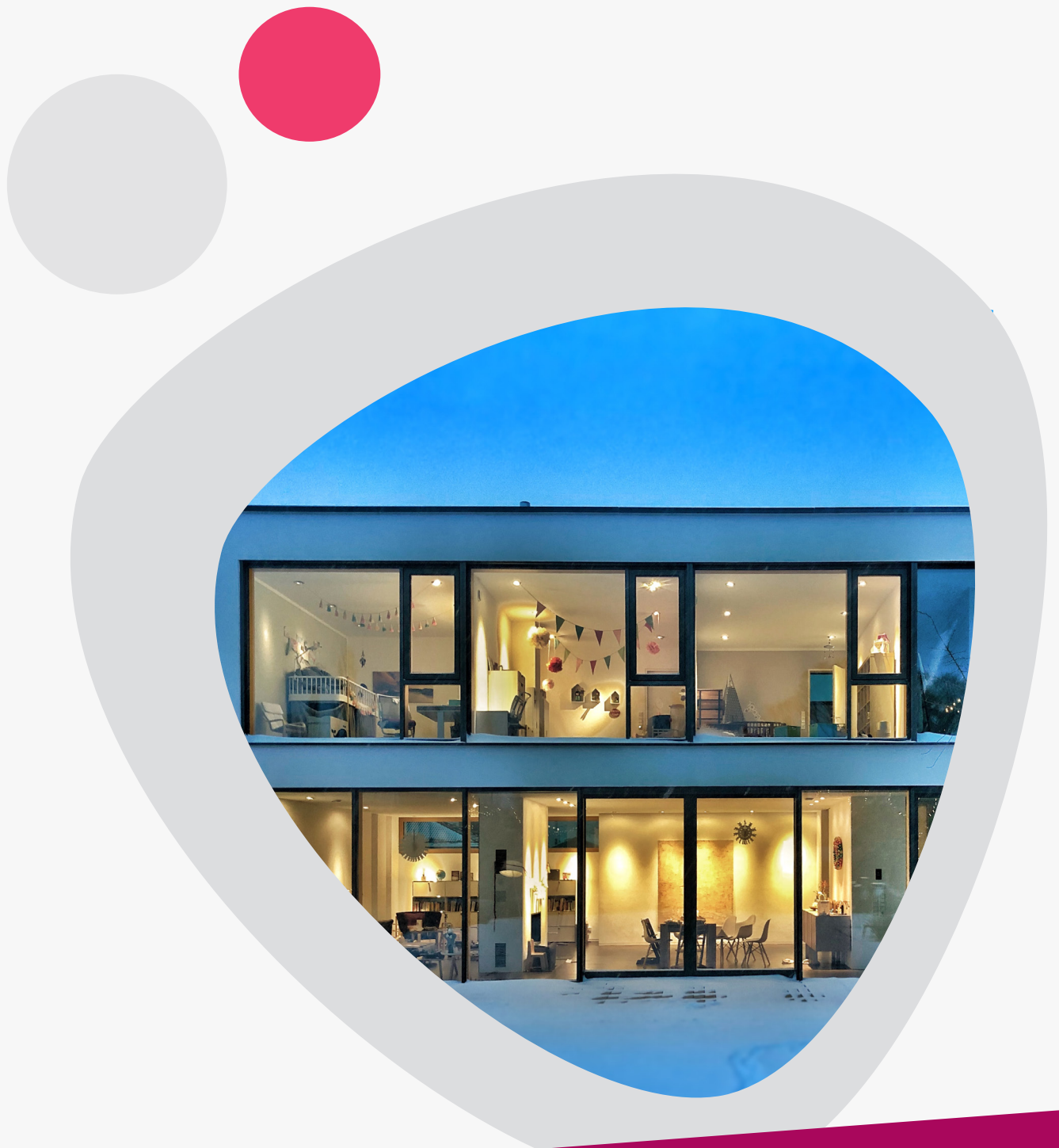


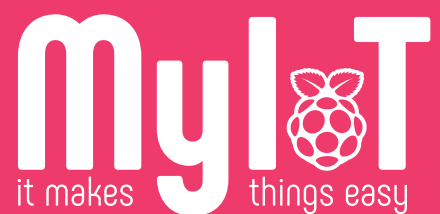
# PRIMA ACCENSIONE

## CARATTERISTICHE GENERALI



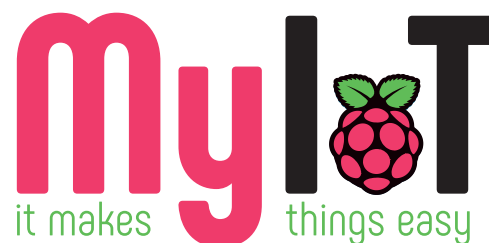
[info@italsensor.com](mailto:info@italsensor.com)

[www.italsensor.com](http://www.italsensor.com)



# INDICE

<b>PAG. 2</b>	PRIMA ACCENSIONE E CARATTERISTICHE GENERALI
<b>PAG. 3</b>	PANNELLO SUPERIORE
<b>PAG. 3</b>	CONNETTORE ALIMENTAZIONE / INTERFACCIA RS485
<b>PAG. 4</b>	NOTE PER IL PRIMO UTILIZZO
<b>PAG. 4</b>	PROGRAMMI DI ESEMPIO PER VERIFICA FUNZIONI DEL MODULO
<b>PAG. 4</b>	PULSANTE RST
<b>PAG. 5</b>	COMUNICAZIONE SERIALE RS485
<b>PAG. 5</b>	MODULO RTC
<b>PAG. 6</b>	DESCRIZIONE DELLE PORTE GPIO E DEL CONNETTORE DI ESPANSIONE CN2
<b>PAG. 6</b>	INTERFACCIA DI ESPANSIONE



## PRIMA ACCENSIONE E CARATTERISTICHE GENERALI

Il modulo, a richiesta, può essere fornito già equipaggiato della microSDHC montata e precaricata con sistema operativo Raspbian “Stretch” e della batteria per il modulo RTC.

- Si rimuove il gancio di blocco per guida DIN dal fondello di chiusura della custodia.
- Si rimuove, con delicatezza, il fondello nero dalla parte superiore della custodia facendo leva con un cacciavite a taglio.
- Si estrae il gruppo elettronica dalla custodia.
- Si inserisce la batteria CR1025 per il modulo RTC (Attenzione a orientare il polo negativo della batteria in contatto con il circuito stampato ed il polo positivo in contatto con l'alloggiamento metallico della batteria)
- Si inserisce la microSDHC nell'apposito alloggiamento (sistema con bloccaggio meccanico).
- Si inserisce il Common Module nell'apposito alloggiamento.
- Se sono utilizzate le linee di uscita aggiuntive si inserisce il connettore femmina con i relativi cavetti già cablati nella scheda di espansione e si estrae il pannello serigrafato superiore dalla custodia.

**(ATTENZIONE: sganciare la scheda di espansione dal circuito stampato principale prima di inserire il connettore di uscita, per evitare di sforzare, e quindi rovinare i contatti della scheda)**

- Si introduce il modulo nella custodia facendo prima passare (se utilizzati) i cavetti della scheda di espansione attraverso l'apertura presente in cima alla custodia, in seguito si inserisce e si blocca il pannello serigrafato di chiusura.
- Si inserisce il fondello del modulo chiudendo così la custodia.
- Si inserisce nel fondello il gancio per l'innesto sulla guida DIN.
- Si alimenta il modulo seguendo lo schema di connessione serigrafato sulla parete della custodia.
- Inizialmente il led blu (ON) presente sulla scheda di uscita inizierà a lampeggiare (lampeggia anche nel caso in cui il modulo venga alimentato senza che sia inserita la scheda Compute Module).
- Dopo alcuni secondi il led blu (ON) smetterà di lampeggiare, questo indica che il processo di boot è andato a buon fine, la scheda Compute Module può quindi essere raggiunta attraverso connessione LAN (ad esempio utilizzando WinSCP per gestire il trasferimento di file da e verso il modulo oppure Putty per la connessione SSH ed accedere alla console del Compute Module, utilizzando WinSCP è anche possibile effettuare la connessione SSH alla console direttamente da WinSCP).
- Il modulo RPICM3 può essere fornito con microSDHC e batteria RTC preinstallate:

- microSDHC da 8 GB, classe 10
- sistema operativo Raspbian “Stretch”
- IP statico preconfigurato: 192.168.2.228 (file /etc/dhcpd.conf)
- SSH abilitato anche per utente root (user: root, password: ital)
- server web Apache2, PHP7 (v 7.0.33) e database Mariadb (v 10.1.37) già caricati ed operativi
- interfaccia seriale ttyAMA0 già configurata per utilizzo diretto dal Compute Module (disattivato con sole output alla partenza del sistema), l'uscita è sulla porta RS485
- RTC già configurato ed operativo (batteria CR1025, Li/MnO<sub>2</sub>, già installata)
- utility rpicm3extended precaricata e subito utilizzabile per la gestione del modulo
- esempi di utilizzo (uscite aggiuntive, gestione watchdog, uso RS485, ...) scritti in Python

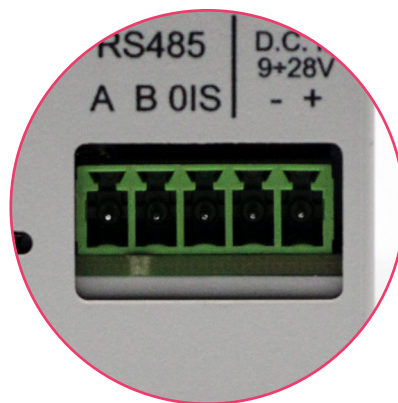
- Alimentazione DC 9÷28V (protezione contro inversione polarità attraverso fusibile ripristinabile)
- Interfaccia RS485 half-duplex da 1200 a 115200 baud con gestione automatica della direzione di trasmissione e meccanismo fail-safe.
- n°1 porta Ethernet LAN 10/100 conforme standard IEEE802.3/802.3u
- n°2 porte USB2.0
- n°1 porta micro USB interna per la programmazione della memoria eMMC relativa al modulo CM3
- n°8 porte di uscita con driver push-pull non optoisolati, funzione 3-state attivabile su due gruppi di 4 linee, protezione contro sovratemperatura, corrente di uscita massima per canale 50 mA, frequenza massima in uscita 50 kHz. Alimentazione interfaccia a +5V, lo 0V è in comune con lo 0V dell'alimentazione del modulo e deve essere necessariamente collegato esternamente al modulo stesso. **ATTENZIONE: attualmente non è implementata la protezione contro inversione di polarità sull'alimentazione** (presto sarà disponibile un'interfaccia push-pull con le stesse caratteristiche di base, ma opto isolata e con alimentazione estesa a 5÷30V).



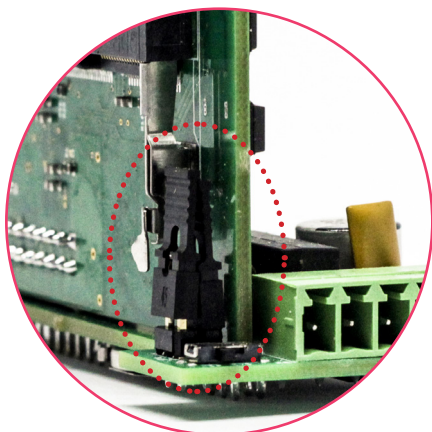
- TX:** led verde (trasmissione dati su RS485)
- RX:** led rosso (ricezione dati su RS485)
- ON:** led blu (attività Compute Module)
- L1:** led giallo (utilizzo utente)
- L2:** led giallo (utilizzo utente)
  
- RST:** accesso per pulsante RST (vedere nel testo)
  
- CN2:** connettore espansione n°8 uscite aggiuntive (1...8) ed alimentazione interfaccia di uscita (9:0V, 10:+5V)

### CONNETTORE ALIMENTAZIONE / INTERFACCIA RS485

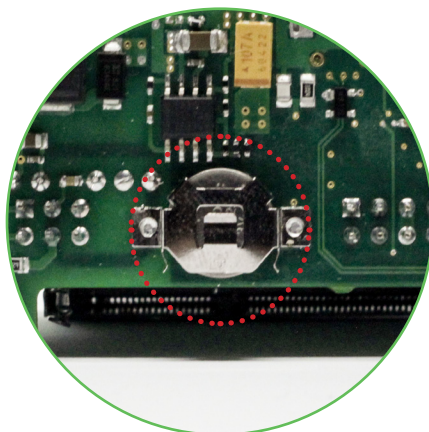
Connettore alimentazione modulo ed interfaccia RS485 (OIS: riferimento isolato RS485 rispetto a 0V modulo). L'apertura a sinistra del connettore permette di accedere ad un interruttore che consente di inserire (se spostato verso il connettore di alimentazione) un terminatore da 100 ohm tra le linee A e B dell'interfaccia RS485. All'interno del modulo, attraverso due jumper, è possibile inserire o disinserire i resistori di pull-up e pull-down da 620 ohm sulle linee A e B dell'interfaccia RS485.



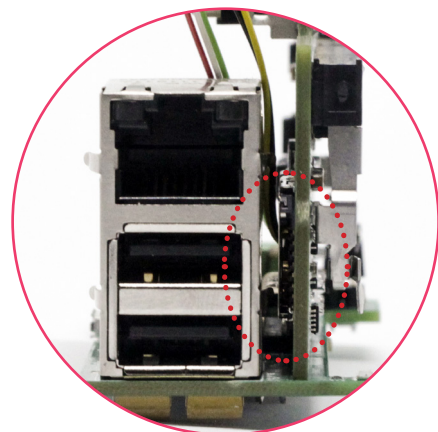
### Interfaccia Rs485 Impostazione Jumper



### Batteria RTC



### microSHDC





## NOTE PER IL PRIMO UTILIZZO

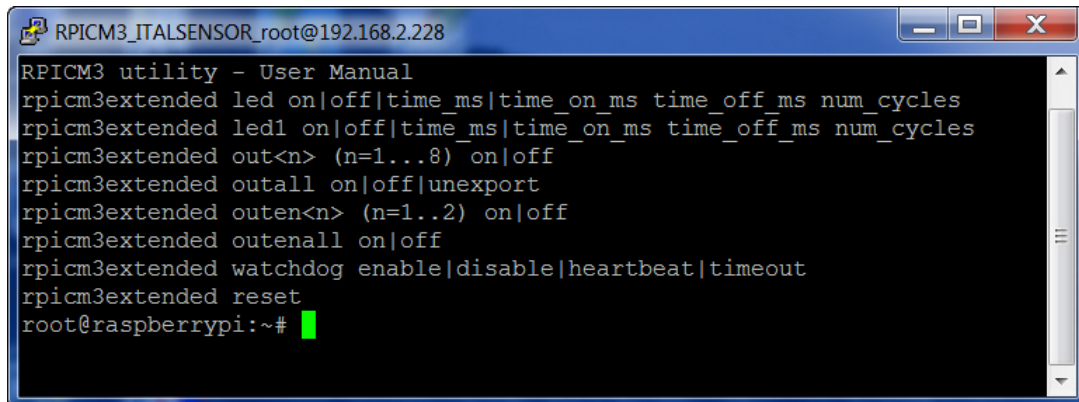
Nella cartella:

```
/home/test
```

sono presenti degli esempi scritti in Python (verificate con versione 2.7.13) utilizzabili come base di partenza per la scrittura delle proprie applicazioni.

Tutte le funzionalità del modulo possono essere gestite attraverso una apposita applicazione di sistema richiamabile da linee di comando. Per visualizzare la guida in linea scrivere:

```
rpim3extended
```



Per la gestione delle uscite sul modulo di espansione si hanno le seguenti opzioni:

- led: accende (on) o spegne (off) il led L1
- led1: accende (on) o spegne (off) il led L2
- out<n>: imposta a "1" (5V) o a "0" (0V) la linea n (n=1..8) di uscita
- outall: imposta a "1" (5V) o a "0" (0V) tutte le linee di uscita
- outen<n>: abilita (on) disabilita (off) il primo o il secondo gruppo delle 4 linee in uscita
- outenall: abilita (on) oppure disabilita (off) tutte le uscite ponendole in alta impedenza (3-state)

## PROGRAMMI DI ESEMPIO PER VERIFICA FUNZIONI DEL MODULO

FILE (/home/test )	UTILIZZO
testout.py	Demo utilizzo uscite interfaccia espansione e led
tstGPIO18.py	Portando GPIO18 ad 1 attiva la procedura di reset dopo 60s
tstGPIO25.py	Lettura stato linea GPIO25 (stato del pulsante RST)
tstHWWDT.py	Esempio di utilizzo del watchdog hardware

Ogni demo può essere interrotto durante l'esecuzione premendo nella console la combinazione di tasti ^C (CTRL+C). Maggiori dettagli nei commenti contenuti nei file.

## PULSANTE RST

Premendo o lasciando a riposo il pulsante RST per un tempo inferiore a 4s il microcontrollore attiva/disattiva rispettivamente la linea GPIO25 (funzionamento verificabile con lo script demo tstGPIO25.py).

Se RST è premuto per un tempo superiore a 4s il led blu (ON) inizia una sequenza di due lampeggi, se dopo questo periodo di tempo il pulsante viene rilasciato il microcontrollore attende 60s trascorso questo intervallo di tempo inizia la sequenza di reset del Compute Module.

Se il pulsante RST continua a rimanere premuto per almeno altri 10s il led blu lampeggia velocemente ed il rilascio del pulsante provocherà il reset immediato del Compute Module.

## COMUNICAZIONE SERIALE RS485

Baud rate compreso tra 1200 e 115200 baud. Interfaccia isolata e protetta con meccanismo di fail-safe, la direzione di trasmissione è gestita automaticamente dal microcontrollore presente nel modulo. Sono disponibili due file di esempio:

FILE (/home/test )	UTILIZZO
<b>tstserial_ma.py</b>	Test comunicazione seriale su RS485 (master)
<b>tstserial_sl.py</b>	Test comunicazione seriale su RS485 (slave)

Per eseguire il codice dimostrativo della porta RS485 si devono interconnettere tra loro due moduli.

Sul primo modulo, identificato come master, sarà eseguito lo script **tstserial\_ma.py**; questo script inizia ad effettuare la trasmissione di un carattere oppure di una serie di caratteri compresi tra un codice di partenza ed uno di arrivo attendendo una risposta dal modulo slave.

Sul secondo modulo, identificato come slave, dovrà essere eseguito lo script **tstserial\_sl.py**; lo script resta in attesa e ritrasmette immediatamente (echo) i caratteri ricevuti dal master.

La connessione tra i due moduli si effettua collegando rispettivamente i pin A e B del connettore del modulo master ai corrispettivi A e B del modulo slave



Per avere una trasmissione/ricezione coerente si deve impostare lo stesso valore di baud rate per entrambi gli script.



Per entrambi i codici di esempio è disponibile un help di utilizzo richiamabile invocando lo script senza argomenti.



Gli script di esempio relativi alla trasmissione/ricezione seriale RS485 sono stati scritti pensando ad una gestione dei dati di tipo binario, non solo quindi per la trasmissione di dati relativi ai codici ASCII dei caratteri visualizzabili.

### Esempio 1: trasmissione continua singolo valore/carattere

- Per trasmettere il carattere "A" alla velocità di 115200 baud con una pausa di 1s tra invii successivi utilizzare sul modulo master il comando:

```
python tstserial_ma.py 115200 65 65 1
```

- Per attivare la ricezione sul modulo slave utilizzare il comando:

```
python tstserial_sl.py 115200
```

### Esempio 2: trasmissione continua su un intervallo di valori/caratteri

- Per trasmettere a 115200 baud con una pausa di 100ms tra invii successivi tutti i caratteri aventi codice ASCII compreso tra 0 e 255 utilizzare sul modulo master il seguente comando:

```
python tstserial_ma.py 115200 0 255 0.1
```

- Per attivare la ricezione sul modulo slave utilizzare il seguente comando:

```
python tstserial_sl.py 115200
```

## MODULO RTC

Il modulo RTC (basato sul circuito integrato MCP79410) è un orologio hardware separato dall'ora di sistema che permette di mantenere le informazioni di data ed ora anche a modulo non alimentato. Si interfaccia al Compute Module attraverso le linee I2C del modulo stesso che quindi sono riservate per questo utilizzo specifico.

Per leggere la data e l'ora memorizzata nel modulo RTC scrivere:

```
sudo hwclock -r
```

Nel caso in cui sia necessario impostare data ed ora del modulo RTC, allineandola all'ora di sistema (visibile con il comando **date**), sarà sufficiente scrivere:

```
sudo hwclock -w
```

## RIEPILOGO E DESCRIZIONE DELLE PORTE GPIO E DEL CONNETTORE DI ESPANSIONE CN2

Determinate porte GPIO del Compute Module sono utilizzate per scopi specifici.

GPIO Compute Module	Direzione	Funzione
GPIO2		Linea I2C (SDA) utilizzata per modulo RTC
GPIO3		Linea I2C (SCL) utilizzata per modulo RTC
GPIO23		Riservato: utilizzo futuro
GPIO24		Riservato: utilizzo futuro
GPIO14	OUT	TXD – Linea trasmissione seriale
GPIO15	IN	RXD – Linea ricezione seriale
GPIO16	OUT	1: led LD1 ON / 0: led LD1 OFF
GPIO34	OUT	1: led LD2 ON / 0: led LD2 OFF
GPIO25	IN	1: RST attivato / 0: RST a riposo
GPIO18	OUT	1: attiva procedura di reset del Compute Module dopo 60”
GPIO22	OUT	1: attiva funzionalità watchdog hardware
GPIO27	OUT	Segnale di heartbeat per watchdog hardware, richiesto almeno un cambio di stato ogni 60”
GPIO17	IN	Linea forzata ad 1 dal microcontrollore quando, con hardware watchdog attivato (GPIO22), il microcontrollore non rileva almeno un cambio di stato della linea GPIO27 entro un intervallo di tempo di 60”, dopo un ulteriore attesa di 60” viene eseguito il reset del sistema forzando a zero la linea RUN del Compute Module.
GPIO11	OUT	1: attiva uscite linee 1,2,3,4 / 0: linee 1,2,3,4 in 3-state
GPIO21	OUT	1: attiva uscite linee 5,6,7,8 / 0: linee 5,6,7,8 in 3-state

Porte GPIO interne

GPIO Compute Module	Direzione	Funzione	Pin CN2
GPIO36	OUT	Impostazione stato uscita linea 1	1
GPIO20	OUT	Impostazione stato uscita linea 2	2
GPIO37	OUT	Impostazione stato uscita linea 3	3
GPIO35	OUT	Impostazione stato uscita linea 4	4
GPIO38	OUT	Impostazione stato uscita linea 5	5
GPIO13	OUT	Impostazione stato uscita linea 6	6
GPIO12	OUT	Impostazione stato uscita linea 7	7
GPIO26	OUT	Impostazione stato uscita linea 8	8
-	-	Linea alimentazione uscita espansione: 0V	9
-	-	Linea alimentazione uscite espansione: +Vin=5V	10

Assegnazioni per gestione interfaccia di uscita e connettore di espansione CN2



### INTERFACCIA DI ESPANSIONE

La linea dello 0V relativa all'interfaccia di espansione è in comune con l'alimentazione del modulo, i driver di uscita non separano galvanicamente i GPIO del Compute Module dalle linee di uscita, ma realizzano un'interfaccia in grado di lavorare a 5V, protetta termicamente, con una corrente massima di 50 mA per canale e frequenza di commutazione fino a 50 kHz.