

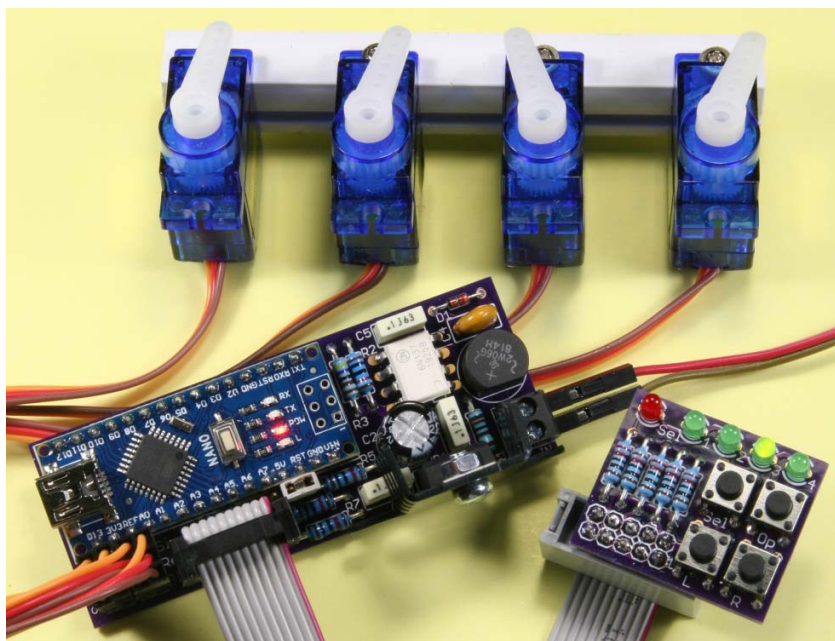
## Construya su propio decodificador Quad Servo DCC - Parte 1

### Introducción

Después de decidir usar servos para mover los desvíos en una pequeña maqueta HO que estoy planeando construir, las pruebas con varios decodificadores comerciales de accesorios DCC fueron bastante decepcionantes. Quería poder configurar el movimiento de cada servo de forma fácil y precisa a mano, antes de su instalación en la maqueta, y luego poder ajustar los movimientos del servo una vez instalado en su lugar.

Encontré las prestaciones ofrecidas por los decodificadores comerciales bastante incómodas de usar, especialmente una vez que los servos y el decodificador estaban en su lugar debajo del tablero de la maqueta, y parecía bastante difícil configurar el servo con precisión. De ahí que, inspirado en el trabajo de Geoff Bunza en decodificadores de accesorios basados en torno a los módulos Arduino, decidí construir el mío propio con la capacidad de configurar y operar cuatro servos. El resultado final, exitoso, es un decodificador de accesorios DCC que puede ser construido por cualquier persona con habilidades razonables de soldadura y los conocimientos básicos de informática para cargar software en un módulo Arduino, por un costo de menos de \$ 20, incluidas placas de circuito impreso de diseño personalizado.

El decodificador Quad Servo DCC (QSDD) en realidad consta de dos módulos separados: el decodificador en sí mismo, que se conecta a la vía de la maqueta y acciona hasta cuatro servos asociados, y un pequeño teclado desmontable que se conecta al decodificador y se usa para ajustar manualmente el ángulo del servo y para establecer su velocidad de movimiento.



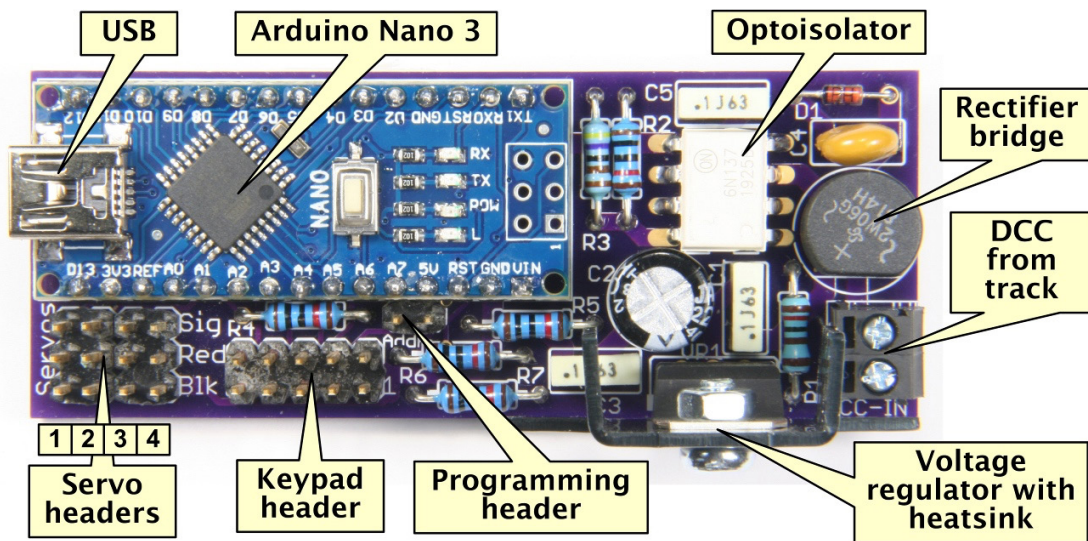
*Todas las fotografías del autor*

El teclado también se puede utilizar para operar los servos a mano, utilizando los pulsadores instalados. Para mayor facilidad, además de poder conectarse directamente al decodificador, el teclado puede, alternativamente, conectarse a través de un cable plano de hasta 40 pulgadas de largo (1 metro), lo que le permite ver y configurar sus desvíos cuando el decodificador y los servos están fuera de la vista, bajo la maqueta.

Tenga en cuenta que, una vez configurado, no es necesario mantener el teclado enchufado. Si lo desea, puede desconectarlo completamente de la placa decodificadora. Seguidamente, el decodificador moverá los servos para operar sus desvíos en respuesta a los comandos DCC estándar enviados a la vía desde su central, utilizando controladores de mano o software adecuado (como [JMRI Panel Pro](#) o mi propia aplicación [A-Track](#)) ejecutándose en su computadora.

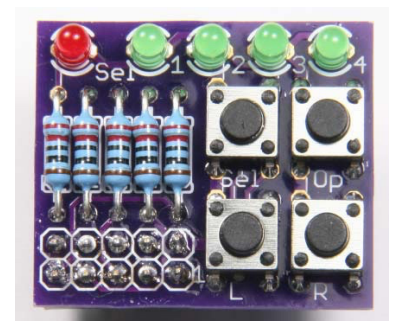
### Estructura y funcionamiento

La conexión de la placa decodificadora a la vía de la maqueta a través del bloque de terminales DCC-IN proporciona alimentación al resto de los circuitos a través del puente rectificador y del regulador de voltaje de + 5V. La señal DCC, que lleva paquetes de comando desde su central, también se lleva al Arduino Nano 3 a través de un chip optoacoplador 6N137, un dispositivo que protege el microprocesador y los circuitos asociados del un voltaje de la vía DCC potencialmente destructivo (aproximadamente 14 voltios CA).



Cuatro cabezales de tres pines proporcionan conexión a pequeños servos analógicos como el Tower SG90, y un bloque de 10 pines en doble fila permite la conexión separada de la placa de teclado, cuando sea necesario, para configurar las características operativas del servo. También hay un pequeño cabezal de conexión de programación –de dos pines– para instalar un puente (*jumper*) cuando se desea establecer direcciones DCC para cada uno de los servos adjuntos. El puente se puede “aparcar” en uno de los pines del cabezal durante el uso normal del decodificador.

El teclado que se conecta al bloque de 10 pines en doble fila incluye cuatro pulsadores etiquetados como **Se1** (Seleccionar), que se utiliza al configurar los movimientos del servo, **Op** (Operate) que selecciona qué servo controlar, **L** (izquierda) y **R** (derecha) que hacen que el servo seleccionado se mueva. También hay cinco diodos emisores de luz (LED), uno de color rojo **Se1** (Seleccionar) y cuatro verdes marcados como **1**, **2**, **3** y **4**, que le muestran qué servo está seleccionado e indica la operación en curso. Los detalles del uso del teclado se cubrirán en la Parte 2 de este artículo.



### Construyendo el Decodificador Quad Servo DCC

Si desea construir uno o más Decodificadores DCC Quad Servo (QSDD), hay un par de placas de circuito impreso (PCB) disponibles en OSH Park, una pequeña empresa ubicada en el lago Oswego, Oregón. Suministran PCB en múltiplos de tres placas, tres PCB de decodificador por US \$ 15,45 y tres PCB de teclado por US \$ 5,25, incluido el envío gratuito a cualquier lugar del mundo.

Puedes ver lo que OSH Park puede ofrecer, y echar un vistazo a la pareja de placas PCB, siguiendo estos enlaces para el codificador QuadServo-DCCD ([https://oshpark.com/shared\\_projects/CjngC3qb](https://oshpark.com/shared_projects/CjngC3qb)) y QuadServo- PCB del teclado ([https://oshpark.com/shared\\_projects/7ATX5aqB](https://oshpark.com/shared_projects/7ATX5aqB)).

Si desea solicitar un conjunto de PCB (en múltiplos de tres), haga clic en el botón denominado "Order Board" junto a cada placa, introduzca su dirección de correo electrónico, nombre y una contraseña de su elección para crear una cuenta con OSH Park; luego siga su proceso de pedido. Puede pagar con un tarjeta de crédito o mediante PayPal. Sus placas se fabricarán y entregarán en dos o tres semanas dependiendo de dónde se encuentre en el mundo.

Si prefiere utilizar un proveedor de PCB alternativo, en lugar de hacer clic en "Order Board", simplemente haga clic en "Download" para descargar una copia del archivo adecuado en formato Eagle (.brd) que puede enviarlo luego a su proveedor preferido.

Tenga en cuenta que ni A-Train Systems ni yo tenemos ninguna otra conexión con OSH Park mas que como un cliente muy satisfecho de sus servicios.

Las piezas necesarias para construir un QSDD completo (decodificador más teclado) incluidos los cables, se enumeran en las tablas siguientes:

Part - Decoder	Reference	Quantity	Value
Diode Rectifier Bridge	BR1	1	W005G
Capacitor - Polyester	C1, C3, C5	3	100nF
" - Electrolytic	C2	1	220uF 35V
" - Disc Ceramic	C4	1	270pF
Resistor - Metal Film, 0.25 Watt	R1	1	1K2
" "	R2, R4 - R7	5	10K
" "	R3	1	4K7
Diode	D1	1	1N4148
Voltage Regulator - 5 Volt	VR1	1	LM7805
Optoisolator	OK1	1	6N137
Arduino Module	M1	1	Nano-3
Terminal Block - 3.5mm pitch	DCC-IN	1	1 x 2
Pin Header - 0.1" (2.54mm) pitch	JP1 - JP4	4	1 x 3
"	JP5, JP6	2	1 x 5
"	JP7	1	1 x 2
Heatsink - to fit TO220 package		1	
USB Cable - A Plug to Mini B Plug		1	

Part - Keypad	Reference	Quantity	Value
Resistor - Metal Film, 0.25 Watt	R1 - R5	5	220R
Light-Emitting Diode - Red	LED1	1	3mm Red
Light-Emitting Diode - Green	LED2 - LED5	4	3mm Green
Tactile Switch	S1 - S4	4	6mm
Socket Header - 0.1" (2.54mm) pitch	JP1 - JP2	2	1 x 5
IDC Socket Ribbon Cable Connector		1	10-way (2 x 5)
IDC Plug Ribbon Cable Connector		1	Box Header (2 x 5)
Ribbon Cable - 0.05" (1.27mm) pitch		As reqd	10-way

Los proveedores sugeridos para las piezas enumeradas anteriormente son RS Components (<https://uk.rs-online.com/web/>) o Farnell (<https://uk.farnell.com/>) para usuarios del Reino Unido o Newark ([www.newark.com/](http://www.newark.com/)) para usuarios de EE.UU. (parte de la misma empresa que Farnell). Mouser ([www.mouser.com/](http://www.mouser.com/)) o Digikey ([www.digikey.com/](http://www.digikey.com/)) son fuentes alternativas en EE.UU, aunque sus precios tienden a ser un poco más altos que los de Newark. Tanto Mouser como Digikey también tienen operan en Europa, pero aún tienden a tener precios más altos que RS Components o Farnell.

El módulo Arduino Nano 3 se puede obtener mejor de uno de los muchos proveedores que operan en eBay, aunque tendrá que decidir quién le dará el mejor servicio basado en los comentarios de clientes anteriores.

La siguiente tabla muestra los números de pieza sugeridos para cada componente QSDD de cada proveedor. Haga clic en el número de pieza para ver la página web adecuada con detalles del componente :

Ref-Decoder	RS Cmps	Farnell	Newark	Mouser	Digikey
BR1	<a href="#">7082668</a>	<a href="#">1861434</a>	<a href="#">70AC6544</a>	<a href="#">625-B380C1000G-E4</a>	<a href="#">B250C1000G-E4/51GI-ND</a>
C1, C3, C5	<a href="#">3121469</a>	<a href="#">2429342</a>	<a href="#">18AC7634</a>	<a href="#">80-R82DC3100AA50J</a>	<a href="#">399-19335-ND</a>
C2	<a href="#">7111264</a>	<a href="#">8126690</a>	<a href="#">62W6211</a>	<a href="#">80-ESK227M035AG3AA</a>	<a href="#">P5166-ND</a>
C4	<a href="#">7167226</a>	<a href="#">2860060</a>	<a href="#">57AC2084</a>	<a href="#">594-S271K43SL0N6TK5R</a>	<a href="#">BC2679CT-ND</a>
R1	<a href="#">1650230</a>	<a href="#">9341226</a>	<a href="#">95W7689</a>	<a href="#">71-CCF071K20GKE36</a>	<a href="#">S1.2KCACT-ND</a>
R2, R4 - R7	<a href="#">1651031</a>	<a href="#">9341110</a>	<a href="#">95W7695</a>	<a href="#">71-CCF0710K0JKE36</a>	<a href="#">S10KCACT-ND</a>
R3	<a href="#">1650319</a>	<a href="#">9341951</a>	<a href="#">95W7764</a>	<a href="#">71-CCF074K70GKE36</a>	<a href="#">S4.7KCACT-ND</a>
D1	<a href="#">7390290</a>	<a href="#">2675146</a>	<a href="#">05AC0533</a>	<a href="#">512-1N4148</a>	<a href="#">1N4148FSCT-ND</a>
VR1	<a href="#">7147780</a>	<a href="#">1467758</a>	<a href="#">89K1366</a>	<a href="#">511-L7805CV</a>	<a href="#">497-1443-5-ND</a>
OK1	<a href="#">8051267</a>	<a href="#">2453244</a>	<a href="#">31Y6274</a>	<a href="#">859-6N137M</a>	<a href="#">160-1791-ND</a>
M1	eBay	eBay	eBay	eBay	eBay
DCC-IN	<a href="#">8971332</a>	<a href="#">3882615</a>	<a href="#">68C9065</a>	<a href="#">651-1985807</a>	<a href="#">277-6043-ND</a>
JP1 – JP4	<a href="#">2518632</a>	<a href="#">1593422</a>	<a href="#">08N6751</a>	<a href="#">517-929834-01-24-RK</a>	<a href="#">3M9457-24-ND</a>
JP5, JP6	"	"	"	"	"
JP7	"	"	"	"	"
Heatsink	<a href="#">7124257</a>	<a href="#">1611415</a>	<a href="#">81F046</a>	<a href="#">532-507302B00</a>	<a href="#">HS115-ND</a>
USB Cable	eBay	eBay	eBay	eBay	eBay

Ref-Keypad	RS Cmps	Farnell	Newark	Mouser	Digikey
R1 – R5	<a href="#">1650814</a>	<a href="#">9341528</a>	<a href="#">95W7736</a>	<a href="#">71-CCF07220RJKE36</a>	<a href="#">S220CACT-ND</a>
LED1	<a href="#">1780909</a>	<a href="#">1581111</a>	<a href="#">14N9386</a>	<a href="#">859-LTL-4211N</a>	<a href="#">160-1139-ND</a>
LED2 – LED5	<a href="#">1808502</a>	<a href="#">1581114</a>	<a href="#">14N9374</a>	<a href="#">859-LTL-4231N-1</a>	<a href="#">160-1958-ND</a>
S1 – S4	<a href="#">3786476</a>	<a href="#">1555982</a>	<a href="#">95M4260</a>	<a href="#">688-SKHHAM</a>	<a href="#">450-1650-ND</a>
JP1 – JP2	<a href="#">8277728</a>	<a href="#">1593490</a>	<a href="#">08N6807</a>	<a href="#">710-61301021821</a>	<a href="#">S7108-ND</a>
IDC Socket	<a href="#">8323483</a>	<a href="#">2215247</a>	<a href="#">45W6459</a>	<a href="#">517-D89110-0131HK</a>	<a href="#">732-2102-ND</a>
IDC Plug	<a href="#">6741205</a>	<a href="#">4139045</a>	<a href="#">94F7977</a>	<a href="#">710-61201025821</a>	<a href="#">732-5452-ND</a>
Ribbon Cable	eBay	eBay	eBay	eBay	eBay

#### NOTAS.-

1. Es posible que pueda obtener localmente componentes equivalentes a un costo menor, utilizando los detalles disponibles para cada parte sugerida, haciendo clic en los enlaces de arriba (asumiendo que tiene conocimientos de electrónica suficientes para comprender las especificaciones).

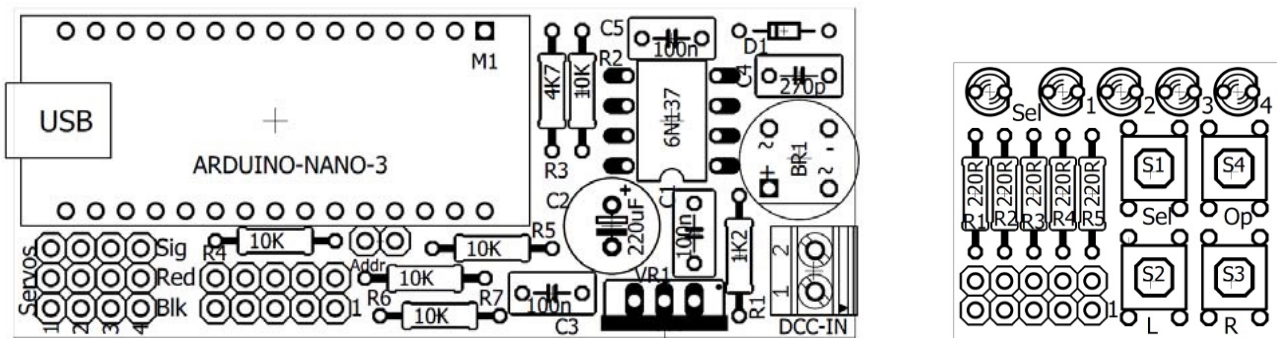
Aunque eBay es una buena fuente de componentes, especialmente para pequeñas cantidades de componentes pasivos, cabezales de conexión, conectores diversos o hardware, debe tener cuidado con las piezas de muy bajo costo, ya que éstas son a menudo de baja calidad (o pueden ser rechazos deficientes de los fabricantes).

2. El costo total de las piezas para un solo QSDD debe ser inferior a 20 dólares estadounidenses (16 libras esterlinas, ó 18,5 euros), incluida la PCB, pero podría ser mucho más si todos los componentes se solicitan a los principales proveedores.

3. Comprar componentes electrónicos por separado o en pequeñas cantidades es mucho más caro que comprar a granel (en cantidades de 10 o más), por lo que vale la pena considerarlo detenidamente al decidir cuántos módulos podría construir, y luego adquirir todos los componentes necesarios en una sola compra. Esto también reducirá los gastos de envío.

Una vez que haya adquirido sus PCB y un kit completo de componentes, el siguiente paso es iniciar el montaje. Si no tiene experiencia en la soldadura de componentes electrónicos, debería echar un vistazo primero a una o dos de las guías disponibles en Internet (como en [www.makerspaces.com/how-to-solder/](http://www.makerspaces.com/how-to-solder/)) y algún video entre la multitud de ellos disponibles en YouTube, aunque no hay nada mejor que conseguir una placa de cobre de uno de los proveedores de componentes y practicar con cables de soldadura (y algunos componentes de repuesto) antes abordar las PCB del módulo real.

Use soldadura con núcleo de resina sólo en forma de alambre delgado; nunca use soldadura con un fundente ácido (como el que se vende para fontanería), y utilice un soldador de punta fina con una potencia máxima de 25 W. Todas las soldaduras han de hacerse lo más rápido posible para evitar dañar las PCB y los componentes. ¡El mayor enemigo de la electrónica es el calor! La posición y el valor de cada componente se muestra en los siguientes diagramas tanto para el decodificador como para teclado (tomado de los archivos de diseño de PCB) .

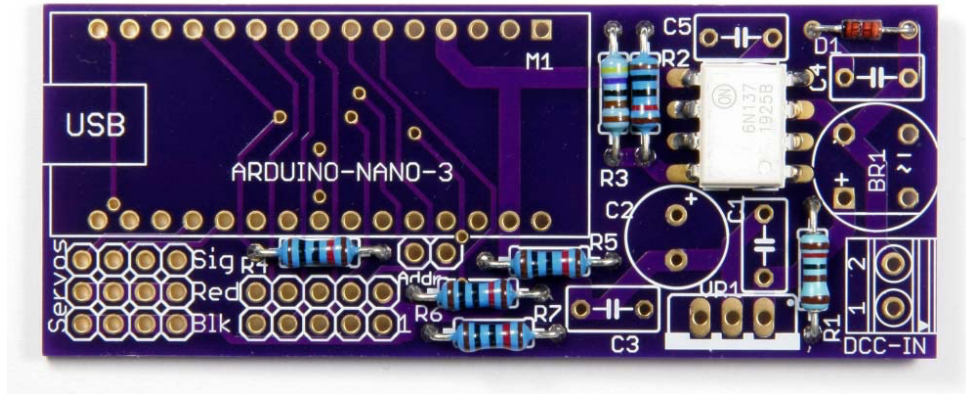


Primero coloque los componentes de menor altura en las placas de circuito impreso, es decir, el diodo (D1) y las resistencias (R1-R7) de modo que, cuando dé la vuelta a la placa PCB y la coloque para soldar el componente en el parte inferior, los componentes no se caigan de los orificios. Asegúrese de que el diodo esté instalado en el decodificador de la manera correcta, como se muestra en la serigrafía de la PCB. No importa de qué manera se coloquen las resistencias en la PCB puesto que no tienen polaridad.

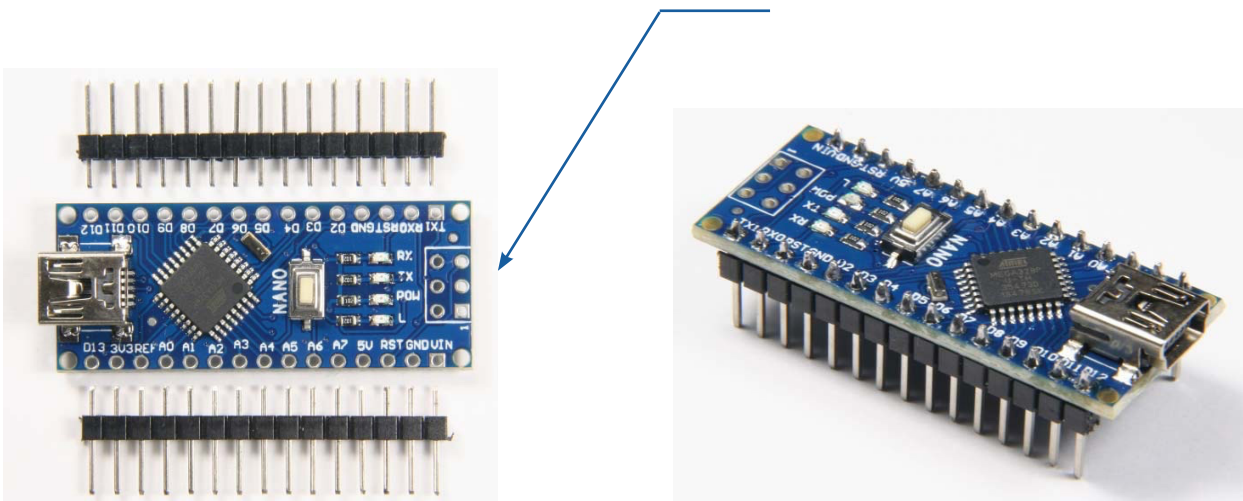
Un consejo aquí es soldar sólo una patilla de cada componente, luego voltear la PCB y verificar que todos los componentes siguen estando debidamente alineados en la placa. Si no, hágalo así: fundiendo brevemente la unión soldada mientras sostiene el componente plano, pero tenga cuidado con los dedos (¡los cables soldados están calientes!) antes de soldar las patillas restantes del componente. Corte las patillas sobrantes del componentes al ras de las uniones soldadas en la parte inferior de la PCB utilizando unas tenacillas de corte.

Cuando ajuste el optoacoplador 6N137 (como los siguientes componentes más altos), asegúrese de que la muesca o el punto en un extremo del chip está hacia el centro de la PCB, nuevamente como lo indica la marca de la placa, y que todos los pines atraviesen los orificios de la PCB, sin que ninguno quede doblado debajo del componente.

Suelde primero dos pines opuestos diagonalmente en el chip y luego verifique que el dispositivo aún esté nivelado respecto a la PCB. Si no está bien nivelado, es fácil fundir la soldadura en un pin de una esquina mientras empuja el dispositivo hacia abajo a la posición correcta.

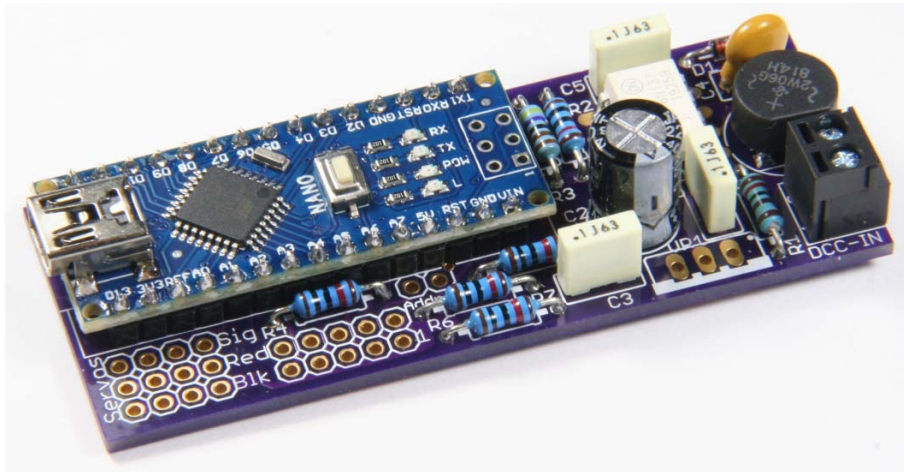


Si compra el módulo Arduino Nano 3 “sin soldar”, deberá instalar las tiras de pines en el módulo antes de colocarlo en la PCB. Al hacer esto, suelde los pines de la esquina primero, y luego verifique que las tiras estén ajustadas a escuadra y a ras con el módulo antes de soldar el resto de pines, y finalmente suelde el conjunto completo a la PCB. Como siempre, realice las juntas de soldadura en el menor tiempo posible, para minimizar el calor aplicado al módulo. Tenga en cuenta que en este montaje no es necesario colocar el bloque adicional de 6 pines (generalmente también suministrado) en el extremo más alejado del conector USB del módulo.



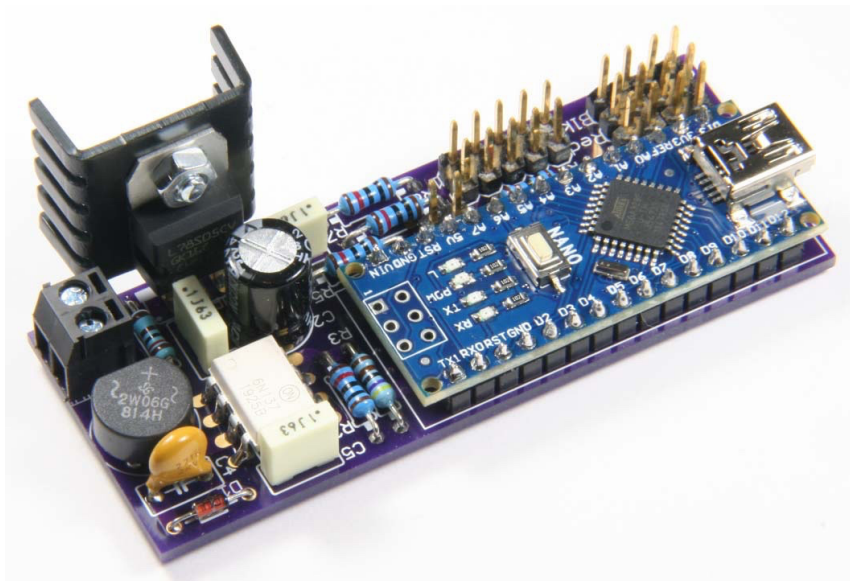
En esta etapa, y antes de soldar el Arduino Nano a la placa PCB del decodificador, puede conectarlo a uno de los puertos USB de su ordenador, usando el cable que se indica en la tabla de piezas del decodificador, y verificar que el módulo se enciende correctamente; normalmente verá dos LED encendidos, uno constante (etiquetado POW) y uno parpadeando (etiquetado como L), dependiendo del estado interno actual del módulo Nano. No es necesario que tenga instalado el software Arduino en su ordenador para realizar esta verificación.

Coloque y suelde el módulo Nano en la placa decodificadora, asegurándose de que el conector USB del módulo está hacia el borde de la placa, seguido de los condensadores (C1-C5), puente rectificador (BR1) y bloque de terminales. Como componente más alto, el condensador de 220uF (C2) se instalará en último lugar, y debe asegurarse de que esté montado de la manera correcta, con la raya que indica el terminal negativo (-) hacia el borde más cercano de la PCB. Los otros condensadores se pueden instalar de cualquier forma en la placa. El borde plano del puente rectificador debe estar al lado del terminal cuadrado.

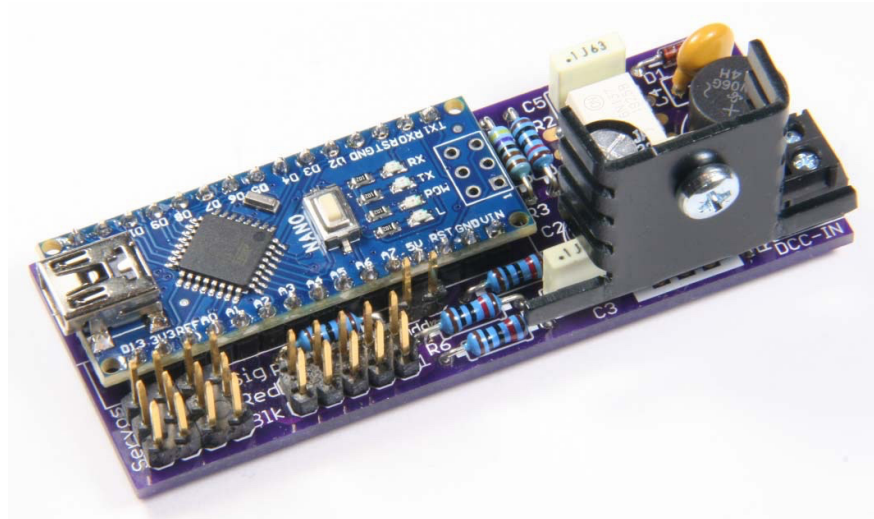


Ahora coloque los bloques de pines. Aunque puede comprar tiras individuales que coincidan con los tamaños utilizados, es mucho más barato comprar una sola tira larga de al menos 24 pines y encajarla con cuidado al tamaño requerido. Tenga cuidado con comprar las tiras demasiado baratas, en eBay por ejemplo, ya que el plástico utilizado tiende a romperse fácilmente exponiendo el pasador del extremo y que, a menudo, no lo mantiene en posición.

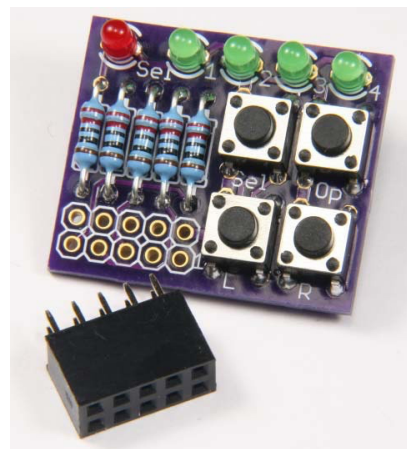
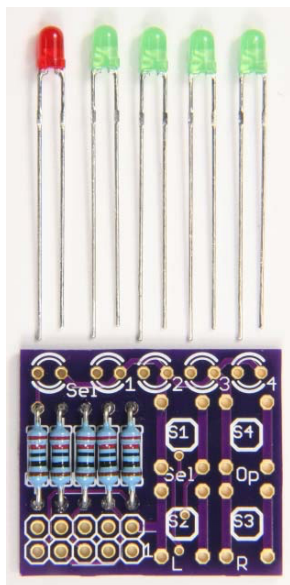
Finalmente, coloque el regulador de voltaje con la lengüeta de metal hacia afuera. Se recomienda un disipador de calor, –como se indica en la tabla anterior de piezas del decodificador–, fijado al regulador de voltaje para reducir su temperatura de funcionamiento a un nivel seguro para el componente.



El disipador de calor identificado en la lista de piezas se puede instalar con las aletas orientadas hacia el regulador de voltaje, como se muestra arriba, o si endereza con cuidado el par inferior de aletas (tenga cuidado con el aluminio blando agrietando y rompiendo las aletas del cuerpo principal del disipador), con el aletas mirando hacia el centro de la PCB.



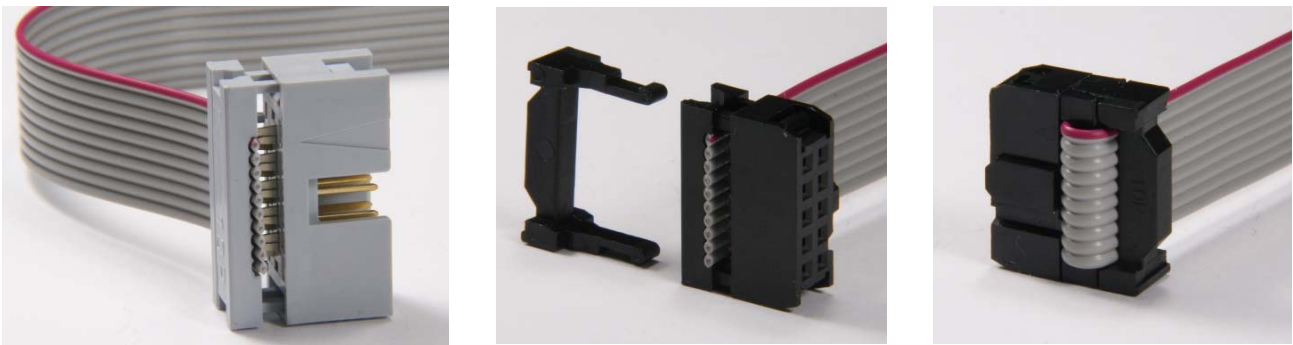
En la PCB del teclado, asegúrese de que los LED estén orientados con su patita más corta hacia el título (Sel , 1 , 2 , 3 , 4), y tenga en cuenta que los LED son particularmente sensibles al calor, especialmente cuando se instalan a ras de la PCB, por lo que no se demore con su soldador. El único punto a tener en cuenta es que el cabezal de enchufe de 2 x 5 se instala en último lugar, después de los cuatro botones, y se coloca en la **parte inferior de la placa**.



Inspeccione cuidadosamente las placas terminadas para verificar que todas sus uniones soldadas estén brillantes, y que la soldadura ha atravesado los orificios de la PCB hasta el lado del componente de la placa. Compruebe también que no haya puentes de soldadura entre las almohadillas de cobre o los pines de los componentes en ninguna parte de cualquier lado de la PCB. Se recomienda encarecidamente el uso de una lente de mano x5 ó x10 aumentos, o una lupa de joyero para esta inspección.

El teclado está diseñado para que pueda conectarse a la placa decodificadora ya sea directamente o mediante un cable plano. Si bien puede comprar cables adecuados listos para usar, es sencillo ensamblar el suyo propio utilizando los componentes enumerados. Después de cortar el cable plano de 10 vías a la longitud requerida, y asegurándose de que los extremos queden cuadrados, coloque los extremos en el aislamiento de desplazamiento de los conectores (IDC) como se muestra en las imágenes siguientes. Una presión firme del dedo generalmente será suficiente para sujetar las dos partes de cada conector en el cable temporalmente, con las espigas simplemente mordiendo el aislamiento del cable.





Coloque cada conector por turno entre las mordazas de un tornillo de banco y aplique con cuidado la presión suficiente para cerrar las dos partes de cada conector sobre el cable y así completar la unión. Finalmente, doble el cable plano sobre la parte superior del conector de enchufe y deslice el puente anti-tracción en su lugar, como se muestra arriba (los conectores de enchufe de cabezal de caja no parecen estar fácilmente disponibles con puente anti-tracción).

Ahora ya puede conectar el teclado al bloque de 2 x 5 pines en el decodificador, ya sea directamente o usando el cable plano descrito anteriormente, y conectar el cable USB especificado (enchufe A a enchufe Mini B) de su ordenador al módulo Arduino Nano. Si todo está ensamblado correctamente, el Arduino Nano debe encender dos de sus LED integrados, uno fijo y otro intermitente (dependiendo del estado interno actual del Nano), listo para recibir su programa de software (un "sketch").

## Software para el decodificador Quad Servo DCC

Como se dijo anteriormente, el artículo de Geoff Bunza en el foro *Model Railroad Hobbyist (MRH)*, "SMA20 Low Cost 17 Channel DCC Decoders" (<https://mrhmag.com/node/24316>) inspiró el desarrollo de este decodificador y, para cualquiera que no esté familiarizado con el hardware y software Arduino que forman la base de todos estos proyectos, el artículo más completo de Geoff en la edición de diciembre de 2016 de *MRH* "A modeler's introduction to the Arduino" (<http://mrhpub.com/2016-12-dec/online/html5/index.html?page=132&noflash>) proporciona un muy buen comienzo.

Los archivos proporcionados en Bonus Extras para el MRH de diciembre de 2016 (<https://mrhmag.com/magazine/mrh-2016-12-dec/bonus-extras>) incluyen una guía para configurar el Arduino Integrated Development Environment (IDE) en su ordenador, incluidos los enlaces a la web de Arduino para descargas ([www.arduino.cc/en/Main/Software](http://www.arduino.cc/en/Main/Software)) y una guía para comenzar ([www.arduino.cc/en/Guide/HomePage](http://www.arduino.cc/en/Guide/HomePage)).

También hay enlaces a tutoriales:

([https://create.arduino.cc/projecthub/Arduino\\_Genuino/getting-started-with-the-arduino-desktop-ide-623be4](https://create.arduino.cc/projecthub/Arduino_Genuino/getting-started-with-the-arduino-desktop-ide-623be4)) y cómo configurar el hardware necesario, incluido el Arduino Nano ([www.arduino.cc/en/Guide/ArduinoNano](http://www.arduino.cc/en/Guide/ArduinoNano)) que se utiliza aquí en el QSDD.

Una vez que tenga el IDE de Arduino instalado y configurado en su ordenador, y haya dominado los conceptos básicos al trabajar con algunos de los ejemplos proporcionados, debe utilizar la opción Incluir Función de Biblioteca del menú Sketch en el IDE para agregar una biblioteca adicional al sistema.

Esta es la Biblioteca NmraDcc:

([www.arduinolibraries.info/libraries/nmra-dcc](http://www.arduinolibraries.info/libraries/nmra-dcc)) desarrollado por Alex Shepherd, Wolfgang Kuffer, Geoff Bunza, Martin Pischky, Franz-Peter Müller y Sven (littleyoda) que permite a un sistema Arduino interpretar los comandos DCC. Esta biblioteca también se puede descargar directamente desde GitHub (<https://github.com/mrrwa/NmraDcc>) donde se pueden encontrar muchos ejemplos de utilización de la biblioteca.

Ahora puede descargar el sketch <sup>(1)</sup> para el QSDD desde la sección de descarga de Arduino de mi A-Train Systems website ([www.a-train-systems.co.uk/arduino/download](http://www.a-train-systems.co.uk/arduino/download)) a cualquier carpeta conveniente en su ordenador, y luego abrirlo en el IDE de Arduino.

Además de la biblioteca NmraDcc, el sketch llama a la biblioteca Arduino Servo estándar (ya cargado como parte del IDE) para admitir operaciones de servo en lugar de la Biblioteca SoftwareServo utilizada por Geoff Bunza en sus aplicaciones, ya que esta última no permite el grado de control sobre servos individuales que se requiere para la configuración manual.

Un error común, observado en la mayoría de los sketch y bibliotecas de ejemplo que he encontrado, es que contienen muy poca información de cómo funcionan o cómo usan las instalaciones que ellos proponen. Por lo tanto, he intentado incluir comentarios sobre el propósito de cada sección del esquema QSDD y cómo se utilizan las diversas funciones incluidas. Además, como se describirá en la Parte 2 de este artículo, puede utilizar el Monitor Serial del IDE para mostrar mensajes explicativos y confirmaciones de sus acciones a medida que procede a configurar el decodificador. Verifique que el boceto se compile con las bibliotecas incluidas, haciendo clic en el botón Verificar (✓) en la barra de herramientas de Arduino IDE. Es más probable que cualquier error se deba a que el IDE de Arduino no pudo encontrar las bibliotecas requeridas donde se espera que estén, o que la configuración de la placa, el procesador y el puerto en el menú Herramientas no están configurados correctamente.

Suponiendo que todo esté bien, asegúrese de tener el cable USB conectado desde su ordenador al Arduino Nano, luego haga clic en el botón Cargar (→) para transferir el sketch compilado al módulo decodificador. Debería ver los LED Tx y Rx de Arduino Nano iluminados en diferentes secuencias de tiempo a medida que avanza la transferencia. Cuando se completa la carga, los cinco LED del teclado deben encenderse en secuencia, y luego apagarse de nuevo en el mismo orden, para confirmar que el código se ha transferido correctamente y ejecutar su proceso de inicialización normal.

Después de esto, debería ver el LED rojo **Sel** del teclado iluminarse brevemente de nuevo. Esto significa que los ajustes predeterminados para las variables de configuración del decodificador (CV) se han cargado en el memoria permanente (EEPROM) del Arduino Nano. Esto solo ocurre la primera vez que se enciende el decodificador inicializado, hasta el momento en que elija restablecer todas las CV a sus valores predeterminados, como se explica en la sección técnica de la Parte 2 de este artículo. Una vez que se ha cargado el sketch, el QSDD retendrá todo el código incluso cuando se encuentre apagado, por lo que ahora puede quitar el cable USB del Arduino Nano antes de conectar hasta cuatro servos al decodificador, listos para el ajuste de sus movimientos.

Cada servo se conecta a uno de los cuatro grupos de tres pines en la placa decodificadora, asegurándose de que el cable negro o marrón de cada servo está conectado al pin más cercano al borde de la placa. Conectar el servo al revés no causará ningún daño, pero el servo no funcionará.

Aunque el decodificador se puede alimentar desde su computadora a través del cable USB, sólo los modernos puertos USB-3 tienen capacidad suficiente para alimentar también los servos; es probable que un puerto USB-2, más antiguo, sea sobrecargado tan pronto como comience a conducir un servo, una acción que probablemente reiniciará el decodificador deteniendo así cualquier impulso adicional a los servos.

Por lo tanto, ahora debe conectar la alimentación de la vía desde su sistema DCC a los dos terminales DCC-IN del decodificador. Si no es conveniente conectar su sistema DCC en este punto, entonces en su lugar, se puede utilizar un suministro de entre 9 y 16 voltios, ya sea **CA** o **CC** <sup>(2)</sup>, incluso una batería de 9 voltios trabajarán.

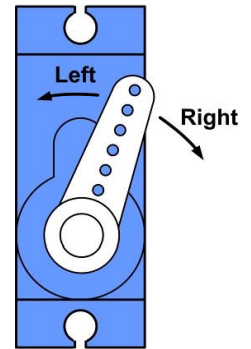
En esta etapa, recomendaría que los servos conectados **no se ajusten** a los desvíos de su maqueta o estén, al menos, desconectados de cualquier tipo de vínculo (alambre, resorte) que esté utilizando para impulsar los desvíos, ya que el servo oscila (con movimientos hacia la izquierda o hacia la derecha) ya que los valores predeterminados cargados en el software pueden no ser apropiados para su configuración particular. En casos extremos, el servo o el enlace, o ambos, podrían dañarse. Es más seguro tener los servos libres para girar con, idealmente, un brazo de accionamiento básico instalado (horn) de modo que sea fácil ver dónde se encuentra actualmente la posición del servo, tal y como se muestra en el diagrama siguiente, por ejemplo.

---

1.- El texto se refiere a la versión 4.6 del sketch; sin embargo la versión 5.2, que es el **software actualizado** tiene un funcionamiento un poco diferente. Esta versión actualizada también se puede utilizar con el hardware que se describe en esta Parte 1.

2.- Corriente Continua : **DC** por sus siglas en inglés.

En este artículo, los movimientos del servo se describirán como “Izquierda” y “Derecha” como se ve en la imagen con el tornillo donde se monta el brazo de accionamiento hacia abajo, como se muestra a la derecha (mirando el servo desde arriba). Tan pronto como el decodificador tenga alimentación, todos los LED del teclado deben encenderse uno tras otro y luego apagarse en el mismo orden, y todos los servos conectados deben moverse en el sentido de las agujas del reloj a la posición predeterminada más a la derecha establecida en el software (20 grados a la derecha del centro).



Para reiniciar el decodificador, y así reiniciar el programa cargado (sketch) de nuevo, normalmente se presionaría el botón de reinicio en el Arduino Nano. Sin embargo, con el teclado conectado directamente en placa decodificadora, el botón de reinicio no es de fácil acceso, por lo que, como alternativa, simplemente presione y mantenga presionado el botón **L** en el teclado; seguidamente presione el botón **Op**, y luego suelte ambos para reiniciar el Arduino Nano.

Para operar un servo, presione y suelte el botón **Op** en el teclado. Esto iluminará el LED verde **1** para mostrar que el servo 1 está seleccionado.

Presione el botón **L** para mover el servo 1 a su posición límite izquierda, a la velocidad establecida, o el botón **R** para mover el servo 1 a su posición límite derecha. No se producirá ningún movimiento, por supuesto, si el servo ya está posicionado en el límite seleccionado.

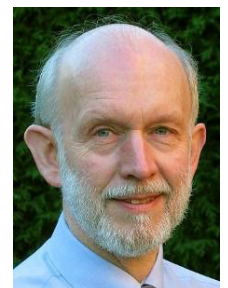
Al presionar de nuevo el botón **Op**, se encenderán los LED verdes **2, 3 y 4** a su vez, seleccionando los servos 2, 3 y 4. Cuando se ilumina el LED verde correspondiente, el servo seleccionado se puede operar usando los botones **L** y **R**, como se describe. Una última pulsación del botón **Op** apagará todos los LED y detendrá el funcionamiento manual del servo.

Por supuesto, también puede operar los servos enviando un comando de accesorio con la dirección apropiada desde su central DCC. Con el sketch cargado, se asignan a los cuatro servos direcciones predeterminadas de 1, 2, 3 y 4, respectivamente.

Detalles completos de cómo programar direcciones de servo de su propia elección en el decodificador y sobre cómo configurar los ángulos precisos del servo para adaptarse a los vínculos del desvío en su maqueta, se darán en la Parte 2 de este artículo: le da tiempo durante el próximo mes para construir su propio decodificador Quad Servo DCC.

## Dr. Terry Chamberlain

Terry Chamberlain se introdujo en el modelismo ferroviario, casi por accidente, en la década de 1990, cuando respondió a una solicitud de algunos aficionados modelistas en California para construir un Sistema DCC basado en una computadora personal Atari, y tuvo que construir un diseño simple para demostrar que todo funcionaba. Finalmente, el proyecto se convirtió en A-Track, una aplicación de Windows que proporciona soporte informático completo para toda gama de sistemas NCE DCC, con instalaciones similares al decodificador de JMRI Pro y Panel Pro.



Terry es un ingeniero electrónico profesional y pasó la mayor parte de su carrera en la industria de defensa del Reino Unido diseñando y gestionando el desarrollo de grandes sistemas informáticos en tiempo real para el Royal Navy. Ahora que se ha jubilado, todavía espera construir el diseño de tala y minería que ha estado planeando durante años (después de varias visitas a Colorado), pero sigue distrayéndose con nuevos proyectos de informática y electrónica para maquetas de ferrocarriles.