

DESARROLLO DE UN SISTEMA ALTERNO DE CONTROL DE UN PODIO ROBÓTICO CON 4 GRADOS DE LIBERTAD USANDO EL PROGRAMA EVIR DE NAVEGACIÓN REMOTA DESDE UN COMPUTADOR Y JOYSTICK.

INFORME DE AVANCE:

1. Estudio: Para el desarrollo de un sistema alternativo de control del podio robótico, primero se obtienen los requerimientos generales del sistema, esto se hizo recurriendo a la documentación existente del proyecto lo cual se resume en los siguientes requisitos:

- Se requiere el control de 4 servomotores DC con retroalimentación por encoder incremental usando PWM.
- Se requiere la implementación de un algoritmo de control para los motores que permita movimiento fluido y sin vibraciones (control de trayectoria).
- Se requiere poder controlar el podio a través del puerto UART desde un servidor.
- Se requiere el envío de las posiciones de las articulaciones hacia el servidor por medio de un puerto UART.

El diagrama de bloques del sistema se muestra en la Figura 1.

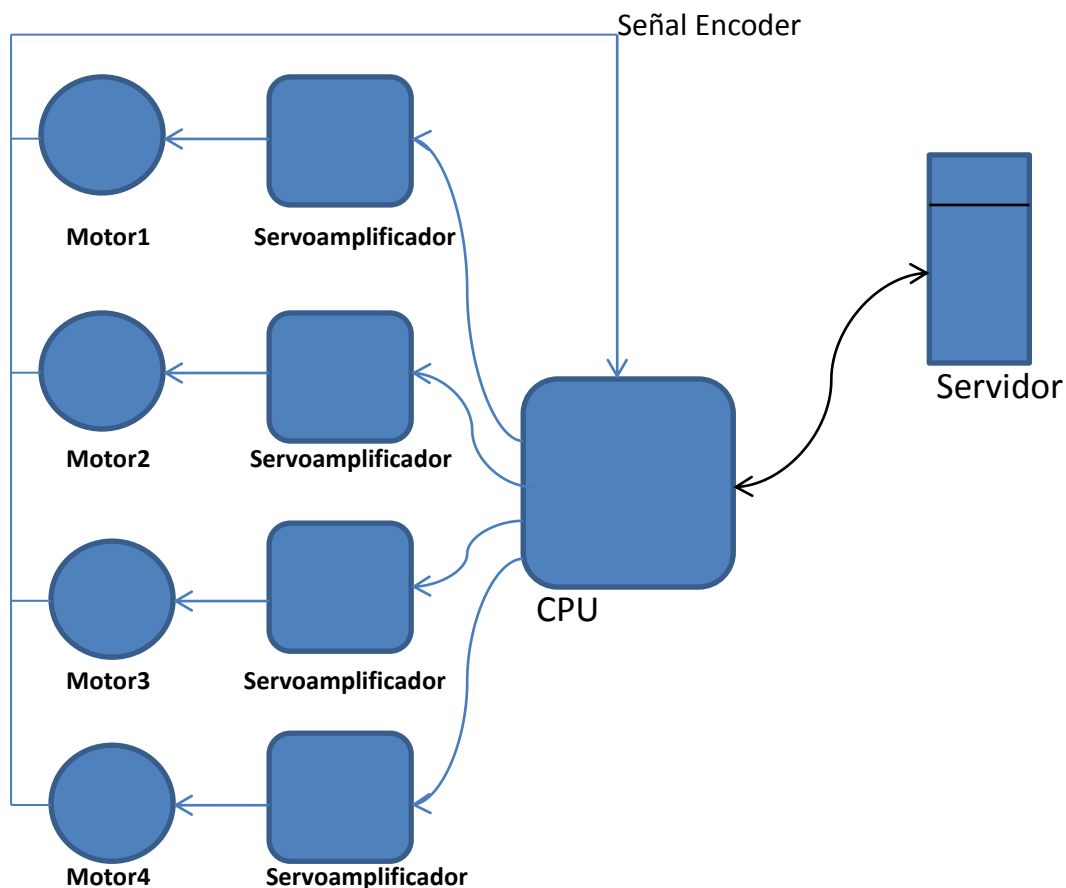


Figura 1. Diagrama de bloques del sistema

2. Selección de la arquitectura: Una vez estudiados los requerimientos se procede con la elección de la arquitectura lo cual implica la selección de un sistema electrónico que cumpla con lo establecido en el ítem 1. Para lo cual se elige el procesador STM32F407 porque cumple con las siguientes características:

- Core: ARM® 32-bit Cortex®-M4 CPU with FPU, Adaptive real-time accelerator (ART Accelerator™) allowing 0-wait state execution from Flash memory, frequency up to 168 MHz, memory protection unit, 210 DMIPS/1.25 DMIPS/MHz (Dhrystone 2.1), and DSP instructions
 - Memories
 - Up to 1 Mbyte of Flash memory
 - Up to 192+4 Kbytes of SRAM including 64-Kbyte of CCM (core coupled memory) data RAM
 - Flexible static memory controller supporting Compact Flash, SRAM, PSRAM, NOR and NAND memories
 - LCD parallel interface, 8080/6800 modes
 - Clock, reset and supply management
 - 1.8 V to 3.6 V application supply and I/Os
 - POR, PDR, PVD and BOR
 - 4-to-26 MHz crystal oscillator
 - Internal 16 MHz factory-trimmed RC (1% accuracy)
 - 32 kHz oscillator for RTC with calibration
 - Internal 32 kHz RC with calibration
 - Sleep, Stop and Standby modes
 - V_{BAT} supply for RTC, 20×32 bit backup registers + optional 4 KB backup SRAM
 - 3×12-bit, 2.4 MSPS A/D converters: up to 24 channels and 7.2 MSPS in triple interleaved mode
 - 2×12-bit D/A converters
 - General-purpose DMA: 16-stream DMA controller with FIFOs and burst support
 - Up to 17 timers: up to twelve 16-bit and two 32-bit timers up to 168 MHz, each with up to 4 IC/OC/PWM or pulse counter and quadrature (incremental) encoder input
 - Debug mode
 - Serial wire debug (SWD) & JTAG interfaces
 - Cortex-M4 Embedded Trace Macrocell™
 - Up to 140 I/O ports with interrupt capability
 - Up to 136 fast I/Os up to 84 MHz
 - Up to 138 5 V-tolerant I/Os
 - Up to 15 communication interfaces
 - Up to 3 × I²C interfaces (SMBus/PMBus)
 - Up to 4 USARTs/2 UARTs (10.5 Mbit/s, ISO 7816 interface, LIN, IrDA, modem control)
 - Up to 3 SPIs (42 Mbits/s), 2 with muxed full-duplex I2S to achieve audio class accuracy via internal audio PLL or external clock
 - 2 × CAN interfaces (2.0B Active)
 - SDIO interface
 - Advanced connectivity
 - USB 2.0 full-speed device/host/OTG controller with on-chip PHY
 - USB 2.0 high-speed/full-speed device/host/OTG controller with dedicated DMA, on-chip full-speed PHY and ULPI
 - 10/100 Ethernet MAC with dedicated DMA: supports IEEE 1588v2 hardware, MII/RMII
 - 8- to 14-bit parallel camera interface up to 54 Mbytes/s
 - True random number generator
 - CRC calculation unit
 - 96-bit unique ID
 - RTC: subsecond accuracy, hardware calendar
- Principalmente notamos que el procesador tiene puertos 4 UART, PWM, 140 puertos I/O, 168MHz de operación, 1Mbyte de FLASH. En síntesis todo lo que se necesita en un solo microcontrolador.

Para el proyecto se compra el sistema de desarrollo STM32F4DISCOVERY (Figura 2) el cual contiene toda la circuitería necesaria para el proyecto + programador en una sola tarjeta.



Figura 2. Tarjeta Discovery. Imagen tomada de <http://www.st.com/web/catalog/tools/FM116/SC959/SS1532/PF252419>

3. Diseño de hardware adicional: Los niveles de voltaje entre las tarjetas servoamplificadoras, encoder, UART y el CPU son diferentes, se hace necesario el diseño de un circuito acondicionador que “*nivela*” las tensiones para que se pueda lograr la comunicación sin daños en el sistema, para lo cual se diseña un circuito (SHIELD) que se acopla a los terminales de pin de la tarjeta DISCOVERY (FIGURA 3)

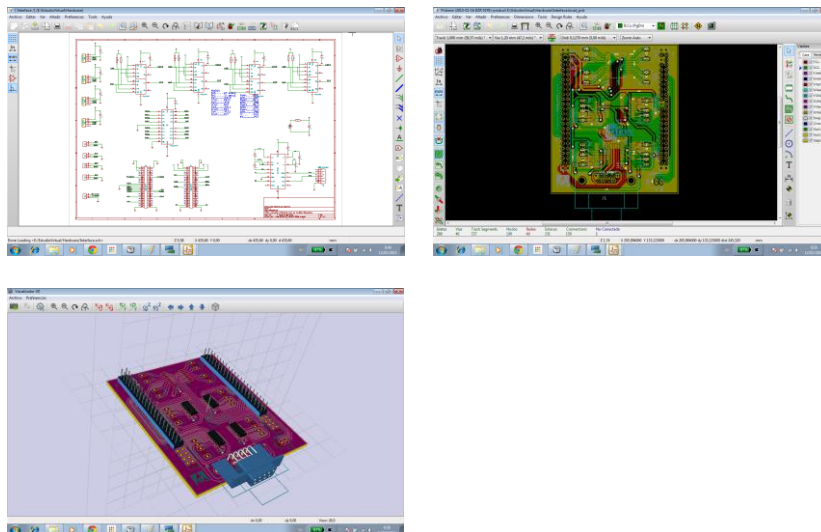


Figura 3. Circuito de interfaz entre la tarjeta DISCOVERY y el podio robótico.

4. Programación del hardware: Se comienza con la implementación de un intérprete de javascript en el microcontrolador STM32F4 para lo cual se descarga e instala el binario ESPRUIINO de <http://www.espruino.com/> el cual es gratuito y libre.

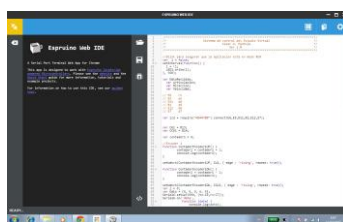


Figura 4. Entorno de desarrollo en javascript para microcontroladores de Espruino.