

Rivisitazione della popolare Arduino Nano, rispetto alla quale implementa in più l'architettura della nostra Fishino con tanto di connettività WiFi, caricabatteria ecc.



GUPPY: FISHINO SI FA NANO

di MASSIMO DEL FEDELE

Come vi abbiamo anticipato nell'articolo in cui abbiamo presentato la nostra Fishino MEGA (scheda compatibile con Arduino MEGA, contraddistinta da un gran numero di I/O e molta memoria di programma e/o RAM rispetto alla versione UNO) abbiamo sviluppato una board che può essere considerata l'altra faccia della medaglia, in quanto è la versione più essenziale della "saga" Fishino. Si tratta della Fishino Guppy, il cui nome deriva da un pesciolino ben conosciuto agli appassionati di acquari, che abbiamo voluto assegnarle in virtù delle ridottissime dimensioni della scheda, che è la più piccola Fishino. Malgrado sia la versione più miniaturizzata (come la Nano è per Arduino) la nostra piccola Guppy presenta funzionalità e caratteristiche di tutto rispetto (come potete leggere riquadro a pagina seguente). Le periferiche incluse nella board sono quelle delle Fishino UNO e MEGA, eccetto che per il modulo RTC, che fisicamente non

trova posto nel PCB che è stato voluto, in sintonia con la filosofia della Arduino Nano, il più piccolo possibile. Rispetto alla Fishino UNO, però, abbiamo aggiunto tutto lo stadio d'alimentazione in tecnologia switching e la possibilità di alimentare la board con una batteria ai polimeri di litio, cosa estremamente interessante perché si presta a realizzare progetti portatili, se non, addirittura, indossabili.

SCHEMA ELETTRICO

Passiamo ora alla descrizione dello schema elettrico, che si può considerare diviso nei seguenti blocchi:

- stadio di alimentazione;
- stadio caricabatteria;
- interfaccia USB;
- controller Atmega328p;
- adattatori di livello;
- modulo WiFi;
- connettore per microSD-Card.

CARATTERISTICHE TECNICHE

- 100% compatibile con Arduino Nano
- Scheda WiFi a bordo
- Connettore per microSD a bordo
- Tripla alimentazione: batteria LiPo, porta USB e connettore di alimentazione esterno (6,5-20 volt)
- Stadio di alimentazione switching ad elevata efficienza, con step-up per ottenere i 5 volt dalla batteria e step-down per ricavarli dall'ingresso esterno,
- Sezione 3,3 volt potenziata per gestire il modulo WiFi, la microSD ed eventuali ulteriori componenti esterni
- Caricabatteria a bordo con indicazione dello stato di carica

SCHEMA ELETTRICO

Siccome, eccezione fatta per i primi due blocchi (lo stadio di alimentazione e il caricabatteria), lo schema elettrico della Guppy è molto simile, quasi identico a quello già presentato per la board Fishino UNO, ci limiteremo a descrivere in dettaglio solo i primi due stadi e daremo una spiegazione sommaria dei restanti, evidenziando le differenze rispetto alle precedenti schede. Per gli approfondimenti del caso rimandiamo all'articolo dove abbiamo descritto Fishino UNO (fascicolo n° 198 di Elettronica In). Lo schema elettrico cui faremo riferimento in questa descrizione è illustrato nella pagina qui accanto.

L'alimentazione

A differenza di quanto fatto nella Fishino Mega, nella Guppy non abbiamo utilizzato uno stadio Sepic, il quale, pur permettendoci di risparmiare un componente e di poter alimentare il circuito sul pin VIN con una tensione minima di 3,5 volt, si è rivelato troppo instabile a causa delle dimensioni ridotte della scheda.

Nel primissimo prototipo avevamo infatti utilizzato tale tipologia, unita ad un PCB a due layer; il tutto funzionava perfettamente finché non veniva utilizzato il modulo WiFi, dopodiché i disturbi provocati da questo "rientravano" nel feedback dell'alimentazione provocando instabilità.

Siamo quindi ritornati sui nostri

passi optando per un doppio converter: uno step-up per l'ingresso batteria ed uno step-down per l'alimentazione dall'ingresso VIN. Per evitare l'instabilità possibile comunque anche con queste tipologie, abbiamo poi optato per un PCB a quattro layer, con i piani di massa ed alimentazione all'interno, in modo da creare una barriera che impedisce ai forti segnali impulsivi di trasmettere le proprie interferenze attraverso il circuito stampato stesso.

Con questi accorgimenti la nostra Guppy è quindi diventata estremamente stabile, anche se il prezzo da pagare è che non è possibile autocostruire la scheda con metodi artigianali; pubblichiamo comunque schema elettrico e piano di montaggio per consentirvi di conoscere a fondo l'hardware della board.

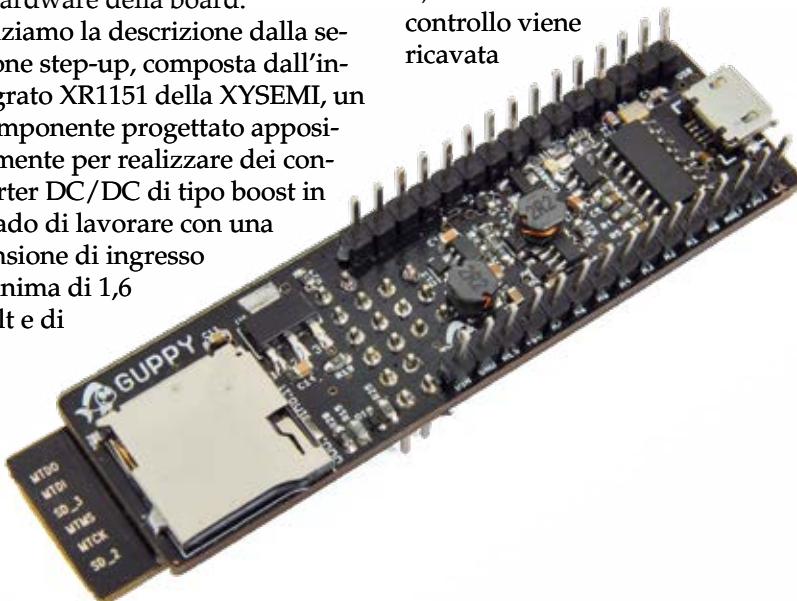
Iniziamo la descrizione dalla sezione step-up, composta dall'integrato XR1151 della XYSEMI, un componente progettato appositamente per realizzare dei converter DC/DC di tipo boost in grado di lavorare con una tensione di ingresso minima di 1,6 volt e di

fornire in uscita 5 volt con 800 mA di corrente, se alimentato con 3,3 volt in ingresso. È quindi ottimale per accettare l'alimentazione da un una comune batteria LiPo 1s (3,7 volt, quindi a singola cella).

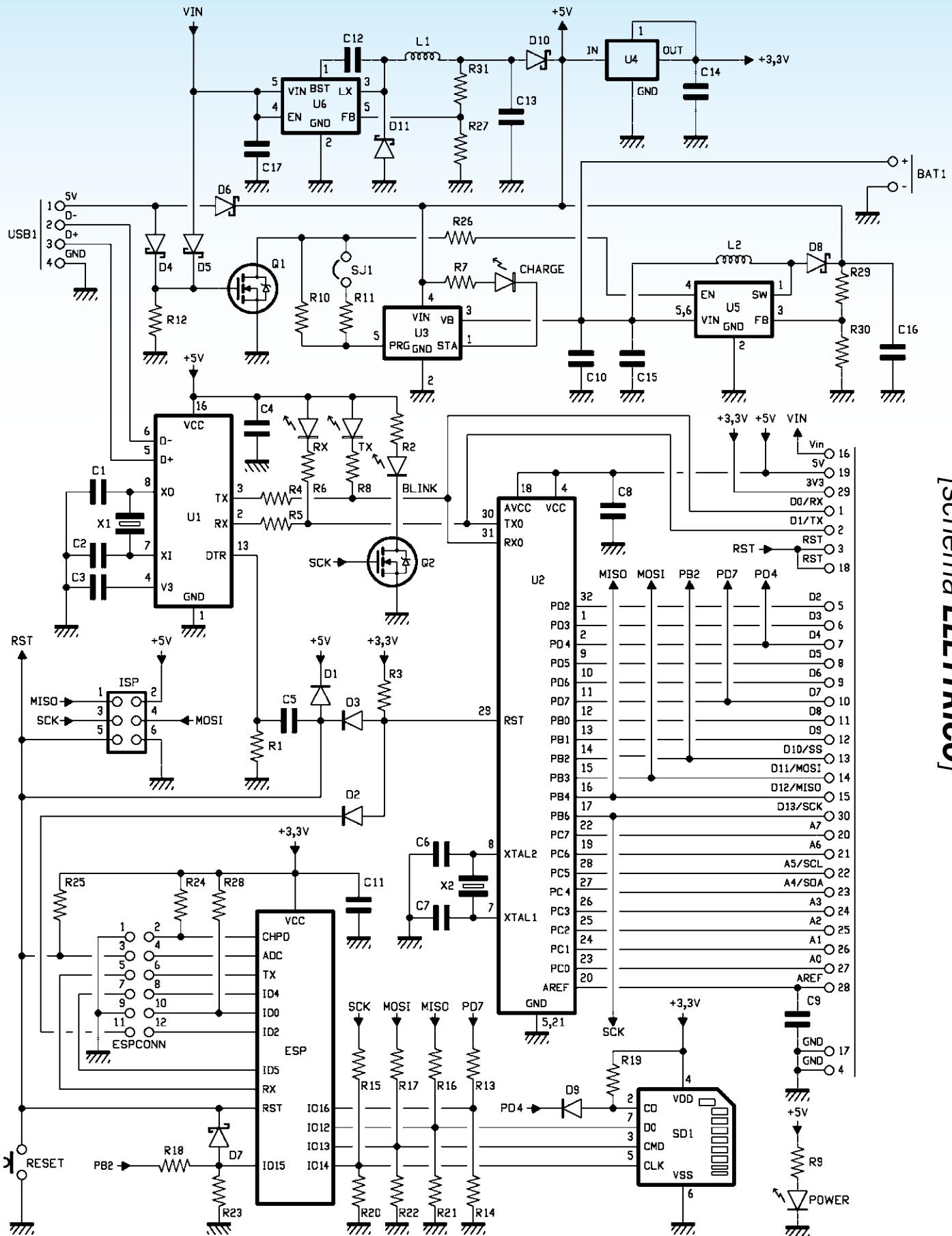
L'integrato contiene al suo interno tutti gli stadi necessari: un generatore PWM, un interruttore CMOS con una resistenza Rdson estremamente bassa (tipicamente 100 milliohm), la necessaria ciruiteria di retroazione (feedback) per mantenere stabile la tensione in uscita ed un sistema di avvio progressivo (soft-start) per evitare picchi di corrente eccessivi nei componenti demandati alla commutazione della corrente d'uscita.

L'integrato lavora ad una frequenza tipica di 1,2 MHz e quindi consente l'utilizzo di componenti (induttanza e condensatori) di dimensioni estremamente contenuti. Opera in modo asincrono, quindi richiede un diodo esterno di commutazione (D12) che nel nostro caso è uno Schottky con bassissima tensione diretta per non penalizzare l'efficienza.

Il circuito di retroazione è calibrato per una tensione di 0,6 V; la tensione di controllo viene ricavata



[Schema ELETTRICO]



dal partitore costituito dalle resistenze di precisione R29 ed R30 ed applicata nell'ingresso FB. La differenza di potenziale in uscita viene stabilizzata sul valore dato dalla formula:

$$V_{out} = 0.6V \cdot (1 + \frac{R_{29}}{R_{30}}) = 0.6V \cdot (1 + \frac{97.6K}{13.3K}) = 5.003 \text{ Volt}$$

La tensione viene riportata direttamente alla linea dei 5V senza alcun diodo aggiuntivo, essendo già presente il diodo D8 e l'interruttore interno all'integrato, disattivabile tramite l'ingresso EN dello stesso, sul quale ritorniamo più avanti.

Una nota: l'XR1151 è fatto per lavorare con tensioni molto basse, e quindi si danneggia se all'ingresso vengono applicati più di 6 volt; questo lo differenzia dal componente utilizzato nello stadio SEPIC della Fishino Mega, che in ingresso accettava tensioni ben più elevate. È quindi assolutamente da evitare l'applicazione di tensioni superiori ai 5÷5,5 V all'ingresso batteria, pena la distruzione del componente e di ciò che viene alimentato da esso. Proseguiamo con la sezione step-down, che ruota attorno all'integrato AOZ1280 dell'Alpha&Omega Semiconductor; questo componente accetta in ingresso tensioni da 3 volt fino a 26 volt ed è in grado di fornire un'uscita minima di 0,8 volt. La frequenza di lavoro è elevatissima, 1,5 MHz, quindi anche qui possiamo utilizzare componenti di dimensioni estremamente ridotte. Come nel caso visto per lo step-up, l'AOZ1280 contiene tutti i componenti necessari per realizzare un buck converter asincrono, ma richiede un diodo esterno di blocco, siglato D11 nello schema elettrico.

L'integrato dispone di un ingresso di compensazione che, tramite il condensatore C12 da 10 nF, permette di stabilizzare la rete di

retroazione in tutte le condizioni di carico.

L'ingresso EN (enable) in questo caso non viene utilizzato, ragion per cui lo abbiamo collegato direttamente alla tensione in ingresso, il che abilita il convertitore non appena viene alimentato.

L'ingresso di retroazione (FB, ossia feedback) è in questo caso calcolato per una tensione di riferimento interna pari a 0,8 volt; la tensione in uscita risulta quindi pari a:

$$V_{out} = 0.8V \cdot (1 + \frac{R_{31}}{R_{32}}) = 0.8V \cdot (1 + \frac{68.1K}{12.1K}) = 5.3 \text{ Volt}$$

La Vout è stata volutamente scelta leggermente superiore ai 5 volt per compensare la caduta sul diodo Schottky in uscita D10, che si aggira sugli 0,2÷0,3 volt; tale diodo è indispensabile per evitare che la tensione proveniente dallo step-up rientri nel componente e vada ad influire sull'ingresso VIN.

Detto ciò, andiamo ad analizzare la sezione di commutazione degli ingressi di alimentazione, differente rispetto allo schema delle altre Fishino; essa ruota attorno ai due diodi Schottky D4 e D5, al MOSFET a canale N Q1 e alle resistenze R12 e R26. In assenza di tensione sugli ingressi VIN o USBVCC, il gate di Q1 viene mantenuto a massa tramite la resistenza R12, spegnendo quindi il componente. La sua uscita si trova quindi libera di fluttuare ed è mantenuta a livello prossimo all'alimentazione dall'integrato carica batteria (che vedremo in seguito) U3; l'ingresso EN (enable) dell'XR1151 verrà quindi a trovarsi a livello alto (tramite la resistenza R26) e quindi entrerà in funzione. L'uscita a 5 volt non potrà rientrare sui pin VIN o USBVCC a causa della presenza dei diodi D6 e D10.

Applicando, per contro, una tensione su uno degli ingressi VIN o USBVCC, il MOSFET Q1 verrà polarizzato positivamente attraverso il diodo D4 o D5 e quindi entrerà in conduzione, portando a massa l'ingresso EN dell'XR1151 spegnendolo.

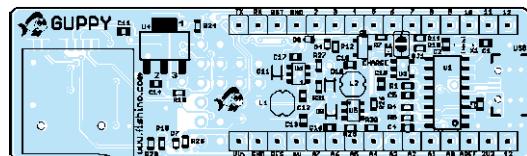
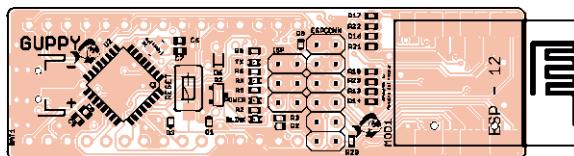
Se la tensione all'ingresso VIN è minore di circa 6 volt, all'uscita del convertitore Step-Down sarà presente una tensione inferiore ai 5 volt, cosa che farà entrare in conduzione il diodo D6 attivando quindi l'alimentazione via USB; per contro, se sui VIN applichiamo una tensione sufficiente, l'uscita del buck converter AOZ1280 sarà sufficiente a portare il diodo D6 in interdizione e quindi disattivare anche l'alimentazione dalla porta USB. Abbiamo quindi ottenuto una commutazione completamente automatica delle tensioni di alimentazione, in questo modo: se è presente una tensione su VIN o sul plug USB, la batteria verrà sconnessa dal circuito e ne verrà contemporaneamente attivata la carica, come vedremo in seguito.

Se non è presente alcuna tensione esterna, entrerà in azione la batteria e, contemporaneamente, verrà disattivato il circuito di carica.

L'ultima parte che analizziamo, ma non ultima per importanza, è costituita dal regolatore a 3,3 volt che fornisce la tensione ai moduli WiFi e alla scheda microSD. Questa parte circuitale, per esigenze di spazio è stata realizzata tramite un consueto regolatore lineare, ma di capacità superiore a quello della Arduino Nano; è infatti in grado di erogare in uscita continuativamente oltre 800 mA di corrente, contro i 500 mA di picco del regolatore di tensione originale della Nano.



[piano di MONTAGGIO]



R1, R2, R4, R5, R8, R18: 1 kohm (0603)

R3, R10, R12, R14, R19, R24÷R26, R28: 10 khom (0503)

R6: 2,4 kohm (0603)

R7, R9: 470 ohm (0603)

R11: 2,7 kohm (0603)

R13, R23: 3,3 kohm (0603)

R15÷R17: 680 ohm (0603)

R20÷R22: 1,5 kohm (0603)

R27: 12,1 kohm (0603)

R29: 97,6 kohm (0603)

R30: 13,3 kohm (0603)

R31: 68,1 kohm (0603)

C1, C2, C6, C7: 22 pF ceramico (0603)

C3, C5: 1 µF ceramico (0603)

C4, C8, C9, C11: 100 nF ceramico (0603)

C10: 4,7 µF ceramico (0603)

C12: 10 nF ceramico (0603)

C13÷C17: 22 µF ceramico (0603)

D1÷D5, D7, D9: RB521S SOD523

D6, D8, D10, D11: PMEG3020EJ

SOD323

X1, X2: Quarzo 16 Mhz

Q1, Q2: 2N7002 SOT23

L1, L2: Bobina 2,2 µH CD32

TX: LED Blu (0603)

RX: LED Giallo (0603)

BLINK: LED Bianco (0603)

POWER: LED Verde (0603)

CHARGE: LED Rosso (0603)

U1: CH340G SOIC16

U2: ATMEGA328P-AU TQFP32-08

U3: MCP73831 SOT23-5

U4: NCP1117ST33 SOT223

U5: XR1151 SOT23-6L

U6: AOZ1280 SOT23-6L

RESET: Microswitch

SD1: Connettore micro-SD

ESP: Modulo Wi-Fi ESP8266

USB1: Connettore micro-USB

BAT1: Connettore JST 2mm 2 vie

Varie:

-Strip maschio 2x3 vie

-Strip maschio 6 vie (2 pz.)

-Strip maschio 15 vie (2 pz.)

-Circuito stampato S1270

Stadio caricabatteria

Questo stadio ruota attorno al notissimo MCP73831 (U3) che abbiamo già trattato in vari progetti e che quindi non stiamo a rispiegare. L'unica parte degna di nota è il circuito di attivazione della carica, che fa capo ad alcuni componenti già visti nel paragrafo riguardante la commutazione dell'alimentazione: Q1, D4, D5 ed R12.

Abbiamo infatti già visto che quando è presente una tensione di alimentazione esterna su VIN o USBVCC, il MOSFET Q1 viene portato in conduzione e quindi il suo drain viene a trovarsi collegato a massa. Questo, oltre che spegnere lo step-up XR1151 come visto in precedenza, collega a massa la resistenza R10 (con eventualmente in parallelo la R11 se scelta la carica a 500 mA),

attivando così la ricarica.

Il diodo led rosso CHARGE si accende durante la carica e si spegne quando essa è completa, in assenza di batteria o quando il caricabatteria è disattivato.

INTERFACCIA USB

Lo stadio d'interfaccia USB della Guppy è identico a quello utilizzato a bordo delle schede Fishino UNO e Fishino Mega e gravita attorno all'ormai noto CH340G, un'integrato in grado di realizzare un completo convertitore USB/Seriale con tutti i relativi segnali di controllo, dei quali noi utilizziamo solo i canali dati (RX e TX) ed il segnale DTR per l'autoreset della Fishino Guppy quando viene programmata dal PC mediante l'IDE Arduino.

Per problemi di spazio, qui non

abbiamo previsto la possibilità di disattivare l'autoreset, funzione, peraltro, quasi mai utilizzata.

Per una descrizione più approfondita di questo modulo rimandiamo all'articolo relativo dove abbiamo pubblicato il progetto della Fishino UNO dove è stato descritto in dettaglio (fascicolo n° 198).

MICROCONTROLLORE

Questo stadio risulta praticamente identico a quello della scheda Fishino UNO e utilizza lo stesso microcontrollore, ossia l'ATmega328P; differisce per l'utilizzo di due ingressi analogici aggiuntivi, vale a dire A5 ed A6, che non sono presenti nel layout della "UNO". Per il resto non sono state introdotte modifiche, quindi anche in questo caso qui rimandiamo alla descrizione dettagliata

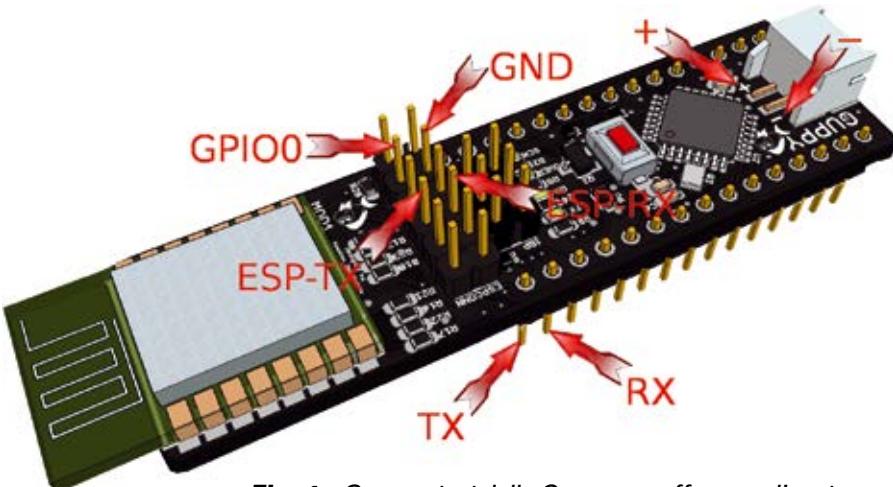


Fig. 1 - Connessioni della Guppy per effettuare l'aggiornamento del firmware.

reperibile nell'articolo descrittivo della Fishino UNO.

Una sola nota riguardante il bootloader : pur essendo questa scheda compatibile con l'Arduino Nano, abbiamo preferito installare il bootloader di Arduino UNO, per il semplice motivo che quest'ultimo occupa 1,5 kByte di Flash in meno e consente quindi di caricare sketch di dimensioni maggiori rispetto a quanto possibile con il bootloader della Nano. Pertanto è necessario, nell'IDE, selezionare come scheda Arduino UNO e non Arduino Nano, altrimenti l'IDE non riconoscerà il prodotto.

Nell'improbabile eventualità che si necessiti la piena compatibilità anche dal lato bootloader è comunque possibile caricare sul Guppy il bootloader originale di Arduino Nano.

Prestate attenzione alla piedinatura del connettore ISP, che nella Guppy è "rovesciata" rispetto a quella della UNO; ciò per seguire la piedinatura originale della Arduino Nano. Vi diciamo questo perché a chi scrive è capitato di "bruciare" una scheda Nano per aver collegato il connettore a rovescio, quindi, in caso di dubbio, vi consigliamo di consultare lo schema elettrico ed il layout del PCB prima di eseguire la programmazione del bootloader!

STADIO ADATTATORE DI LIVELLO

Anche in questo stadio non vi sono modifiche rispetto alle schede Fishino precedenti, fatto salvo il valore delle resistenze (da R13 a R23) che è dimezzato rispetto a quello delle corrispondenti resistenze di Fishino UNO e Fishino MEGA, per garantire una maggior immunità ai disturbi; tale requisito è stato imposto dalle dimensioni estremamente ridotte della scheda. Ad un'impedenza minore, infatti, corrisponde una maggior immunità ai disturbi esterni al costo di un piccolo aumento dei consumi durante il funzionamento del WiFi o della scheda microSD, da considerarsi comunque del tutto trascurabile. Per la descrizione dettagliata del funzionamento vi rimandiamo anche qui all'articolo relativo al Fishino UNO; qui indichiamo soltanto che gli stadi sono costituiti da partitori resistivi

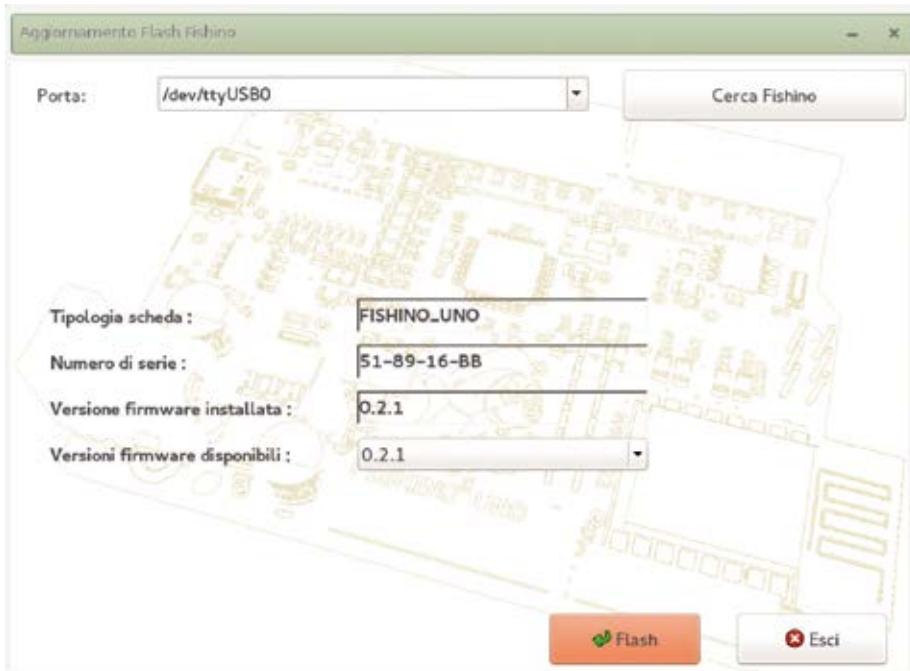
che abbassano il valore massimo della tensione in uscita dall'Atmega dai 5 volt a circa 3,3 Volt, per non danneggiare i componenti a tensione più bassa; in senso inverso non occorre alcuna traslazione, essendo i 3,3 volt in grado di pilotare direttamente le logiche a 5 volt, dato che i pin sono 5V-tolerant.

MODULO WIFI

Come nella Fishino UNO e nella Fishino MEGA, la sezione di interfaccia wireless ruota attorno al modulo ESP12-F (l'ultima uscita della serie 12) basata sul chip WiFi ESP8266; la particolarità del modulo qui utilizzato è che ha un firmware studiato ad-hoc per comunicare con l'Atmega328 tramite interfaccia SPI, ben più performante della seriale che il firmware di fabbrica prevede di utilizzare per l'interfacciamento. A fine articolo riporteremo il procedimento di sostituzione di tale firmware con quello che occorre per il colloquio su SPI (la procedura è comunque presente anche sul sito web di riferimento del progetto, www.fishino.it). Maggiori dettagli sul modulo WiFi li trovate negli articoli dedicati a Fishino UNO e Fishino MEGA.



Fig. 2



CONNETTORE PER MICROSD

Anche questo stadio non è stato modificato rispetto alle Fishino MEGA e Fishino UNO, per cui rimandiamo chi volesse approfondirne il funzionamento agli articoli relativi a tali schede.

AGGIORNAMENTO DEL FIRMWARE DEL MODULO WIFI

Come per i modelli precedenti, suggeriamo di eseguire un aggiornamento del firmware al primo utilizzo e comunque periodicamente, perché è un programma in continuo sviluppo ed intimamente connesso con le librerie fornite. La procedura di aggiornamento è praticamente identica a quella dei precedenti modelli di Fishino; differiscono le connessioni da effettuare, che sono quelle illustrate nella Fig. 1. La procedura di aggiornamento è semplificata da un programma apposito, disponibile sia per la piattaforma Windows che per Linux, che esegue l'operazione in modo completamente automatico ed a prova di errore. I passi da compiere per l'aggiornamento sono i seguenti.

1. Caricare uno sketch che non

utilizzi la porta seriale. L'esempio base BLINK (quello che fa lampeggiare il led sulla scheda) è perfetto; tale passaggio serve ad evitare interferenze tra lo sketch caricato ed il collegamento seriale tramite l'ATmega e l'ESP; se il programma di flash non rileva Fishino, al 99% il problema è uno sketch sbagliato caricato.

2. Collegare la porta TX di Fishino Guppy con la porta ESP-TX sul connettore ESP, e la porta RX della Fishino Guppy con la porta ESP-RX sul connettore ESP (vedere la Fig. 1).
3. Collegare la porta GPIO0 a massa tramite un cavo o un ponticello sempre sul connettore ESP (vedere figura a fianco).
4. Collegare la Guppy al PC (o premere il pulsante di RESET, se già connessa).
5. Lanciare il programma **FishinoFlasher**, assicurandosi che il PC sia connesso ad Internet.

Se i collegamenti sono stati eseguiti correttamente, il programma rileverà la porta a cui è connessa la Guppy, determinerà il modello e la versione del

firmware attualmente installata, si collegherà ad un server remoto e scaricherà la lista dei firmware disponibili, mostrando l'ultimo ma permettendo comunque la selezione delle versioni precedenti (Fig. 2) nel caso si voglia fare un downgrade. Facendo clic sul pulsante "Flash" verrà avviata la procedura di aggiornamento alla fine della quale apparirà un messaggio di conferma. Per terminare il programma occorre fare clic sul pulsante "Esci".

Nel caso la Guppy non venga rilevata automaticamente, è possibile provare a selezionare la porta manualmente. È comunque probabile che siano stati commessi degli errori nei collegamenti.

La selezione manuale risulta indispensabile nel raro caso in cui più di una Fishino sia connessa contemporaneamente al PC, nel qual caso il primo viene rilevato automaticamente ma resta la possibilità di sceglierne un altro. Una volta terminata la procedura è sufficiente eliminare i tre collegamenti e **Fishino** sarà pronto per l'uso con il nuovo firmware.

Bene, abbiamo concluso: siete pronti per sviluppare le vostre applicazioni con la Guppy!



per il MATERIALE

La board Guppy (cod. Guppy) è in vendita presso Futura Elettronica al prezzo di Euro 33,90.

Il prezzo si intende IVA compresa.

Il materiale va richiesto a:
Futura Elettronica, Via Adige 11,
21013 Gallarate (VA)
Tel: 0331-799775 · Fax: 0331-792287
<http://www.futurashop.it>