el humano o con otros sistemas inteligentes desarrollados también por el autor (uno que bloquea las jugadas ganadoras del adversario; y el otro que primero

busca bloquear, luego ganar y si no, juega aleatoriamente).

Finalmente, el autor aclara que no pudo desarrollar un jugador que jugará perfectamente Tres en Línea pero logró que realizaran un 100 % de jugadas legales. Este proyecto tiene por nombre Neural Net Training for Tic Tac Toe.

La empresa ADIT Software desarrolló un software inteligente para jugar Tres en Línea, donde utiliza 200 redes neuronales con nueve neuronas de entrada, nueve de salida y diez ocultas. Cada una realizó 100 juegos (con otra asignada automáticamente) de los cuales en 50 comenzaba. Luego, utilizó algoritmos genéticos para escoger la mejor. Según la empresa: este sistema era fuerte atacando pero a veces fallaba defendiendo; ganaba un número razonable de veces pero también perdía mucho incluso después de 1000 generaciones. Este software se llama MENACE y el sistema siempre juega de primero. No especifica fecha de creación.

En 1999 se desarrolló un robot para jugar Tres en Línea, el cual fue llevado al festival MindFest de inteligencia artificial que se desarrolló en el Instituto Tecnológico de Massachussets (MIT) de Estados Unidos en el mismo año. Este proyecto fue realizado por M. Beri, G. Ferrari y M. Ferrari, y consiste en un robot construido con materiales de Lego® Mindstorms. El robot coloca la pieza correspondiente a su jugada en el campo de juego, detecta cuando es su turno además si hay ganador o no. Aunque el robot no puede aprender, este realiza su jugada basándose en algunas estrategias limitadas cuyos autores decidieron no publicar. El proyecto responde al nombre de A Lego®Mindstorms Tic-Tac-Toe Player.

Bases Teóricas

Mecánica

Leyes de Newton

En la segunda mitad del siglo XVII, Sir Isaac Newton formuló tres leyes fundamentales en las que se basa la ciencia mecánica. Estas son enunciadas por F. Beer et al (1990) como sigue:

1. PRIMERA LEY. Si la fuerza resultante que actúa sobre una partícula es cero, la partícula permanecerá en reposo (si originalmente estaba en reposo) o se moverá con velocidad constante en una línea recta (si originalmente estaba en movimiento).
2. SEGUNDA LEY. Si la fuerza resultante que actúa sobre una partícula no es cero, la partícula tendrá una aceleración proporcional a la magnitud resultante y en la dirección de ésta.
3. TERCERA LEY. Las fuerzas de acción y reacción de cuerpos en contacto tienen la misma magnitud, la misma línea de acción y sentidos opuestos.

Equilibrio

Se dice que un cuerpo está en equilibrio cuando la resultante de las cargas externas y reacciones que actúan sobre el cuerpo es igual a cero; esto es, la suma de todas las fuerzas (o de sus componentes) actuando en cualquier dirección posible debe ser igual a cero. Además, también debe ser cero la suma de los momentos de todas las fuerzas respecto a cualquier eje.

Para que una estructura o parte de ella esté en equilibrio bajo la acción de un sistema de cargas, debe satisfacer las seis ecuaciones de equilibrio de la Estática. Utilizando los ejes cartesianos x , y y z , las ecuaciones pueden escribirse como sigue

$$\sum F_x = 0; \sum F_y = 0; \sum F_z = 0; \quad (1)$$

Estas ecuaciones no pueden demostrarse algebraicamente; sólo describen la afirmación de Sir Isaac Newton: para cada acción sobre un cuerpo en reposo, hay una reacción igual y opuesta.

Una estructura puede encontrarse en equilibrio estable o inestable. Según J. Kinney (1981), una estructura en equilibrio estable indica que ésta podrá soportar cualquier sistema concebible de cargas aplicadas. Es importante destacar que aunque una estructura sea estable para soportar uno o varios conjuntos de cargas, si no es estable para cualquier conjunto de cargas entonces se dice que se encuentra en equilibrio inestable. Frecuentemente, una estructura en equilibrio inestable será estable bajo un sistema particular de cargas aplicadas.

Modelo y análisis matemático para el diseño estructural del brazo

Según J. Kinney (1981), “una estructura articulada está compuesta de eslabones o barras, que se suponen conectadas por pasadores carentes de fricción en las juntas”. Con esta descripción, es claro que un brazo mecánico es una estructura articulada. El modelo matemático del brazo articulado se describe en la figura uno.

Se puede notar que: el pasador a permite la rotación de la estructura en el eje y; el pasador b permite desplazamientos de la estructura perpendiculares al eje z; y, el pasador c permite girar el eslabón B perpendicularmente al eje z.

$$\sum F_x = 0; \sum F_y = 0; \sum F_z = 0; \quad (2)$$

En la figura 1, a pesar de que el punto h se desplaza en tres dimensiones, los eslabones A y B realizan sus desplazamientos en un plano cualquiera denotado por

yx. Asimismo, las cargas que actúan sobre la estructura son pesos cuyas líneas de acción también se encuentran en el plano yx.

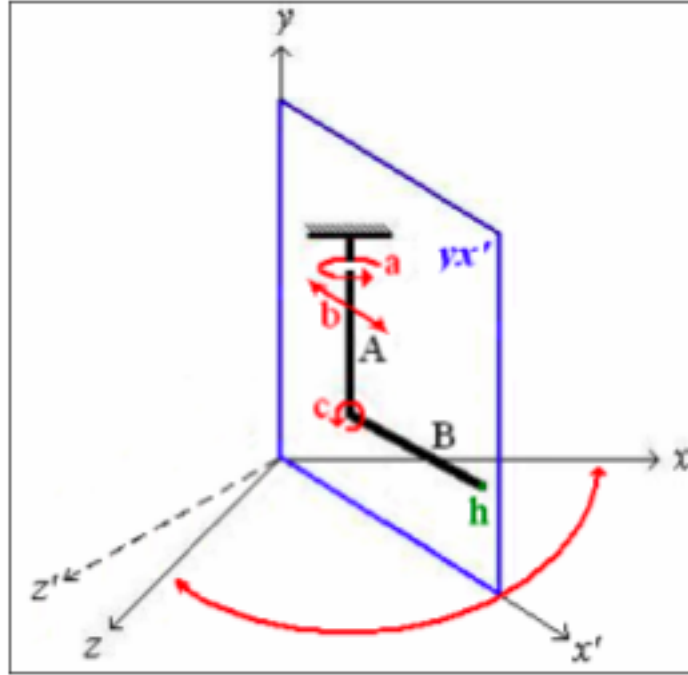


Ilustración 1. Modelo Matemático del Brazo Mecánico

Dado que las cargas, eslabones y desplazamientos relativos, se encuentran en el mismo plano (cargas coplanares), es posible analizar la estructura en dos dimensiones. En este caso, J. McCormac et al. (1994) establece que

“ la suma de las fuerzas en las direcciones x y y, así como la suma de los momentos respecto a un eje perpendicular al plano debe ser cero. Definiendo así las siguientes ecuaciones de equilibrio para esta estructura ”

$$\sum F_x = 0; \sum F_y = 0; \sum M_z = 0 \quad (3)$$

El brazo mecánico es una estructura en equilibrio inestable, y utilizando motores eléctricos paso a paso se logra anular la fuerza de inercia. Para lograr que

la estructura se desplace como indica el modelo matemático (figura 1), se puede utilizar la siguiente configuración

Esta estructura se puede descomponer y así analizar cada eslabón por separado, esto es porque tanto el eslabón A como el B son cuerpos en equilibrio. Así, el diagrama de cuerpo libre para el eslabón B es

donde P_1 es el peso del eslabón aplicado en su centro de gravedad, L_1 es la distancia de P_1 al punto de apoyo R_B , P_2 es el peso de la carga, L_2 es la distancia de P_2 al punto de apoyo R_B , P_3 es el peso de la contrapesa, y L_3 es la distancia de P_3 al punto de apoyo R_B .

Para que el cuerpo se encuentre en equilibrio se debe cumplir que

$$\sum F_y : P_1 + P_2 + P_3 - R_B = 0 \quad (4)$$

$$\sum M_{R_B} : P_1 \times L_1 + P_2 \times L_2 - P_3 \times L_3 = 0 \quad (5)$$

Ahora, el diagrama de cuerpo libre para el eslabón A es como sigue:

donde P_4 es el peso del motor M_1 , L_4 es la distancia de P_4 al punto de apoyo R_A , P_5 es el peso del motor M_2 , L_5 es la distancia de P_5 al punto de apoyo R_A , P_6 es el peso del eslabón A aplicado en su centro de gravedad, L_6 es la distancia de P_6 al punto de apoyo R_A . P_7 es el peso del eslabón B, y L_7 es la distancia de P_7 al punto de apoyo R_A .

Para que este cuerpo esté en equilibrio se debe cumplir entonces que

$$\sum F_y : P_4 + P_5 + P_6 + P_7 - R_A = 0 \quad (6)$$

$$\sum M_{R_A} : -P_4 \times (L_4 \sin \theta) - P_5 \times (L_5 \sin \theta) + P_6 \times (L_6 \sin \theta) + P_7 \times (L_7 \sin \theta) = 0 \quad (7)$$

Observando la ecuación de momento en el punto de apoyo RA, se observa que $\sin \theta$ es común en todos los elementos. Esto significa que el cuerpo se encontrará en equilibrio para cualquier ángulo.

Ya establecidas las ecuaciones de equilibrio para esta estructura, el diseño puede ser llevado a cabo.

Engranajes

Los engranajes son utilizados para transmitir potencia de un eje a otro. Véase la siguiente figura.

La relación de engranajes viene dada por $n = N_P/N_E$ donde N_P es el número de dientes del piñón y N_E es el número de dientes del engranaje.

La velocidad de salida (Engranaje) respecto a la de entrada (Piñón) está determinada por $\omega_E = n\omega_P$. Y, el par motor de salida es determinado por $T_E = T_P/n$.

Electrónica

Antes de explicar lo referente al microcontrolador 8951 es necesario aclarar que no se trata de un manual para su uso, se trata de la teoría en que se basó este trabajo de grado incluyendo algunos detalles importantes basados en previas experiencias del autor.

Microcontrolador 8951

El microcontrolador 8951 pertenece a la familia de microcontroladores MCS-51. Según M. Martos et al., esta familia consiste en cuatro miembros compatibles pin a pin que son el 8031, 8051, 8751 y 8951; los cuales se diferencian en la memoria de programa ROM.

Según A. Vega (1999), el microcontrolador 8951 tiene las siguientes caracte-

rísticas: CPU de 8 bits como parte central, 32 líneas bidireccionales de entrada y salida (4 puertos), 128 bytes de memoria RAM, 2 Contadores / Temporizadores de 16 bits, 5 estructuras de interrupción con dos niveles de prioridad, 64 KBytes de espacio para programa y 64 Kbytes de espacio para datos. La estructura física se observa a continuación.

Puerto P0

Este puerto se puede utilizar para convertirlo en bus de datos o direcciones cuando el pin EA es conectado a 0v (0 lógico), este se encarga de la parte baja del bus cuando se utiliza memoria externa. Es importante señalar que este puerto no es adecuado para emitir señales de 5v (1 lógico) ya que las salidas del mismo están en alta impedancia.

Puerto P1

Es un puerto bidireccional de 8 bits cuya etapa de salida puede manejar corrientes equivalentes a cuatro cargas TTL LS. Este puerto no tiene funciones secundarias.

Puerto P2

Al igual que el puerto P1, es un puerto bidireccional de 8 bits y puede manejar corrientes equivalentes a cuatro cargas TTL LS. Este puerto tiene como función secundaria la de suministrar la parte alta de la dirección cuando es necesario recurrir a memoria externa.

Puerto P3

Igual que los puertos anteriores, pero la función secundaria de los pines son: P3.0/RxD, entrada para la comunicación serial; P3.1/TxD, salida para la comunicación serial; P3.2/INT0, entrada para la interrupción externa 0; P3.3/INT1, entrada para la interrupción externa 1; P3.4/T0 y P3.5/T1, entradas respectivas de conteo para los Timer 0 y 1; P3.6/WR y P3.7/RD, salidas respectivas que indican la escritura o lectura de la memoria externa.

Los pines pueden ser configurados independientemente, es decir, si no se utilizarán las funciones secundarias de algunos pines, estos pueden ser utilizados libremente por el programador para otros fines.

Otros pines

Los pines Vcc y Vss son para la alimentación del microcontrolador, deben conectarse respectivamente a +5v y a 0v. Los pines correspondientes a XTAL1 y XTAL2 son para conectar el cristal oscilador que determinará la frecuencia con que funcionará el reloj del microcontrolador, generalmente se emplean cristales osciladores de 11.059 MHz o 12 MHz. Es importante tomar en cuenta la frecuencia de oscilación tanto para las rutinas de retardo como para la comunicación serial y temporizadores, ya que los ciclos de máquina dependen de esta frecuencia.

El pin RST sirve para reiniciar el microcontrolador cuando se le envía una señal alta (1 lógico) durante dos ciclos de máquina, el proceso de reinicio causa una interrupción en la dirección 00h.

El pin Vpp/EA sirve para acceder a memoria externa. Cuando se utiliza memoria externa para almacenar las instrucciones a ejecutar por el microcontrolador, se debe conectar a 0 lógico. Cuando esto ocurre, los puertos P0 y P2 desempeñan las funciones secundarias. Ahora, en caso de que las instrucciones

hayan sido almacenadas en la memoria interna del microcontrolador, deberá conectarse a 1 lógico.

El pin PROG/ALE fija el byte bajo de la dirección durante el acceso a memoria externa y, cuando no se utiliza memoria externa, emite un rango constante de 1/6 de la frecuencia del oscilador que, según A. Vega (1999), puede ser utilizado para cronometrar.

Finalmente, el pin PSEN habilita la lectura para memoria de programas externos.

Memoria RAM Interna

El microcontrolador 8951 posee cuatro bancos de ocho registros cada uno, donde cada registro está conformado por 8 bits. También posee 16 bytes de memoria que son direccionable bit a bit lo cual permite utilizar banderas adicionales, y 80 bytes más para uso general.

La memoria del microcontrolador 8951 se segmenta de la siguiente manera Registros: desde la dirección 00h hasta la dirección 07h se ubican los registros R0 a R7 del banco 0; de 08h a 0Fh, los registros R0 a R7 del banco 1; de 10h a 17h los registros R0 a R7 del banco 2; y de 18h a 1Fh los registros del banco 3. Suman un total de 32 bytes destinados a registros. A todos se pueden acceder pero es necesario modificar los bits RS0 y RS1 del registro PSW; por defecto los valores son 0 y 0, lo cuál indica que se está trabajando en el banco 0. Para trabajar con el banco 1 se debe establecer RS0=0 y RS1=1; para el banco 2, RS0=1 y RS1=0; y para el banco 3, RS0=1 y RS1=1.

Direccionable bit a bit: desde la dirección 20h hasta la dirección 2Fh son celdas a las que se puede acceder bit a bit (e.g. 20h.0, 2Fh.7); la cantidad total es de 16 bytes.

Memoria de uso general: desde la dirección 30h hasta la dirección 7Fh, lo cual suma 80 bytes para uso general.

El área de memoria a partir de la dirección 80h hasta la dirección FFh, pertenece los registros de interrupciones, configuración de los temporizadores, acumulador, etc.

Comunicación Serial

La comunicación serial se implementa utilizando cualquiera de los temporizadores para generar la velocidad de transmisión/recepción de datos. Tanto para enviar como para recibir datos se debe hacer uso del registro SBUF. Este registro almacena 1 byte que será recibido o transmitido. Un aspecto importante, es que tanto el microcontrolador como el PC deben ser configurados a la misma velocidad de transmisión/recepción.

Un computador personal permite conexiones, a través del puerto serial, a las siguientes velocidades: 110, 300, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200 bps, utilizando Microsoft HyperTerminal; por lo que al momento de configurar la velocidad de transmisión/recepción en el microcontrolador, se debe aplicar la siguiente fórmula (A. Vega (1999))

$$BaudRate = 2^{SMOD} \times Freq/32 \times (256 - THx) \quad (8)$$

donde Baud Rate es la velocidad en bps; Freq es la frecuencia de oscilación (determinada por el cristal); SMOD determina el divisor de frecuencia (0 para 1/32 ó 1 para 1/64); y THx, es el valor de recarga para el temporizador 0 ó 1. Para aplicar la fórmula: se conoce la velocidad de transmisión/recepción de datos (e.g. 9600 bps), se conoce la frecuencia de oscilación (e.g. 11.059 MHz) y el bit SMOD es determinado a nivel de programación (e.g. 0). Para este caso, es necesario despejar la variable THx a fin de determinar la velocidad de transmisión/recepción de datos.

Es importante señalar que el resultado de THx debe ser aproximado a un entero menor que 256 debido que para asignar el valor de recarga es necesario un entero no mayor a 8 bits (capacidad de THx). En caso de usar del temporizador 1 en modo 2 y para las especificaciones expuestas en el párrafo anterior, al despejar THx el resultado es 253.00005. Esto indica que para lograr configurar la comunicación serial a una velocidad de transmisión/recepción de 9600 bps, utilizando un cristal oscilador de 11.059 MHz y SMOD = 0, el valor con que debe ser cargado TH1 es FDh (253 en decimal).

La comunicación serial, con las especificaciones expuestas, funciona sin contratiempos. Esto se debe a que 253.00005 es muy aproximado a 253, el problema que puede surgir es al momento de utilizar una velocidad distinta a 9600 bps o al utilizar un cristal oscilador con una frecuencia distinta a 11.059 MHz. Para esto, es necesario volver a aplicar la fórmula con las nuevas especificaciones y, en caso de obtener un valor (THx) no muy aproximado a un entero (e.g. 253.5) se debe probar con diferentes configuraciones, ya que este valor es determinante para lograr la velocidad de transmisión/recepción.

Otro ejemplo, si se desea trabajar con un cristal oscilador de 12 MHz es necesario bien modificar el modo de operación del temporizador o la velocidad de transmisión/recepción, ya que el valor obtenido para TH1 (en caso de utilizar el temporizador 1) no es muy exacto. Esto se puede resolver utilizando una velocidad de 1200 bps, SMOD = 0, temporizador 1 en modo 2; ya que al momento de aplicar la fórmula, el valor de TH1 es $229.95 \approx 230$ (E6 en hexadecimal).

Por último, pero no menos importante, para la comunicación serial con una computadora es necesario utilizar el circuito integrado MAX232 (en caso de usar la norma RS-232) ya que los niveles lógicos del microcontrolador son TTL y la computadora utiliza $0 = -12v$ y $1 = 12v$. Este circuito integrado funciona como interfaz: cuando recibe datos del PC estos son convertidos a los niveles lógicos del microcontrolador y, cuando se envían datos desde el 8951, estos son convertidos a niveles lógicos del computador.

Otros Componentes

Latch

Los latch son utilizados para funcionar como memoria externa volátil, aunque permiten almacenar poca información (el 74LS175 almacena 4 bits y sus negados) estos pueden ser utilizados para expandir los puertos del microcontrolador. Particularmente, el 74LS175 consta de cuatro entradas, ocho salidas (las cuatro entradas y sus negados), un pin para limpiar la memoria (CLR) y otro para determinar cuando el latch debe almacenar la información.

El diagrama del latch 74LS175 se puede observar a continuación

Motores Paso a Paso

La característica más importante de los motores paso a paso es que permiten un movimiento controlado, lo cual permite ajustar la velocidad a fin de evitar fuerzas de inercia. Entre otras características se puede mencionar que son precisos, fuertes y de fácil control (respecto a los servomotores).

Los motores paso a paso pueden ser bipolares o unipolares. Los bipolares requieren el uso de un puente H para su manejo, ya que funcionan con voltajes positivos y voltajes negativos (e.g. +12v, -12v). Los unipolares no necesitan de puente H, ya que trabajan con voltaje positivo y tierra (e.g. +5v y 0v). Estos dos tipos de motores pueden diferenciarse físicamente por la cantidad de cables: los bipolares poseen cuatro cables y los unipolares poseen cinco o seis cables.

En este trabajo de grado es de interés los motores unipolares, por lo cual se explican a continuación.

Los motores unipolares funcionan mediante la activación de bobinas, las cuales al ser activadas generan un campo magnético que permiten hacer girar al piñón cierto ángulo. Este ángulo viene determinado por el fabricante del

motor.

Para hacer girar a un motor paso a paso, es necesario conseguir la secuencia de activación y desactivación de bobinas; pero primero se debe conocer cual es el punto común de las bobinas (cuando el motor tiene 5 cables, tiene un solo punto común; cuando son 6 cables tiene dos puntos comunes, uno para cada par de bobinas).

El punto común se consigue midiendo la resistencia (Ω) entre cada uno de los cables respecto a los demás; cuando en un cable la resistencia es igual respecto a cualquier otro cable, significa que es un punto común. El valor de esta resistencia también lo especifica el fabricante.

Una vez localizado el (los) punto(s) común(es), se puede controlar el motor de dos maneras: a) conectando el punto común al voltaje que especifica el motor, o b) conectando el punto común a tierra (0v). En la primera, la activación de bobinas se debe realizar conectando el cable de la bobina respectiva a tierra (0v). En la segunda, es necesario conectarlo al voltaje indicado por el fabricante.

Ya establecida la conexión al punto común, el movimiento del motor se puede realizar de tres formas distintas: (a) bobina en bobina, (b) de dos bobinas en dos bobinas, o (c) utilizando medio paso (Half Stepping).

Las secuencias de activación se reflejan en el cuadro 1.

El cuadro 1 describe la secuencia de activación de bobinas para hacer girar el motor en una dirección. En cuanto a estas formas se puede decir que: (a) la forma de bobina en bobina no es muy usada, ya que de esta forma el motor tiene menos torque que utilizándolo de dos en dos bobinas, (b) de dos en dos bobinas permite dar pasos en el ángulo que especifica el fabricante y con más torque que la forma anterior, (c) utilizando medio paso, cada paso que da el motor es la mitad del ángulo que especifica el fabricante.

Es importante destacar que para el uso de motores paso a paso con el micro-controlador, no se deben conectar los cables del motor directamente a los pines ya que los requisitos de corriente para los motores paso a paso generalmente es

elevado (alrededor 1 Amp). Para su uso se recomienda utilizar transistores (e.g. TIP- 122).

Cuadro 1.

Secuencias Paso a Paso

Bobino	Bobina por bobina			
on	Bobina A	Bobina B	Bobina C	Bobina D
on	On	Off	Off	Off
on	Off	On	Off	Off
on	Off	Off	On	Off
on	Off	Off	Off	On

Fuente: Rada (2005).

Cuadro 2.

Título del cuadro

Audio Name	Sum of Extracted Bits						
Police	5	-1	5	5	-7	-5	3
Midnight	7	-3	5	3	-1	-3	5
News	9	-3	7	9	-5	-1	9

Fuente: Rada (2005).

Teoría de Juegos

El objetivo de la teoría de juegos es analizar el comportamiento estratégico de los jugadores. Según R. Fischer (2000), “La teoría de juegos examina el comportamiento estratégico de jugadores que interactúan motivados por la maximización de la utilidad y que saben que los otros participantes son racionales”.

Varoufakis (2001) define la teoría de juegos como

“El análisis del comportamiento racional bajo circunstancias de interdependencia estratégica, cuando la mejor estrategia de un individuo depende de lo que sus oponentes probablemente hagan. comportamiento racional bajo circunstancias de interdependencia estratégica, cuando la mejor estrategia de un individuo depende de lo que sus oponentes probablemente hagan.comportamiento racional bajo circunstancias de interdependencia estratégica, cuando la mejor estrategia de un individuo depende de lo que sus oponentes probablemente hagan.comportamiento racional bajo circunstancias de interdependencia estratégica, cuando la mejor estrategia de un individuo depende de lo que sus oponentes probablemente hagan.comportamiento racional bajo circunstancias de interdependencia estratégica, cuando la mejor estrategia de un individuo depende de lo que sus oponentes probablemente hagan.comportamiento racional bajo circunstancias de interdependencia estratégica, cuando la mejor estrategia de un individuo depende de lo que sus oponentes probablemente hagan.comportamiento racional bajo circunstancias de interdependencia estratégica, cuando la mejor estrategia de un individuo depende de lo que sus oponentes probablemente hagan.”

En la teoría de juegos siempre se habla sobre lo inteligente y lo racional que es un jugador. Según B. Slantchev (2004) un jugador racional es aquel que, coherentemente, toma decisiones en búsqueda de un objetivo bien definido; y un jugador inteligente es aquel que, sabiendo lo que los demás jugadores saben, puede hacer las mismas inferencias que ellos pueden hacer.

De las definiciones anteriores se puede inferir que:

1. La teoría de juegos analiza el comportamiento estratégico de los jugadores ante los posibles eventos que ocurren en el juego.
2. El comportamiento racional de cada jugador viene dado por la búsqueda de maximizar su utilidad.
3. El comportamiento inteligente de cada jugador está determinado por la capacidad de inferir la calidad de una estrategia teniendo en cuenta que los demás jugadores tienen la misma capacidad.
4. Un jugador racional e inteligente jugará basándose en la maximización de su utilidad tomando en cuenta las jugadas que pueden hacer los demás jugadores para

maximizar sus respectivas utilidades.

La representación de un juego en forma normal viene dada por una matriz de pagos que contiene las utilidades que generará cada combinación de estrategias por cada uno de los jugadores. Por ejemplo, el juego piedra, papel ó tijeras es un juego de suma cero en el que dos jugadores eligen un objeto (Piedra, Papel ó Tijeras) y las reglas son: Piedra le gana a Tijeras, Tijeras le ganan a Papel, Papel le gana a Piedra y, si ambos eligen el mismo objeto entonces es un empate; su representación en forma normal se muestra en el siguiente cuadro.

Cuadro 3.

Titulo del cuadro

Jugares I y II	Piedra	Papel	Tijeras
Piedra	(0,0)	(-1,1)	(1,-1)
Papel	(1,-1)	(0,0)	(-1,1)
Tijeras	(-1,1)	(1,-1)	(0,0)

Fuente: Rada (2005).

Como se puede observar en el cuadro 2, la forma normal muestra sólo los resultados representados en términos de la función utilidad de cada jugador (I, II) que se generan para cada combinación de estrategias. La función de utilidad (1,-1) indica que gana el jugador I y pierde el jugador II, (-1,1) gana II y pierde I, y (0,0) implica que I y II empatan. Es de notar que, dada cualquier combinación de estrategias, si se suma la utilidad de A con la utilidad de B, el resultado siempre es cero. Esto indica que Piedra, Papel ó Tijeras es un juego de suma cero.

Bases Legales

Es completamente ilegal. Esto es un ejemplo de definición: **algoritmo evolutivo**. Esto es un ejemplo de plural **algoritmos evolutivos**.

Definición de Términos Básicos

Algoritmos Evolutivos son algoritmos dentro de la Computación Evolutiva que se inspiran en los mecanismos biológicos de selección, cruce, reproducción, y mutación para conseguir soluciones óptimas o sub-óptimas a problemas de optimización. Están basados en la evolución de poblaciones donde cada individuo representa una solución y existe una **función de aptitud** que determina la calidad de la solución y guía el proceso evolutivo. 25

Sistema de Hipótesis

Existe la hipertesis que L^AT_EX será la raíz de los próximos sistemas operativos.

Operacionalización de las Variables

Operar variables.

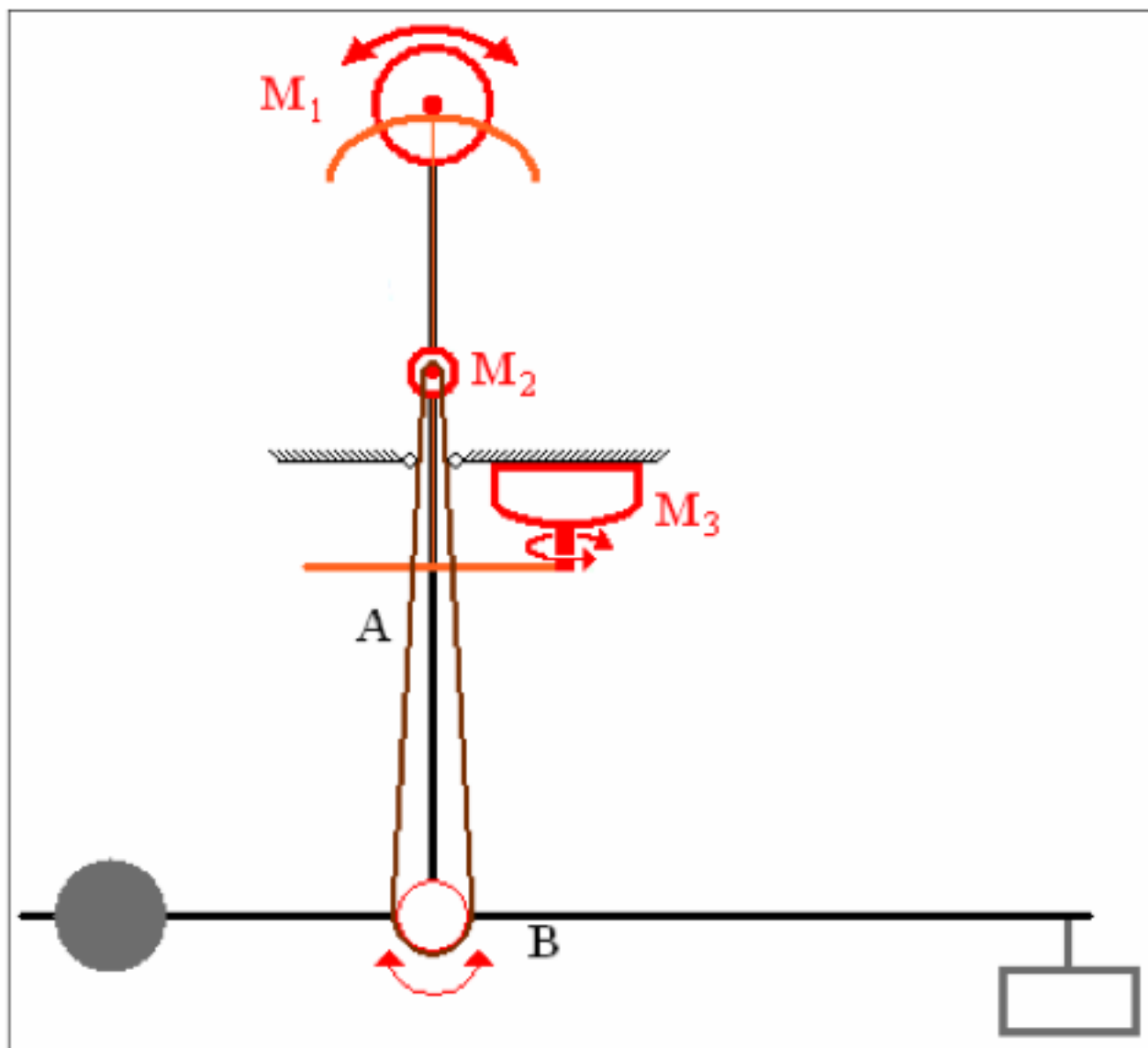


Ilustración 2. Brazo Mecánico con Motores Paso a Paso

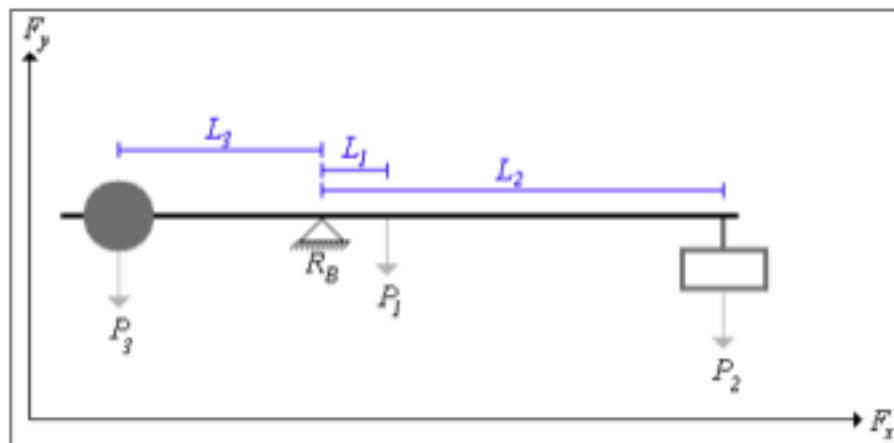


Ilustración 3. Diagrama de Cuerpo Libre – Eslabón B

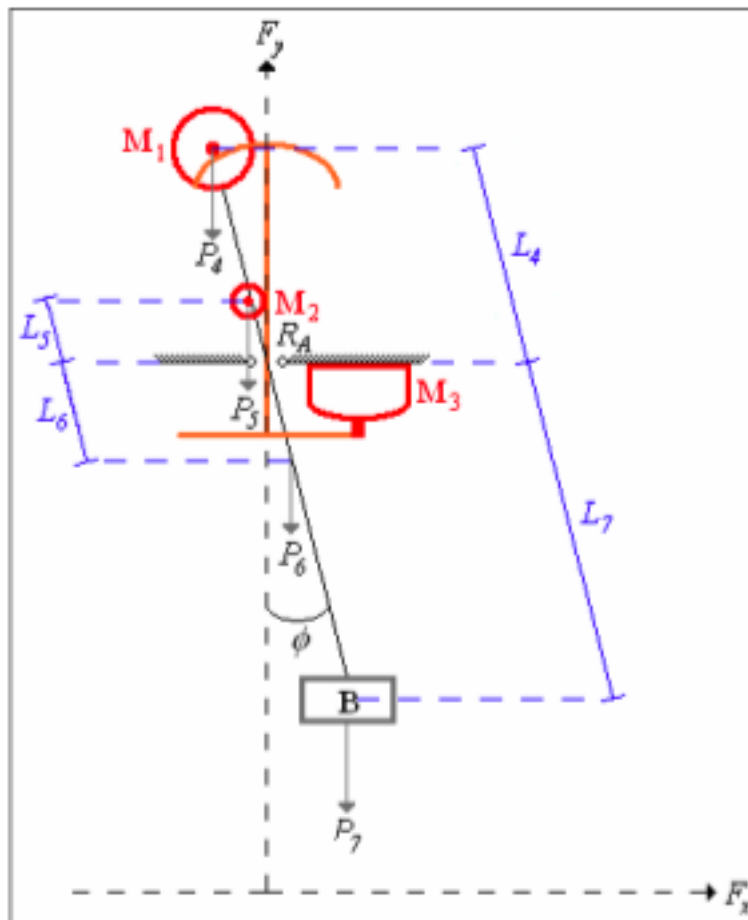


Ilustración 4. Diagrama de Cuerpo Libre – Eslabón A

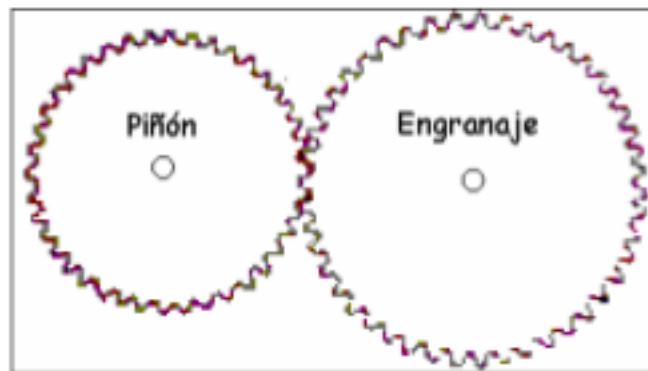


Ilustración 5. Piñón – Engranaje

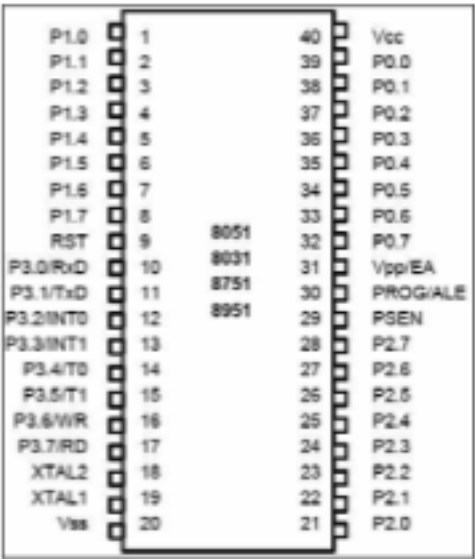


Ilustración 6. Microcontrolador MCS-51

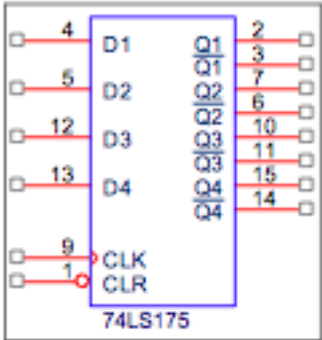


Ilustración 7. Diagrama – 74LS175

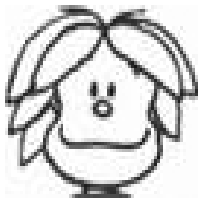


Gráfico 1. Esto debería ser un gráfico (e.g. barras)



Gráfico 2. Esto debería ser un gráfico (e.g. de líneas)

CAPÍTULO III

MARCO METODOLÓGICO

Tipo de Investigación

Esto es experimental según ?.

Cuadro 4.

Título del cuadro

Audio Name	Sum of Extracted Bits						
Police	5	-1	5	5	-7	-5	3
Midnight	7	-3	5	3	-1	-3	5
News	9	-3	7	9	-5	-1	9

Fuente: El autor.

Población y Muestra

En el cuadro 4 se puede observar que sirvió la etiqueta

Diseño de la Investigación o Procedimiento

Técnicas e Instrumentos de Recolección de Datos

Técnicas de Procesamiento y Análisis de los Datos

CAPÍTULO IV

RESULTADOS

CAPÍTULO V

CONCLUSIONES

CAPÍTULO VI

RECOMENDACIONES

ANEXOS

ANEXO A

(Curriculum Vitae)

ANEXO B

(Instrumento de Recolección de Datos)