

# Vehicle hacking con ArduECU

Ecco il progetto Kickstarter per una centralina Arduino-based per auto e ambienti "difficili"

**N**egli ultimi tempi si parla sempre più spesso di veicoli più o meno intelligenti e di pari passo è cresciuto l'interesse degli appassionati verso il vehicle hacking, sia passivo (la semplice raccolta dati) sia attivo (controllo del veicolo stesso). Il progetto **ArduECU** punta esattamente in questa direzione. Attivo su Kickstarter all'indirizzo <http://kck.st/2p9zfx0>, si one come obiettivo quello di soddisfare l'esigenza di **collegare un device programmabile di classe Arduino all'interfaccia CAN** (standard diffusissimo di comunicazione nato quasi vent'anni fa) dei veicoli o dei macchinari. ArduECU è un Arduino a tutti gli

effetti, ma in un packaging protetto meccanicamente e, soprattutto, elettricamente. Il CAN Bus di un veicolo ne trasporta tutte le informazioni di stato. **Con ArduECU è quindi possibile leggere tutti i parametri del motore** (per esempio la posizione della farfalla all'aspirazione, il carico istantaneo, la temperatura dell'aria in ingresso, ecc.) anche solo per salvarli su un supporto permanente per poi fare un'analisi comportamentale sul profilo del guidatore. Quando leggerete questa notizia speriamo che il progetto su Kickstarter avrà già raggiunto il suo target di finanziamento, ma anche se purtroppo così non fosse, semplicemente l'idea di una "centralina" dotata di CAN Bus, robusta e certificata



» I modelli più utilizzati da appassionati e sviluppatori hanno tutti già disponibili le prime build di *Lineage OS* in versione *Experimental* o *Nightly*. Per avere le stabili occorrerà aspettare

con le specifiche IP69K per gli ambienti umidi e difficili, per non parlare del fatto che

è programmabile ed espandibile, è davvero eccezionale. [LXP](#)

## Una memoria che ci ha portato sulla Luna

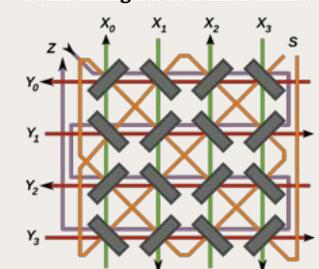
**N**ella storia del calcolo, prima meccanico e poi elettronico, una delle maggiori esigenze è sempre stata quella della memorizzazione più o meno permanente dei dati di lavoro. **Prima delle attuali memorie a stato solido i calcolatori hanno usufruito delle tecnologie più disparate** (e, a volte, disperate!): valvole termoioniche, cilindri metallici, nastri di carta perforata, ampole cilindriche piene di mercurio, relè, condensatori e nuclei di ferrite (core in inglese, da cui il termine coredump che conosciamo un po'

tutti). **Le memorie a nuclei di ferrite si costruiscono davvero facilmente, sono relativamente robuste e soprattutto sono permanenti**, non hanno cioè bisogno di alimentazione per conservare le informazioni, il che è un vantaggio enorme. Il principio di funzionamento è molto semplice (basta vedere lo schema in figura): degli anelli metallici magnetizzabili sono attraversati da fili elettrici. Combinando l'intensità e la polarità della corrente nei vari fili della memoria si può (ri)magnetizzare ogni anello in un dato verso del campo magnetico e anche leggerne lo stato. Si realizza così una

matrice di bit indirizzabile a memorizzazione permanente. **Il programma Apollo con cui la NASA ha mandato l'uomo sulla Luna usava proprio una memoria del genere** sotto forma di treccia (rope memory) flessibile. Se vogliamo provare l'ebbrezza di salvare dati come faceva l'agenzia aerospaziale americana, possiamo visitare [www.corememoryshield.com](http://www.corememoryshield.com): due intraprendenti appassionati di elettronica e informatica, Ben North e Oliver Nash, hanno creato uno shield per Arduino che gestisce una matrice di "ben" 32 bit! Naturalmente l'utilità pratica di un progetto

di questo tipo non è certo altissima ma sperimentare dal vivo il funzionamento di una tecnologia che ci ha portato sulla Luna è senza prezzo. [LXP](#)

Crediti immagine: Konstantin Lanzet



» L'intricato percorso dei fili che combinando gli effetti elettromagnetici magnetizzano e smagnetizzano i core