

Shield Irrighino - in kit

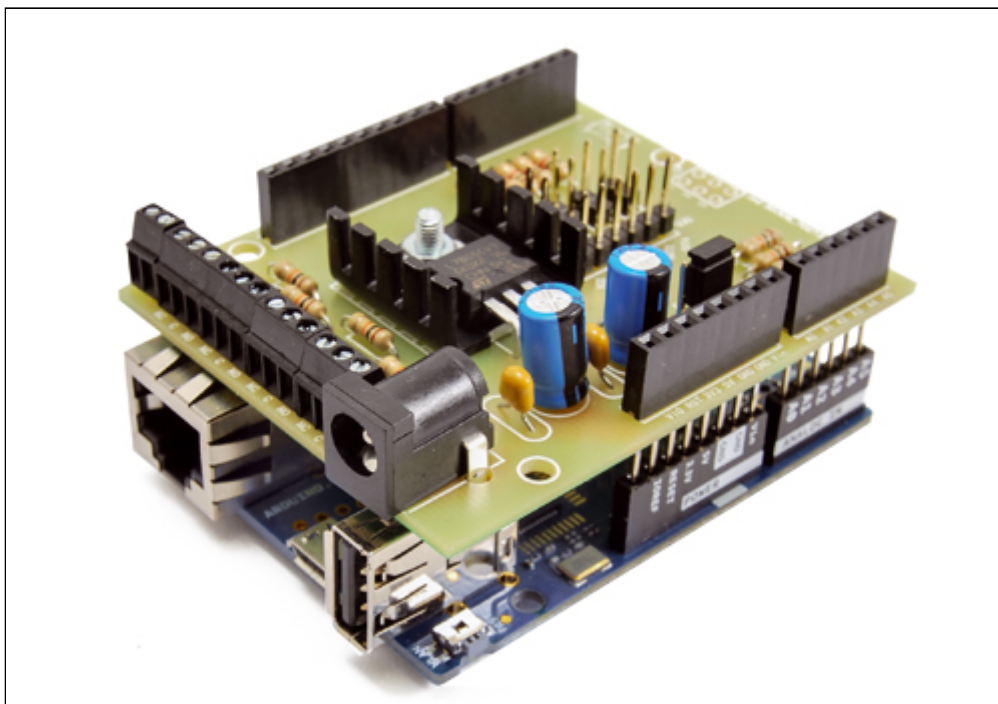
Prezzo: 10.66 €

Tasse: 2.34 €

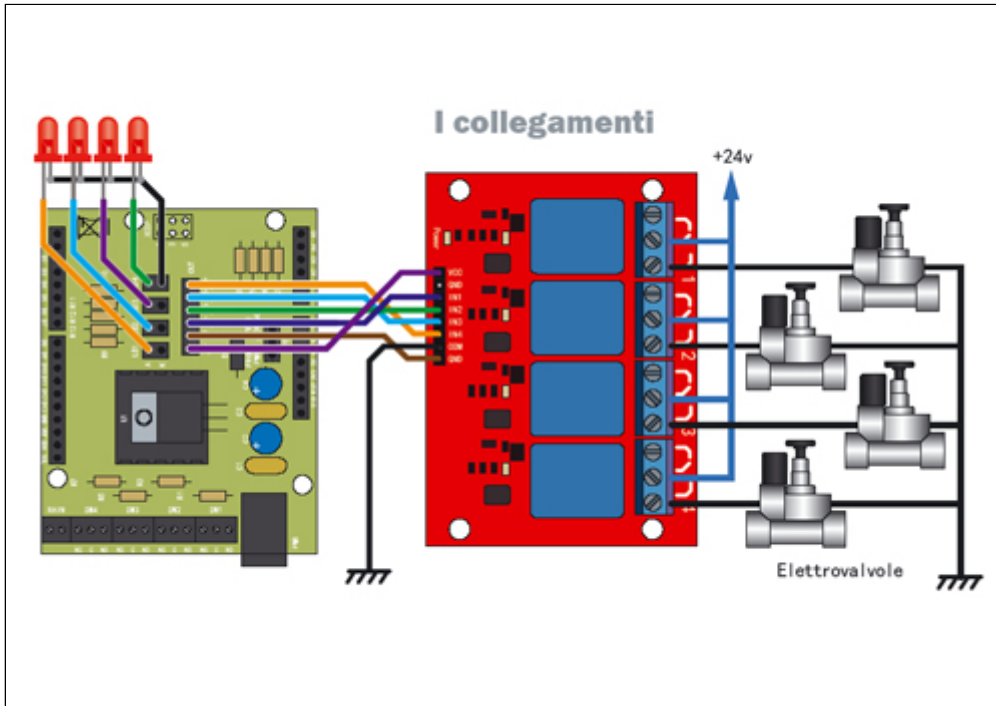
Prezzo totale (con tasse): 13.00 €



Lo shield Irrighino è una centralina di irrigazione basata su Arduino Yún e interfacciabile a una scheda a relè (max. 4) per il controllo di elettrovalvole. È in grado di gestire un numero configurabile di uscite, un sensore per la pioggia ed è programmabile tramite un'interfaccia web raggiungibile da qualsiasi PC o smartphone collegato alla rete domestica. Particolarmente adatto per annaffiare in modo automatico le nostre piante e i nostri giardini. Alimentazione: da 8 a 15 Vdc. **N.B.** Arduino Yún e la scheda a relè non sono comprese nel kit (vedere prodotti correlati).

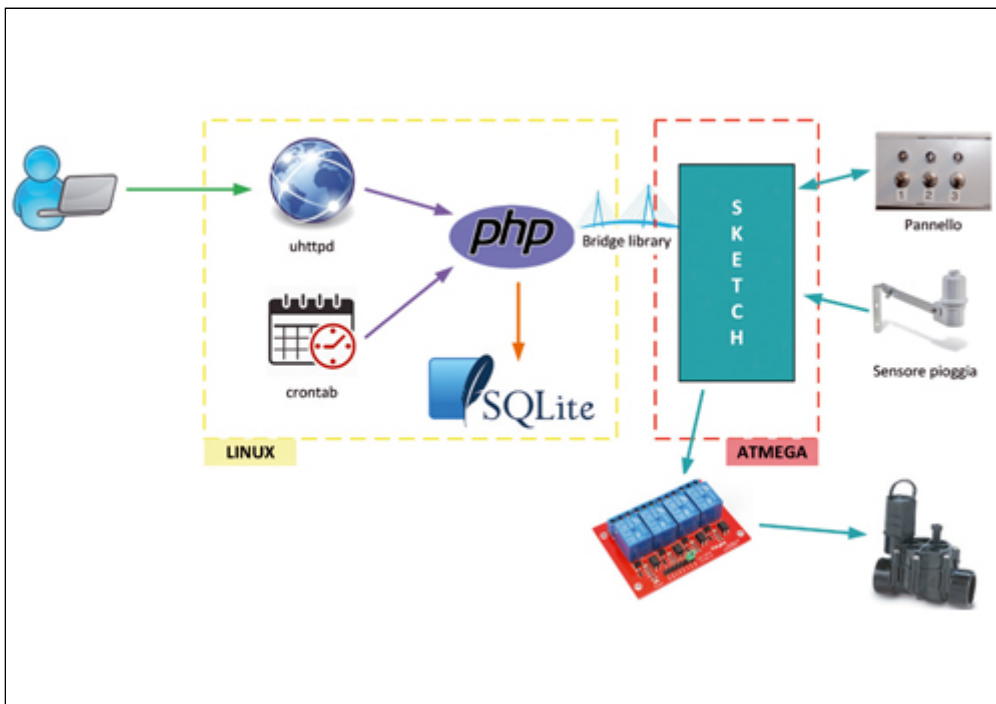


Collegamenti

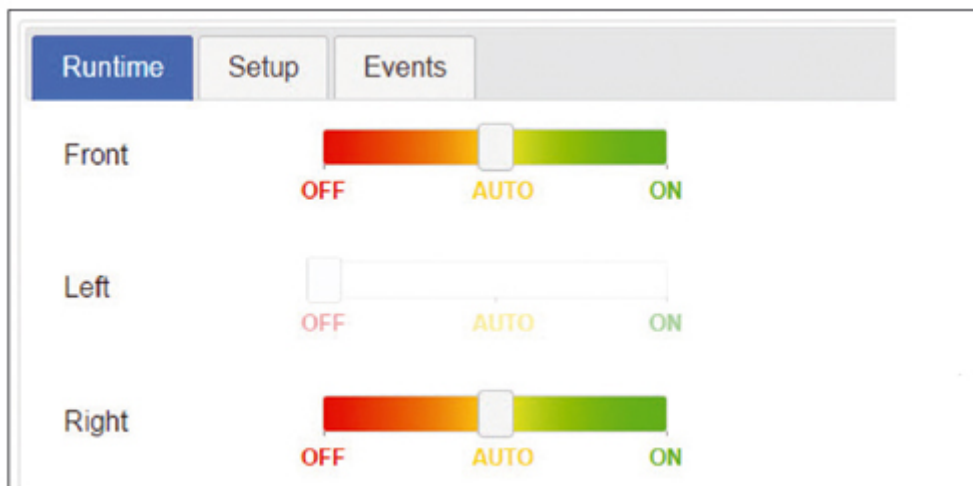


Sopra, lo schema di cablaggio dello shield con la scheda a 4 relé che utilizziamo per comandare fino a quattro elettrovalvole; lo schema è valido a prescindere dalla tensione di funzionamento delle elettrovalvole, che può essere da 12 a 220 Vca.

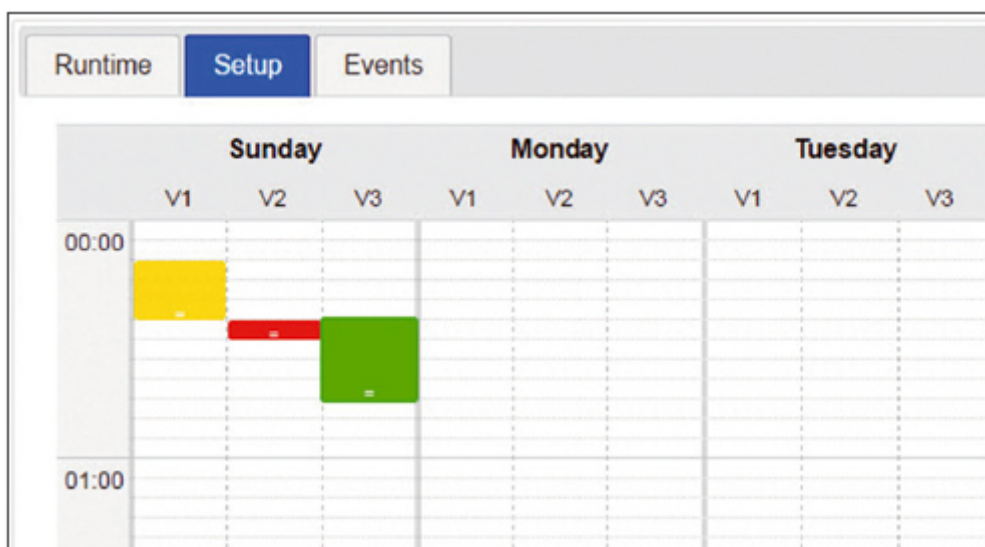
Utilizzo



Come si utilizza la centralina di irrigazione nella modalità remota: da un PC aprite il browser Internet e nella barra degli indirizzi puntate l'indirizzo web http://<ip_arduino_yun>/sd/irrigino; dalla pagina che si aprirà potrete accedere alla schermata di gestione della nostra centralina. L'interfaccia è divisa in tre tab: Runtime, da dove è possibile comandare manualmente le uscite, Setup, dalla quale è possibile programmare le attivazioni su base settimanale, ed Events, dove viene visualizzato l'elenco degli eventi (attivazioni, modifiche alle configurazioni...) più recenti. Ogni uscita può essere impostata in tre stati: ON - l'uscita viene attivata; OFF – l'uscita viene disattivata o AUTO – l'uscita segue la programmazione impostata. Se l'uscita viene configurata tramite il pannello esterno, i relativi comandi sull'interfaccia web saranno disattivati: il pannello ha infatti priorità maggiore.



La fase di programmazione è molto semplice: gli eventi sono raccolti in una tabella nelle cui colonne sono rappresentati i giorni della settimana e le singole uscite (V1...). Ogni riga rappresenta un intervallo di 5 minuti: cliccando con il mouse su una cella è possibile definire un evento di attivazione. Tramite il mouse, l'evento può essere ridimensionato (per modificarne la durata) e spostato, anche su un'altra uscita. Cliccando su un evento, questo viene selezionato e attorno ad esso viene visualizzato un bordo più scuro che ne evidenzia la selezione. È possibile selezionare più eventi o deselectionarli facendo nuovamente clic su di essi mentre si preme il tasto Shift. Infine premendo il tasto CANC è possibile eliminare tutti gli eventi selezionati.



Sebbene il progetto che potete scaricare da Github sia configurato per funzionare out of the box insieme allo shield che abbiamo realizzato, Irrighino è stato pensato per essere facilmente personalizzabile in modo da poterlo adattare ad ogni necessità. Aprendo il file include.php, presente all'interno della cartella php, è possibile modificare il numero di uscite, il loro nome e il colore assegnato, come illustrato nella figura seguente. È anche possibile modificare il comportamento in caso di pioggia (stop immediato delle schedulazioni in corso e/o nessuna nuova attivazione) o disabilitarlo completamente nel caso non sia previsto nel vostro impianto. Anche i collegamenti tra Arduino e i dispositivi esterni sono configurabili modificando lo sketch. Tutto il mapping tra i dispositivi e i pin cui sono collegati è stato definito in un apposito file, chiamato config.h. È quindi possibile modificare i pin cui sono collegati relé, interruttori, LED e sensore di pioggia semplicemente indicando, sempre nel file config.h, i pin di Arduino desiderati. Ricordate, nel caso applichiate ad Arduino altri shield oltre quello di Irrighino, di verificare sempre prima di assegnare gli I/O in config.h che non siano già utilizzati dalle rispettive applicazioni. Accertatevi sempre che il numero di uscite configurato nell'interfaccia web corrisponda al numero di output (OUT_x) definiti nello sketch e ricordatevi anche di aggiornare lo sketch principale. Infine, se non prevedete l'installazione di un pannello di controllo esterno è possibile inibire le funzioni associate al controllo manuale mettendo a false la costante USE_SWITCHES.

```
30 // ----- Output configuration -----
31
32 define ("OUTPUTS_NUMBER", 2);
33
34 $outputs = array();
35 $outputs[0]["name"] = "Giardino";
36 $outputs[0]["baseColor"] = "#FFD800";
37 $outputs[0]["borderColor"] = "#E5BF00";
38
39 $outputs[1]["name"] = "Fiori balcone";
40 $outputs[1]["baseColor"] = "#FF0000";
41 $outputs[1]["borderColor"] = "#E00000";
42
```

```
irrighino  config.h  debug_functions.c

// outputs, digital PINs connected to the relays
#define OUT_1      2
#define OUT_2      3
#define OUT_3      4

// leds, digital PINs that directly drive the panel leds
#define LED_1      7
#define LED_2      6
#define LED_3      5

// switches, analog PINs that read the 3-way panel switches
#define SWITCH_1   0
#define SWITCH_2   1
#define SWITCH_3   2

// rain sensor, digital PIN connected to the rain sensor
// (basically a 2-way switch, normally closed)
#define RAIN_SENSOR 8
```

- [Sketch e Website](#)