

Configuración y análisis del sistema de transmisión y recepción de texto plano a través del Picosatélite CubeSat Kit y el MHX2420 SL

Juan Fernando Balarezo Serrano

Departamento de Eléctrica y Electrónica, Escuela Politécnica del Ejército. Av. El Progreso S/N, Sangolqui, Ecuador

Resumen: El presente artículo describe la configuración, manejo del software e integración del Picosatélite *CubeSat Kit*, entre el software necesario para llevar a cabo las tareas antes mencionadas, destaca el ambiente de desarrollo *CrossWorks* que sirve como una herramienta completa para la programación del microcontrolador MSP430 de *Texas Instruments*, el cual es el cerebro y núcleo del Picosatélite, mediante el cual se configuran y administran las diferentes interfaces, herramientas y funciones que el *CubeSat Kit* posee e integración del Sistema Operativo en Tiempo Real *Salvo*. Finalmente se configuran los módems MHX 2420 para establecer una aplicación de comunicación inalámbrica en la banda de 2.4GHz, que transmita texto plano en tiempo real.

1. Configuración de los Módulos MHX2420 Mediante el CubeSat Kit

La configuración a través del CubeSat Kit consiste en programar el microcontrolador Texas Instruments MSP430, integrado al tablero de desarrollo, de tal forma que este envíe los comandos AT de configuración al módulo de comunicaciones MHX2420 acoplado a dicho tablero. Para realizar esta configuración es necesario tener instalado un compilador en C que sea compatible con el microcontrolador, en este caso manejamos el Ambiente de Desarrollo Crossworks MSP430 de Rowley Associates, el cual está dedicado exclusivamente para la serie del microcontrolador con el que se está trabajando.

Dentro de la programación del Tablero de Desarrollo de CubeSat Kit se creó una tarea mediante la cual se programa y configura el módulo MHX2420 como se explica a continuación:

- Inicio de la tarea, función tipo void, no devuelve ningún valor al llamarla.
- Se envía mensajes a la interfaz de debug, la cual es la interfaz RS232 del Tablero de Desarrollo del CubeSat Kit, para indicar que procedimientos se están llevando a cabo de manera informativa. Esto se hace mediante una de las funciones propias del

CubeSat Kit que incluye en sus librerías, dicha función se ejecuta de la siguiente forma: `user_debug_msg(STR_TASK_MHX_TALK “Mensaje”);` y de forma automática el mensaje es enviado a la interfaz RS232 del Tablero de Desarrollo del CubeSat Kit. Estos mensajes se pueden visualizar conectando una computadora a la interfaz RS232 mediante un cable serial directo y usando un terminal ASCII.

- Se ingresa un retardo de 5 segundos incluido un tiempo aleatorio. Se debe evitar ingresar un retardo en 0 puesto que esto terminaría con la tarea. Para llamar a retardos de tiempos se usa la función `OS_Delay(TiempodeRetardo);` por ejemplo si deseamos insertar un retardo de 1 segundo se lo hace de la siguiente forma: `OS_Delay(100)`. Todas las funciones que comiencen con OS son propias del sistema operativo en tiempo real (RTOS) SALVO.
- Se verifica si la interfaz MHX RF (Es la interfaz de comunicación entre el CubeSat Kit y el MHX2420) está siendo usada mediante un semáforo binario propio del RTOS SALVO.
- Se habilita la comunicación hacia el módulo MHX2420 y se enciende el mismo. Mediante las funciones `csk_mhx_open()` y `csk_mhx_pwr_on()` respectivamente. Las funciones que empiezan con csk son propias de las librerías del CubeSat Kit. Y, se coloca el valor 1 a la variable `csk_status.mhx_connected`, la cual sirve de indicador que se estableció comunicación con el módulo MHX2420.
- Se envía un mensaje de aviso a la interfaz RS232 del Tablero de Desarrollo indicando que se estableció comunicación y se encendió el módulo MHX2420. Y se espera 1 segundo para que encienda el módulo y se notifica mediante el respectivo mensaje.
- Mediante el envío del comando `mhx` al módulo MHX2420 se lo coloca en modo de comandos, y esperamos un segundo por la respuesta del módem. Mediante la línea `csk_uart1_puts(“Comando”)` se envía un comando al MHX2420.
- Se envía el comando `AT&F7&K0` al módem MHX2420 de tal forma que se lo configura como modo Esclavo, mediante el comando `AT&F7` que configura automáticamente los registros para que el equipo trabaje en dicho modo y los demás registros como potencia de salida y tasas de transmisión que no definen el tipo de modo de

operación del módem son configurados con sus valores por defecto. Adicionalmente añadimos &K0 al comando para deshabilitar el handshaking de hardware.

- Una vez configurado el módem como Esclavo, salimos del modo de comandos enviando el texto ATA al MHX2420 y cambiamos a modo de transmisión de datos para que empiece la aplicación de envío de texto plano desde el CubeSat Kit al módulo MHX2420 remoto una vez establecido el enlace entre los módulos.
- Se realiza el envío de datos en forma de texto plano, se envían mensajes informativos, y se llama a la tarea propia del CubeSat Kit que realiza la medición de temperatura en grados centígrados del microcontrolador, la cual es `onchip_temp_deg_c()`, el resultado de la misma es enviada al remoto. Para enviar datos al remoto se usa la línea `sprintf(strTmp, "Texto a enviar")`. Los datos se envían al módem MHX2420 para que el mismo los envíe mediante el enlace inalámbrico al módulo remoto y a su vez los datos son enviados a la interfaz RS-232 del Tablero de Desarrollo del CubeSat Kit con el fin de constatar la integridad de los datos recibidos en el equipo remoto, es decir comprobar que los datos enviados mediante el enlace inalámbrico sean los mismos que el microcontrolador genera.
- Después de haber enviado el texto al remoto, se vuelve al modo de comandos mediante el envío del comando `+++` al MHX2420.
- Se apaga el segmento de transmisión del módem MHX2420.
- Se deshabilita la comunicación entre el CubeSat Kit hacia el módulo MHX2420 y finaliza la tarea.

En resumen, esta tarea sirve para establecer comunicación con el módulo MHX2420 acoplado en el tablero y establecer el enlace inalámbrico con un módem remoto.

Después de un retardo de 5s la tarea intentará establecer comunicación con la interfaz MHX. Cuando se lleva a cabo con éxito se enciende el módulo MHX2420 y se establece el enlace inalámbrico con el equipo remoto, se transmiten los datos (texto plano) de un módem al otro, luego se apaga el módulo MHX2420 y se libera la comunicación hacia la interfaz MHX. Si un modem MHX2420 remoto está conectado inalámbricamente, los LEDs RSSI del Tablero

de Desarrollo del CubeSat Kit se encienden durante los 5s que el MHX2420 está encendido.

2. Consideraciones de Software y Hardware

Dentro del hardware del *CubeSat Kit*, contamos con el dispositivo que permite cargar la configuración creada mediante el ambiente de desarrollo *CrssWorks* en el microcontrolador MSP430, es propio de *Texas Instruments* y su nombre es JTAG. El dispositivo JTAG es necesario para comunicar la computadora con el Tablero de Desarrollo del *CubeSat Kit*, específicamente el microcontrolador MSP430, para cargar la configuración realizada en el ambiente de desarrollo *CrossWorks*. Dicho dispositivo es propio de *Texas Instruments* y está incluido dentro de los elementos del *CubeSat Kit*, el modelo del equipo es MSP-FET430UIF, el cual es una herramienta de emulación que permite iniciar el desarrollo de aplicaciones en el microcontrolador MSP430. Incluye una interfaz USB la cual sirve para programar y depurar el sistema del MSP430 a través de su interfaz JTAG.

Posee una memoria flash en la cual se almacena la configuración que va a ser cargada en el microcontrolador en tiempo real, la cual puede ser borrada y reprogramada las veces que sea necesario, incluso sin necesidad de fuentes externas de poder ya que se alimenta mediante el puerto USB de la computadora puesto que requiere únicamente entre 1.8 a 3.6 V y 100mA [4].

3. Configuración del Microcontrolador MSP430 de Texas Instruments.

Se elaboraron archivos de código fuente y cabeceras para desarrollar una aplicación que transmite datos (texto plano) generados en el microcontrolador mediante un módem MHX2420 inalámbrico, el cual se comunica en la banda de 2.4GHz, usando *Frequency Hopping Spread Spectrum*, con un módem idéntico (MHX2420), tales elementos código y cabecera son: archivos de código fuente: `main.c`, `task_cmd.c`, `task_mhx.c`; cabeceras: `events.h`, `tasks.h`, `main.h`, `salvocfg.h`

A su vez la aplicación permite que el Tablero de Desarrollo del *CubeSat Kit* pueda llevar a cabo el reconocimiento de caracteres mediante la interfaz RS232 que posee usando un terminal ASCII de una computadora conectada al Tablero. Esta interfaz también despliega

información de los procesos que se están llevando a cabo en el microcontrolador, en el *CubeSat Kit* y en el sistema en general.

4. Configuración de los módems MHX vía comandos AT

El módulo MHX2420 integrado al tablero de desarrollo de Microhard Corp. está configurado como Maestro mediante el comando AT&F6 que configura automáticamente los registros para que el equipo trabaje en dicho modo y los demás registros como potencia de salida y tasas de transmisión que no definen el tipo de modo de operación del módem son configurados con sus valores por defecto. Adicionalmente digitamos el comando AT&K0 para deshabilitar el handshaking de hardware.

La configuración del módulo MHX2420 acoplado al CubeSat Kit mediante el envío del comando AT&F7&K0, configura los registros del módem con los valores por defecto para que el mismo trabaje en modo Esclavo y se deshabilita el handshaking de hardware en ambos dispositivos.

Tabla. 3: Configuración y Registros modo Maestro

Descripción	Registro	Valor
Modo de operación	S101=0	Modo Maestro
Tasa de Transmisión Inalámbrica	S103=2	172800 bps
Tiempo de Salto de Frecuencias	S109=9	20 ms
Tamaño Máximo del Paquete	S112=255	255 bytes
Dirección de Destino	S140=2	2 (Remoto)
Modo de Transmisión Serial	S142=0	Mediante la interfaz RS232
Brillo de los LEDs(%)	S149=100	100%
Velocidad de Transmisión de le Interfaz Serial	S102=7	9600 bps
Dirección de Red	S104=1234567890	1234567890
Potencia de Transmisión(dBm)	S108=30	30 dBm
Formato de los Datos (Interfaz RS232)	S110=1	8N1
Número de Retransmisión de Datos	S113=5	5

Descripción	Registro	Valor
Tipo de Red	S133=1	Punto a Punto

Tabla. 4: Configuración y Registros modo Esclavo.

Descripción	Registro	Valor
Modo de operación	S101=2	Modo Esclavo
Tasa de Transmisión Inalámbrica	S103=2	172800 bps
Dirección de Unidad	S105=2	2
Tiempo de Salto de Frecuencias	S109=9	20 ms
Tamaño Máximo del Paquete	S112=255	255 bytes
Dirección de Destino	S140=1	1 (Maestro)
Modo de Transmisión Serial	S142=0	Mediante la interfaz RS232
Brillo de los LEDs(%)	S149=100	100%
Velocidad de Transmisión de la Interfaz Serial	S102=7	9600 bps
Dirección de Red	S104=1234567890	1234567890
Potencia de Transmisión(dBm)	S108=30	30 dBm
Formato de los Datos (Interfaz RS232)	S110=1	8N1
Número de Retransmisión de Datos	S113=5	5
Tipo de Red	S133=1	Punto a Punto

5. Integración Hardware del *CubeSat Kit*

Una vez configurado el microcontrolador del Tablero de Desarrollo del CubeSat Kit, debemos acoplar el módulo MHX2420 a dicho tablero y la antena tipo *Rubber Duck* a dicho módem, también se debe conectar una computadora mediante un cable serial DB9 hembra-hembra a la interfaz RS232 (de ser necesario se debe usar un conversor de interfaz de DB9 a USB), como se muestra en la figura 1.



Fig. 1: Hardware del *CubeSat Kit* integrado.

6. Análisis de Resultados

Las pruebas realizadas indicaron que todo lo que se configuró, en el ambiente de desarrollo *CrossWorks* y fue aplicado al microcontrolador MSP430 para que administre y controle el sistema *CubeSat Kit*, funcionó correctamente, mediante el uso de terminales ASCII, tanto en la interfaz RS232 del *CubeSat Kit* como en la interfaz RS232 del Tablero de Desarrollo de *Microhard Corporation*, se observó la entrega de datos sin ningún tipo de errores o alteraciones, con lo que la integridad de los datos no fue afectada en ningún momento mientras se estableció el enlace inalámbrico.

El módulo MHX2420, realiza un salto de frecuencias cada 20ms según lo configurado en los registros de los módems.

El MHX2420 posee 202 canales de frecuencia en los que establece la comunicación inalámbrica y en los cuales realiza el salto de frecuencias. El primer canal está ubicado en la frecuencia 2.4016GHz con separaciones de 400kHz entre cada canal, siendo la frecuencia del último canal 2.4820GHz. La frecuencia (F) de cada canal es:

$$F = 2401.6 + [(n - 1) \times 0.400] \text{ MHz} \quad [1]$$

En la tabla 5 se puede ver a qué canal (n) pertenecen las frecuencias capturadas en el analizador de espectros.

$$n = [(F - 2401.6) / 0.400] + 1 \quad [2]$$

Tabla 5: Frecuencias capturadas y su canal correspondiente.

Frecuencia(MHz)	Canal
2415.5	36
2435.5	86
2440.5	98
2465.5	161

El texto plano que se recibiría es como se muestra en la figura 2.

```

COM3 - PuTTY
0:0000001222 task_mhx_talk:  CubeSat Kit ESCUELA POLITECNICA DEL EJERCITO ...
0:0000001320 task_mhx_talk:  Autor: Juan Fernando Balarezo
0:0000001420 task_mhx_talk:  Directores:
0:0000001520 task_mhx_talk:  Ing. Dario Duque
0:0000001620 task_mhx_talk:  Ing. Vanessa Vargas
0:0000001721 TEMPERATURA MEDIDA EN EL CHIP: 24C.

```

Fig. 2: Datos recibidos en el modulo MHX2420 remoto.

7. Conclusiones

Se ha configurado e integrado el Picosatélite *CubeSat Kit*, basado en el RTOS (Sistema Operativo en Tiempo Real) *Salvo* usando el microcontrolador TI-MSP430, con módems MHX 2420 en la banda de 2,4 GHz, desarrollando una aplicación de transmisión de texto plano en la banda mencionada con la tecnología *Frequency Hopping Spread Spectrum*. Donde además se desarrolló una aplicación de lectura de caracteres usando la interfaz RS232 del Tablero de Desarrollo del *CubeSat Kit*, la cual a su vez también está configurada de tal forma que entrega datos sobre el estatus de los procesos que lleva a cabo el sistema.

Se ha configurado el software del sistema embebido del *CubeSat Kit* usando el Tablero de Desarrollo con el microcontrolador RISC-16 bits MSP430 de *Texas Instruments*.

Mediante el RTOS *Salvo* se desarrollaron tareas que trabajan en tiempo real, en ambientes asincrónicos y multitarea para el *CubeSat-ESPE*. Se configuraron los módems MHX2420 acoplados al Tablero de Desarrollo del *CubeSat Kit* y al Tablero de Desarrollo de *Microhard Corporation* de tal forma que se estableció un enlace inalámbrico en la banda de 2.4GHz.

8. Bibliografía

- [1] Kalman, A.E., SALVO Users Manual. Version 4.2.2. USA 2010. Pumpkin, Inc. Tomado de Internet: <http://www.pumpkininc.com/content/doc/man>

ual/SalvoUserManual.pdf

[2] MHX 2420 Operating Manual.
Microhard Systems. Canada 2007.

[3] CubeSat Kit User Manual. USA 2005.
Pumpkin Inc. Tomado de Internet:
<http://www.cubesatkit.com/docs/cubesatkitmanual.pdf>

[4] MSP430 Hardware Tools User's Guide.
2009. Texas Instruments. Tomado de Internet:
<http://www.ti.com/lit/ug/slau278i/slau278i.pdf>

9. Biografía



Juan Fernando Balarezo, nace en la ciudad de Quito el 12 de Octubre de 1987.

En el año 2005, obtiene el título del Bachillerato Internacional en el Colegio Intisana. Actualmente es egresado de la facultad de Electrónica de la ESPE y se encuentra presentando su proyecto de grado.

Realizó sus pasantías en la empresa Iseyco C.A. donde actualmente está trabajando, empezó en el Departamento Técnico y en la actualidad labora como Asistente de Ingeniería en el área de Preventa de la empresa.