# Condizionatore di segnale Daisy

## Introduzione

Daisy è un condizionatore di segnale basato su ESP32 in grado di gestire fino a 4 celle di carico estensimetriche in contemporanea, grazie ai due ADC HX711 a doppio canale di cui dispone. La particolarità di questa scheda è la sua elevata immunità ai disturbi, ottenuta grazie a particolari soluzioni hardware ed intelligenti algoritmi di filtraggio. Inoltre, Daisy si integra perfettamente negli impianti di automazione già esistenti grazie ai diversi protocolli di comunicazione implementati.

## Specifiche

Le principali caratteristiche tecniche di Daisy sono le seguenti:

* Alimentazione +24V;
* 4 canali per la gestione di celle di carico;
* 7 uscite digitali (di cui 2 PWM) 24V;
* 5 ingressi digitali 24V;
* Real Time Clock batterizzato;
* Supporto per scheda microSD esterna;
* **Protocollo di comunicazione ModBus RTU**;
* Protocollo di comunicazione CanBus;
* Protocollo di comunicazione MQTT.

## Protocollo ModBus

Il protocollo ModBus implementato è di tipo RTU ed ha i seguenti parametri di comunicazione:

* Indirizzo Slave = 1;
* Baud Rate = 9600;
* Parity = None.

I registri messi a disposizione sono i seguenti:

|  |  |  |  |
| --- | --- | --- | --- |
| **Holding registers** | **Data type** | **Address for Modbus requests** | **Description** |
| cell1 MS | uint\_16 | 0 | Cell1 reading value |
| cell1 LS | uint\_16 | 1 | Cell1 reading value |
| cell2 MS | uint\_16 | 2 | Cell2 reading value |
| cell2 LS | uint\_16 | 3 | Cell2 reading value |
| cell3 MS | uint\_16 | 4 | Cell3 reading value |
| cell3 LS | uint\_16 | 5 | Cell3 reading value |
| cell4 MS | uint\_16 | 6 | Cell4 reading value |
| cell4 LS | uint\_16 | 7 | Cell4 reading value |
| pesoTot MS | uint\_16 | 8 | Total weight (already filtered) |
| pesoTot LS | uint\_16 | 9 | Total weight (already filtered) |
| pesoCalib MS | uint\_16 | 10 | Calibration sample weight |
| pesoCalib LS | uint\_16 | 11 | Calibration sample weight |
| Diagnostic | uint\_16 | 12 | System diagnostic data |
|  |  |  |  |
| **Coils** | **Data type** | **Address for Modbus requests** | **Description** |
| TareCommand | bool | 0 | Tare command rising edge logic |
| CalibCommand | bool | 8 | Calibration command rising edge logic |
|  |  |  |  |
|  |  |  |  |
| Diagnositc::descriptor | bit | Value |  |
| implemented | 0 | cell1\_ok=0, cell1\_not\_ok=1 |  |
| implemented | 1 | cell2\_ok=0, cell2\_not\_ok=1 |  |
| implemented | 2 | cell3\_ok=0, cell3\_not\_ok=1 |  |
| implemented | 3 | cell4\_ok=0, cell4\_not\_ok=1 |  |
| implemented | 4 | Memory: working=0, not working=1 |  |

### Protocollo di tara

Per realizzare la tara occorre impostare il valore Coil0=1.

Si noti che il comando di tara funziona in logica rising-edge. Occorre quindi riportare Coil0=0 prima di lanciare un nuovo comando di tara.

### Protocollo di calibrazione

Per realizzare la calibrazione occorre impostare il valore Coil8=1.

Si noti che il comando di calibrazione funziona in logica rising-edge. Occorre quindi riportare Coil8=0 prima di lanciare un nuovo comando di calibrazione.

In particolare, il processo di calibrazione si compone dei seguenti passaggi:

1. Esecuzione della tara con la bilancia scarica: Coil0=0 -> Coil0=1;
2. Posizionamento del peso di calibrazione sulla bilancia;
3. Inserimento del valore del peso di calibrazione nei registri: PesoCalibMS (10) e PesoCalibLS (11);
4. Esecuzione della calibrazione: Coil8=0 -> Coil8=1.

## Cablaggio