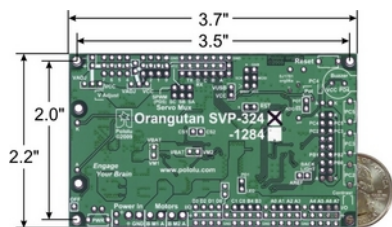


Pololu Orangután SVP-324.Guía de usuario



1. Introducción	2
Especificaciones & Hardware de base	2
Sistemas operativos soportados	3
2. Contactar con Pololu	3
3. Diagrama	4
4. Componentes y pins del módulo	5
Alimentación & Conexiones de motores	5
USB Alimentación	5
Motores	5
Entradas/salidas & Alimentación	6
LCD	6
Pulsadores	6
Buzzer	6
Trimpot	6
Conector de programación	7
Líneas auxiliares I/O & Salidas alimentación	7
Servo Demultiplexor	7
Voltaje ajustable (VADJ)	7
LEDs	8
5. Empezar con el Orangután SVP	8
Instalación de los controladores Windows	8
Uso de los programas demo	10
Programación del Orangután	10
Montaje de la versión en Kit	12
6. AVR Tabla de asignación de pins según función	14
7. AVR Tabla de asignación de pins según el pin	14
8. Uso del puerto de comunicación USB	15
Comunicación con el PC	16
Comunicación con el AVR	16
9. Uso del puerto serie TTL	17
10. Referencias	17
11. Tabla de funcionamiento de motores	18
12. USB alimentación	18
13. Firmware	18
Determinar la versión de firmware	19
14. Actualización de firmware	19
Instalar el controlador bootloader en el PC	19
Cargar el nuevo firmware	20

1. Introducción



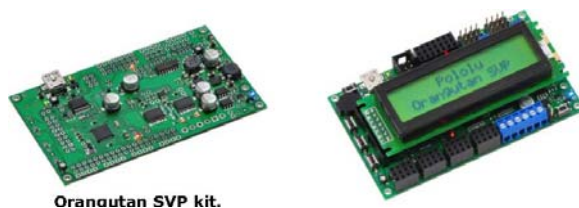
El controlador **Orangután SVP robot** es una completa solución para pequeños y medianos robots que funcionan con una tensión de entre 6 y 13.5 V. El módulo está diseñado alrededor de un microcontrolador AVR Atmel ATmega324PA con una frecuencia de reloj a 20 Mhz (32KB flash, 2KB RAM, y 1KB EEPROM) y va acompañado de todos los complementos necesarios para la conexión de periféricos que soporten las aplicaciones diseñadas: controladora dual de motores de 2

A continuos (6 punta) por canal, un demultiplexor para el fácil control de hasta 8 servos mediante hardware PWM, una pantalla removible LCD de 16x2 con luz de fondo, un trimmer como potenciómetro, un buzzer para sonidos y música, además de tres pulsadores y dos leds programables por el usuario. La placa contiene 21 I/O libres de las cuales 12 pueden usarse como entradas analógicas, y dos reguladores de voltaje (switching buck step-down) – uno para el bus de 5V y otro ajustable desde 2,5V hasta el 85% del VIN – cada uno capaz de proporcionar 3A, suficientes para trabajar con fuerza al añadir servos, sensores y otros periféricos.

A más del microcontrolador AVR el Orangután SVP dispone de una PIC18F14K50 MCU auxiliar que da soporte al procesador principal y se utiliza como programador AVR ISP integrado por lo que no se requiere de uno externo para trabajar. Este microcontrolador auxiliar permite una conexión USB para establecer comunicación con el PC y su firmware permite varios trabajos en paralelo con el micro principal. Por ejemplo, este PIC puede leer los codificadores incrementales sin afectar al procesador principal, o configurar las cuatro entradas para utilizarse como entradas adicionales analógicas. El ATmega 324 puede leer los datos directamente del auxiliar a través del módulo SPI. Un cable **USB-A a mini-B** viene incluido con el módulo.

Como el Orangután SVP tiene acceso directo al microcontrolador AVR le hace compatible con todo el software diseñado para los microcontroladores Atmel, incluyendo el entorno **AVR Studio** y el **compilador WinAVR GCC C/C++**. Nosotros te proveemos de un extenso surtido de **librerías de software** que te permitirán hacer más fácil el trabajo de programación, incluyendo el microcontrolador auxiliar.

Usando estas librerías te sentirás más cómodo al escribir tu código para la LCD, los botones, el control de motores y el funcionamiento de los servos. Estas librerías vienen con programas de ejemplo para los periféricos del Orangután SVP.



Especificaciones & Hardware de base

- Dimensiones: 3.70" × 2.20" (93x54 mm)
- Voltaje de entrada: 6 – 13.5 V
- Atmel ATmega324PA AVR (32 KB flash, 2 KB SRAM, 1024 bytes EEPROM) a 20 Mhz
- Programador integrado USB AVR ISP (cable USB-A a mini-B incluido)
- 2 puertos para motores bidireccionales (2 A continuo por canal, 6 A máximo por canal)
- 8-salidas multiplexadas de hardware PWM del micro AVR's para el control de hasta 8 servos.
- 21 líneas de entrada/salida libres.
 - 17 desde el ATmega324PA, de las que 8 pueden ser entradas analógicas.
 - 4 líneas del PIC auxiliar que pueden ser entradas analógicas o entradas para los codificadores (encoders)
- 2 UART's de hardware
- Pantalla removible de LCD con 16-caracteres × 2-líneas con fondo retroiluminado.
- Regulador primario de 5V capaz de proporcionar 3A.
- Regulador secundario (2.5 V – 85% del VIN) capaz de proporcionar hasta 3A.
- Buzzer conectado a una de las salidas hardware PWM.

- 3 pulsadores de usuario.
- 2 leds de usuario.
- Botón de alimentación (push-on/push-off) y botón de reset.
- La potencia del circuito permite añadir botones extra con opción de apagado automático.
- Procesador auxiliar (conectado vía SPI) que proporciona:
 - Lectura del voltaje de la batería.
 - Lectura del potenciómetro de usuario.
 - Conexión USB.
 - Programación In-System-Programming del procesador principal.
 - Habilidad para la lectura de dos codificadores incrementales (quadrature encoders).
- Orangután SVP totalmente ensamblado (también en kit).

Sistemas operativos soportados

La conexión USB del Orangután SVP's trabaja bajo Microsoft Windows XP, Windows Vista, Windows 7, y Linux y permite la programación del AVR directamente desde el PC o también puedes comunicarte (nivel TTL) con dispositivos serie con el ordenador.

Bajo Linux, los tres puertos COM creados por el SVP pueden aparecer como dispositivos con los nombres /dev/ttyACM0, /dev/ttyACM1 y /dev/ttyACM2 (los números dependen de cuantos dispositivos ACM estén conectados) y usando un programa de terminal (como kermi) para enviar o recibir bytes a través de los puertos.

La conexión USB del Orangután SVP's no es compatible con ninguna versión de Mac OS.

Nota: No es necesario usar la conexión USB del Orangután, si tienes un programador AVR ISP también puedes programar el AVR a través del conector de 6 pins que está cerca del conector USB. En este caso no importa el sistema operativo del ordenador siempre y cuando trabaje con ese programador.

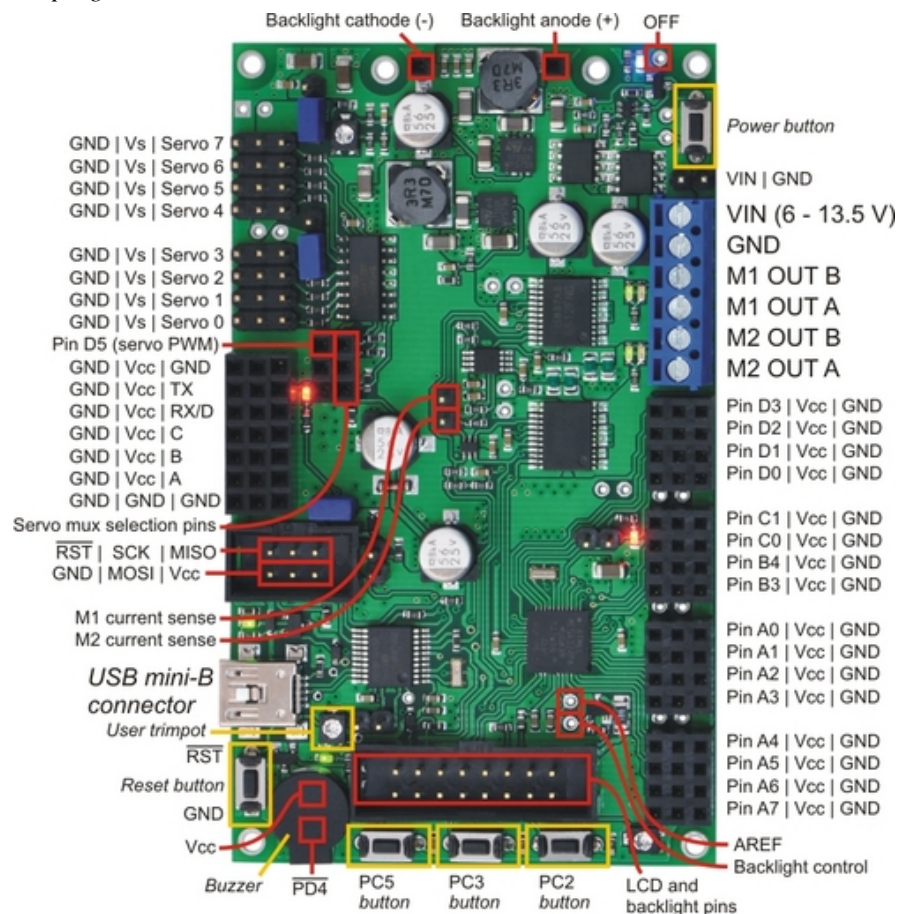
2. Contactar con Pololu

Repasa la página del **Orangután SVP-324 robot controller** para obtener información adicional, incluyendo imágenes, ejemplos de código y notas de aplicación. Puedes buscar entre las librerías para trabajar con el hardware de la placa y acceder a los códigos de ejemplo que hay en **Pololu AVR Library**.

Nos encantaría conocer de tus proyectos y tu experiencia con las controladoras Orangután Robot.

Puede ponerse en contacto con todos nosotros directamente o posteando en nuestro foro que hemos creado al efecto.

Cuéntanos lo que hicimos bien, lo que podríamos mejorar, lo que te gustaría ver en el futuro, o cualquier otra cosa que nos quieras decir.



3. Diagrama

Pololu Orangutan SVP Reference Diagram

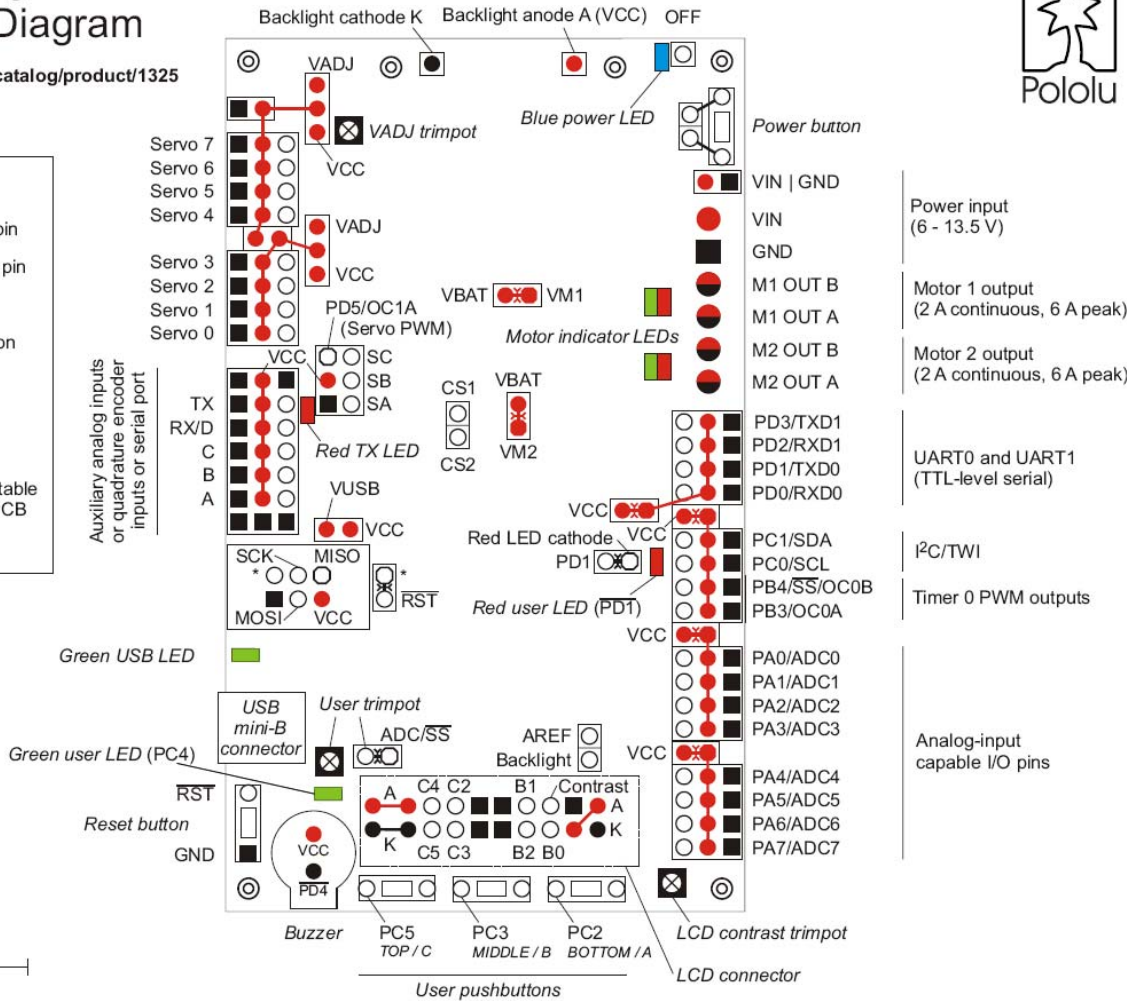
<http://www.pololu.com/catalog/product/1325>

© 2009 Pololu Corporation



Key

- Ground pin
- Positive power pin
- Negative power pin
- I/O pin
- Power connection
- ⊗ Trimpot
- ⊙ Mounting hole
- ▭ LED
- ⊗ Jumper with cuttable connection on PCB bottom



Scale: 1.77:1



4. Componentes y pins del módulo

El Orangután SVP contiene un microcontrolador programable AVR ATmega324PA que se conecta a dos controladores de motores para el funcionamiento directo de motores DC, a una pantalla LCD de 16x2, a un buzzer, a tres pulsadores de usuario, a dos leds de usuario y a un demultiplexador para el control de hasta 8 servos. El AVR está conectado a un procesador auxiliar (PIC18F14K50) que permite el acceso al voltaje de la batería, a un trimmer de 10k y a cuatro entradas adicionales. El procesador auxiliar también nos sirve para la programación del microcontrolador principal haciendo innecesario el uso de uno externo aunque también puede usarse. El micro auxiliar permite la comunicación USB entre el AVR y el PC y actúa como un conversor USB-a-serie.

Estas y el resto de características del módulo están etiquetadas en la figura. Muchos de los puntos de conexión están indicados en la cara inferior de la placa como se puede ver. Las dimensiones del mismo son de 3,7" × 2,2" (54x93 mm.), con cuatro agujeros de anclaje de 0,08" para 4 tornillos localizados a 0,1" de los extremos de la placa.

Alimentación & Conexiones de motores

La alimentación y las conexiones de los motores están situadas en el extremo inferior derecho del módulo. La entrada de corriente (VBAT) puede estar comprendida entre 6-13.5 V a partir de la cual un regulador genera los 5V (VCC) necesarios para el funcionamiento de la lógica del dispositivo.

El Orangután SVP dispone de dos circuitos controladores TB6612FNG, uno para cada salida de motor. Cada salida puede proporcionar 2A continuos y de forma breve hasta 6A. Si no tomas medidas adicionales para mantener frío el motor durante su funcionamiento, como puede ser el uso de un disipador de calor, el utilizar la corriente continua durante mucho tiempo provoca que el circuito se caliente pero gracias al sistema termorregulador incluido hará que se deshabilite si nos excedemos.

Por defecto, el circuito controlador de motores se alimenta de una fuente externa (VBAT).

Sin embargo, puedes desconectar la fuente de alimentación externa de los controladores de motor mediante el corte de la conexión marcada en la parte inferior de la placa $\overline{\text{VBAT-VM1}}$ y $\overline{\text{VBAT-VM2}}$.

Esto permite conectar otra fuente de alimentación para los motores como VADJ (véase más abajo). La controladora de motor tienen un rango de operación de entre 4,5 a 13,5 V, por lo que su fuente de alimentación deberá estar en ese rango y ser capaz de suministrar toda la fuerza que los motores empleados puedan necesitar.

USB Alimentación

Cuando se conecta a un ordenador, la conexión USB proporciona los 5V de alimentación. Si una fuente de alimentación externa está presente, la unidad se filtra de la fuente de alimentación externa y no la obtiene del USB. Si sólo la conexión USB está presente, entonces por defecto, el procesador auxiliar es alimentado desde USB, pero el AVR y los pines de alimentación VCC no se activarán. Está disponible una opción para alimentar todo el dispositivo desde el USB. Vea la Sección 11 para más información

Motores

La controladora de motores funciona con dos de las salidas PWM del AVR, desde el temporizador Timer2 de 8 bits para el control de velocidad y de dos salidas digitales para el control de la dirección. Esto le permite alcanzar velocidades de motor variables mediante hardware PWM en lugar de tener que crear un largo procesado por software para ese control. Puedes controlar los motores usando las funciones de la sección **Orangután Motors** de la Pololu AVR C/C++ Library.

Para cada motor el Orangután SVP tiene un circuito sensor que produce una salida de voltaje proporcional a la corriente necesitada por los motores (850 mv/A). Las salidas correspondientes de estos circuitos están etiquetadas como CS1 y CS2, accesibles por un conector de dos vías que se encuentra en la cara superior en el centro de la placa.

Entradas/salidas & Alimentación

Hay dieciséis líneas I/O accesibles por el usuario a través de conectores hembra de 4×3 y de 0,100” a lo largo del borde de la placa. Cada línea está asociada con un pin de alimentación y otro de masa para facilitar la conexión de sensores: el pin exterior es masa, el pin del medio es alimentación y el pin interior es el de señal que se conecta directamente a una línea I/O del AVR.

Para cada banco hay cuatro pins de I/O y puedes configurar el voltaje de alimentación que se suministra por el pin del medio (alimentación). De forma predeterminada, los pines de alimentación están conectados a VCC (5 V). Puedes cortar en la parte inferior de la placa para desconectarlos de VCC. Esto permite conectarlos a través de uno de los agujeros a una fuente diferente, como VBAT, que también está disponible en la placa.

El total de corriente disponible de VCC (5V) es de 3A lo que significa que puedes alimentar los servos y otros periféricos directamente desde el regulador de voltaje

LCD

El Orangután SVP se suministra con una pantalla LCD removible de 16x2 caracteres con luz de fondo que usa el conocido **HD44780 parallel interface**. Puedes conectar una LCD diferente con un cable apropiado. El Atmega324 dedica cuatro líneas de I/O para los datos en DB4-DB7 (la pantalla LCD está configurada en modo 4 bits) y tres líneas de control RS, RW y EN. Date cuenta que las líneas de datos están compartidas con los botones de usuario y los leds. Puedes hacer funcionar la LCD mediante las funciones de la sección **Orangután LCD** de la Pololu AVR C/C++ Library.

La luz de fondo de la LCD se apaga poniendo en bajo la línea BACKLIGHT. El brillo de la pantalla de la LCD se puede ajustar mediante la conexión a una línea de salida PWM libre.

El pin AREF del AVR está disponible junto al pin de la luz de fondo.

Pulsadores

El Orangután SVP tiene cinco pulsadores en total: el de alimentación on/off en la cara superior a la derecha del conector de motores, el de reset en posición contraria (izquierda, en el otro extremo) y tres pulsadores debajo del de reset. Recuerda que el pulsador de alimentación desconecta toda la corriente a la placa en cambio el pulsador de reset desconecta únicamente el AVR (resetea el microcontrolador). Los botones de usuario van a los pins PC5, PC3 y PC2. Pulsando uno de los botones el pin (en pull-up a través de una resistencia) de línea asociada se pone a GND. Para detectar las pulsaciones de los mismos podemos usar la sección **OrangutanPushbuttons** de la Pololu AVR C/C++ Library. La librería tiene en cuenta la configuración de los pins como entradas, habilita las resistencias de pull-up internas del AVR y tiene en cuenta los rebotes de los contactos mecánicos del pulsador.

Buzzer

El Orangután SVP viene con un buzzer controlado por el pin PD4. Si alternas el estado de la salida del pin producirás una frecuencia de sonido. Usa las funciones en la sección **OrangutanBuzzer** de la Pololu AVR C/C++ Library para tocar notas en background (usando hardware PWM) mientras el resto del procesador realiza otras tareas.

Trimpot

El Orangután SVP viene con un potenciómetro de 10k localizable más abajo del conector USB y cerca del conector de la LCD. El trimpot está conectado al procesador auxiliar, que también mide la salida de voltaje y le pasa nota al AVR.

Puedes desconectar el trimpot del procesador auxiliar cortando la conexión **POT-ADC/SS** en la parte inferior de la placa. Tendrás dos opciones con esta línea libre: usarla como entrada analógica de propósito general o conectarla a una de las líneas libres del AVR y usarla como línea esclava SPI para el procesador auxiliar comunicándote con otro periférico SPI.

Conector de programación

El Orangután SVP tiene un conector de 6-pin para programación situado en la parte superior izquierda. Tienes la opción de usar un programador externo AVR ISP (in-system programmer) o un programador compatible **Pololu USB AVR Programmer** para el AVR. Pero no es necesario ya que el SVP lleva un procesador auxiliar que se usa como AVR ISP programmer del AVR.

Por defecto, el pin 5 del conector de programación, etiquetado con un asterisco (*), está conectado a la línea AVR's ***-ReSeT**, necesario para la programación por ISP desde un dispositivo externo. También puedes desconectar estos dos pins cortando en el sitio marcado y usar las líneas como opciones para tener más líneas.

Líneas auxiliares I/O & Salidas alimentación

El Orangután SVP tiene cinco líneas auxiliares de entrada/salida conectadas al microcontrolador auxiliar. Cada una de estas líneas va asociada con un pin de alimentación (VCC) y uno de masas distribuidas como las anteriores, es decir la de señal al interior de la placa. La línea TX es la línea de transmisión vía serie. Transmite bytes nivel-TTL recibidos desde el "Pololu Orangután SVP TTL Serial Port". Las líneas A, B, C, y D/RX pueden configurarse para diferentes fines. Pueden funcionar como tres entradas analógicas más a la línea de recepción serie, como cuatro entradas analógicas o como entradas para los codificadores. Ver la sección **OrangutanSVP** de Pololu AVR C/C++ Library para más información.

Servo Demultiplexor

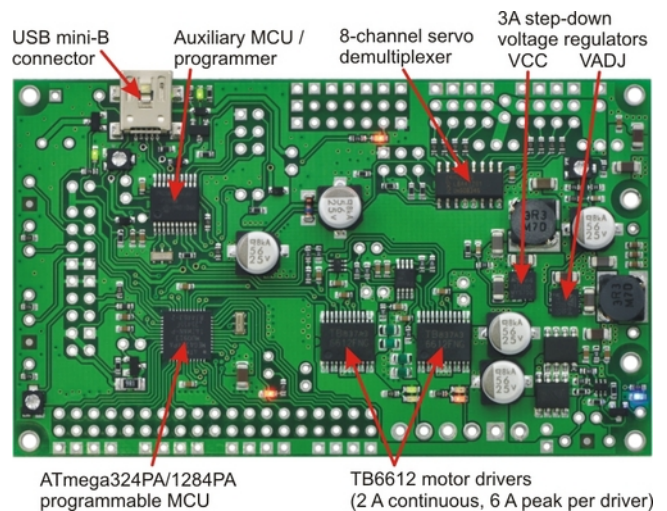
El hardware está situado en la parte superior derecha del Orangután SVP y permite el control de hasta 8 servos sin sacrificar muchas líneas ni ciclos del procesador. Puedes controlar los servos utilizando las funciones en la sección **OrangutanServos** de la Pololu USB AVR C/C++ library.

La señal de entrada del demultiplexor está conectada al pin PD5 del AVR. Si no usas servos puedes utilizar esta línea para otros fines o como salida PWM.

Los tres pins de salida del multiplexor (SA, SB, y SC) están disponibles, en un conector de 3x2 al lado del multiplexor, y están conectadas a las líneas libres de I/O del AVR, permitiendo el intercambio entre servos. Los pins de salida tienen resistencias de pull-up, por lo que si usas menos de cuatro servos algunas de ellas se desconectan.

Los ocho pins de salida están disponibles en dos conectores machos de 4x3. Estas líneas tienen corrientes limitadas por resistencias. Cada salida va asociada a un pin de alimentación y otro de masa para facilitar la conexión de los servos: el pin exterior es masa, el del medio alimentación y el interior la señal. Para cada banco de servos puedes configurar la alimentación que saldrá del pin, usando los jumpers. Puedes alimentar los servos desde VCC, VADJ (ver abajo), o desde una alimentación separada.

Orangután SVP con el hardware etiquetado.



Voltaje ajustable (VADJ)

A más del regulador de 5V para VCC, el Orangután SVP tiene un regulador con voltaje ajustable. Ambos reguladores proporcionan una corriente de 3 A. El regulador ajustable emite corriente desde una entrada externa (VBAT), y produce una salida denominada VADJ. El trimpot situado en la esquina de arriba a la derecha, determina el VADJ. Esta corriente estará en el rango de valores de entre 2,5 V y el 85% del valor de VIN.

En general, es ventajoso para la energía de los servos y otros dispositivos de alta potencia el VADJ (en lugar de VCC), porque si los periféricos necesitan demasiado corriente la corriente del AVR no se verá afectada.

LEDs

Orangután SVP LEDs.

El Orangután SVP viene con 9 LEDs:

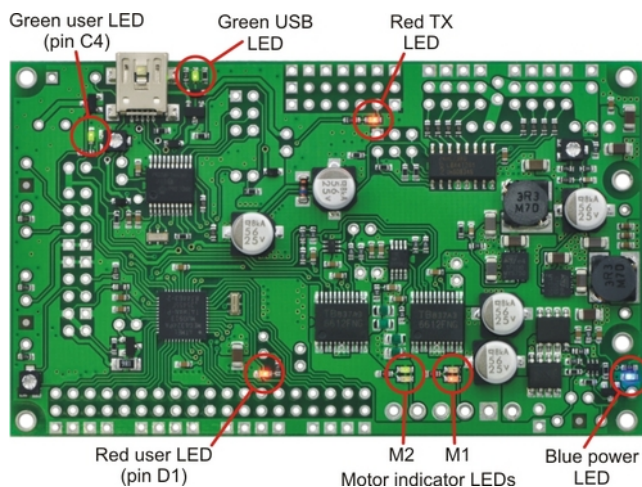
Uno **azul** de alimentación cerca del pulsador. Cuatro para los motores en las salidas de motores, uno **verde** indica movimiento en un sentido (el voltaje de salida B está alto en relación al A), el otro **rojo** indica lo contrario (voltaje bajo en B en relación al alto en A).

Uno **rojo** de usuario situado cerca de los bancos de pins de entrada/salida en el borde inferior que está conectado a la línea PD1. Se enciende si la línea está en salida baja. Luego si la línea PD1 es la línea de transmisión serie del UART0 (TXD0) dicho led parpadea en cuanto haya transmisión desde el AVR. Puede desconectarse cortando la línea marcada como **PD1-LED** en la otra cara de la placa.

Un led **verde** de usuario se encuentra cerca del trimpot y del buzzer. Se enciende si PC4 está como salida alta. Recuerda que la línea del PC4 también se utiliza por la LCD para datos por lo que parpadeará cuando se actualice la LCD. Otro **verde** se encuentra cerca del conector USB. Este led está controlado por el procesador auxiliar e indica el estado de la conexión USB. Cuando se desconecta o el dispositivo está en modo Suspend (porque el ordenador lo suspende) el led se apagará.

Cuando conectas el dispositivo al ordenador vía USB el led verde se enciende, parpadeando lentamente. Hay un parpadeo continuo mientras espera recibir un mensaje del PC indicando que se ha hecho la conexión correctamente. Después de este mensaje, el led se queda encendido y solo parpadea si hay actividad.

Otro led **rojo** se encuentra cerca del conector con la línea TX del procesador auxiliar. Este led está conectado a la línea TX y parpadea cuando el auxiliar transmite bytes vía TTL-serie desde el ordenador. También parpadea cuando la alimentación del procesador auxiliar indica malas condiciones de inicio. DOS parpadeos indican el disparo del brown-out reset, es decir la alimentación VDD del procesador es inferior a 3.0v. Es necesario revisar las conexiones de alimentación y el voltaje de la batería y estar seguro de que no se consume demasiada energía con los dispositivos conectados.

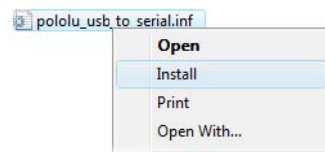


5. Empezar con el Orangután SVP

Si usas Windows XP, necesitas tener instalado el Service Pack 3 o Hotfix KB918365 antes de proceder con los controladores para el Orangután SVP. Antes de conectar el Pololu Orangután SVP robot al ordenador deben estar instalados los controladores:

Instalación de los controladores Windows

1. Bájate el *Pololu USB-to-serial driver for Windows* (4k inf)
2. Botón derecho en pololu_usb_to_serial.inf y selecciona "Install".
3. Después de "Install", Windows avisa que el controlador no está testeado por Microsoft y recomienda parar. Sigue, clic "Continue Anyway" (Windows XP) o "Install this driver software anyway"



(Windows Vista).

4. Windows Vista: Después de instalar INF, el ordenador configura automáticamente los mismos en cuanto conectes el Orangután SVP, en este caso no es necesario realizar ninguna acción.
5. Windows XP: Después de que INF file se ha instalado, sigue los pasos 5-9 para cada nuevo Orangután SVP que conectes al ordenador.
6. Conecta el dispositivo al puerto USB. El Orangután SVP muestra tres dispositivos en uno, en un ordenador con XP y detecta los tres nuevos dispositivos mostrando el mensaje “Found New Hardware Wizard” tres veces. Cada vez que “Found New Hardware Wizard” aparece, sigue los pasos 6-9.
7. Cuando se muestre “Found New Hardware Wizard” selecciona “No, not this time” y clic “Next”.
8. En la segunda pantalla de “Found New Hardware Wizard”, selecciona “Install the software automatically” y clic “Next”.
9. Windows XP mostrará que el controlador no está testeado por Microsoft y recomienda parar la instalación. clic “Continue Anyway”.
10. Cuando haya finalizado “Found New Hardware Wizard”, clic “Finish”. Después de esto otra ventana aparece hasta tres veces para el Orangután SVP. Sigue los pasos 6-9 para cada vez.
11. Si usas Windows XP y tienes problemas de instalación, la causa de los mismos puede encontrarse en algún fallo del fichero antiguo de Microsoft’s usb-to-serial driver usbser.sys. Versiones del controlador anteriores a la 5.1.2600.2930 no funcionan con el Orangután SVP. Comprueba la versión del driver en la pestaña “Details” o en “Properties” de C:\Windows\System32\drivers\usbser.sys. Para tener la versión correcta del driver necesitas instalar el **Service Pack 3** o **Hotfix KB918365**.

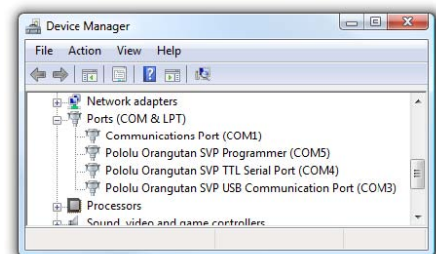
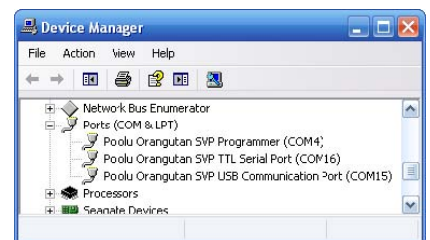
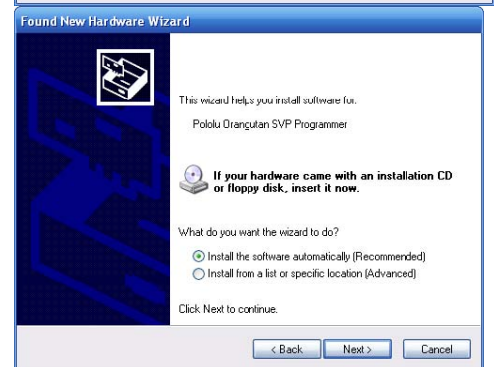
Después de instalarlos vete a Device Manager y expande el “Ports (COM & LPT)” veras tres nuevos COM ports:

Pololu Orangután SVP Programmer (Sección 5.c), Pololu Orangután SVP USB Communication Port (Sección 8), y el Pololu Orangután SVP TTL Serial Port (Sección 9). En paréntesis después de los nombres veras el puerto (Ej. “COM5” o “COM6”).



Existe algún software que no reconoce la numeración alta de los puertos COM. Así por ejemplo AVR Studio solo reconoce del COM1 al COM9. Si ves necesario cambiar la numeración de los mismos usa el Administrador de dispositivos. Abre “Propiedades” y clic en el botón “Advanced...”

de la pestaña “Port Settings”. Desde esta ventana de dialogo puedes cambiar el COM Port asignado al dispositivo.



Uso de los programas demo

El Orangután viene con un programa demo pre-grabado que muestra algunas características y prueba la placa para comprobar que trabaja correctamente.

Cuando conectas el Orangután por primera vez, el led rojo de usuario parpadea cerca de una vez por segundo. Si tienes el buzzer conectado oírás un beep. Si está conectada la LCD, mostrará el mensaje “Pololu Orangután SVP”, y luego aparece “Demo Program”, indicando que está en funcionamiento el programa de demostración. Si ves el led rojo parpadear y no aparece nada en la LCD necesitarás ajustar el potenciómetro de brillo de la pantalla. Cuando esté funcionando correctamente pulsa el botón Middle central (marcado como PC3 sobre la placa) para entrar en el menú principal. Pulsa el botón Top (marcado como PC5) o el botón Bottom (marcado como PC2) para desplazarse por el menú y presiona el del medio para confirmar la selección o para salir de la demo.

Hay nueve demos accesibles desde el menú:

1. Entradas analógicas: Esta demo muestra las lecturas de los voltajes del Orangután SVP's 13 entradas analógicas como barras gráficas. Las entradas están en el orden: PA7, PA6, PA5, PA4, PA3, PA2, PA1, PA0, trimpot, A, B, C, D. Pulsa el botón TOP para habilitar/deshabilitar las resistencias de pull-up de PA0—PA7.
2. Voltaje batería: Muestra el voltaje de la batería en milivoltios.
3. Entradas digitales: Esta demo muestra las lecturas de las entradas digitales accesibles por el usuario. Las resistencias de pull-up están activadas por lo que cada línea lee 1 si no hay nada conectado. Si conectas algo en estas líneas aparecerá un cero (procura no producir un pequeño cortocircuito)
4. LEDs: Parpadean los leds verde y rojo de usuario.
5. Trimpot: Muestra en la LCD la posición del trimpot situado en el extremo superior izquierdo de la placa, con un número de entre 0 a 1023. Al mostrar el valor, hace parpadear el led y al mismo tiempo suena una nota correspondiente al valor en esa frecuencia. Para mover el trimpot utiliza un destornillador pequeño de 2mm.
6. Motores: Los motores funcionan según el botón bottom o top o ambos a la vez para ambos motores. Los motores giran con una velocidad gradual; en tus programas puedes cambiar de velocidad más rápidamente. Una pulsación corta del botón produce que las letras correspondientes a los motores pasen a ser minúsculas y que estos giren en sentido contrario.
7. Música: Toca una melodía mientras saca un mensaje desplazándose por la pantalla. Es para demostrar la habilidad del Orangután SVP de tocar música en segundo plano.
8. Timer: Un simple contador. Pulsa el botón para iniciar o parar el conteo. El contador continúa mientras exploras otras demos.
9. USB: Demuestra la conexión USB entre el Orangután SVP y el PC. Conecta el Orangután SVP al USB. Algunos bytes se desplazan desde el Pololu Orangután SVP USB Communications Port (Sección 8) y se muestran en la pantalla y se repiten en el PC si has arrancado un programa de terminal en el PC. El led rojo indica el estado de la línea DTR de control y el led verde el estado de la línea RTS.

El código fuente de esta demo se encuentra en la **Pololu AVR C/C++ Library**. Después de bajar y desempaquetar el fichero zip el programa está en la carpeta `examples\atmega324p\svp-demo-program`.

Programación del Orangután

Para programar el microcontrolador Atmega-324 del Orangután SVP's, es necesario un compilador de software de programación del AVR ISP.

1. Compilador:
 - a. Si tienes Windows, recomendamos instalar **WinAVR**. WinAVR es una suite libre, open-source de herramientas de desarrollo para la familia de los micros AVR incluyendo un

compilador GNU C/C++ para AVR (avr-gcc). Lee las siguientes instrucciones de instalación de la página de WinAVR's sourceforge.net.

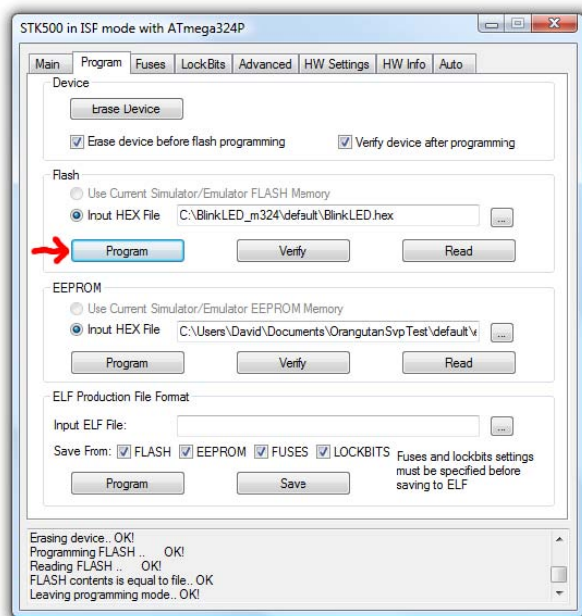
- b. Para programar un Orangután bajo Linux necesitas instalar cuatro paquetes de software que se pueden bajar de sus respectivas websites. Si tienes Ubuntu Linux estos paquetes se encuentran en el repositorio "Universe".
 - **gcc-avr**: el compilador GNU C compiler, para los AVR.
 - **avr-libc**: la librería para tener acceso a las funciones específicas de AVR
 - **binutils-avr**: herramientas para convertir código objeto a ficheros hex.
 - **avrdude**: el software para el funcionamiento del programador.

2. AVR ISP programming software: Atmel ofrece un entorno de desarrollo integrado (IDE) para la programación de los AVR llamada **AVR Studio**. Este paquete trabaja en conjunción con el compilador **WinAVR C/C++ GCC** y contiene el soporte para trabajar con el programador AVR ISP. Por favor sigue las instrucciones de instalación de Atmel para el **AVR Studio**. Otra alternativa es programar libremente las líneas de comandos con la aplicación **avrdude**, que forma parte del paquete de **WinAVR**.

Para el primer test recomendamos escoger la programación del Orangután con un simple programa que hace parpadear el LED rojo en el pin PD1: **BlinkLED_m324.zip** (7k zip)

En Linux debes mover Makefile a linux/ arriba del directorio, compilar usando el comando *make* y programar usando el comando *make program*.

En Windows puedes usar el AVR Studio para programarlo. Abre el proyecto pulsando dos veces en BlinkLED.aps. Conecta el Orangután al USB. Enciéndelo desde el botón power (se enciende el LED azul). En Tools menú, selecciona Program AVR > Connect.... Selecciona la plataforma AVRISP.



Selecciona el puerto COM Port del "Pololu Orangután SVP Programmer" que puedes ver con el Administrador de dispositivos en la lista "Ports (COM & LPT)". Pulsa el botón Connect.... Si AVR Studio realiza la conexión con el Orangután SVP's programmer, se mostrará una ventana de dialogo con el modo *STK500*. En la sección *Flash* de la pestaña Program, busca el fichero hex. del proyecto (debe estar creado en el directorio por defecto de cuando iniciaste el proyecto). Asegúrate de que el AVR está encendido (LED azul on). Ahora clic en el botón Program y si todo se ha hecho bien verás el LED rojo del Orangután parpadeando cada segundo.

Como segundo paso, recomendamos instalar la **Pololu AVR C/C++ Library**, la cual provee las funciones para interactuar con todo el hardware integrado en el Orangután, incluyendo el procesador auxiliar, así como varias demos para demostrar como trabajan estas

funciones. Es bastante sencillo y ya podrás empezar a trabajar con él.

Para más detalles de como empezar con el AVR Studio, incluyendo pantallas, mira la **Pololu USB AVR Programmer User's Guide**. Esta guía es específica y hay muchas secciones en donde se usa el AVR Studio y que son necesarias para poder trabajar y programar el Orangután SVP'. Una diferencia notable, sin embargo, está en que el programador del Orangután SVP's ignora el **parámetro de frecuencia** del AVR Studio's ISP y **siempre programa a 2 MHz**. No obstante, el Orangután SVP's programmer prevé mediante los ajustes de algunos parámetros, el reloj del AVR. Esto se hace para evitar que accidentalmente uses un ajuste incorrecto del reloj y dejes el AVR sin poder programarse.

Si tienes un programador externo puedes utilizarlo con el Orangután SVP si lo enchufas al conector de 6-pin AVR ISP localizado cerca del conector USB.

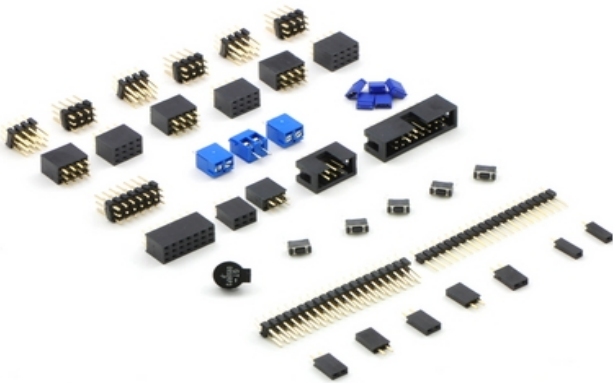
Montaje de la versión en Kit.

Esta sección es una guía para la conexión de los diferentes cabezales que vienen con la versión en kit de la Orangután SVP.

El Orangután SVP kit contiene estos dispositivos:

- Seis bloques de 3×4 macho
- Seis bloques de 3×4 hembra
- Un bloque macho de 3×7
- Uno de 3×7 hembra
- Tres bloques de 2-pin y 3.5mm terminal
- Cinco bloques azules para jumpers
- Dos bloques de 2×3 hembras
- Uno de 2×3 macho enmarcado
- Uno de 2×8 macho enmarcado
- Un buzzer
- Cinco botones pulsadores
- Una tira de 2×20 macho
- Una tira de 1×20 macho
- Cinco de 1×2 hembras
- Dos 1x1 hembras

La imagen de abajo muestra los puntos para instalar estos cabezales y terminales y estos son:
Hardware incluido con el kit de Orangután SVP.



Punto 1.- Pulsador de alimentación.

Para poner en marcha el AVR necesitamos un botón a conectar en el sitio indicado (o un jumper en el lugar 11). El Orangután SVP PCB tiene dos juegos de agujeros paralelos junto al botón power. Puedes soldar un pulsador en este punto o realizar un punto de conexión propio mediante el uso de un cabezal de 1x2 o directamente una soldadura. Si el orangután no tiene fácil acceso dentro de tu robot, puedes instalar un interruptor temporal en un lugar más conveniente y mediante unos cables hacer la conexión a ese cabezal.

Punto 2.- Entrada alimentación

Hay dos puntos de conexión para la entrada de alimentación 6–13.5V. Puedes instalar el bloque de terminales de 3.5mm y además a su derecha, puedes usar un bloque de 1x2 para conectar directamente un **pack de baterías** directamente.

Punto 3.- Salida de motores.

Debes poner dos bloques de terminales de 3.5mm para acomodar los dos motores.

Punto 4.- Bloques entradas/salidas de usuario

Los pins de usuario de I/O del AVR se dividen en cuatro bloques. El kit trae 4 bloques de 3x4 que debes soldar en la parte marcada como 4 en la figura.

Punto 5.- Retroiluminación y conector AREF

Si quieres poner en off la retroiluminación de la LCD o conectarla a una salida externa de voltaje tienes que soldar un conector de 1×2 en este sitio.

Punto 6.- Conector LCD

Los pins de conexión de la LCD con el AVR para su control se realizan con un cabezal de 2x8 que se corresponden con los 16 pins dibujados en el diagrama. En una pantalla de **16×2 caracteres con retroiluminación** (incluida en la versión ensamblada pero no en esta versión) la conexión debe estar configurada teniendo en cuenta el diagrama y las líneas A y K en la cara opuesta de la placa. Si vas a usar una pantalla diferente consulta la hoja de características y **Orangután SVP Reference Diagram**.

Punto 7.- Pulsadores de usuario.

Los tres pulsadores de usuario se conectarán aquí.

Punto 8.- Buzzer.

El buzzer en este punto.

Punto 9.- Pulsador Reset.

El pulsador de reset aquí.

Punto 10.- Conector SPI/Programación

En la versión ensamblada hay un cabezal de 2×3 soldado en este punto que permite el uso de un programador ISP AVR externo para programar el AVR por si no quieres usar el que viene integrado en la placa. También permite el acceso a los pins SPI del AVR y posibilita la conexión de dispositivos que usen este tipo de comunicación. Hay un cabezal hembra de 2×3 para ese fin.

Punto 11.- VCC-VUSB jumper

Para alimentar la placa desde USB, instala un cabezal macho de 1×2 en este punto y usa un bloque azul para conectar el VUSB-VCC. Mira la sección 11 para más información y consejos de uso de este jumper.

Punto 12.- Bloques entrada/salida procesador auxiliar

Este bloque es para el procesador auxiliar y las líneas A, B, C, D/RX, y TX, junto con las conexiones a GND y VCC. Puedes soldar un 3×7 macho o hembra según las necesidades.

Punto 13.- Salidas de corriente de los motores

Si quieres acceder a las salidas analógicas de voltaje del circuito de alimentación de cada uno de los controladores de los motores debes soltar un conector hembra de 1x2 en esta posición.

Punto 14.- Multiplexor para los servos

Para acceder a las entradas del multiplexor de servos, soldar un conector hembra o macho de 2×3.

Punto 15.- Servo salidas

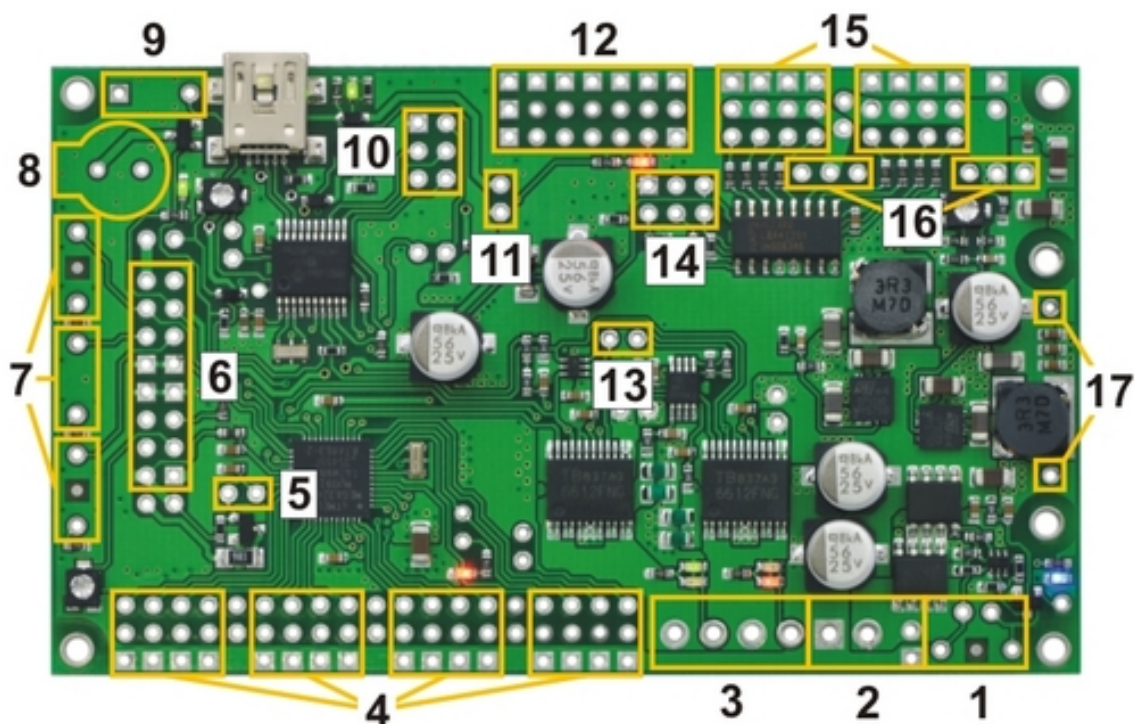
En este punto se insertarán los cabezales de pins machos para los dos bloques de servos. Para eso se dispone de dos bloques de 3x4 y que se corresponden con los conectores de 3 pins de los servos.

Punto 16.- Servo selección alimentación

Para usar los puertos para servos debes conectar una alimentación (no hay alimentación por defecto). Debes instalar dos conectores machos de 1×3 junto con los bloques azules para alternar la alimentación entre VCC-servos-VADJ.

Punto 17.- LCD conexión de retroiluminación y soporte de la estructura

Si vas a montar una pantalla LCD de 16×2 en el Orangután SVP, esta debe ser como en la versión ensamblada, debes soldar dos pins hembras en este punto que servirán como soporte de la misma y como alimentación de la luz de fondo.



6. AVR Tabla de asignación de pins según función

Función	ATmega324 Pin
Entradas analógicas/ digitales I/O (x8)	PA0 – PA7
Digital I/O (x8)	PB3, PC0, PC1, PD0 – PD4, PD5**
Salida digital	PB4*
Salidas PWM (x2)	PB3, PB4
LED rojo de usuario	PD1
LED verde de usuario	PC4
UART0	PD0, PD1
UART1	PD2, PD3
Pulsadores de usuario (Bottom, Middle, Top, x3)	PC2, PC3, PC5
motor 1 (dirección, velocidad)	PC7, PD7
motor 2 (dirección, velocidad)	PC6, PD6
buzzer	PD4
LCD control (RS, R/W, E)	PB0, PB1, PB2
LCD datos (4-bit: DB4 – DB7)	PC2 – PC5
SPI líneas de control del procesador auxiliar (x3)	PB5, PB6, PB7
ICSP líneas de programación (x3)	PB5, PB6, PB7
ADC pin de referencia	AREF
Botón de reset	RESET

*: El pin PB4/SS como entrada puede interferir la comunicación con el procesador auxiliar, por lo que recomendamos usarlo como salida digital. Ver más información en la Sección 6.1 de Pololu USB AVR C/C++ User's Guide.

** : PD5 es una I/O digital mientras no se use para trabajar con los pulsos de los servos.

7. AVR Tabla de asignación de pins según el pin

Pin	Orangután Función	Notas / Funciones alternativas
PA0	entrada analógica / I/O digital	ADC entrada canal 0 (ADC0)
PA1	entrada analógica / I/O digital	ADC entrada canal 1 (ADC1)
PA2	entrada analógica / I/O digital	ADC entrada canal 2 (ADC2)
PA3	entrada analógica / I/O digital	ADC entrada canal 3 (ADC3)
PA4	entrada analógica / I/O digital	ADC entrada canal 4 (ADC4)
PA5	entrada analógica / I/O digital	ADC entrada canal 5 (ADC5)
PA6	entrada analógica / I/O digital	ADC entrada canal 6 (ADC6)
PA7	entrada analógica / I/O digital	ADC entrada canal 7 (ADC7)
PB0	LCD línea de control RS	Timer/Counter 0 External Counter Input (T0)
PB1	LCD línea de control R/W	Divided system clock output (CLKO)
		Timer/Counter 1 External Counter Input (T1)
PB2	LCD línea de control E	Interrupción externa 2 (INT2)
		Analog Comparator Positive Input (AIN0)
PB3	digital I/O	Timer0 PWM output A (OC0A)
		Analog Comparator Negative Input (AIN1)
PB4	Salida digital*	Timer0 PWM output B (OC0B)
		SPI Slave Select Input (SS)
PB5	línea control procesador auxiliar	SPI Master Output/Slave Input (MOSI)
		ISP línea de programación
PB6	auxiliary processor control line	SPI Master Input/Slave Output (MISO)
		ISP línea de programación

PB7	línea control procesador auxiliar	SPI Clock (SCK)
		ISP línea de programación
PC0	digital I/O	I2C/TWI clock line (SCL)
PC1	digital I/O	I2C/TWI input/output data line (SDA)
PC2	LCD línea datos DB4	Pulsador usuario (pressing pulls low)
PC3	LCD línea datos DB5	Pulsador usuario (pressing pulls low)
PC4	LCD línea datos DB6	LED verde de usuario (high turns LED on)
PC5	LCD línea datos DB7	Pulsador de usuario (pressing pulls low)
PC6	M2 línea control dirección	
PC7	M1 línea control dirección	
PD0	digital I/O	USART0 input pin (RXD0)
PD1	Digital I/O	LED rojo de usuario (nivel bajo LED on; se puede juntar con un jumper)
		USART0 pin de salida (TXD0)
PD2	digital I/O	USART1 pin de entrada (RXD1)
		Interrupción externa 0 (INT0)
PD3	digital I/O	USART1 pin de salida (TXD1)
		Interrupción externa 1 (INT1)
PD4	buzzer	Timer1 PWM output B (OC1B)
PD5	digital I/O	Salida servo pulsos (SPWM) Timer1 PWM output A (OC1A)
PD6	M2 línea control velocidad	Timer2 PWM output B (OC2B)
PD7	M1 línea control velocidad	Timer2 PWM output A (OC2A)
AREF	ADC pin de referencia	
AVCC	Voltaje para Port A y ADC no accesible para el usuario	
RESET	Reset botón de entrada reset (resetea la MCU)	
XTAL1	20 MHz entrada reloj no accesible al usuario	
XTAL2	20 MHz entrada reloj no accesible al usuario	

*: El pin PB4/SS como entrada puede interferir la comunicación con el procesador auxiliar, recomendamos se use como **salida digital**. Ver Sección 6.1 de Pololu USB AVR C/C++ User's Guide.

8. Uso del puerto de comunicación USB

La conexión USB del Orangután SVP's habilita un puerto serie virtual llamado Pololu Orangután SVP USB Communication Port que permite dos vías de comunicación entre el PC y el Orangután SVP's AVR a la velocidad de unos 20 kilobytes por segundo.

Los bytes enviados desde el ordenador son transmitidos a través del procesador auxiliar y están disponibles por el AVR. Los bytes que envía el AVR pueden ir al ordenador. Como complemento a las dos líneas de comunicación de datos se pueden usar las líneas de control de estado RTS y DTR. El estado de estas líneas (entre 0/1) están controladas por el ordenador y ser leídas por el AVR. Muchos programas de terminal ponen la línea DTR a 1 cuando conectamos por el puerto serie y la ponen a 0 en cuanto desconectamos. También podemos usar la línea DTR para determinar cuando se establece la conexión entre el programa terminal y el Orangután, y hacer que el estado del Orangután dependa de esta conexión.

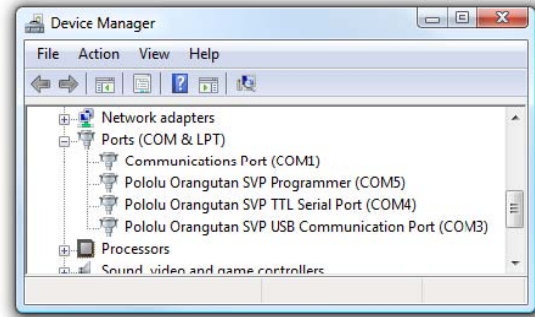
En el PC, los controladores USB del SVP Orangután permiten que la comunicación para el sistema operativo se establezca como puerto serie estándar, permite utilizar los programas de la terminal actual y el software diseñado para puertos serie. Para el AVR, la librería AVR Pololu la C / C++ ofrece funciones de comunicación de serie para usar el puerto.

Comunicación con el PC

Para usar el Puerto de comunicación USB primero debes saber el número de puerto que le ha sido asignado por el sistema operativo.

Para ello, abre el Administrador de dispositivos en Panel de control/Sistema expande la lista de “Ports (COM & LPT)” y comprueba el puerto asignado al Pololu Orangután SVP USB Port. En Windows, el mismo puerto puede estar asociado con otro dispositivo, por lo que deberás cambiarlo manualmente desde las propiedades del puerto (ver Sección 5.a).

Windows Vista device manager muestra el Pololu Orangután SVP.

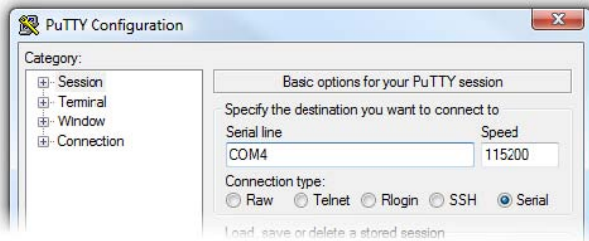


Para determinar el puerto en Linux, escribe `ls /dev/ttyACM*`. El puerto será uno de los dispositivos listados allí. Si esta en tres puertos, el USB Communication Port será `/dev/ttyACM1` (el programador será `/dev/ttyACM0` y el por TTL serie port será `/dev/ttyACM2`). Si ves más de tres puertos comprueba con `dmesg` en cual esta conectado el Orangután SVP; El segundo puerto es USB Communication Port. En Linux, el nombre del puerto depende de cuantos dispositivos usan el controlador USB CDC ACM para crear un puerto serie virtual y al mismo tiempo la conexión del Orangután SVP.

Después de determinar el puerto ya podrás hacer uso del mismo.

Hay varios programas de terminal disponibles y libres incluyendo PuTTY (Windows o Linux), Tera Term (Windows), y Br@y Terminal (Windows). Puedes usar el que quieras pero antes tienes que configurarlo con el puerto que corresponda.

PuTTY es un programa terminal de Windows que permite enviar y recibir bytes por el puerto serie..



Si necesitas enviar y recibir bytes del tipo non-ASCII usa la utilidad Pololu Serial Transmitter para Windows .

Los usuarios avanzados crean miniaplicaciones y para ello prefieren el programa **kermit** (Windows o Linux).

Puedes escribir un programa que use el puerto serie.

El Microsoft .NET framework para Windows

contiene una `SerialPort` class que hace fácil la lectura y escritura de bytes por el puerto serie. Este es un ejemplo de código C# .NET que usa el USB Communication Port:

```
// Escoge puerto y velocidad en baudios.
// El puerto puedes encontrarlo en el Administrador de dispositivos.
// La velocidad en el puerto USB Communication es irrelevante.
System.IO.Ports.SerialPort port = new System.IO.Ports.SerialPort("COM4", 115200);
// Conexión del puerto
port.Open();
// Pon la línea DTR a 1.
port.DtrEnable = true;
// Transmite dos bytes al AVR: 0x61, 0x62
port.Write(new byte[]{0x61, 0x62}, 0, 2);
// Espera a recibir un byte por la línea RX.
int response = port.ReadByte();
// Muestra que byte se ha recibido.
MessageBox.Show("Received byte: " + response);
// Pon la línea DTR a 0.
port.DtrEnable = false;
// Desconecta el puerto para que otros programas puedan usarlo.
port.Close();
```

Comunicación con el AVR

La escritura de programas para el puerto USB Communications en el AVR la encontramos en dos secciones diferentes de la librería de Pololu AVR C/C++.

Las funciones específicas de la sección del SVP permiten las funciones: `rts_enabled()` y `dtr_enabled()` que devuelven el estado de estas líneas de control. La documentación completa se encuentra en la

Sección 12 de la **Pololu AVR Library Command Reference** . También está en Sección 6.i de la **Pololu AVR C/C++ Library User's Guide** .

La sección Serial Port Communication de la librería contiene varias funciones para la comunicación serie y todas ellas trabajan con el USB Communications port. Más documentación se encuentra en la Sección 9 de **Pololu AVR Library Command Reference** y en Sección 6.j de **Pololu AVR C/C++ Library User's Guide**

9. Uso del puerto serie TTL

La conexión USB del Orangután SVP's tiene un puerto serie llamado Pololu Orangután SVP TTL Serial Port que permite la comunicación en dos vías con un PC y con dispositivos serie TTL-level y velocidades de entre 300 bps a 115200 bps. Este puerto serie puede usarse tanto con un PC como con cualquier otro dispositivo externo y el AVR. Puede usarse también para depurar la comunicación entre el AVR y otro dispositivo externo. Si vas a realizar la comunicación directa entre PC y AVR se recomienda usar el USB Communications Port (ver sección 8).

Los bytes enviados desde el PC a este puerto se transmiten por la línea TX. Los bytes recibidos van por la línea D/RX al ordenador. Los bytes pueden transmitirse como 8 bits, no invertidos, sin paridad y 1 stop bit (8,N,1). Para usar el puerto serie TTL debes saber el nº de puerto asignado por el sistema operativo.

Para determinar el puerto abre /Panel de Control/Sistema y administrador de dispositivos y expande el "Ports (COM & LPT)", y allí encontrarás el Pololu Orangután SVP TTL. Si estuviera compartido con tres dispositivos puedes cambiarlo en propiedades (mira Sección 5.a).

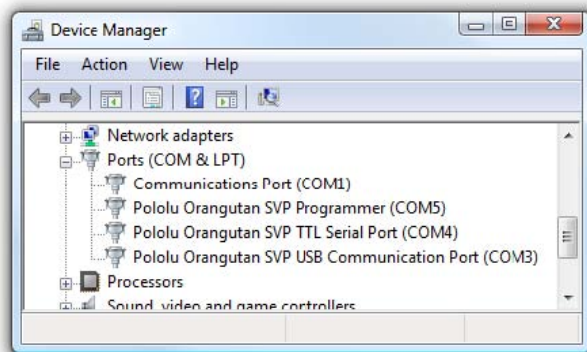
Windows Vista administrador dispositivos.

En Linux, el nombre del puerto depende de cuantos dispositivos usan el controlador USB CDC ACM para crear un puerto serie virtual y al mismo tiempo la conexión del Orangután SVP.

Después de determinar el puerto ya podrás hacer uso del mismo.

Como ya hemos citado hay varios programas de tipo terminal disponibles y libres incluyendo PuTTY (Windows o Linux), Tera Term (Windows), y Br@y Terminal (Windows).

Puedes usar el que quieras pero antes deberás de configurarlo e indicarle el puerto que vas a utilizar para realizar la comunicación.



10. Referencias

Esto es una lista de referencias usadas:

- WinAVR : libre, código abierto C/C++ compilador para AVR's
- AVR Studio : IDE para desarrollo libre del Atmel que usa WinAVR como compilador
- ATmega324PA documentación : ATmega324PA hoja de datos y documentación adicional
- Pololu AVR Librería referencia comandos : información detallada de cada función de la librería.
- Pololu AVR Guía de usuario : instrucciones para la instalación y uso de la librería con ejemplos de código.
- AVR Libc Página principal
- Tutorial: AVR Programación en Mac

Mira la pestaña de referencias del producto Orangután en la página Web que incluye ejemplos par el AVR Studio y novedades.

Finalmente puedes comentarios en el forum correspondiente.

11. Tabla de funcionamiento de motores

Input		Orangutan SVP output			
PD7, PD6	PC7, PC6	M1A, M2A	M1B, M2B	motor effect	LED on
H	H	L	H	“forward”*	green
H	L	H	L	“reverse”*	red
L	H or L	L	L	brake low	none

Recuerda que el concepto “forward” (atrás) es arbitrario y para lo único que sirve es más describir que la rotación del motor se hace en sentido inverso al actual.

Motor 1 se controla por los pins PD7 (Ej. OC2A) y PC7.

Motor 2 se controla por los pins PD6 (Jr. OC2B) y PC6. Pins PD6 y PD7 están conectados al ATmega324's como salidas PWM de 8-bits, las cuales permiten configurar velocidades variables a los motores por temporizadores hardware en lugar de software. Esto libera la CPU para realizar otras tareas mientras la velocidad del motor se mantiene controlada automáticamente por los timers.

Para el control de motores en la Orangután SVP's usa las funciones de la sección **OrangutanMotors** de Pololu AVR C/C++ Library, o escribe tu propio código.

12. USB alimentación

Cuando conectas al PC la conexión USB suministra una alimentación de 5 V. Si hay una fuente externa conectada, la unidad se desconecta de la fuente externa por lo que no afecta al USB. Si solo la alimentación procedente del USB está presente el procesador auxiliar se alimenta del USB pero el AVR y los pins VCC de la placa no tienen alimentación.

Hay una opción disponible para alimentar toda la placa con el USB y consiste en instalar un jumper que conecte Vusb a Vcc. Cuando se alimenta de esta manera el regulador de voltaje absorbe algo de corriente, resultando un VBAT de entre 4.6 – 5.0 V y un VCC alrededor de 4.3 – 4.6 V. En cualquier caso VCC puede ser insuficiente para que se dispare la detección del brown-out en el AVR. En cualquier caso si necesitas alimentar el AVR con el USB puedes cambiar el parámetro detección brown-out para tolerar voltajes bajos.

El uso de un jumper **VCC-VUSB** puede causar problemas porque el Orangután con la alimentación del USB no tiene fuerza suficiente en caso de necesitar más de 100 mA para trabajar. Si la demanda de corriente desde USB es grande, el suministrador principal desconectará la alimentación del puerto. En resumen que el AVR alimentado por el USB no trabajará si precisa mucha corriente.

13. Firmware

El firmware es un programa que utiliza el procesador auxiliar del Orangután SVP's para comunicarse con el Atmel y con el PC via USB además de tareas auxiliares como control de carga de batería. Este programa puede actualizarse para la corrección de errores y añadir nuevas prestaciones.

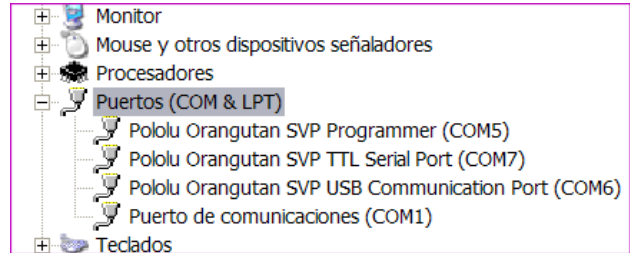
La versión 1.01 corrige algunos errores detectados con el soporte de los encoders de cuadratura al usar el procesador auxiliar. Si usas el procesador auxiliar para las salidas de los encoders recomendamos que actualices la versión para no tener problemas con las lecturas, en caso contrario no es necesario.

La versión 1.00 del Firmware del procesador auxiliar del SVP es la original. Creada en Noviembre de 2009. La versión 1.01 está actualizada con fecha 15 de enero del 2010. Esta versión corrige algunos errores relacionados con el soporte para los encoders ya que en ocasiones provocaba pérdidas de contador al trabajar con dos encoders.

Determinar la versión de firmware

Para determinar la versión que tiene el procesador auxiliar en un PC con Windows, haremos:

1. Conectar el Orangután SVP al PC vía USB.
2. Abre el Administrador de dispositivos (Panel de Control/Sistema/Administrador de dispositivos).
3. Entra en la lista de “Puertos (COM & LPT)” y doble click para ver los puertos utilizador por el Pololu Orangután y doble clic en Pololu Orangután SVP USB Comuni.....
4. En la pestaña Detalles, selecciona el “Identificadores de hardware” en la casilla desplegable.
5. El primer valor mostrado debe ser algo así: USB\VID_1FFB&PID_0087&REV_0100&MI_02. El número después de REV_ es el código actual. Si la revisión es “0100” tienes la versión de 1.00. Si es “0101” tienes la actual firmware 1.01.



Para determinar la versión en un PC con Linux seguiremos los pasos siguientes:

1. Conecta el Orangután SVP al PC vía USB.
2. Corre el comando: `lsusb -v -d 1ffb:0087 | grep bcdDevice`. Mostrará una línea con la versión del firmware y el número que corresponda.

14. Actualización de firmware

Para actualizar el firmware del procesador auxiliar, primero debes descarga la última versión con el nombre de: [Firmware versión 1.01 for the Orangután SVP](#) (27k fichero.pgm)—del 15 de Enero.

Instalar el controlador bootloader en el PC

1. Conecta el Orangután SVP al PC vía USB.
2. Apaga el Orangután SVP's AVR. El led azul se apagará. Tienes que quitar el jumper **VCC-VUSB** si está puesto.
3. La línea del bootloader se muestra en la imagen. Debes conectar esta línea a masa de la siguiente manera. Puedes usar un cable que se conecte a masa por un lado y en el otro extremo conectado con el agujero marcado según la figura, teniendo en cuenta de no tocar ninguna de las partes cercanas. Masa está disponible en varios sitios del SVP (ej. GND de alimentación); comprueba el diagrama del [Orangután SVP Reference Diagram](#) (82k pdf) si no estás seguro. Tendrás que quitar la pantalla de la LCD para acceder a la línea del bootloader (Cuida con las patillas de apoyo de la LCD, situadas en el extremo derecho).

Línea bootloader del Orangután SVP's.



4. Aparecerá que ha encontrado un nuevo dispositivo llamado “*Pololu org06a Bootloader*”.
 - **Windows 7, Vista y Linux:** El controlador para el bootloader se instala automáticamente.
 - **Windows XP:** Cuando aparezca “Encontrado nuevo Hardware”, sigue los pasos 6–8 de la **sección 5. Empezar con el Orangután SVP** para su instalación y para que puedas trabajar con él.
5. Una vez esté instalado el “Pololu org06a Bootloader”, el led verde del USB LED parpadeará como un corazón (dos veces), y verás una entrada para el bootloader en el apartado “Puertos (COM & LPT)” con el nombre “Pololu org06a Bootloader (COM8)”. Recuerda el COM asignado para lo que viene ahora.

Cargar el nuevo firmware

1. Usando un programa de terminal (como [Br@y Terminal](#)) inicia una conexión con ese puerto. En Windows, puedes conocer el puerto a usar (ej. COM8) mirando, como hemos dicho antes, en el administrador de dispositivos. En Linux, puedes saberlo (ej. /dev/ttyACM0) usando el comando `dmesg`. Puedes usar cualquier velocidad de baudios.
2. Teclea las letras, una a una, y en minúsculas desde el terminal para enviar: **fwbootload**. Después de enviar cada letra el bootloader contesta repitiéndola en MAYUSCULAS. Después de terminar la secuencia de letras veras algo así como “**fFWwBBoOoOtTlLoOaAdD**” en la pantalla del programa de terminal.
3. Envía en minúsculas la letra “s”. El bootloader tardará unos segundos en borrar el firmware y seguidamente contesta con una letra MAYUSCULA “S”. Cuida de **NO DESCONECTAR** el dispositivo del PC ni el cable de linea de programación del bootloader durante estos pasos, hasta que la actualización esté completamente realizada.
4. Ahora envía el contenido del fichero firmware descargado. El fichero es un fichero de texto (ASCII) que puede editarse con el notepad y proceder a copiar y pegar en el programa terminal. Alternativamente con el Br@y Terminal puedes usar el botón de “Send File”.
5. Mientras se está enviando el contenido del fichero.pgm el bootloader contesta con una secuencia de caracteres (“...”). El proceso puede tardar alrededor de 5 segundos. Una vez realizada la operación de actualización se mostrará el símbolo (“”) para indicar el fin del proceso.
6. Desconectar el USB del SVP, desconecta el cable a masa, conecta la pantalla LCD y al volver a encenderlo ya puedes usar el nuevo firmware. Puedes comprobar la nueva versión tal como has hecho en el paso *Determinar la versión de firmware*.



Si tienes algún problema ponte en contacto con nosotros <http://www.pololu.com/contact>.