

PAC 7

Tareas realizadas

Software de LPC1769

BlueGEAR 1.0

Este software conecta con VGEAR mediante protocolo VAP para solicitar datos de la ECU. Por otra parte atiende peticiones VAP recibidas por Bluetooth mediante el módulo HC05. Por último muestrea el ADC para comprobar las aceleraciones del acelerómetro MMA7036.

Toda la ejecución está supervisada por un Watchdog de manera que cuando el microcontrolador se bloquea es reiniciado.

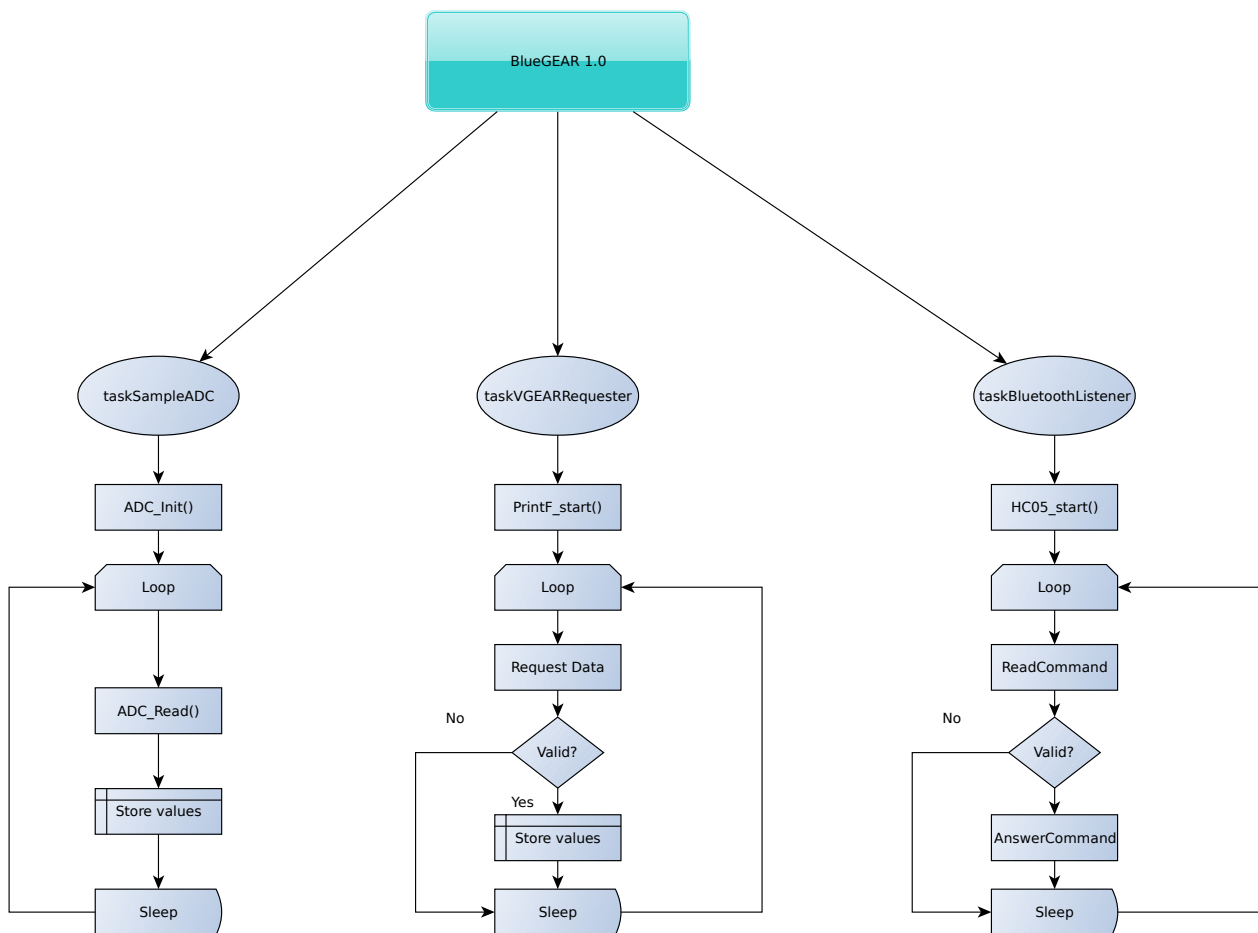


Figura 1: BlueGEAR 1.0.

BlueGEAR 1.1

BlueGEAR 1.1 no se conecta a VGEAR sino que utiliza el módulo KWP2000 de la biblioteca LPC1769_UOC_Library para conectarse a la ECU de la Suzuki M800.

LPC1769_UOC_Library

Esta biblioteca se ha diseñado siguiendo un esquema orientado a objetos permitiendo usar los módulos desde diferentes puntos y con diferentes dispositivos de manera segura (protección con mutex).

Los drivers se inicializan asociándoles una UART y es el llamador el responsable de proporcionar este identificador de UART para usar las funciones del driver.

Para no repetir código no se han encapsulado las funciones de escritura de manera que se tiene que usar el módulo Printf en los drivers HC05 y KWP2000.

Como consecuencia de este diseño no se tiene encapsulada toda la funcionalidad en los módulos y resulta más incómodo para el usuario de la biblioteca.

Driver HC05

Se ha implementado un driver para el módulo HC05 con control del “keyPin” y del “powerPin”.

Driver ADC

Se implementa un driver para el uso del conversor analógico digital que integra el LPC1769.

Driver Timer

Este driver permite usar el Timer0 del LPC1769. Se puede ampliar para usar el resto de timers.

VGEAR

Consiste en dos módulos:

vgear_command.c. Se encarga de parsear comandos VAP recibidos por Bluetooth.

vgear_request.c. Se encarga de solicitar datos a VGEAR mediante protocolo VAP.

KWP2000

Este módulo es un “port” de VGEAR de Arduino. Se encarga de implementar la conexión al bus ISO14230 de la Suzuki M800. Todos los tiempos han sido ajustados respecto a lo probado en VGEAR.

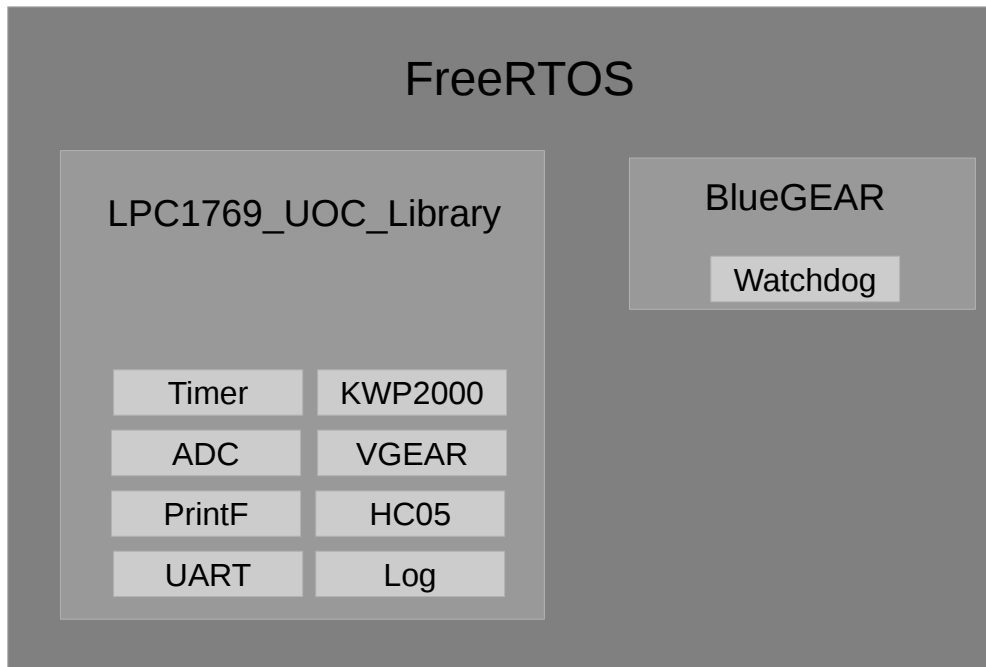


Figura 2: Esquema del software

Android

VGEAR Android 1.0

Esta aplicación para Android está escrita con Qt5. Se conecta a BlueGEAR 1.0 mediante Bluetooth para solicitar datos por protocolo VAP de la ECU y los muestra en pantalla. Además usa el GPS del dispositivo para comprobar los datos recibidos. Adicionalmente usa el acelerómetro del dispositivo para mostrar las aceleraciones.

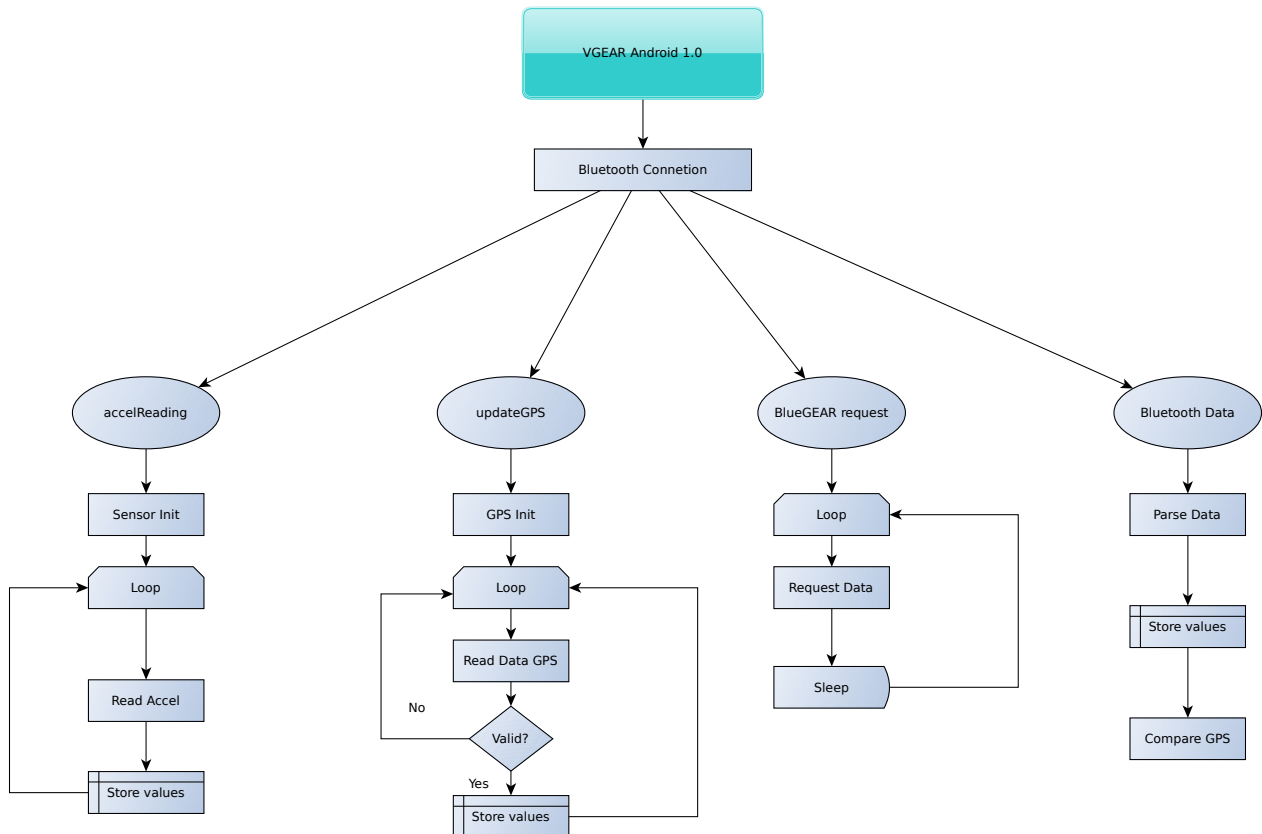


Figura 3: VGEAR para Android 1.0.

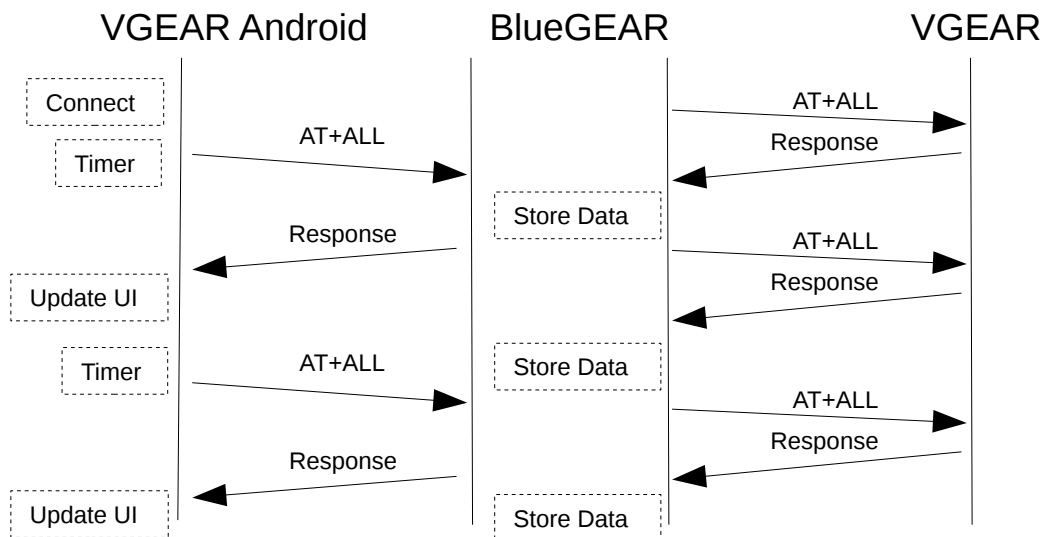


Figura 4: Comunicación entre VGEAR 1.0, BlueGEAR 1.0 y VGEAR.

VGEAR Android 1.1

Esta versión realiza las mismas acciones que VGEAR Android 1.0 pero reduciendo la latencia en las comunicaciones gracias a que no solicita datos sino que espera que BlueGEAR 1.1 los envíe de manera activa.

Además contiene mejoras gráficas, una organización mejor de la información y una barra de progreso para mostrar la apertura de la mariposa de admisión.

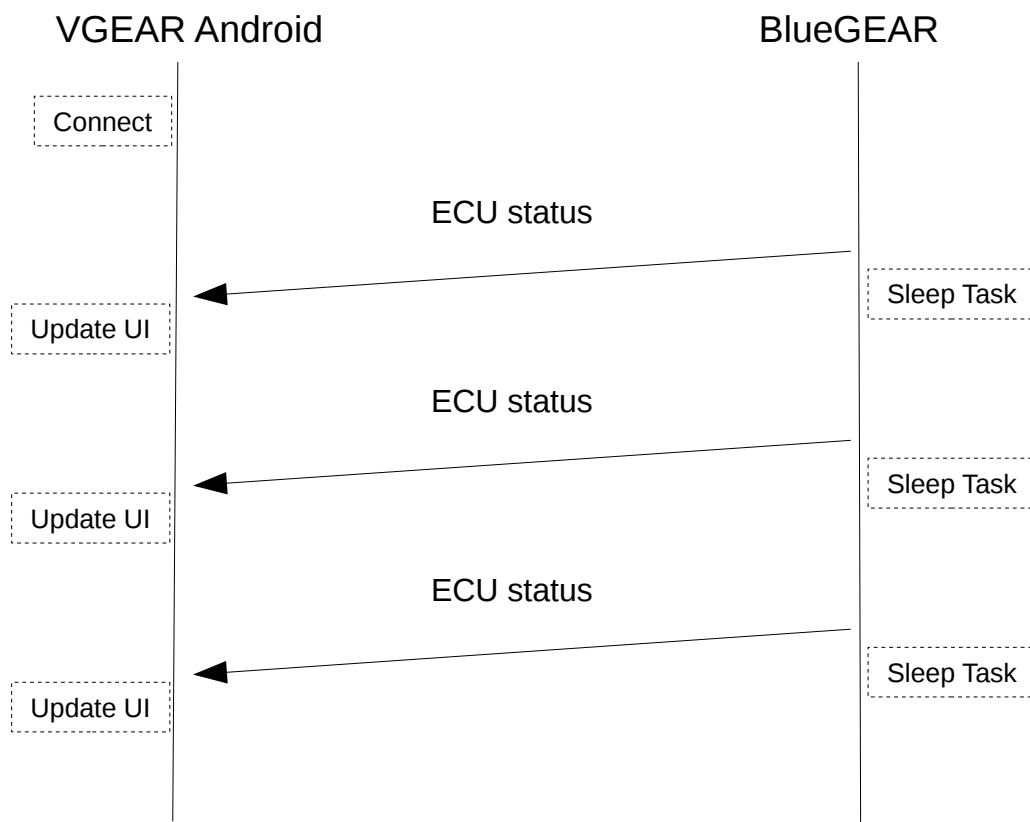


Figura 5: Comunicación entre VGEAR para Android 1.1 y BlueGEAR 1.1.

Testing

Para verificar el correcto funcionamiento de los driver se han realizado diversas aplicaciones de prueba en el LPC1769.

Para verificar la comunicación serie y el protocolo VAP se han realizado programas de prueba tanto en Arduino como en LPC1769. Una vez verificado se portó el código a VGEAR para Android.

Autoevaluación

Puntos complicados

Uno de los puntos más complejos fue la conexión con Bluetooth mediante Qt5 en Android. Se tuvo que usar la interfaz JNI de Java para poder acceder al listado de dispositivos conectados ya que Android no ofrece esa interfaz en su NDK. En concreto el método `getKnownDevices` de `BluetoothClient.cpp` implementa esta funcionalidad.

Otro punto complicado fue averiguar que al linkar con `redlib` en `semihost` las interrupciones pueden ser deshabilitadas provocando que, por ejemplo, la lectura de la UART fuera errónea.

Bugs y errores

No se han detectado errores.

Warnings

Existen diferentes warnings de castings que se han solucionado usando `stdint.h`.

Existe otro warning acerca de que `sprintf` descarta el cualificador `volatile` pero en ese contexto no hay problema ya que está mutexado.

Existen otros warnings acerca de variables declaradas que no se usan, en concreto se refieren a parte del protocolo VAP que no ha sido implementado.

Tiempo dedicado

- 2h Ampliación VGEAR Arduino con VAP
- 40h BlueGEAR 1.0
- 48h BlueGEAR 1.1
- 24h VGEAR Android 1.0
- 6h VGEAR Android 1.1
- 3h Software de test

Conclusión

Se han cumplido los objetivos del PFC. Todo el software se ha probado exhaustivamente y cumple la funcionalidad propuesta desde un principio.