

Sistema eChook

1 Objectivo

O sistema eChook, apesar da sua utilidade, constitui um gasto desnecessário por já se possuir a maior parte dos componentes além de ter um preço considerável. Assim, foram investigadas alternativas a este sistema.

2 Descrição do eChook

A descrição do sistema eChook como apresentado no site é: *It consists of an Arduino based PCB sold through the Greenpower online store and an (Android Only) app that displays and logs the data. If the phone has a data connection it can also push the data to cloud services enabling real time telemetry from the car to the pits - or anywhere with an internet connection!*

O sistema é open source e é possível encontrar [documentação](#) e descrições de hardware e software no [GitHub](#).

O eChook permite fazer análise em tempo real dos seguintes parâmetros:

Corrente [I]	Throttle Input [%]	Gear Number and Ratio
Tensão Total [V]	Travões [On/Off]	Temperatura 1 [°C]
Tensão Bateria 1 [V]	RPM do Motor [RPM]	Temperatura 2 [°C]
Tensão Bateria 2 [V]	Velocidade da roda [m/s]	Posição GPS
Amp Hora [Ah]	Distância [m]	Número de Voltas

Table 1: Logs eChook

2.1 Hardware

Em termos de hardware o sistema contém uma *custom PCB* com as interfaces para os sensores, um Arduino Nano e um módulo *Bluetooth* para ligar à app Android. A placa é alimentada pelas baterias de 24V do carro. É necessária a instalação do software *RS DesignSpark PCB* para abrir os ficheiros que descrevem a placa.

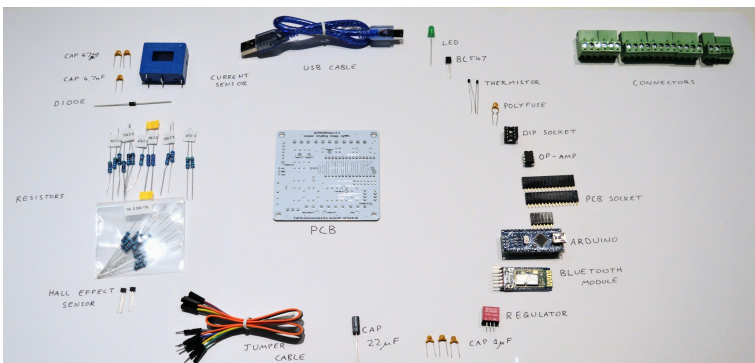


Figure 1: Componentes.

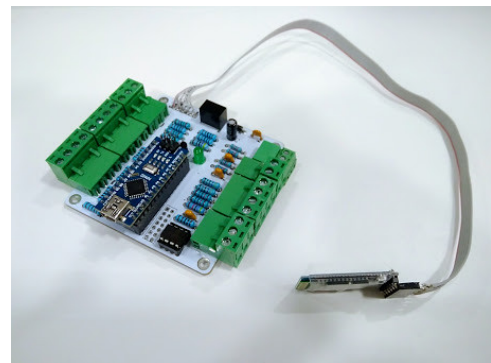


Figure 2: Montagem Final.

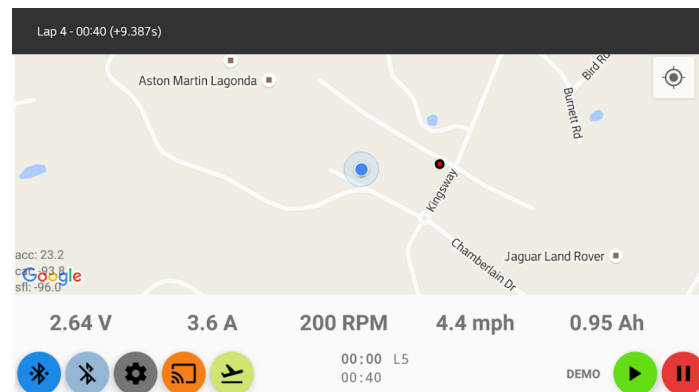
2.2 Software

O eChook usa o Arduino Nano, estando o código disponibilizado completo para utilização, sendo apenas necessário fazer o download de algumas bibliotecas. A aplicação para Android estabelece a conexão entre o telemóvel e o arduino através de *Bluetooth* e é usada como um display. É recomendada a utilização deste telemóvel no carro de forma a evitar problemas de conectividade. Ao fazer o upload do código para o arduino é preciso também fazer um setup deste módulo de *Bluetooth* e se o telemóvel conseguir ligar-se a uma rede WiFi ou por dados móveis a aplicação consegue fazer o upload da informação em tempo real para a internet. É também possível enviar para o *Google Drive* ou para a *DropBox* um ficheiro *.csv* com um resumo da sessão.



GPS

Uma das funcionalidades da aplicação do eChook é a posição no mapa, que precisa de calibração, e o número de voltas efectuadas. Para isto, antes da corrida é marcada uma posição de "observador" no mapa no meio do circuito perguntando depois se as voltas serão feitas no sentido horário ou anti-horário. Quando o carro começa a mover-se a aplicação conta o tempo e analisa em graus a posição do carro em relação ao "observador". Sempre que o carro passa por essa posição uma volta é adicionada.



Telemetria

De momento o eChook usa o *dweet.io*, serviço com o qual não tem qualquer afiliação, de forma a permitir a visualização dos dados em tempo real. A aplicação do eChook envia os dados para um canal *dweet* e qualquer pessoa com acesso ao ID fornecido pode ver os dados de um determinado carro. O problema desta plataforma é o facto de os dados serem públicos sendo, por isso, a localização GPS impedida como forma de protecção. Também é possível fazer o host de um website em específico modificando a *Source* no código fornecido no GitHub. A demo é algo deste género cujo design é feito posteriormente.

2.3 Vantagens e Desvantagens

Como referido acima, o eChook constitui um sistema já completamente implementado, sendo também possível adicionar ou modificar tanto o código como o hardware se pretendido.

Além do preço, o seu maior problema é o facto de usar *Bluetooth* para ligar o arduino ao telemóvel, o que pode não ser a forma de conexão mais fiável durante uma corrida.

3 Alternativas

3.1 Bluetooth

O *Bluetooth* é o método usado pelo sistema eChook e, apesar de ser uma possibilidade, não será abordado, mesmo que outras formas de comunicação sem fios sejam descritas.

3.2 GSM Module

Este módulo oferece a possibilidade de a partir do arduino mandar sms's ou fazer chamdadas, mas para uma transmissão de dados tão rápida torna-se imprático, mas pode ser explorado para casos específicos. Além disto, permite a utilização de dados móveis e uma conexão à internet sem ter de recorrer a um módulo de WiFi.

No fundo, o módulo consitui um shield que funciona como um telemóvel do ponto de vista de uma operadora de telecomunicações, sendo, por isso, necessário um cartão SIM funcional com um contracto ou acordo com essa mesma operadora. No seguinte link, [GSM Shield](#), encontra-se uma explicação mais detalhada do processo de implementação, bem como código para o arduino.

Os preços deste módulo rondam os 12 Euros.



Figure 6: Bluetooth Shield.

3.3 Arduino WiFi Module

O módulo WiFi estabelece uma ligação entre o arduino e a internet sendo depois possível enviar informação e fazer conexão com um telemóvel. Alguns modelos deste módulo possuem um slot para um cartão micro-SD. A partir daqui será necessário uma aplicação ou servidor de forma a receber os dados do arduino que serão abordadas mais tarde.

Os preços do módulo rondam os 7 Euros.

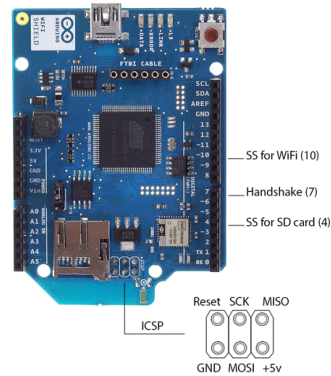


Figure 7: WiFi Shield.

Nos seguintes links há duas implementações desta funcionalidade: [Arduino WiFi Connection](#) e [Arduino WiFi Shield](#).

3.4 "On The Go" - OTG Cable

A ligação através de um cabo é a ideal, uma vez que a probabilidade de perda de conexão é muito menor em comparação com uma ligação por *Bluetooth* ou WiFi.

Neste link está uma implementação da ligação de um arduino por cabo com um telemóvel android, mas apenas para enviar dados: [Arduino Connection with Android Studio](#). Esta [Aplicação do Android Studio](#), que precisa de modificações para receber dados, envia, da mesma forma, informação e comandos para o arduino.

Este cabo custa mais ou menos 4 Euros e são necessários o cabo OTG e um cabo USB para ligar ao telemóvel.

3.5 Conexão

Será sempre necessário ligar o arduino a um telemóvel presente no carro, por exemplo por uma aplicação criada para o Android, ou directamente a um web server como no sistema eChook de forma a ter acesso aos dados em tempo real.

Web Server

O *Google Firebase* permite a recepção de dados em tempo real directamente do arduino através do módulo de WiFi. É necessário um anfitrião e uma chave para haver uma conexão. Numa parte deste [tutorial](#) há uma explicação de como ler dados do arduino através desta plataforma.

Similar ao *Google Firebase* pode ser usado o *Ubidots* que permite também ligar ao telemóvel.

Há também implementações que conectam o arduino através de WiFi ao *Google Spreadsheets*. Nesta caso é necessário trabalhar em *JavaScript*.

Aplicação

O Android Studio permite esta funcionalidade para um só telemóvel com a possibilidade de criar uma forma de enviar para vários telemóveis. Exemplo de implementação: [Android Communicator](#)

Muitas implementações usam o *MIT App Inventor* que usa também programação por blocos como no seguinte [tutorial](#). Esta plataforma cria um aplicação que é usada no Android e permite a comunicação com o arduino.

3.6 Screen Recorder - Solução um pouco idiota

Há algumas aplicações já feitas para o Android que permitem ligar o arduino pelo cabo OTG e usar o telemóvel como se fosse um display LCD, ou para simplesmente ver os dados em tempo real. De forma a não ter de enviar a informação em si para um segundo telemóvel ou servidor há a possibilidade de, através de um programa do estilo TeamViewer, aceder aos dados em tempo real. Esta solução apenas elimina o trabalho de implementação da aplicação ou servidor e na realidade é mais uma espécie de back-up preguiçoso.

4 Conclusão

Idealmente seria usado um cabo como ligação de forma a garantir que não há problemas de conexão. Apesar disto, no fundo, se fosse possível garantir que tanto o módulo GSM como o WiFi funcionam sem falhas é uma solução mais prática por ser possível ter os dados já "tratados" sem ter de existir um intermediário. A implementação pode variar em nível de dificuldade, sendo uma aplicação em Android Studio relativamente complicada por ter vários componentes como o receber dados e enviá-los para um telemóvel ou servidor, com várias mini tarefas pelo meio. Por outro lado, o servidor, quer associado a uma aplicação, quer directamente ligado ao arduino, usa linguagens de programação menos usuais, pelo menos no contexto de cadeiras do Técnico. Mesmo assim, com tutoriais de Youtube ou parecido não representa um obstáculo assim tão grande.