

**CONTROL
DE MECANISMOS
POR ORDENADOR**

Ezequiel Álvarez Jáñez y María Isabel Pizarro Pizarro

Í N D I C E

INTRODUCCIÓN	247
I. JUSTIFICACIÓN DE LA ACTIVIDAD	249
II. OBJETIVOS	250
III. CONTENIDOS	252
Conceptuales	252
Procedimentales	254
Actitudinales	255
IV. ORGANIZACIÓN DE LAS FASES	257
V. METODOLOGÍA UTILIZADA	258
VI. FECHAS Y PERÍODOS DE REALIZACIÓN	259
VII. DESARROLLO DE LA ACTIVIDAD	260
VIII. CRITERIOS Y PROCEDIMIENTOS DE EVALUACIÓN	261
IX. LA POSIBLE PROLONGACIÓN DEL TRABAJO	262
GUÍA DIDÁCTICA	263

Introducción

Es evidente que nos movemos en un mundo cada vez más dominado por el ordenador y sus aplicaciones. Una de esas aplicaciones es el control de mecanismos por ordenador.

Este hecho, unido a la proliferación de ordenadores y material informático en general en las aulas de nuestros institutos nos ofrece la posibilidad de enseñar a los alumnos unos conceptos y aplicaciones que hace no mucho tiempo sólo eran pensables y ejecutables a nivel de últimos cursos de carreras técnicas.

Encontramos además en este trabajo la virtud de ser altamente motivador para los alumnos y acercarles a unos conocimientos cada vez más necesarios en el entorno del individuo.

El profesional docente jamás debe olvidar la relación que guardan sus enseñanzas con la realidad circundante a la cual se deben. Además nos ha parecido interesante utilizar esta realidad para llegar a ciertos objetivos propuestos por medio de los contenidos adecuados.

Debido al avance de las tecnologías, se espera que en un futuro no muy lejano incluso se apliquen estas técnicas utilizadas en este proyecto, para el control de seguridad de las viviendas.

I. JUSTIFICACIÓN DE LA ACTIVIDAD

Desde la aparición de la tecnología como asignatura obligatoria en la Educación Secundaria Obligatoria (E.S.O) se han dado diversas respuestas a la metodología usada en el desarrollo propuesto en los materiales curriculares.

Nosotros hemos aprovechado una herramienta muy potente y de alto valor educativo como es un lenguaje de programación, **LOGO**, para desarrollar nuestra experiencia de innovación educativa.

El fin último de nuestro trabajo, sería sentar las bases de la programación de una asignatura, optativa de 4º de E.S.O., con un carácter interdisciplinar a caballo entre las áreas de **TECNOLOGÍA** e **INFORMÁTICA**.

La actividad en sí, va encaminada al control de diversos mecanismos de la vida cotidiana mediante el uso del ordenador, en la cual son los propios alumnos quienes diseñarán y crearán el software necesario para llevar el control de una actividad concreta. Además los alumnos distribuidos en **GRUPOS DE TRABAJOS** realizarán la construcción de sus propios mecanismos.

La actividad habitúa a los alumnos en la utilización del ordenador.

La idea ha partido de la demanda por parte de un grupo de alumnos de un trabajo de profundización en proyectos tecnológicos controlados por ordenador.

II. OBJETIVOS

Los objetivos que se pretenden con esta actividad los podríamos englobar en dos grandes grupos: **Experimentación de la actividad y objetivos didácticos derivados de la actividad.**

Dentro del primer grupo de objetivos podríamos detallar los siguientes:

- 1.- Conocimiento y puesta a punto de los recursos necesarios: ordenador, tarjeta controladora, lenguaje de programación LOGO y operadores tecnológicos.
- 2.- Realización de diversos programas y construcción de los prototipos correspondientes que den solución a problemas planteados. Se tratará de resolver problemas tipo: un semáforo, juego de luces, ascensor.
- 3.- Estudiar la adecuación de los contenidos establecidos al desarrollo de las capacidades de los alumnos que cursan 4º de E.S.O.
- 4.- Conocer el presupuesto de la actividad y si es asumible por el centro en el caso de extender la actividad en forma de asignatura optativa.

En el segundo grupo que podríamos considerar como objetivos de cara a los alumnos que realizan la actividad, es decir los objetivos didácticos, están los siguientes:

- 1.- Ser capaz de abordar un problema real en el cual se puede implementar una solución construyendo un mecanismo y diseñando un programa de control
- 2.- Ser capaz de realizar sencillos programas de control usando el lenguaje de programación logo.
- 3.- Ser capaz de diseñar y construir un prototipo en el cual intervengan operadores mecánicos, eléctricos, electrónicos, etc.
- 4.- Ser capaz de trabajar en equipo, coordinándose unos alumnos con los otros, creando un ambiente agradable de trabajo, con el cual se da respuesta al problema planteado.

- 5.- Ser capaz de planificar una solución con el tiempo suficiente y para ello hacer relación de materiales y herramientas necesarias y una secuencia de las actividades a realizar.
- 6.- Ser capaz de descubrir problemas en su entorno susceptibles de ser abordados con la dinámica de trabajo propuesto.
- 7.- Ser capaz de crear un clima grato de trabajo, aportando soluciones propias y aceptando las de los demás siempre bajo una crítica constructiva.
- 8.- Ser capaz de mantener un ambiente de trabajo limpio y ordenado, y cuidar los elementos de trabajo (aula, ordenador, tarjeta controladora, mecanismos de construcción, etc.)

Con los anteriores objetivos se cubre suficientemente el campo de capacidades que se pretende desarrollar en el alumno, como son motrices, afectivas, cognitivas de relación interpersonal, inserción social, todas ellas conllevan al desarrollo del alumno.

III. CONTENIDOS

Los contenidos comprenden todos los aprendizajes que los alumnos deben alcanzar para progresar en las direcciones que marcan los fines de la educación en la etapa, para lo que es preciso estimular comportamientos adquirir valores actitudes y habilidades de pensamiento además de conocimientos.

Los contenidos que pretendemos conseguir los hemos dividido en tres tipos, que detallaremos a continuación.

Conceptuales

- 1.- Distinguir cada uno de los elementos necesarios para el control de un proyecto (ordenador, tarjeta controladora, mecanismos a controlar, etc)
- 2.- Saber cada una de las partes de un ordenador tanto a nivel hardware como software.
- 3.- Los componentes principales de un ordenador: memorias, U.C.P., periféricos y dispositivos de almacenamiento tanto interno como externo.
- 4.- Manejo de los comandos del sistema operativo MS-DOS, relacionados con la gestión de ficheros (crear, borrar, imprimir, grabar), directorios y ejecución de programas.
- 5.- Lenguajes de programación. Lenguaje LOGO.
- 6.- Primitivas del LOGO.
- 7.- Entradas y salidas.
- 8.- Primitivas de control de la tarjeta controladora.
- 9.- Principales primitivas de LOGO:

En este contenido se describen las primitivas del ACTILOGO divididas en grupos, y en el orden alfabético dentro de cada grupo.

- DISCOS Y ARCHIVOS:
 - CARGA • CIERRA • CIERRATODO
 - GUARDA • LEECAR • TECLA?
- PALABRAS, LISTAS Y VARIABLES:
 - FRASE • HAZ
- FUNCIONES LOGICAS:
 - NO • O • Y
- OPERADORES LOGICOS:
 - < • > • =
- FUNCIONES MATEMÁTICAS:
 - POTENCIA • SUMA
- OPERADORES ARITMÉTICOS:
 - + • - • * • /
- PROCEDIMIENTOS Y SENTENCIAS:
 - EDITA • ALTO • ESCRIBE
 - ESPERA • FIN • PARA
 - SI
- CONTROL DE PANTALLA:
 - BP • BT • PONCURSOR
- MISCELANEOS:
 - ADIOS

10.- Principales primitivas de control de la tarjeta:

- Salidas digitales (con inversión de polaridad (al motor) y sin inversión (bombillas, relés)).
- Salidas analógicas.
- Entradas digitales.
- Entradas analógicas

11.- Sistema de representación de la información. Sistema binario, sistema decimal y conversión entre ambos.

- 12.- Estructura de un programa.
- 13.- Los procedimientos y recursividad.
- 14.- Operadores de salida (motores, bombillas y electroimanes).
- 15.- Operadores de entrada. Digitales (pulsadores), analógicos (sensor de posición (potenciómetro), sensores de temperatura (termistancia) y luminosidad (LDR)).
- 16.- Conexión de los distintos operadores.
- 17.- Técnicas de construcción fijas y desmontables.
- 18.- Operadores mecánicos (engranajes, reductoras, tornillos sinfín, poleas, correas, piñón-cremalleras, levas).

Procedimentales

- 1.- Estudiar el problema planteado estructurando en fases su resolución:
 - Diseño y planificación
 - Construcción del mecanismo y programación del mismo.
 - Evaluación y elaboración de memoria.
- 2.- Analizar el problema. Identificar los elementos necesarios para su construcción.
- 3.- Diseñar el prototipo a controlar.
- 4.- Planificar las herramientas y materiales.
- 5.- Diseñar del programa de control para este mecanismo.
- 6.- Construir el prototipo diseñado.
- 7.- Conectar el prototipo a la controladora a través de sus entradas y salidas.
- 8.- Preparar el equipo de control:
A nivel físico:
 - Conexión de la controladora al ordenador.
 - Encendido del ordenador y controladora.

A nivel lógico:

- Instalación del programa LOGO en su versión ACTILOGO.
- Instalación del programa controladora para actilogo.
- Cargar el actilogo.
- Cargar los comandos de control de la controladora.
- Editar el programa que controla el mecanismo.

9.- Comprobar el funcionamiento del prototipo.

10.-Depuración del programa y prototipo.

11.-Búsqueda de posibles mejoras.

12.-Realización de la memoria de trabajo en la cual se incluirán los siguientes anexos:

- Listado del programa.
- Planos del prototipo.
- Esquemas eléctricos y de conexión.
- Descripción del funcionamiento.
- Relación de materiales y herramientas.
- Presupuesto.
- Valoración del trabajo realizado.
- Aplicación de la mismas técnicas a otros problemas.

Actitudinales

- 1.- Actitudes positivas y creativas para resolver los problemas prácticos, con confianza en la propia capacidad de alcanzar resultados palpables.
- 2.- Actitud inquisitiva, abierta y flexible en la exploración y desarrollo de sus propias ideas.
- 3.- Curiosidad y respeto hacia las ideas de otras personas.
- 4.- Gusto por el orden y método en el trabajo, perseverando ante las dificultades encontradas.
- 5.- Disposición e iniciativa personal para organizar y participar solidariamente en tareas de equipo.
- 6.- Gusto por el orden y la limpieza en la presentación de los trabajos.

- 7.- Valoración de otros lenguajes de comunicación distintos al habitual como son el binario y el lenguaje de programación.
- 8.- Interés por conocer el papel que desempeña el conocimiento tecnológico en distintos trabajos y profesiones y por estudiar y elaborar su orientación vocacional y profesional.

IV. ORGANIZACIÓN DE LAS FASES

Para realización del trabajo, dada la dificultad y novedad que supone para los alumnos, hemos creído conveniente dividirlo en una serie de fases que expondremos a continuación:

FASE I: Durante esta primera fase la pretensión global ha sido que los alumnos adquieran los contenidos relacionados con los siguientes temas:

- Equipos utilizados.
- Lenguaje de programación.
- Construcción de prototipos desmontables.

FASE II: En esta fase se ha pretendido que los alumnos consigan un hábito de trabajo para este tipo de proyectos. Para ello, se ha propuesto una serie de problemas, por ejemplo, «**Un semáforo con la siguiente secuencia, 10 segundos rojo, 10 segundos verde y 2 segundos ámbar**», en el cuales los alumnos ha construido el prototipo y mecanismo de control trabajando los contenidos procedimentales.

FASE III: En dicha fase se ha propuesto a los alumnos la realización de un trabajo libre. Los alumnos deben buscar problemas de su entorno que puedan ser resueltos con esta metodologías y ser ellos mismos quienes realicen todo el proyecto.

FASE IV: Los alumnos realizarán una memoria de trabajo a partir de sus actividades de enseñanza-aprendizaje, en la cual aparezca al menos los puntos enumerados en el contenido procedimental número 12.

V. METODOLOGÍA UTILIZADA

La metodología usada ha sido variada, descriptiva, resolución teóricas de problemas, y trabajos prácticos (proyectos), todas ellas se han llevado a cabo de forma conjunta.

La forma básica de proceder ha sido en todo momento en equipo de trabajo formado por seis alumnos. A lo largo de la actividad se ha fomentado una metodología activa y participativa basada en el diálogo tanto entre alumnos como alumnos y profesores.

Durante el desarrollo nos hemos propuesto un modelo de aprendizaje basado en la teoría del constructivismo. Para ello, hemos creído conveniente que los alumnos no solo diseñen soluciones teóricas sino que las realicen completamente, construyéndolas y comprobándolas.

Además en el trabajo libre, es el alumno quien crea su propio aprendizaje puesto que el grupo plantea y resuelve el problema en base a unas necesidades detectadas.

Con la metodología propuesta pretendemos conseguir un alto grado de aprendizajes significativos, puesto que, en la mayor parte de los casos, los problemas que se plantean implican la necesidad del aprendizaje de nuevos contenidos y procedimientos. Todo ello unido a la novedad que despierta en los alumnos hace que la motivación sea elevada.

VI. FECHAS Y PERÍODOS DE REALIZACIÓN

La actividad se ha realizado en el I.E.S. Mixto N° 6. de BADAJOZ, en el aula-taller de Tecnología y en el aula Informática, los miércoles de 5 a 8 de la tarde en el período comprendido entre el 15 de diciembre y 15 de mayo del curso 95/96; al amparo del programa de apertura de centros.

VII. DESARROLLO DE LA ACTIVIDAD

Durante el primer mes, se abordó el problema de espacios, horarios, formación de equipo y aspectos teóricos y prácticos relacionados con los equipos necesarios.

Una vez resuelto este problema, con los conocimientos adquiridos se plantearon los siguientes problemas:

- Un semáforo con la siguiente secuencia: 10 s. rojo, 10 s. verde y 2 s. ámbar.
- Un semáforo con la secuencia anterior y preferencia de peatón.
- Dos semáforos en un cruce de calles.
- Juego de luces para un escenario, con las siguientes opciones: iluminación del solista, un coro, todas y la posibilidad de elegir una luz cualquiera.
- Un ascensor con tres pisos.
- Una puerta de garaje automática.
- Detector de velocidad superior a un límite establecido y contabilización del número total de coches que han pasado en un período de tiempo.
- Rótulo luminoso para una empresa.

En el período final se propone el trabajo libre, para el cual los alumnos van a trabajar sobre el control de una vivienda. En esta fase nos encontramos actualmente.

VIII. CRITERIOS Y PROCEDIMIENTOS DE EVALUACIÓN

Los criterios de evaluación nos van a permitir estimar en que medida se están cumpliendo los objetivos marcados para ello hemos considerado como básico los siguientes:

- Resolución en equipo de un problema planteado. La resolución deberá incluir el listado del programa, ejecución del prototipo y memoria.
- Grado de integración de los alumnos y profesores dentro del grupo de trabajo.
- Claridad en la expresión y presentación de los trabajos.
- Cuidado de los materiales.
- Orden y limpieza durante el proceso de trabajo.
- Adquisición de destrezas intelectuales y manuales.
- Respeto de las normas de convivencia establecidas.

Creemos que los procedimientos de evaluación no deben ser algo fuera de la actividad o al final de ella sino que han de estar presente día a día, en las actividades de enseñanza-aprendizaje corrigiendo errores o actuaciones equivocadas y encaminando la experiencia hacia la consecución de los objetivos. Por todo ello los procedimientos que consideramos oportunos son:

- Observación directa de los trabajos realizados señalando errores y posibles mejoras.
- Análisis de las memorias de trabajos presentados.
- Diálogo con los alumnos detectando posibles problemas dentro del grupo.
- Control diario del material utilizado.
- Unidos a los anteriores quizás el mejor procedimiento de evaluación sea el progreso conseguido entre los distintos trabajos puesto que van aumentando en complejidad; para llegar al trabajo libre cuya consecución implica la asimilación de los objetivos.

IX. LA POSIBLE PROLONGACIÓN DEL TRABAJO

El trabajo tiene una clara aplicación en el aula y por tanto prolongación en el tiempo. Es la intención de los profesores participantes en la experiencia diseñar la programación de una asignatura optativa en 4º ESO de clara iniciación profesional para muchos alumnos; al tiempo que supone una materia preparatoria para posteriores estudios académicos o profesionales.

Una vez realizada la mayor parte del trabajo y a la vista de los resultados obtenidos, vemos que se puede plantear la asignatura en nuestro centro como optativa para el curso 96/97. El esfuerzo económico que conlleva no es excesivo, puesto que se dispone de controladoras para ordenador y también de aulas-taller suficientemente equipadas. Habría que realizar eso sí la compra de más unidades para dar respuesta a una posible demanda no prevista.

Es además una realidad el avance del ordenador en el mundo que nos rodea (cada vez más tecnificado), y el tipo de optativa que pretendemos creemos que tendrá cada vez mayor aceptación entre el alumnado.

Por todo ello, estimamos que la **experiencia está siendo altamente positiva** y sus resultados notables en la formación de los alumnos. Así la fase de experimentación finalizará al acabar este trabajo y **el siguiente paso será su realidad en el aula.**

GUÍA DIDÁCTICA

I. INTRODUCCIÓN	267
II. ACTIVIDADES	269
- Jugando a controlar semáforos	269
Control de un semáforo	269
Control de un semáforo con preferencia de peatón	272
Control de dos semáforos	276
- Jugando a controlar luces	281
Iluminación de un escenario	281
Rótulo de reclamo publicitario	288
- Ascensor de tres pisos	292
- Puerta de garaje automática	298
- Detector de tráfico	304
III. GUÍA DE REFERENCIA TÉCNICA	309
- Introducción	309
- Requerimientos HW y SW	309
- Instalación de la aplicación	312
Instalación del Actilogo	312
Instalación de la tarjeta controladora	312
Instalación del programa de aplicación	312
Ejecución del programa	313
IV. BIBLIOGRAFÍA	315

Agradecimientos

Queremos agradecer la colaboración prestada por los alumnos David Espejo Agundez, Miguel Blanco Otano, Pedro Mancha, Pablo Gómez Rodríguez, Manuel Muñoz Marín, Francisco Javier Álvarez Medina, por compartir con nosotros esta experiencia.

Por otro lado, reconocemos la ayuda prestada por Antonio Lozano Timón, catedrático de Tecnología Eléctrica por ayudarnos a dar los primeros pasos, y a Luis del Rosal por la ayuda prestada en la realización del

También queremos agradecer la disponibilidad del centro IES Mixto nº 6 de BADAJOZ por el uso de sus instalaciones para el desarrollo de la actividad.

I. INTRODUCCIÓN

Hemos realizado este proyecto que a continuación detallamos a partir de una experiencia de grupo de trabajo. Dicho grupo se formó por dos profesores de distintas especialidades (TECNOLOGÍA e INFORMÁTICA) y por seis alumnos de 4º ESO; buscando la interdisciplinariedad de áreas curriculares.

La intención ha sido investigar **el control de mecanismos por ordenador mediante el lenguaje LOGO.**

El objetivo principal de este proyecto era analizar la metodología y las actividades de enseñanza-aprendizaje para posteriormente diseñar una asignatura optativa en 4º ESO que tuviese una **interdisciplinariedad tecnología-informática.**

Queríamos saber si el nivel de contenidos era adecuado al desarrollo de capacidades que poseen los alumnos al llegar a este curso.

Del mismo modo pretendíamos identificar problemas que pudieran darse en el transcurso de la actividad, para subsanarlos posteriormente, comprobando si los alumnos respondían con la suficiente motivación.

Por otra parte, queríamos cuantificar si el proyecto podía ser asumible por el centro económicamente en cuanto a la compra de materiales nuevos, y por la disponibilidad de espacios donde desarrollarse.

Para nosotros las respuestas han sido todas favorables y, para cursos posteriores, pensamos lanzar la propuesta de una asignatura optativa en base al estudio que hemos realizado en este proyecto.

Una de las principales razones del éxito de la experiencia radica en que el tema de los ordenadores es para los alumnos algo nuevo que quieren conocer y controlar, ofreciéndoles esta experiencia la posibilidad de realizarlo.

No debemos olvidar que en un mundo como el que habitamos el ordenador ocupa cada vez un puesto más relevante en nuestra sociedad, y es probable que en un futuro no muy lejano, el control automático de nuestro entorno (hogar, oficina, etc) será hecho por ordenador.

El trabajo será un resumen de nuestra experiencia llevada al aula, y tendrá como finalidad que otros compañeros se atrevan a poner en marcha esta experiencia en sus aulas.

Como puede entenderse se trata de un trabajo abierto, que nosotros mismos pensamos seguir completando, a la vez que invitamos a otros docentes a trabajar en este campo.

Lo creemos muy interesante debido a la gran motivación que supone para los alumnos, siempre y cuando se plantee como asignatura optativa en 4º ESO.

Presentamos una serie de actividades que hemos ido realizando con los alumnos de una forma detallada, de modo que quien tenga interés pueda seguirlas y aplicarlas en sus aulas.

II. ACTIVIDADES

JUGANDO A CONTROLAR SEMÁFOROS

CONTROL DE UN SEMÁFORO

1.- Descripción de la actividad

Se quiere controlar un semáforo que tenga la siguiente secuencia de estados:

- Color verde 10 s
- Color ámbar 2 s
- Color rojo 10 s

Este ciclo ha de repetirse indefinidamente o bien hasta que decidamos detenerlo pulsando una tecla.

2.- Análisis del problema

Vamos a necesitar tres bombillas de 5 VCC con una carcasa verde, otra ámbar y otra roja, para simular el semáforo que vamos a controlar.

Para poder realizar el control del semáforo debemos tener durante 10" encendida la bombilla con carcasa verde (B1) y apagadas las otras dos bombillas. Después debe apagarse la B1 y simultáneamente encenderse la bombilla ámbar (B2) permaneciendo así 2". Por último se apaga la bombilla B2 y se enciende la bombilla roja (B3). Este proceso se repetirá indefinidamente, hasta que pulsando una tecla se detenga el proceso.

Esta actividad tiene un funcionamiento muy sencillo de programar, sin embargo llevarlo a la realidad mediante operadores eléctricos y electrónicos conlleva una notable dificultad para la edad de los alumnos.

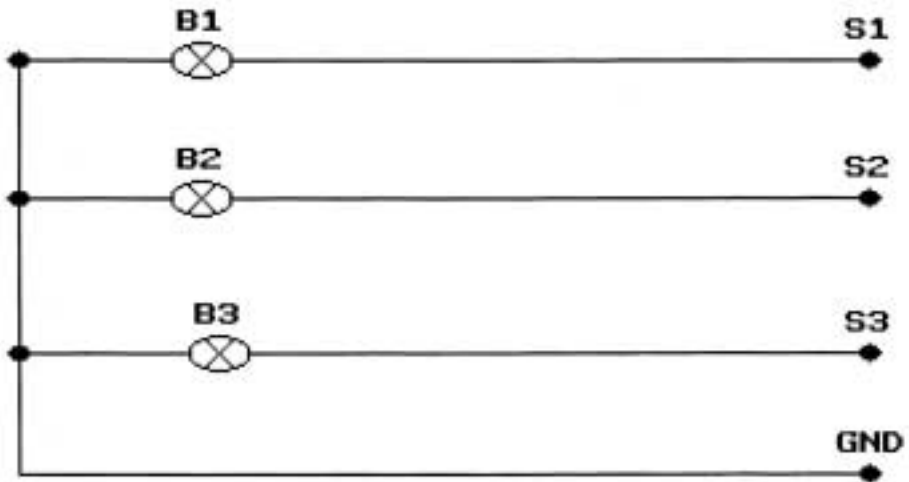
Es importante resaltar esta característica de los problemas a resolver para que los alumnos descubran la enorme ventaja que supone el uso de este tipo de técnica.

3.- Material necesario

- Ordenador, y tarjeta controladora.
- 3 bombillitas de 5VCC con carcasas de color rojo, verde y ámbar.
- Cablecillo de colores y terminales de conexión fácil.

- Lo mejor de todo es tener un equipo de material desmontable, nosotros hemos trabajado con Fischer-Technik.

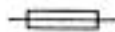
4.- Conexionado



CONEXION



PULSADOR



BOMBILLA 5V



5.- Programación

El procedimiento utilizado para el control de un semáforo es:

PARA SEMA1
SALIDA 1
ESPERA 90
SALIDA 2
ESPERA 12

SALIDA 4
ESPERA 90
SI TECLA? [SALIDA 0 ALTO]
SEMA1
FIN

En este listado vemos las órdenes que se han dado a través de logo para conseguir controlar el semáforo. A continuación vamos a describir de una forma más detallada lo más importante de ellas.

Empezamos creando un procedimiento SEMA1 que va a ser el nombre con el que se llama a este fichero para el control del semáforo, se ha llamado así para diferenciarlo de otras versiones con semáforos que hemos realizado.

SALIDA 1 activa la salida S1 poniendo 5 VCC entre esta salida y la toma de tierra GND, por ello se encenderá la bombilla verde B1 conectada a esta salida.

ESPERA 90 es una orden que actúa de temporizador pues detiene el proceso durante un tiempo de 90 unidades, que en nuestro ordenador corresponden a 10 segundos. Esto depende de la velocidad del ordenador. En este período se mantiene encendida la B1 pues es la que estaba activada.

SALIDA 4 activa la salida S3 encendiendo la bombilla roja, y sólo esa, pues las demás estarán apagadas. Es evidente si escribimos el nº 4 en código binario: 00000100, como vemos el único 1 aparece en la tercera posición por la derecha, es decir la salida S3.

SI TECLA? trabaja como instrucción condicional. Si hemos pulsado una tecla cualquiera en nuestro teclado ejecuta el argumento que se encuentra entre los corchetes, en este caso SALIDA 0 ALTO es decir apaga todo y sale del programa. Para salir sin usar esta opción se puede usar F10.

SEMA1 Actúa aquí una recursividad de cierto nivel. Es curioso e importante que los alumnos entiendan este tipo de instrucciones. Aquí se llama de nuevo a SEMA1, es decir se va a repetir de nuevo toda la cadena de instrucciones de una forma ininterrumpida hasta que alguien pulse una tecla cualquiera. Es la forma, como vemos muy sencilla de conseguir ciclos indefinidos.

FIN la orden fin siempre hay que escribirla al finalizar la edición de un procedimiento.

NOTA Si hemos cometido errores en el teclado o diseño del procedimiento, podemos editarlo escribiendo ED «NOMBREPROCEDIMIENTO y cuando acabemos de editar pulsar F1.

6.- Funcionamiento

El funcionamiento es muy sencillo, pues una vez que escribimos en la pantalla del logo cuando aparezca la interrogación, la orden de sema1 se activa.

?sema1

El semáforo se enciende según la secuencia prefijada, y a la vez se van encendiendo los diodos led rojos correspondientes a las salidas S1 S2 S3.

CONTROL DE UN SEMÁFORO CON PREFERENCIA DE PEATÓN

1.- Descripción de la actividad

Se quiere controlar un semáforo que tenga la siguiente secuencia de estados:

- Color verde 10 s
- Color ámbar 2 s
- Color rojo 10 s

Este ciclo ha de repetirse indefinidamente o bien hasta que decidamos detenerlo pulsando una tecla o bien cuando un usuario pulse un botón situado sobre el soporte del semáforo; en este caso cambiará la secuencia, de modo que dé preferencia al peatón y este pueda pasar sin necesidad de espera.

2.- Análisis del problema

Vamos a necesitar tres bombillas de 5 VCC con una carcasa verde, otra ámbar y otra roja, para simular el semáforo que vamos a controlar. Además nos hará falta un pulsador en que colocaremos en la estructura del semáforo para poder solicitar la preferencia.

Para poder realizar el control del semáforo debemos tener durante 10" encendida la bombilla con carcasa verde (B1) y apagadas las otras dos bombillas. Después debe apagarse la B1 y simultáneamente encenderse la bombilla ámbar (B2) permaneciendo así 2". Por último se apaga la bombilla B2 y se enciende la bombilla roja (B3). Este proceso se repetirá indefinidamente, hasta que pulsando una tecla se detenga el proceso.

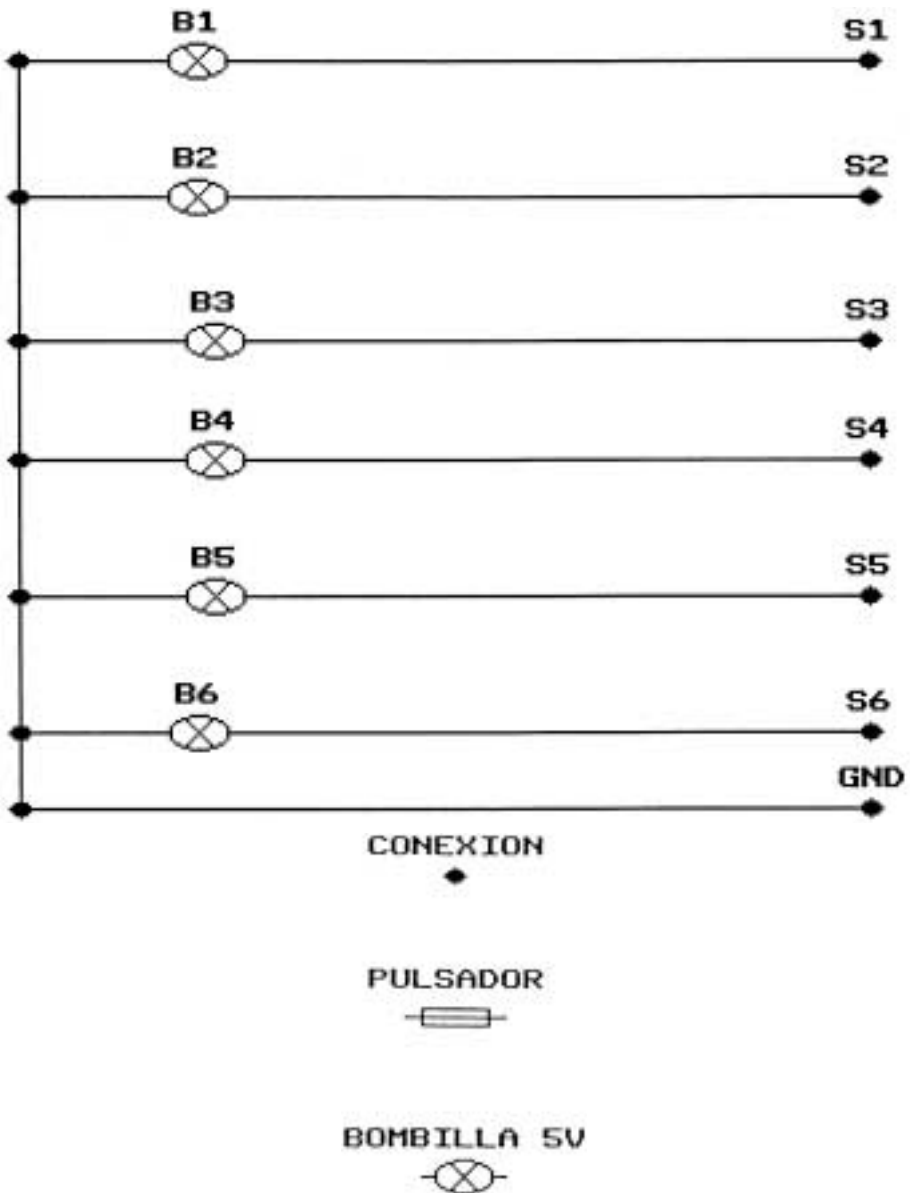
Hasta aquí es igual que el problema anterior. Pero en este además queremos que si llega un peatón y pulsa un botón la secuencia cambie. Si están pasando coches, es decir verde para los coches se pondrá en rojo para ellos de modo que pueda pasar el peatón; mientras que si está en rojo para los coches, se supone que está en verde para el peatón y aunque pulse el botón la secuencia no cambia puesto que ya tiene su preferencia.

Esto lo haremos colocando el pulsador P1 a la entrada digital E1 y realizando el correspondiente programa que después detallaremos.

3.- Material necesario

- Ordenador y tarjeta controladora.
- 3 bombillitas de 5VCC con carcasas de color rojo, verde y ámbar.
- Cablecillo de colores y terminales de conexión fácil.
- Un pulsador.
- Fischer-Technik.

4.- Conexionado





5.- Programación

Los procedimientos utilizados para el control de un semáforo con peatón son:

```
PARA SEMAP  
HAZ «N 1  
VERDE  
FIN
```

```
PARA VERDE  
SALIDA 1  
SI 1 = SD 1 [ROJO]  
ESPERA 10  
HAZ «N :N + 1  
SI :N 9 [AMBAR]  
SI TECLA? [SALIDA 0 ALTO]  
VERDE  
FIN
```

```
PARA ROJO SALIDA 4  
ESPERA 90  
SEMAP  
FIN
```


PARA AMBAR
SALIDA 2
ESPERA 12
SALIDA 4
ESPERA 90
SEMAP
FIN

Vamos a describir el programa haciendo mención especial a lo más importante y a lo novedoso, no comentando órdenes ya explicadas en anteriores programas.

Como podemos comprobar el programa consta de los procedimientos siguientes:

SEMAP
VERDE
ROJO
AMBAR

Esto es habitual en trabajo con logo, pues los procedimientos se comportan como rutinas, o subprogramas que van realizando ciertas funciones, y llamándose a sí mismo o unos a otros van configurando el programa completo.

En el procedimiento SEMAP nos encontramos la primitiva HAZ. Se trata de una asignación, dejando a N inicializada con el valor numérico 1.

El procedimiento VERDE enciende la luz verde del semáforo y es en este procedimiento donde se da la mayor dificultad del programa. Veamos: SD 1 es el valor de la entrada digital 1 (E1). Sólo puede tomar dos valores lógicos, 0 y 1. Si está pulsado llega tensión (5V) a E1 desde la toma de 5VCC e indicará 1. En caso contrario indicará 0.

Como vemos si SD 1 toma el valor 1 la instrucción condicional hace que se ejecute ROJO; es decir cambia la secuencia. Si no vale 1 o sea vale 0 el programa sigue leyendo el procedimiento VERDE. Lo siguiente que hace es esperar un instante e incrementar la variable N y volver a llamarse a sí mismo. A cada paso N se irá incrementando en una unidad y si no deseamos abandonar el programa con SI TECLA? llegará un momento que llegue a 9. En ese momento llama al procedimiento AMBAR. Todo esto se ha realizado así, aunque parezca muy largo es una buena solución, para que en el momento que el peatón pulse su botón de preferencia se ponga en rojo para los coches, pues sino debería esperar hasta el final de la espera de 90 unidades de tiempo.

ROJO simplemente enciende la bombilla B3, de color rojo y vuelve a llamar al procedimiento general SEMAP para que continúe con su secuencia.

AMBAR enciende el ámbar y pasado un tiempo apaga el ámbar encendiendo el rojo y esperar 10", después llama al procedimiento general SEMAP y se continúa la secuencia.

6.- Funcionamiento

El funcionamiento es muy sencillo, pues una vez que escribimos en la pantalla del logo cuando aparezca la interrogación, la orden de semap se activa.

?sema1

El semáforo se enciende según la secuencia prefijada, y a la vez se van encendiendo los diodos led rojos correspondientes a las salidas S1 S2 S3. Mientras no pulse el botón del semáforo todo es como en el trabajo anterior, pero si pulso el botón inmediatamente cambia la secuencia, permitiendo el paso del peatón.

Si pulso una tecla el programa finaliza devolviendo el control al logo.

7.- Actividades de perfeccionamiento

Se pueden proponer al alumno para que éste profundice en el conocimiento de otras actividades basadas en control de semáforo. Por ejemplo sobre este trabajo el alumno puede realizar otro que además tenga un semáforo de peatones, y en el momento que se pulse la preferencia además de cerrarse el paso a los coches, se abra a las personas.

CONTROL DE DOS SEMÁFOROS

1.- Descripción de la actividad

Realizar la sincronización de dos semáforos (S1 y S2) simultáneamente de tal forma que cuando S1 está en verde durante 10 seg. el semáforo S2 se encuentra en rojo durante 10 seg., transcurrido ese tiempo S1 pasa a rojo y S2 pasa a ámbar durante 5 seg.; Transcurrido los 5 seg., S1 continúa en rojo otro 10 seg. más y S2 pasa a verde durante 10 seg.; transcurrido ese período S2 pasa a rojo y S1 cambia a verde a continuación se vuelve a repetir el ciclo. Para interrumpir el ciclo se introduce una tecla por teclado y las luces del semáforo se apagan y los diodos de la tarjeta controladora también.

2.- Análisis del problema

El análisis del problema es la parte más importante del proyecto, ya que si se diseña mal difícilmente puede funcionar.

No hay que olvidar que el análisis del problema hay que estudiarle desde dos puntos de vista uno a nivel físico y otro a nivel lógico.

Para llevar a cabo esta actividad se debe disponer de 6 bombillas y seis carcassas de ellas dos verde, dos rojas y dos ámbar para simular los dos semáforos.

Cada bombilla lleva asociada dos cables con terminales en los extremos, dos terminales se conectan a la bombilla y los otros extremos uno a la entrada **GND** y el otro a la salida correspondiente de la tarjeta controladora.

Las bombillas solo pueden tener dos estados posibles, encendido que toma el valor 1 y apagado que toma el valor 0.

En la siguiente tabla se muestra las posibles combinaciones de los estados de las bombillas que componen los semáforos.

<u>SEMÁFOROS</u>		<u>BOMBILLAS</u>						<u>SALIDAS (DIODOS LED)</u>					
S1	S2	<u>SEMÁFORO1</u>			<u>SEMÁFORO2</u>			S1	S2	S3	S4	S5	S6
		B1	B2	B3	B4	B5	B6						
		V	A	R	V	A	R						
VERDE	ROJO	1	0	0	0	0	1	1	0	0	0	0	1
ÁMBAR	ROJO	0	1	0	0	0	1	0	1	0	0	0	1
ROJO	VERDE	0	0	1	1	0	0	0	0	1	1	0	0
VERDE	ROJO	1	0	0	0	0	1	1	0	0	0	0	1

Los caracteres V, A, R corresponden con cada uno de los colores del semáforo, verde, ámbar y rojo respectivamente.

3.- Material necesario

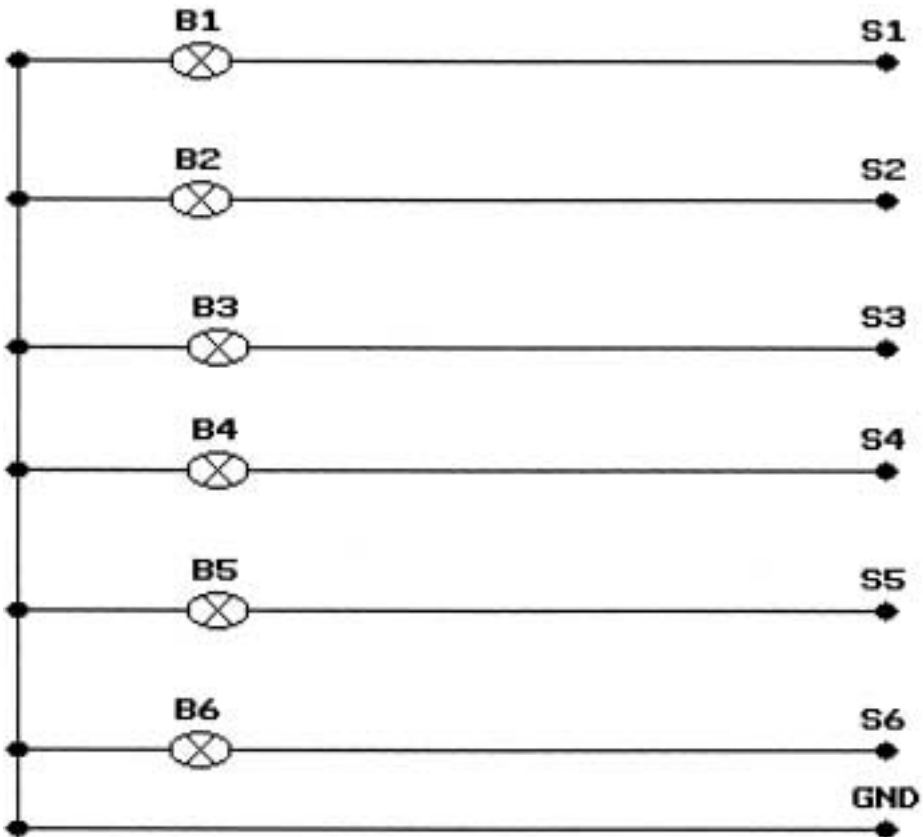
El material necesario para llevar a cabo esta actividad es:

- 6 bombillas de 5 voltios, con 6 carcacas, dos de los siguientes colores: rojo, verde y ámbar.

- 6 cablecillos de colores y con terminales fáciles.
- Soporte de apoyo para simular los semáforos.

Se aconseja realizar el montaje en el aula-taller por los alumnos, sino se dispone de un equipo de ficher-technik.

4.- Conexionado



CONEXION

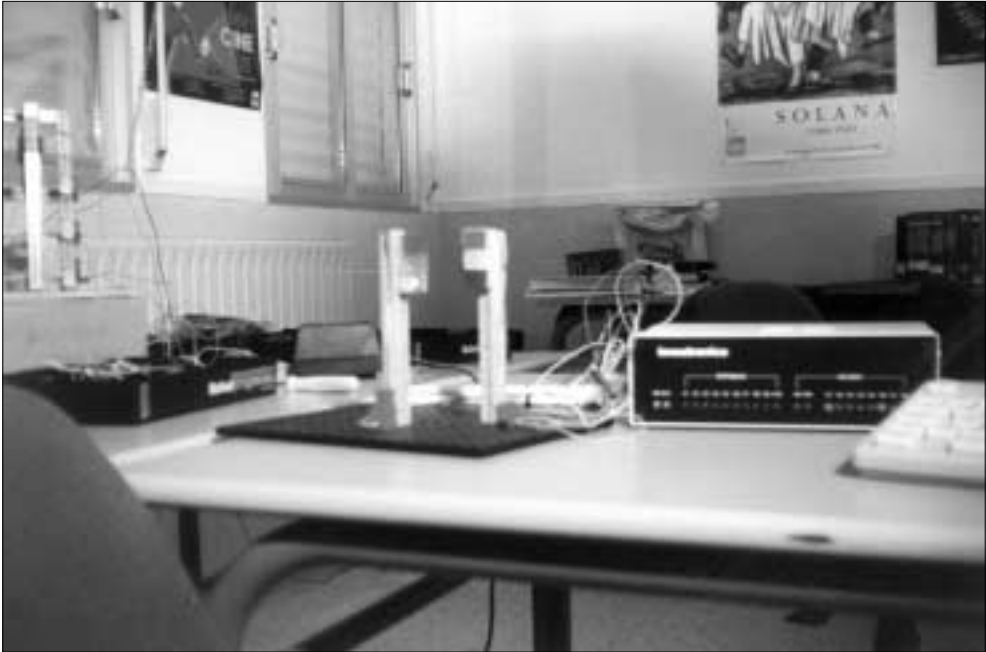


PULSADOR



BOMBILLA 5V





5.- Programación

En este apartado se expondrá el listado del programa con sus diferentes procedimientos, que detallamos a continuación, separados por cada una de las actividades realizadas.

El procedimiento utilizados para el control de dos semáforos es:

```
PARA SEMA2
SALIDA 33
ESPERA 100
SALIDA 34
ESPERA 30
SALIDA 12
ESPERA 100
SALIDA 20
ESPERA 30
SI TECLA? [SALIDA O ALTO]
SEMA2
FIN
```

6.- Breve comentario del programa

El programa se encuentra en un fichero llamado **SEMA2.LOG**, una vez cargado se ejecuta con el procedimiento **SEMA2**. Este procedimiento activa los diodos-led y las bombillas de los semáforo siguiendo la secuencia descrita anteriormente.

En el programa se puede observar que aparecen una primitiva «**SALIDA N**», donde N es un número decimal.

El ordenador trabaja con dos estados posible 0 y 1 debido a la forma en que están implementados sus circuitos electrónicos.

Es por ello, la necesidad de convertir el número binario a decimal dependiendo de las salidas que deseemos activar.

Las salidas corresponden con los diodo-led's que se encuentran en la parte frontal de la tarjeta controladora que deben encenderse en cada instante, estos diodos se representan en binario.

Con 8 led se pueden presentar en base 2, 28 salidas diferentes, es decir, 256, numeradas de 0 a 255.

En el instante uno debe permanecer encendidos los led's S0 y S5 que en binario corresponden al número, $20+25=1+32=33$, el número 33 corresponde con la **SALIDA 33** de la tarjeta controladora.

En el instante dos, las salidas que se encienden son S1 y S5 que corresponden al número binario 34, formado por $21+25=2+32=34$.

En el instante tres, las salidas que se activan son S2 y S3, que equivalen en binario al número 12, es decir, la suma de $22+23$.

En el último instante del ciclo , las salidas que permanecen encendidas son S2 y S4 que corresponden, por el método anterior al número 20.

Para terminar la actividad, al pulsar una tecla cualquiera se encienden todas las salidas de la tarjeta controladora que equivale al número binario 255.

7.- Funcionamiento

La ejecución del programa se repite la siguiente secuencia una serie de veces:

Realizar la sincronización de dos semáforos (S1 y S2) simultáneamente de tal forma que cuando:

- 1.- El **semáforo uno** está en verde durante 10 seg. y el **semáforo dos** se encuentra en rojo durante 10 seg.
- 2.- Transcurrido ese tiempo el **semáforo uno** pasa a rojo y el **semáforo dos** pasa a ámbar durante 5 seg.
- 3.- Al cabo de los 5 seg., el **semáforo uno** continúa en rojo otro 10 seg. más y el **semáforo dos** pasa a verde durante 10 seg.
- 4.- Pasado ese período el **semáforo dos** pasa a rojo y el **semáforo uno** cambia a verde.

A continuación se vuelve a repetir el ciclo. Para interrumpir el ciclo se introduce una tecla por teclado y las luces del semáforo se apagan y los diodos de la tarjeta controladora también, devolviendo el control a la línea de comando del ACTILOGO.

8.- Actividades de perfeccionamiento

Las actividades de perfeccionamiento pueden ser propuesta por el profesor y por los propios alumnos relacionadas con la actividad principal.

Las actividades propuestas por el profesor son entre otras:

- Controlar un cruce con cuatro semáforos.

JUGANDO A CONTROLAR LUCES

ILUMINACIÓN DE UN ESCENARIO

1.- Descripción de la actividad

Consiste en iluminar un escenario de acuerdo con las necesidades de actuación que se este llevando a cabo por el usuario. Esta iluminación debe permitir las siguientes opciones:

- Se deben encender todas las luces del escenario a la vez.
- Una luz del escenario que será la que ilumine al solista de un grupo de música.

- Otra opción que ilumine al coro de un grupo de música.
- Una luz de las ocho posibles elegidas desde teclado por el usuario.

Las luces permanecerán encendidas por un tiempo estimado en 3 segundos.

2.- Análisis del problema

El análisis del problema es la parte mas importante del proyecto, ya que si se diseña mal difícilmente puede funcionar.

No hay que olvidar que el análisis del problema hay que estudiarle desde dos puntos de vista uno a nivel físico y otro a nivel lógico.

Para llevar a cabo esta actividad se debe disponer de 8 bombillas, con carcasas de diferentes colores para simular las luces distribuidas por un escenario.

Cada bombilla lleva asociada dos cables de colores, uno se conecta a la entrada **GND** de la tarjeta y el otro a la salida que tiene asignada en la controladora.

Las bombillas solo pueden tener dos estados posibles, encendido que toma el valor 1 y apagado que toma el valor 0.

En la siguiente tabla se muestra las posibles combinaciones de los estados de las luces de un escenario dependiendo de la acción elegida.

OPCIONES	BOMBILLAS	SALIDAS							
		S8	S7	S6	S5	S4	S3	S2	S1
CORO	B2,B3,B4	0	0	0	0	1	1	1	0
SOLISTA	B1	0	0	0	0	0	0	0	1
TODAS	B1-B8	1	1	1	1	1	1	1	1
1	B1	0	0	0	0	0	0	0	1
2	B2	0	0	0	0	0	0	1	0
3	B3	0	0	0	0	0	1	0	0
4	B4	0	0	0	0	1	0	0	0
5	B5	0	0	0	1	0	0	0	0
6	B6	0	0	1	0	0	0	0	0
7	B7	0	1	0	0	0	0	0	0
8	B8	1	0	0	0	0	0	0	0

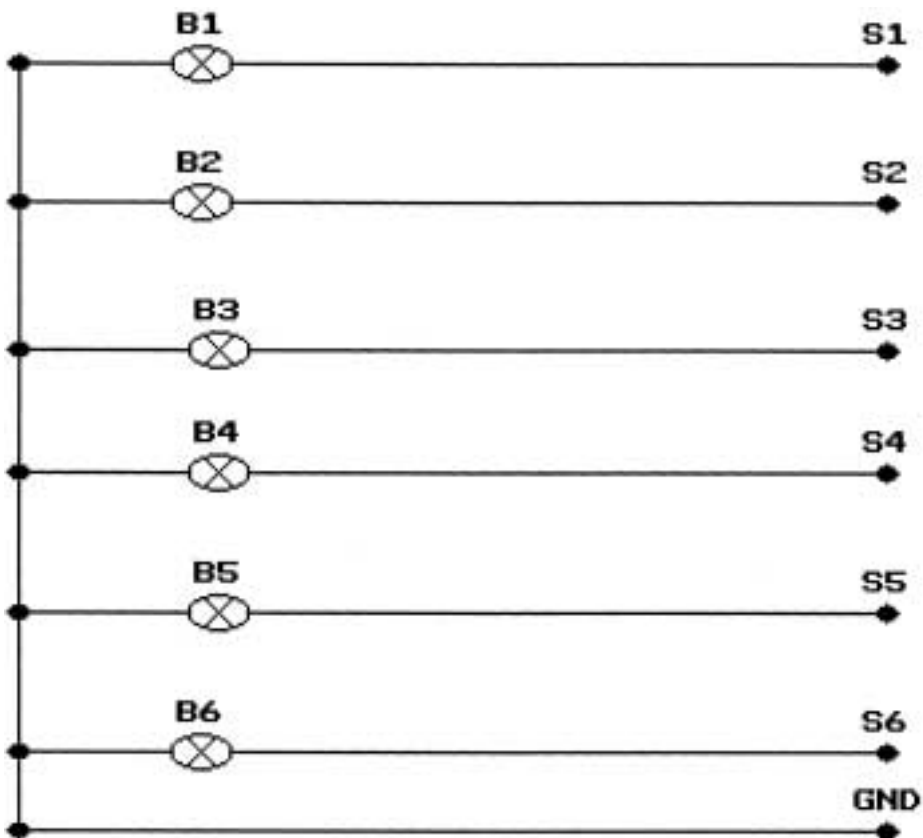
3.- Material necesario

El material necesario para llevar a cabo esta actividad es:

- 10 bombillas de 5 voltios, con 10 carcacas de diferentes colores.
- 10 cablecillos de colores y con terminales fáciles.
- Soporte de apoyo para simular un escenario.

Se aconseja realizar el montaje en el aula-taller por los alumnos, sino se dispone de un equipo de ficher-technik.

4.- Conexionado



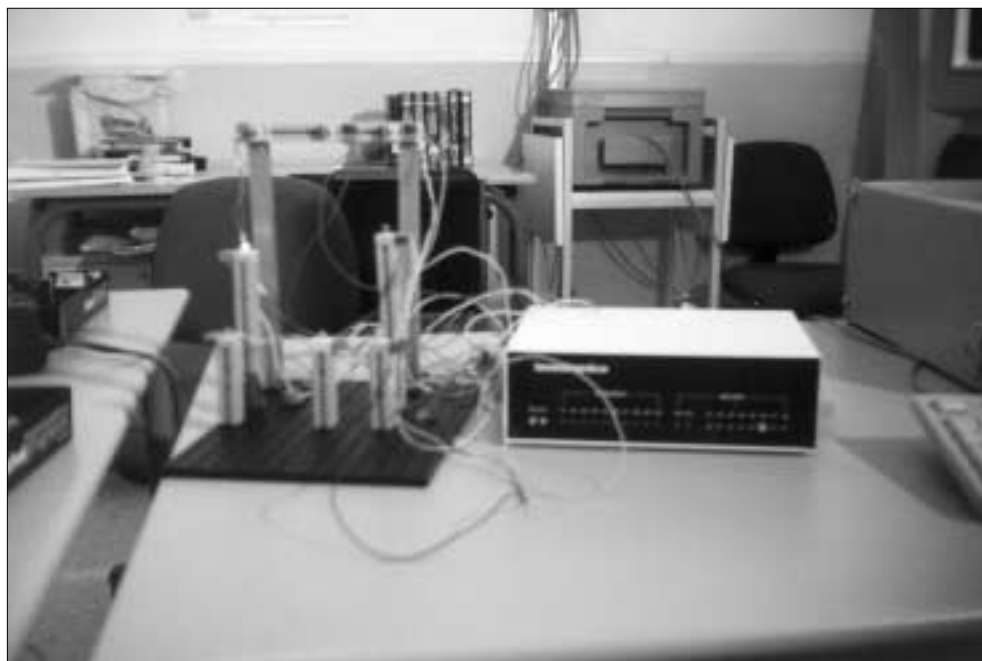
CONEXION



PULSADOR



BOMBILLA 5V



5.- Programación

En este apartado se expondrá el listado del programa que se encuentra en el fichero ESCENA.LOG, con sus diferentes procedimientos, que detallamos a continuación, separados por cada una de las actividades realizadas.

Los procedimientos utilizados para el control de la iluminación de un escenario son los siguientes:

PARA ESCENA

PT

BT

ESPERA 10

PONCURSOR [10 1]

ESCRIBE FRASE [ILUMINACION DEL ESCENARIO]

PONCURSOR [5 5]

ESCRIBE FRASE [1 - TODAS]

PONCURSOR [5 7]

ESCRIBE FRASE [2 - SOLISTA]

PONCURSOR [5 9]

ESCRIBE FRASE [3 - CORO]

PONCURSOR [5 11]

ESCRIBE FRASE [4 - INDIVIDUAL]

PONCURSOR [5 13]

ESCRIBE FRASE [5 - FIN]

PONCURSOR [10 15]

ESCRIBE FRASE [ELIGE UNA OPCION:]

PONCURSOR [29 15]

HAZ «X LEECAR

ES :X

SI «1 = :X [TODAS ALTO]

SI «2 = :X [SOLISTA ALTO]

SI «3 = :X [CORO ALTO]

SI «4 = :X [INDIVIDUAL ALTO]

SI «5 = :X [FINAL]

ESCENA

FIN

PARA CORO

SALIDA 14

ESPERA 10

ESCENA

FIN

PARA SOLISTA

SALIDA 1

ESPERA 10

ESCENA

FIN

PARA TODAS
SALIDA 255
ESPERA 10
ESCENA
FIN

PARA INDIVIDUAL
PONCURSOR [1 18]
ESCRIBE FRASE [INTRODUCE UN N° ENTRE 1 - 8:]
PONCURSOR [31 18]
HAZ «B LEECAR
ES :B
HAZ «A POTENCIA 2 :B - 1
SALIDA :A
ESPERA 40
ESCENA
FIN

PARA FINAL
CIERRATODO
ADIOS
FIN

6.- Breve comentario del programa

Hay que destacar en el programa una serie de primitivas del ACTILOGO que aparecen por primera vez como son:

- **BT:** Borra el texto que hay en pantalla.
- **BT:** Deja toda la pantalla para texto.
- **PONCURSOR:** Cambia la columna y la fila dónde ha de posicionarse el cursor.
- **ESCRIBE:** Visualiza el objeto u objetos indicados como parámetros de entrada.
- **FRASE:** Devuelve una lista compuesta por los objetos de entrada.
- **HAZ:** Hace que el objeto de entrada sea el valor de la variable indicada.
- **LEECAR:** Devuelve el primer carácter teclado.

- **POTENCIA:** Calcula la potencia correspondiente a la base y exponente de entrada.
- **CIERRATODO:** Cierra todos los procedimientos que se encuentran abiertos.
- **ADIOS:** Abandonar el LOGO y salir al sistema operativo.

El programa se encuentra en un fichero llamado ESCENA.LOG que está formado por una serie de procedimientos, como son: ESCENA, CORO, INDIVIDUAL, FINAL, SOLISTA y TODAS. A continuación pasaré a describir cada uno de ellos.

ESCENA: controla toda la actividad y tiene asociados al resto de los procedimientos dependiendo de la opción por la que opte el usuario.

Este crea la pantalla de presentación de las opciones y nos ofrece la posibilidad de elegir una de las opciones. Dependiendo de la opción retorna a un procedimiento u otro.

TODAS: Al elegir la opción 1 se ejecuta este procedimiento, que pone todas las luces a uno, es decir, las enciende, para ello realiza la primitiva SALIDA 255, que corresponde a un número binario. Posteriormente el programa retorna al procedimiento que le fue llamado.

CORO: El funcionamiento es idéntico al procedimiento anterior con la diferencia que la opción elegida es la 3 y el número binario utilizado es 10.

SOLISTA: El funcionamiento es idéntico al procedimiento anterior con la diferencia que la opción elegida es la 2 y el número binario utilizado es 1.

INDIVIDUAL: La dinámica del funcionamiento es idéntica al procedimiento anterior con la diferencia que la opción elegida es la 4 y el número binario utilizado es 1.

Este procedimiento da al usuario la posibilidad de seleccionar la bombilla que quiere encender. Dándole las diferentes salidas.

FINAL: Este procedimiento permite terminar la ejecución del programa.

7.- Funcionamiento

Una vez ejecutado el fichero ESCENA.LOG funciona del siguiente modo:

Aparece el menú principal con sus diferentes opciones y la posibilidad de elegir una de ellas. La acción a realizar será distinta dependiendo de la opción elegida. Las actuaciones a realizar según la opción son:

- OPCIÓN 1: Se encienden todas las luces del escenario y los diodos LED de la tarjeta controladora durante 3 segundos.
- OPCIÓN 2: Se enciende la luz que ilumina al solista y el diodo LED 1 de la tarjeta controladora durante 3 segundos.
- OPCIÓN 3: Se enciende las luces que iluminan al coro y los diodos LED 2,3,4 de la tarjeta controladora durante 3 segundos.
- OPCIÓN 4: Me permite elegir la luz que quiero que se encienda, tecleo el número de la opción y se enciende la luz correspondiente y el diodo LED de la tarjeta controladora que tiene asignado durante 3 segundos y vuelve al menú principal cada vez que se ejecuta una opción.
- OPCIÓN 5: Permite terminar la ejecución del programa y vuelve al sistema operativo MS_DOS.

RÓTULO DE RECLAMO PUBLICITARIO

1.- Descripción de la actividad

Queremos realizar un cartel publicitario, luminoso de modo que sirva de reclamo publicitario, y para ello que encienda y apague las luces que lo formen de modo que atraiga la atención de los posibles clientes.

Hemos elegido un cartel que cuyo texto sea TRAMA 0 que corresponde con el nombre de un negocio de cara al público.

Cada una de las letras del rótulo estará iluminada por una bombilla, de modo que al encenderse cada una de las bombillas se verá iluminada la letra correspondiente.

La secuencia de encendidos y apagados es la siguiente:

- 1.- Se ilumina la T y se espera una instante.
- 2.- Se apaga la T y simultáneamente se enciende la R, y espera un instante.
- 3.- Simultáneamente se apaga la R y se enciende la A, y espera un instante.

- 4.- Se apaga la A y a la vez se enciende la M, y espera un instante.
- 5.- Se apaga la M y se enciende a la vez la A, y espera un instante.
- 6.- Se apaga la A y a la vez se enciende el 0, y se queda ya encendido permanentemente el 0.
- 7.- Se repiten los puntos 1, 2, 3, 4 y 5, quedando encendida permanentemente la bombilla de la A, de modo que al llegar a este punto tendremos encendida la A y el 0.
- 8.- Se repiten los puntos 1, 2, 3 y 4 quedando encendidas las tres últimas bombillas de una forma permanente.
- 9.- Se repiten los puntos 1 2 3 quedando encendidas las cuatro últimas bombillas correspondientes las letras AMA 0
- 10.- Se repiten los puntos 1 y 2 quedando encendido RAMA 0
- 11.- Se enciende la T y ya no se apaga, manteniéndose todo el rótulo iluminado durante un tiempo determinado (5”).
- 12.- Se apagan todas las bombillas del rótulo, quedando completamente oscuro esperando 5”.
- 13.- Se repite esta secuencia otras 2 veces esperando 5” encendido y 5” apagado.
- 14.- Comienza nuevamente el ciclo desde el punto 1 hasta el 13.

2.- Análisis del problema

Cada una de las letras se ilumina con una bombilla. De este modo tendremos seis bombillas B1 B2 B3 B4 B5 B6 que conectaremos a las salidas S1 S2 S3 S4 S5 S6.

Por tanto sólo nos queda definir un programa que resuelva la compleja secuencia que nos hemos propuesto.

Para encender la bombilla B1 hay que activar la salida S1, para la cual la orden es SALIDA 1. Para las demás el código binario no coincide con el código decimal, por lo que para las demás salidas debemos utilizar la conversión ya explicada en los preliminares.

Por ejemplo:

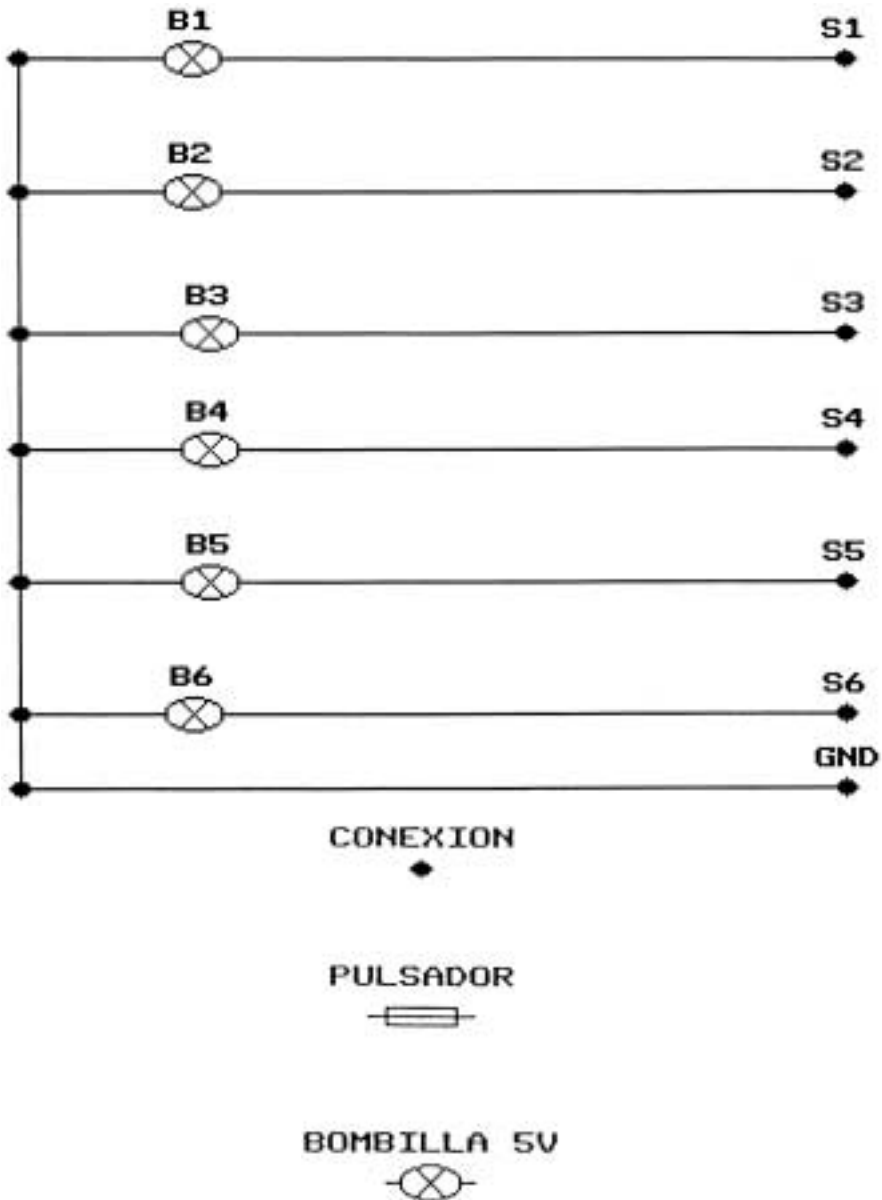
- para encender la B1 escribiré SALIDA 1
- para encender la B2 y apagar la B1 escribiré SALIDA 33 , que supone apagada la B1 y encendida la B2.

Una vez que tengo fijadas algunas bombillas, por ejemplo la A 0, la forma de saber qué salida he de escribir es la siguiente: La A y el 0 son las salidas S5 y S6, para que aparezcan iluminadas he de escribir SALIDA 48, que resultan de sumar $24+25$. Por ello si además quiero que se encienda la T del punto 8, escribiré SALIDA 49.

3.- Material

- Seis bombillitas de 5 VCC, y unos papeles de colores transparentes en los cuales rotular las letras que forman la palabra TRAMA 0.
- Equipo informático común a los demás trabajos
- Piecerío para el soporte donde colocar el rótulo

4.- Conexionado



5.- Programación

A continuación aparece el listado del programa

PARA TRAMA0	SALIDA 52
SALIDA 1	ESPERA 10
ESPERA 10	SALIDA 56
SALIDA 2	ESPERA 10
ESPERA 10	SALIDA 57
SALIDA 4	ESPERA 10
ESPERA 10	SALIDA 58
SALIDA 8	ESPERA 10
ESPERA 10	SALIDA 60
SALIDA 16	ESPERA 10
ESPERA 10	SALIDA 61
SALIDA 32	ESPERA 10
ESPERA 10	SALIDA 62
SALIDA 33	ESPERA 10
ESPERA 10	SALIDA 63
SALIDA 34	ESPERA 10
ESPERA 10	REPITE 3 [CICLO]
SALIDA 36	TRAMA0
ESPERA 10	FIN
SALIDA 40	
ESPERA 10	
SALIDA 48	PARA CICLO
ESPERA 10	SALIDA 0
SALIDA 49	ESPERA 60
ESPERA 10	SALIDA 63
SALIDA 50	ESPERA 60
ESPERA 10	FIN

Podemos ver que este programa sólo tiene dos procedimientos. Ambos muy sencillos, pero que realizan una secuencia muy curiosa y altamente instructiva.

En procedimiento TRAMA0 se van encendiendo y apagando las bombillas conectadas a las salidas correspondientes. Lo que más llama la atención es la forma de ir efectuando las salidas.

SALIDA 1
SALIDA 2
SALIDA 4
SALIDA 8

Lo cual se entiende muy bien si pensamos en el código binario; pues para que se encienda la primera salida he de escribir el nº correspondiente a 20 que resulta ser = 1. Para hacer lo mismo con la salida 2 he de efectuar 21 que es 2. Para sacar la salida 6 he de efectuar $25 = 32$. Para que se mantengan encendidas la S6 y la S1, deberé efectuar la suma $20 + 25 = 33$. Por ello escribiré .

SALIDA 33.

Por todo ello hay una forma más rápida de hacer el programa que se propone en las actividades de perfeccionamiento.

6.- Funcionamiento

El funcionamiento es muy sencillo una vez teclado y cableado según el conexionado se irán encendiendo las luces con la secuencia establecida. Si algo no funciona en el prototipo que sí funciona en el panel indicador frontal, comprobaremos las conexiones.

7.- Actividades de perfeccionamiento

Se puede proponer a los alumnos que realicen de nuevo el programa pero utilizando la primitiva potencia.

ASCENSOR DE TRES PISOS

1.- Descripción de la actividad

Se trata de realizar un ascensor que tenga tres pisos, de modo que al llamarlo desde cualquier piso el ascensor se dirija a ese piso, deteniéndose a su llegada. Será tal que tendrá la posibilidad de ser llamado desde los diferentes pisos por medio de pulsadores.

2.- Análisis del problema

En primer lugar debemos diseñar y ejecutar el modelo de ascensor, con tres plantas, situando en ellas los pulsadores de llamada y también los pulsadores que detecten la posición del ascensor, cuando llegue a la planta correspondiente.

En cuanto a la lógica a seguir en este caso es más compleja, hemos procedido de la forma siguiente:

En primer lugar debemos saber en qué piso se encuentra el ascensor. Es sencillo pues estará en el piso correspondiente al pulsador que esté encendido y por ello tenga el valor lógico 1.

Una vez que sabemos en qué piso se encuentra, las opciones quedan reducidas. Por ejemplo si estoy en el piso 3º sólo puedo bajar al 2º o al 1º. Del mismo modo si me encuentro en el 2º piso puedo o bien subir o bien bajar un piso. Si me encuentro en el primer piso sólo puedo subir uno o dos pisos. El que haga cada una de esas opciones lo iremos indicando con los pulsadores de llamada.

De una forma más resumida diríamos lo siguiente:

- P1. Pulsador de estado piso 1
- P2. Pulsador de estado piso 2
- P3. Pulsador de estado piso 3
- P4. Pulsador de llamada piso 1
- P5. Pulsador de llamada piso 2
- P6. Pulsador de llamada piso 3

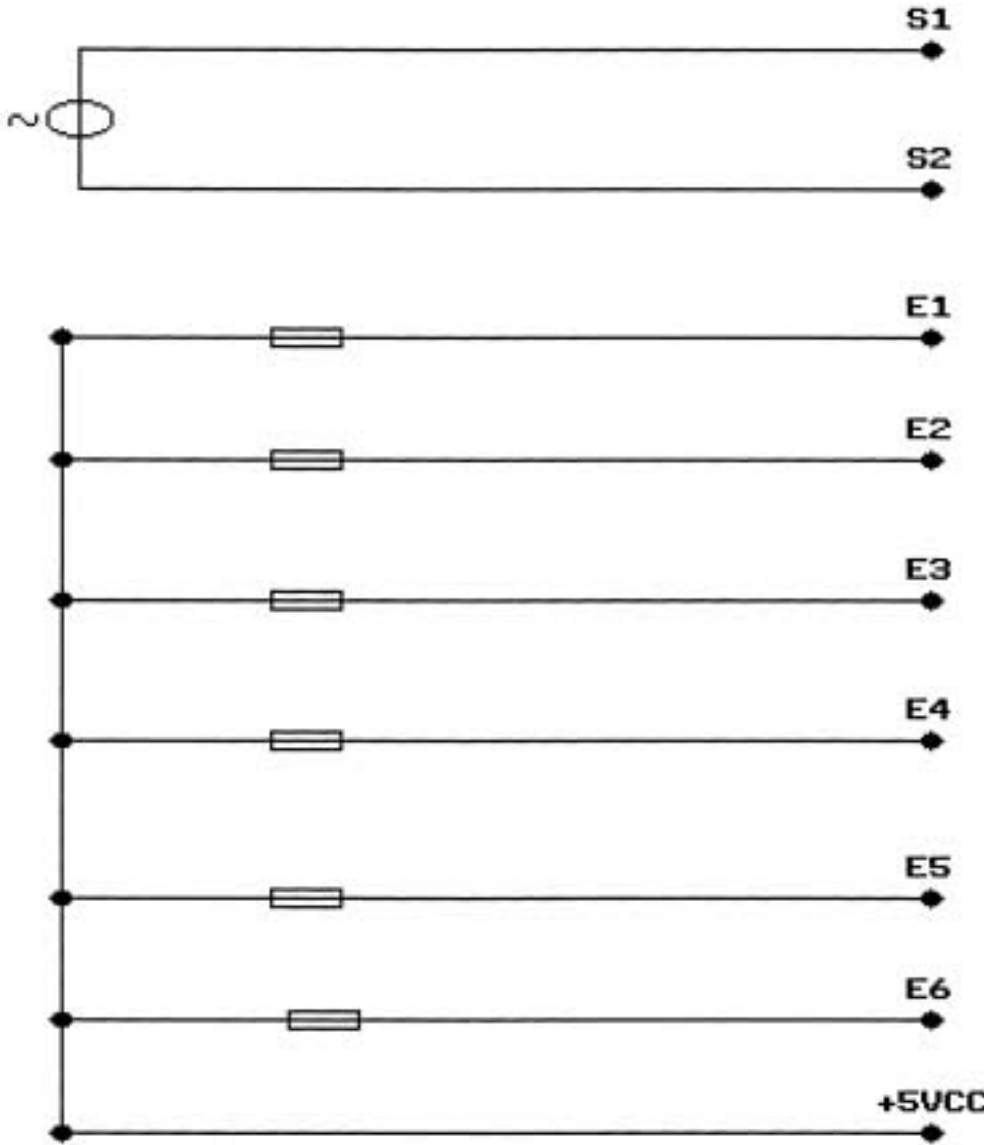
P1	P2	P3	P4	P5	P6	MOTOR	Nº
1	0	0	0	1	0	ON D	1
1	0	0	0	0	1	ON D	2
0	1	0	1	0	0	ON I	3
0	1	0	0	0	1	ON D	4
0	0	1	1	0	0	ON I	5
0	0	1	0	1	0	ON I	6

3.- Material necesario

El material necesario para la realización del trabajo será:

- Equipo informático completo
- Piecerío diverso para la realización del prototipo, en cuanto a su base estructural.
- Seis pulsadores
- Un motor con reductora
- Un mecanismo piñón-cremallera.

2.- Conexionado



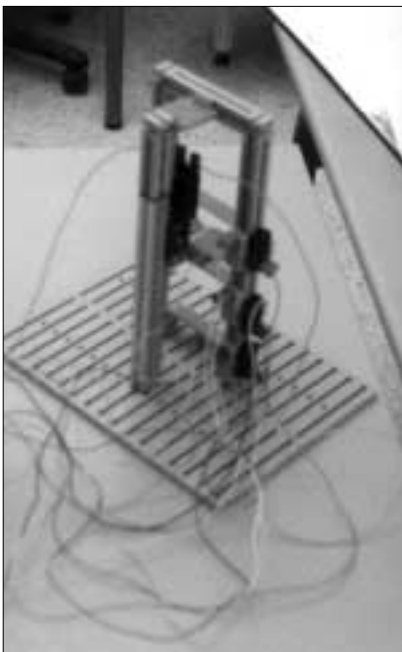
CONEXION



PULSADOR



BOMBILLA 5V



5.- Programación

Los procedimientos necesarios para controlar un ascensor de tres plantas son los siguientes:

```
PARA ASCENSOR
SI 1 = SD 8 [PRIMER]
SI 1 = SD 7 [SEGUNDO]
SI 1 = SD 6 [TERCERO]
SI TECLA? [SALIDA 0 ALTO]
ASCENSOR
FIN
```

PARA PRIMER

SI 1 = SD 3 [BAJADA31 ALTO]

SI 1 = SD 4 [BAJADA21 ALTO]

FIN

PARA SEGUNDO

SI 1 = SD 3 [BAJADA32 ALTO]

SI 1 = SD 5 [SUBIDA12 ALTO]

FIN

PARA TERCERO

SI 1 = SD 5 [SUBIDA13 ALTO]

SI 1 = SD 4 [SUBIDA23 ALTO]

FIN

PARA SUBIDA12

M1 «I SI 1 = SD 4 [M1 «P ALTO]

SUBIDA12

FIN

PARA SUBIDA23

M1 «I SI 1 = SD 3 [M1 «P ALTO]

SUBIDA23

FIN

PARA SUBIDA13

M1 «I SI 1 = SD 3 [M1 «P ALTO]

SUBIDA13

FIN

PARA BAJADA31

M1 «D SI 1 = SD 5 [M1 «P ALTO]

BAJADA31

FIN

PARA BAJADA32

M1 «D SI 1 = SD 4 [M1 «P ALTO]

BAJADA32

FIN

PARA BAJADA21

M1 «D SI 1 = SD 5 [M1 «P ALTO]

BAJADA21
FIN

Como podemos comprobar este programa está compuesto por muchos procedimientos:

ASCENSOR
PRIMERO
SEGUNDO
TERCERO
SUBIDA12
SUBIDA23
SUBIDA13
BAJADA31
BAJADA32
BAJADA21

El primer procedimiento, ASCENSOR es el que detecta el piso desde el que estamos llamando, y en función de ese piso los demás procedimientos se encargan de realizar los movimientos de la plataforma vía motor. Resaltar que mientras no se pulse ningún piso se estará ejecutando ASCENSOR y lo que hará será llamarse a sí mismo lo que equivale a una espera hasta que se pulse una llamada.

El resto de procedimientos es una serie de condicionales del mismo tipo que los del trabaja PUERTA DE GARAJE AUTOMÁTICA.

En todos ellos el propio nombre del procedimiento indica lo que realiza éste.

6.- Funcionamiento

El funcionamiento del programa será tal y como se ha ido describiendo: una vez situada la plataforma del ascensor en una planta determinada, se deberá iluminar el diodo led de entradas correspondiente. Si no es así verificar las conexiones.

Cuando pulse desde el piso al que quiero que vaya la plataforma, se debe iluminar primero el diodo led de entradas correspondiente al piso desde el que he llamado y a la vez ponerse en movimiento la plataforma que se detendrá cuando llegue al piso especificado.

7.- Actividades de perfeccionamiento

Con esta actividad como base aconsejamos realizar trabajos más complejos que enciendan pilotos de señalización en el ascensor cuando esté subiendo que nos indiquen dónde se encuentra.

También se puede hacer un trabajo con el panel de mandos dentro de la caja del ascensor, de modo que suba y baje con él y una vez dentro el posible usuario decida a qué piso ir, como en los ascensores reales.

No obstante, esta actividad en sí, tal y como la hemos descrito ya es bastante compleja para la edad en la que pretendemos trabajar; en cursos posteriores sí serán interesantes mejoras del trabajo.

PUERTA DE GARAJE AUTOMÁTICA

1.- Descripción de la actividad

Se trata de diseñar y construir un puerta de garaje automática que funcione controlado por el ordenador.

El automatismo será tal que:

- Siempre que se acerque un coche que está fuera del garaje la puerta se debe abrir y cuando haya pasado totalmente el vehículo se cerrará automáticamente.
- Siempre que un coche que está dentro del garaje salga acercándose a la puerta se deberá abrir, cerrándose una vez que el coche la haya rebasado completamente.

2.- Análisis del problema

En primer lugar vamos a necesitar construir el modelo de puerta de garaje con piecerío de conexión fácil. La puerta la abriremos con un mecanismo de piñón-cremallera activado por un motor que irá conectado a las salidas a motor S1 S2.

El control del mecanismo, parte más compleja del problema, se resolverá en base a 4 pulsadores. Estos enviarán sus señales lógicas a través de las entradas digitales E1 E2 E3 E4 al ordenador. Dos de los pulsadores actuarán de final de carrera, para que se pare la puerta una vez que esté totalmente abierta o totalmente cerrada.

Otros dos de los pulsadores actuarán como detectores de la proximidad de un vehículo, pues cuando este se acerca los pulsa durante un momento, señal que sirve para que se desencadenen las salidas y la puerta se abra o se cierre.

P1 final de carrera que indica puerta cerrada.

P2 final de carrera que indica puerta abierta totalmente.

P3 pulsador del lado de la calle.

P4 pulsador del lado de dentro del garaje.

Las dos secuencias que se pueden dar son las siguientes:

Secuencia para el coche que va a entrar al garaje

P1	P2	P3	P4	MOTOR	Nº
1	0	0	0	OFF	1
1	0	1	0	ON D	2
0	0	0	0	ON D	3
0	0	0	1	OFF	4
0	1	0	1	ON I	5
0	0	0	0	ON I	6
1	0	0	0	OFF	7
0	0	0	0	OFF	8

Secuencia para el coche que va a salir del garaje

P1	P2	P3	P4	MOTOR	Nº
1	0	0	0	OFF	1
1	0	0	1	ON D	2
0	0	0	0	ON D	3
0	0	0	1	OFF	4
0	1	1	0	ON I	5
0	0	0	0	ON I	6
1	0	0	0	OFF	7
0	0	0	0	OFF	8

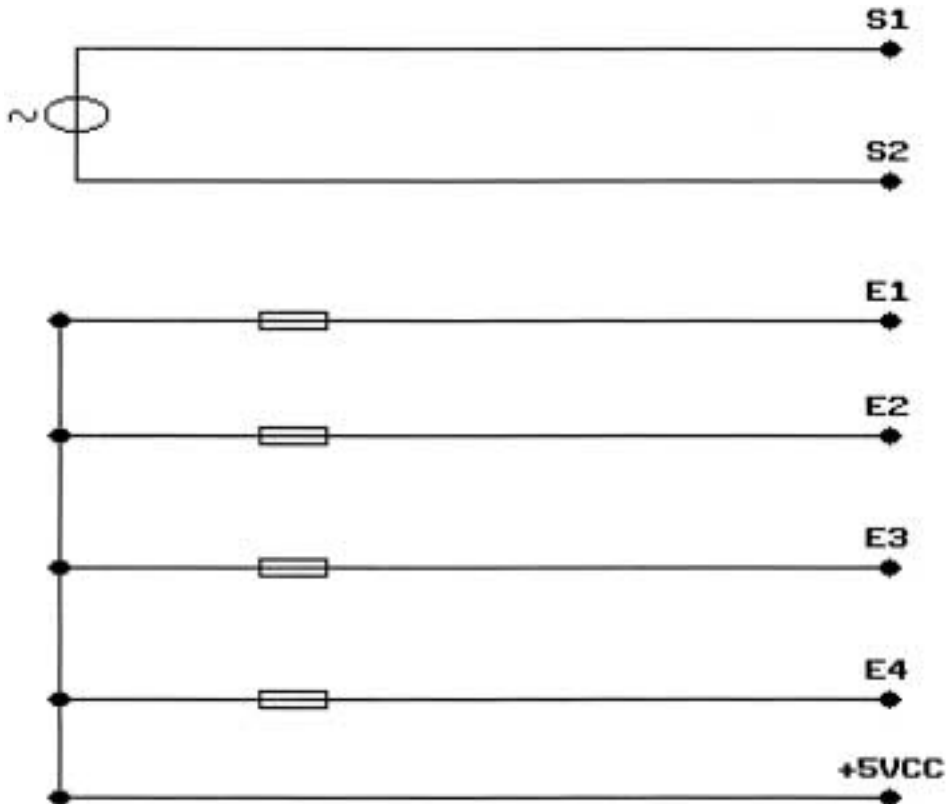
Como podemos ver se trata de una lógica secuencia, pues con todos los pulsadores en 0 unas veces gira el motor a la derecha y otras veces gira a la izquierda, dependiendo de la secuencia anterior. En la tabla hemos usado el siguiente código:

- 1 PULSADO
- 0 SIN PULSAR
- ON D ACTIVADO GIRANDO A LA DERECHA
- ON I ACTIVADO GIRANDO A LA IZQUIERDA
- OFF APAGADO

3.- Material necesario y conexionado

- Equipo informático completo común a los demás trabajos.
- Cuatro pulsadores
- Motor de 5 VCC con reductora.
- Mecanismo piñón-cremallera.
- Piecerío diverso para la construcción del prototipo de la puerta de garaje.

4.- Conexionado



CONEXION



PULSADOR



BOMBILLA 5V



5.- Programación

Los procedimientos necesarios para controlar una puerta de garaje son los siguientes:

PARA GARAJE

SALIDA 0

SI 1 = SD 1 [ABRIR]

SI 1 = SD 2 [CERRAR]

GARAJE

FIN

PARA FINAL

SALIDA 0

CIERRATODO

ADIOS

FIN

```
PARA CIERRA1
M1 «I
SI SD 1 = 1 [M1 «P GARAJE ALTO]
CIERRA1
FIN
```

```
PARA ABRE1
M1 «D
SI SD 2 = 1 [M1 «P GARAJE ALTO]
ABRE1
FIN
```

```
PARA CERRAR
SI SD 3 = 1 [CIERRA1]
SI SD 4 = 1 [CIERRA1]
CERRAR
FIN
```

```
PARA ABRIR
SI SD 3 = 1 [ABRE1 ALTO]
SI SD 4 = 1 [ABRE1 ALTO]
ABRIR
FIN
```

Este programa está formado por los siguientes procedimientos:

```
GARAJE
FINAL
CIERRA1
ABRE1
CERRAR
ABRIR
```

Lo más importante del programa no son las instrucciones novedosas, sino la forma de trabajar con las pocas instrucciones que manejamos.

Para entender mejor el programa, en primer lugar vamos a describir la lógica seguida para resolverlo:

En primer lugar detectamos si la puerta está totalmente abierta (SD 2 =1) o totalmente cerrada (SD 1 = 1). Si la puerta está abierta sólo se puede cerrar; y viceversa, si está totalmente cerrada sólo se puede abrir. Por ello una vez detectada la

posición de la puerta el programa hará el giro en un sentido o en el otro. Todo ello lo hace en función de que accionemos los otros pulsadores.

GARAJE detecta la posición de la puerta. Permite también salir del programa una vez que pulsemos una tecla. Si no está pulsado ninguno de los finales de carrera, el ordenador espera hasta que eso sucede mediante un procedimiento recursivo, llamándose a sí mismo.

ABRIR en caso de que cualquiera de los dos pulsadores se active llama al procedimiento ABRE1.

CERRAR en caso de que cualquiera de los dos pulsadores se active llama al procedimiento CIERRA1.

ABRE1 hace girar el motor a la derecha hasta que se llegue al final de carrera correspondiente, entonces SD 2 vale 1 y se devuelve el control al garaje deteniéndose el motor. El nombre no puede ser ABRE pues abre es palabra reservada por el logo. Es una primitiva de LOGO.

CIERRA1 hace girar el motor a la izquierda hasta que se llegue al final de carrera correspondiente, entonces SD 1 vale 1 y se devuelve el control al garaje y se detiene el motor.

FINAL lo único que hace es poner todas las salidas a cero, desactivando todas las posibles salidas.

6.- Funcionamiento

El funcionamiento del prototipo es sencillo. Con la puerta cerrada aproximamos un coche tocando el pulsador. La puerta debe abrirse. Mientras tanto se enciende un diodo rojo, el S1 o el S2.

Cuando se haya abierto totalmente, el final de carrera indicador se activa, iluminándose el diodo correspondiente, y deteniéndose el motor. Una vez que pasa el coche totalmente toca el pulsador de salida y la puerta comienza a cerrarse. El indicador de entrada correspondiente al pulsador activado se habrá iluminado un instante, y ahora cambiará el diodo rojo, S1 o S2, quedando al revés que estaba mientras se abría la puerta. Cuando se cierra la puerta del todo acciona el final de carrera correspondiente deteniéndose.

Es importante comprobar las conexiones, pues a veces nos dan problemas por falta de contacto. Si se encienden los diodos de salidas y NO los motores o bombilla se debe a errores de conexionado. Si al accionar los pulsadores no se

encienden o apagan los diodos led verdes de las entradas, significa un mal conexionado.

Si se usan conmutadores pulsadores en lugar de interruptores pulsadores, puede suceder que pulsando se apaguen y sin pulsar estén encendidos los diodos led correspondientes. En este caso se mantiene conectado el común y se cambia el otro contacto por el que está libre.

DETECTOR DE TRÁFICO

1.- Descripción de la actividad

Consiste en simular el tráfico entre dos puntos determinados **A** y **B**, calcular el número de coches que pasan por dichos puntos y si la velocidad es superior a una especificada en el programa se deberá encender una luz que esta situada en el punto **B**. El sentido de circulación va desde **A** a **B**.

2.- Análisis del problema

El análisis del problema es la parte mas importante del proyecto, ya que si se diseña mal difícilmente puede funcionar.

No hay que olvidar que el análisis del problema hay que estudiarle desde dos puntos de vista uno a nivel físico y otro a nivel lógico.

Para llevar a cabo esta actividad se debe disponer de 1 bombilla (**B1**) para simular la luz que debe encenderse cuando la velocidad es excesiva, dicha luz se encuentra situada en el punto **B** excesiva. El coche pasa primero por el punto **A** y luego por el punto **B**.

Dos pulsadores (**SD P** y **SD 2**) uno en cada punto que detectan el paso del coche por ellos.

La bombilla lleva asociada dos cables de colores, uno se conecta a la entrada **GND** de la tarjeta y el otro a la salida que tiene asignada en la tarjeta controladora.

Los pulsadores tienen dos terminales fáciles, uno conectado a su entrada correspondiente en la controladora y el otro a la salida de 5 voltios.

Las bombillas solo pueden tener dos estados posibles, encendido que toma el valor 1 y apagado que toma el valor 0.

En la siguiente tabla se muestran las posibles combinaciones de los estados de los pulsadores y la bombilla.

<u>PULSADORES</u>		<u>BOMBILLAS</u>	
SD1	SD2	B1	
1	1	0	VELOCIDAD NORMAL
1	1	1	VELOCIDAD INADECUADA

Puntos:A B

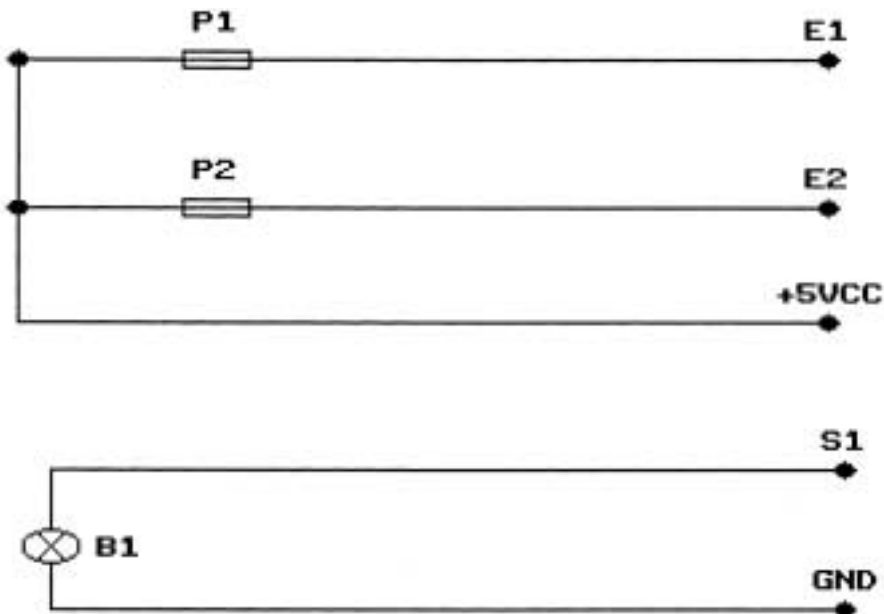
3.- Material necesario

El material necesario para llevar a cabo esta actividad es:

- 1 bombilla de 5 voltios, con 1 carcasa de color.
- 6 cablecillos de colores y con terminales fáciles.
- Soporte de apoyo para simular un escenario.
- 2 pulsadores.

Se aconseja realizar el montaje en el aula-taller por los alumnos, sino se dispone de un equipo de ficher-technik.

4.- Conexionado



CONEXION



PULSADOR



BOMBILLA 5V



5.- Programación

En este apartado se expondrá el listado de la aplicación que se encuentra en el fichero **DENSILOG** con sus diferentes procedimientos, que detallamos a continuación:

PARA TRÁFICO
HAZ «TOT 0
INICIO
FIN


```
PARA FINAL1
PONCURSOR [1 7]
ESCRIBE FRASE [EL NÚMERO TOTAL DE COCHES ES:]
PONCURSOR [30 7]
ES: TOT
ESPERA 30
CIERRATODO
ADIOS
FIN
```

```
PARA INICIO
HAZ «CON1 0
SI 1= SD 1 [HAZ «TOT :TOT + 1 CONTAR]
SI TECLA? [FINAL1]
INICIO
FIN
```

```
PARA CONTAR
HAZ «CON1 :CON1 + 1
SI 1 = SD 2 [BOMBI]
CONTAR
FIN
```

```
PARA BOMBI
SI 40 < :CON1 [INICIO]
SALIDA 1
ESPERA 20
BT
PONCURSOR [1 5]
ESCRIBE FRASE [VELOCIDAD INADECUADA]
SALIDA 0
BT
INICIO
FIN
```

6.- Breve comentario explicativo del programa

El programa se encuentra en un fichero llamado DENS.LOG que está formado por una serie de procedimientos, como son las siguientes:

TRAFICO: Este procedimiento es el ejecutable, el cual solo se utiliza para inicializar una variable a cero y llama otro procedimiento llamado **INICIO**.

INICIO: Este procedimiento detecta si **SD1**, es decir el primer pulsador está a 1 en cuyo caso aumenta el número de coches y llama a otro procedimiento **CONTAR**. Todo ello en es un procedimiento recursivo, es decir que se llama así mismo.

También detecta si se pulsa una tecla desde teclado para finalizar la ejecución del programa.

CONTAR: En él se va contando el tiempo transcurrido desde que $SD1=1$ hasta que $SD2$ pasa a valer 1. en ese instante llama a otro procedimiento llamado **BOMBI**. Para ello se utiliza un procedimiento recursivo.

BOMBI: Este procedimiento compara si la velocidad es inadecuada en cuyo caso mostrará por pantalla el mensaje siguiente «**VELOCIDAD INADECUADA**». Pone a cero todas las salidas y retorna al procedimiento **INICIO**.

7.- Funcionamiento

El ordenador espera a que pase un coche por el punto **A**, transcurrido un tiempo aparece un coche con lo cual el contador de coches se pone a uno, al instante pasa por el punto **B**, se ha contabilizado el tiempo que tarda en llegar al punto **B** si es menor que el especificado en el programa se encenderá la bombilla situada en el punto **B**.

8.- Actividades de perfeccionamiento

Las actividades de perfeccionamiento pueden ser propuestas por el profesor y por los propios alumnos relacionadas con la actividad principal.

Las actividades propuestas por el profesor son entre otras:

- Igual a la actividad planteada añadiéndole al enunciado que hay una dirección y dos sentidos de circulación.
- El enunciado igual que la actividad al que se le debe añadir la posibilidad que entre los dos puntos circulen dos coches a la vez en uno y otro sentido indistintamente.

III. GUÍA DE REFERENCIA TÉCNICA

INTRODUCCIÓN

Este apartado es una guía de instalación del software necesario para la realización del proyecto.

Esta compuesto por una serie de programas como son:

- ACTILOGO
- Programas de control de la tarjeta controladora.
- Los programas de cada una de las actividades.
- El programa de instalación.

REQUERIMIENTOS HW Y SW

Para desarrollar la aplicación es necesario un ordenador compatible con las siguientes características, a nivel hardware y software.

Hardware

Para llevarse a cabo la aplicación es imprescindible los siguientes equipos:

- Un ordenador.
- Una tarjeta controladora.
- Un sistema tecnológico de montaje.

- La controladora.

La controladora va a ser el dispositivo que nos permita convertir las señales en voltaje que nos da el ordenador en el puerto paralelo, en salidas en tensión e intensidad, y también recibir instrucciones del exterior. Por ello este dispositivo está diseñado para su utilización como controlador del accionamiento de pequeños motores de corriente continua y la captación de datos a través de un ordenador compatible, por medio de la salida de la impresora.

La controladora está caracterizada por disponer de un conector externo de fácil acceso en el que están presentes:

- 8 salidas digitales de valor 0 -5 voltios y corriente total de un amperio.
- 2 salidas analógicas, cuyo estado se regula mediante programa entre 0 y 16 valores comprendidos entre 2 y 6.5 voltios. Estas salidas pueden proporcionar una corriente máxima de 1 amperio.
- 10 entradas digitales, cuyo valor lógico es 0 cuando se conectan a 0 voltios y 1 cuando se conectan a 5 voltios.
- 4 entradas analógicas capaces de leer 256 valores de tensión diferentes entre 0 y 5 voltios.
- 1 conexión a 5 VCC.
- 1 conexión a 0 VCC o masa.

El equipo precisa una alimentación de 12 voltios de corriente continua y un consumo en reposo de 200 mA. La fuente de alimentación incluida en el equipo proporciona 12 V y 2 A.

La controladora también tiene una serie de indicadores de estado, mediante diodos led rojos y verdes, en el panel frontal:

- Encendido. Iluminado indica que la alimentación está presente.
- Activo. Programa está activa si está iluminado
- Entradas, de 1 a 10. Su encendido indica que la correspondiente entrada digital está activada.
- Salidas. Las 2 primeras representan el estado de las 2 salidas analógicas. Las 8 siguientes corresponden al de las 8 salidas digitales.

- El ordenador.

Para que el programa funcione correctamente es necesario disponer de un ordenador con las características siguientes:

- Disponer de disco duro.
- Una memoria RAM de 1MB.
- Una tarjeta gráfica.
- Una disketera.
- Teclado.

- El sistema tecnológico de montaje.

El sistema tecnológico de montaje puede ser muy diverso, si bien ha de tener ciertas características comunes.

Debemos disponer de motores de 5 V CC. y lamparitas de 5 VCC; pues son los valores que nos da la controladora a su salida.

Además es conveniente disponer de elementos mecánicos tales como:

- Reductoras diversas.
- Tornillo-sinfín.
- Piñón-cremallera.
- Excéntricas.
- Poleas.
- Correas.
- Engranajes diversos.

Como material eléctrico además del señalado es muy conveniente disponer de:

- Interruptores o conmutadores de pulsador. Si se usa un conmutador, dejando libre una terminal que no sea el común se puede usar como interruptor.
- Cablecillo de colores.
- Terminales de conexión fácil.
- Electroimanes.
- Relés.
- Diodos led
- Resistencias electrónicos.
- LDR.
- Termistencias.

Estos últimos elementos para cuando usemos salidas analógicas.

Como material estructural es bueno disponer de perfiles de aluminio taladrados, de ejes, de contrachapado, de columnas de plásticos y de metal, etc.

A ser posible una forma de no dedicar excesivo tiempo a los montajes, además de abaratar los costes, supone la utilización de kits de piezas desmontables, tipo LEGO, TENTE, ETC. Nosotros hemos usado el piecerío de Fischer-Tecknic.

Software

Las características a nivel software son las siguientes:

- Un sistema operativo versión 5.0 o superiores.
- El programa ACTILOGO.
- Los programas de control de la tarjeta controladora.

INSTALACIÓN DE LA APLICACIÓN

INSTALACIÓN DEL ACTILOGO

Para instalar **ACTILOGO** en el disco duro de su ordenador, si se encuentra éste en un disco, siga los pasos que se indican a continuación:

- Se crea un directorio **ACTILOGO** si no ha instalado la tarjeta controladora con anterioridad pues esta lo crea automáticamente. Para ello teclee **MD ACTILOGO** y pulse <ENTER>.
- Introduzca el disco que contiene el **ACTILOGO** en la unidad A: o B: dependiendo del tipo de unidad asignada, para ello suponemos que es la unidad A.
- Teclee **XCOPY A:\ C:\ACTILOGO /S** y a continuación <ENTER>. Con lo cual ya ha quedado instalado el **ACTILOGO**.

INSTALACIÓN DE LA TARJETA CONTROLADORA

Para instalar **LA TARJETA CONTROLADORA** en el disco duro de su ordenador, si se encuentra éste en un disco, siga los pasos que se indican a continuación:

- Introduzca el disco que contiene **LA TARJETA CONTROLADORA** en la unidad A: o B: dependiendo del tipo de unidad asignada, para ello suponemos que es la unidad A.
- Seleccione la unidad A como área de trabajo. Teclee A: y pulse <ENTER>.
- Teclee **INSTALAR 1** y a continuación <ENTER>. Por último aparece en pantalla instalación finalizada.

INSTALACIÓN DEL PROGRAMA DE APLICACIÓN

Una vez instalado el **ACTILOGO** y la tarjeta controladora, ahora es necesario instalar el programa de aplicación. Para ello se crea un fichero de tipo **BAT** que se encuentra en un disco junto con los programas de aplicación.

Para instalar la aplicación en el disco duro se procederá de la siguiente manera:

- Introduzca el disco que contiene LOS PROGRAMA DE INSTALACION en la unidad A: o B: dependiendo del tipo de unidad asignada, para ello suponemos que es la unidad A.
- Seleccione la unidad A como área de trabajo. Teclee A: y pulse <ENTER>.
- Teclee **INSTALAR** y a continuación <ENTER>. Con lo cual ha quedado instalada la aplicación y se puede ejecutar.

EJECUCIÓN DEL PROGRAMA

Una vez instalado el programa. Para ejecutarlo teclea **PROYECTO** y a continuación <ENTER>. Con lo cual hemos entrado en el editor de comandos del ACTILOGO y ha aparecido su identificativo «?».

Para ejecutar un comando después del comando siempre se debe pulsar <ENTER>.

Posteriormente, una vez encendida la tarjeta controladora tenemos que cargar los programas de esta, para ello teclee

?carga «controla

Una vez cargados los programas la tarjeta controladora, procederemos a cargar el programa de la aplicación que desee. Se puede cargar de dos formas distintas:

- Una del disco duro, tecleando para ello.

?carga «semap

Siendo semap el nombre del fichero donde se encuentra la aplicación.

- Una segunda opción cargándolo del disco, mediante la siguiente primitiva del ACTILOGO.

?carga «A:semap

Una vez cargado el programa se teclea el nombre del programa para su ejecución, siguiendo el ejemplo que nos ocupa se procede de la siguiente forma:

?semap

Una vez ejecutado el programa, controlará el mecanismo para el cual ha sido programado.

IV. BIBLIOGRAFÍA

- “*Controladora para ordenador. Manual de Usuario*” de Alberto de Miguel Álvarez.
- “*Tecnología 3º ESO*”. Ed. Mc Graw Hill.
- “*Manual de LOGO*”. Software de Base.