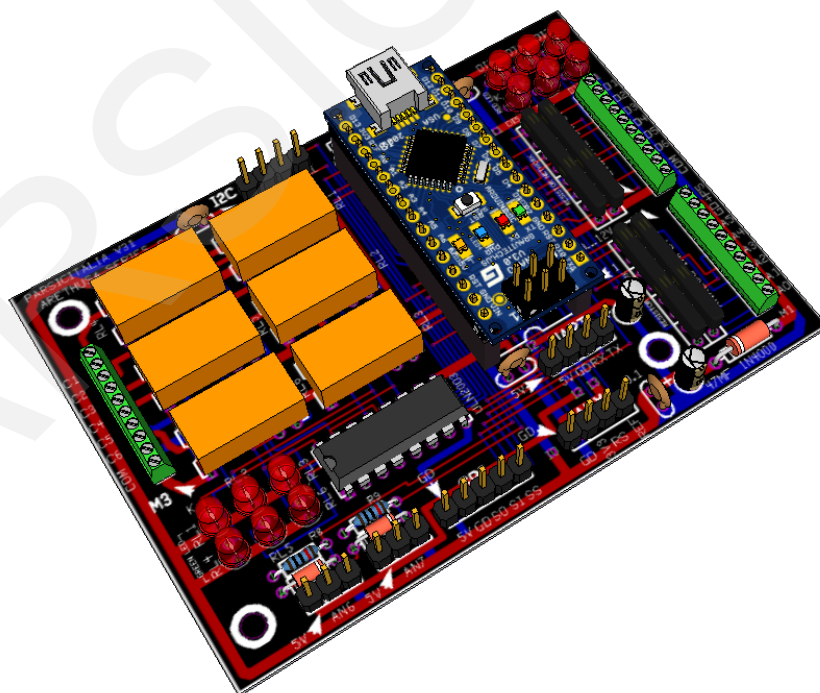




V31an Digital and Analog port expander with ARDUINO Nano



Nota informativa

Le informazioni contenute sul presente manuale tecnico sono state verificate con attenzione. **Parsic Italia** non assume alcuna responsabilità per danni, diretti o indiretti, a cose e/o persone, derivanti da errori, manomissioni e omissioni, e dall'uso improprio del presente manuale .

Prima di eseguire qualsiasi intervento, l'utente si assume ogni responsabilità per l'impiego di questo prodotto **OEM. Parsic Italia, con sede a Savio di Cervia (Ra) , non risponde in alcun modo di possibili danni** materiali e fisici derivanti da tale impiego. **Parsic Italia si riserva il diritto di cambiare o modificare in qualunque momento il contenuto del presente** manuale e/o la modifica del prodotto senza alcun obbligo di avviso. I componenti elettronici ed elettrici impiegati, sono particolari costruttivi dei rispettivi marchi produttori a cui l'utente dovrà fare riferimento attraverso i corrispondenti data book. Il particolare costruttivo del prodotto è proprietà mentale di **Parsic Italia ed è protetto da copyright. E' vietata la** riproduzione, anche parziale, di questo manuale, su qualunque tipo di supporto universalmente conosciuto; la pubblicazione sui circuiti internet, della versione integrale e non modificata, deve prima essere autorizzata da **Parsic Italia**.

Impiego

Questa scheda può essere impiegata in associazione con il sistema a microcontrollore denominato Arduino Nano che necessita di un supporto di espansione degli I/O digitali ed analogici . Questa scheda trova applicazione come scheda accessoria nel sistema di sviluppo Arduino. Per migliori informazioni tecniche sull'utilizzo dei prodotti Arduino si prega di consultare il sito: Arduino.cc

NORMA DI SICUREZZA Avvertenze generali

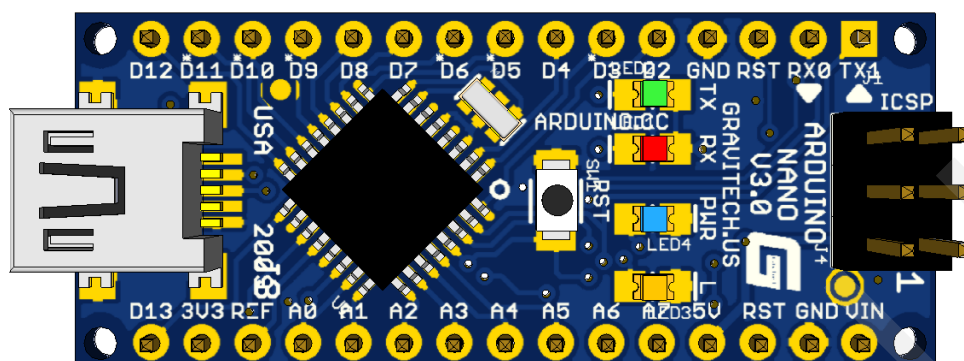
L'impiego di questo dispositivo OEM, è rivolto a personale specializzato e/o qualificato, in grado di interagire con il prodotto in condizione di sicurezza per le persone, macchine ed ambiente, in pieno rispetto delle **Norme di Sicurezza e salute**.

In ambito didattico, gli allievi saranno guidati dal personale docente in grado di indicare le operazioni necessarie per operare in piena sicurezza. L'installazione del prodotto, montaggio, smontaggio, aggiustaggio ,presume la conoscenza, da parte dell'utente, delle **Norme di Sicurezza e delle Norme Tecniche legate al tipo di attività in atto. Pertanto**, saranno adottate tutte le misure necessarie alla protezione ed incolumità personale di chi opera. L'impiego di questo prodotto è consigliato su un sistema elettronico **a limitato preventivo di spesa, e l'operatore** è già edotto sulle problematiche tecniche indotte dalla modifica dei circuiti in cui si opera.

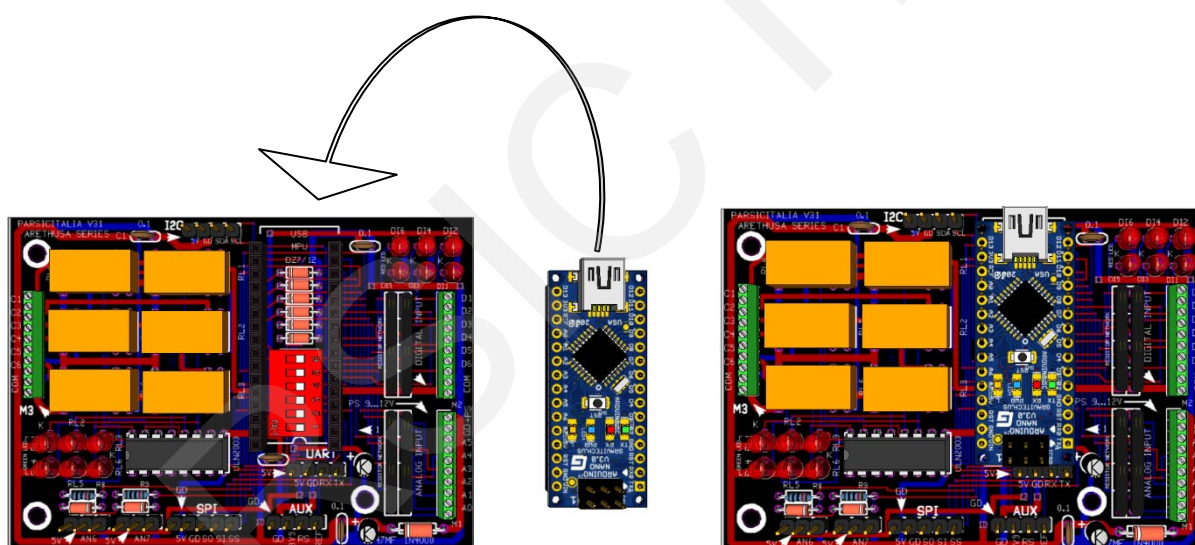
V31 Digital and Analog port expander con Arduino Nano

La scheda è progettata per funzionare con la scheda **Arduino Nano**.

L'Arduino Nano, è una scheda dalle dimensioni molto contenute, basa il suo funzionamento sul microcontrollore **ATmega328**. Con alcune differenze circuitali, ha le stesse funzionalità dell'**Arduino UNO**, ed è stato progettato e prodotto da **Gravitech**.



La scheda Arduino Nano, è dotata di **14 pin di input/output digitali** (6 dei quali possono essere usati come segnali PWM), **8 input analogici**, un quarzo a **16MHz**, un connettore **Mini-B USB**, un connettore per la programmazione **ICSP** ed una serie di accessori necessari per il funzionamento del microcontrollore. La scheda Arduino Nano, si innesta sulla **V31**, posizionando i terminali a spina sul connettore femmina a **32 poli**, facendo attenzione a rivolgere la presa **USB** verso l'esterno, così come si vede in figura :



La scheda Arduino Nano si alimenta per mezzo del cavo **USB**, mentre il supporto **V31** si alimenta con una tensione esterna di **12Vcc**. Quando la programmazione di Arduino è completata, si può sconnettere il cavo USB. La V31 permette di operare in varie modalità e può essere impiegata per applicazioni sperimentali di robotica o **Build Automation**. Se corredata di terminazione **Wi-Fi** o **Bluetooth**, può essere impiegata in abbinamento alle **APP** per **Smartphone** e **Tablet**. Questa scheda è anche indicata per la progettazione di nuovi circuiti o per la didattica scolastica.



Applicazioni della scheda :

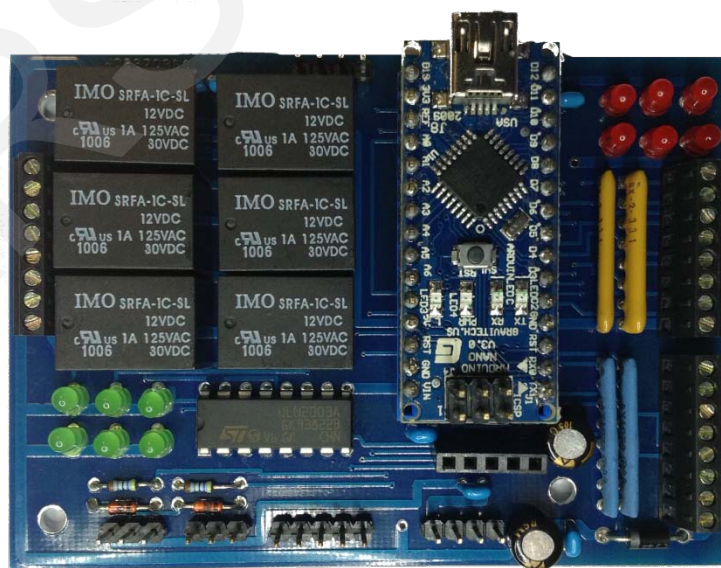
- Sistemi di Automazione, Domotica, Robotica
- Antifurti
- Automotive
- Progettazione
- Didattica

Con Arduino si possono realizzare in maniera relativamente rapida e semplice piccoli dispositivi come controllori di luci, di velocità per motori, sensori di luce, temperatura e umidità e molti altri progetti che utilizzano sensori, attuatori e comunicazione con altri dispositivi.

Caratteristiche tecniche :

- Microcontroller Atmel ATmega328
- 6 ingressi digitali, bidirezionali 30mA, con segnalazioni led
- 6 uscite digitali su relè 1 Amp. con segnalazioni led
- 8 ingressi analogici protetti, risoluzione 10 bit
- Riferimento di tensione interno
- 1 switch selezione segnali
- 1 port SPI
- 1 port I2C
- 1 port UART- Bluetooth
- 1 port AUX
- 2 port ingressi analogici
- Flash memory 32KB
- SRAM 2KB
- EEPROM 1KB
- Clock 16MHz
- Tensione di alimentazione 9-12Vcc
- Dimensioni 90 x 64 x 20 mm

Informazioni tecniche su [Arduino Nano](#) reperibili sulla rete Web: [Schematic](#) ; [Manual](#)



V31AN KIT. Schema di montaggio

La scheda è fornita in due versioni.

La versione **V31AN** è un prodotto preassemblato e pronto all'impiego.

La versione **V31AN KIT** è fornita in scatola di montaggio, con i componenti scolti da saldare al pcb.

Assemblaggio della scheda

L'operazione, molto semplice da eseguire, richiede un minimo di attrezzatura e mezz'ora di applicazione manuale.

Le istruzioni sono contenute nel presente manuale, seguendo le illustrazioni e disegni a colori. Si raccomanda di impiegare **ottimo stagno da laboratorio** ed un saldatore da 25W con punta fine.

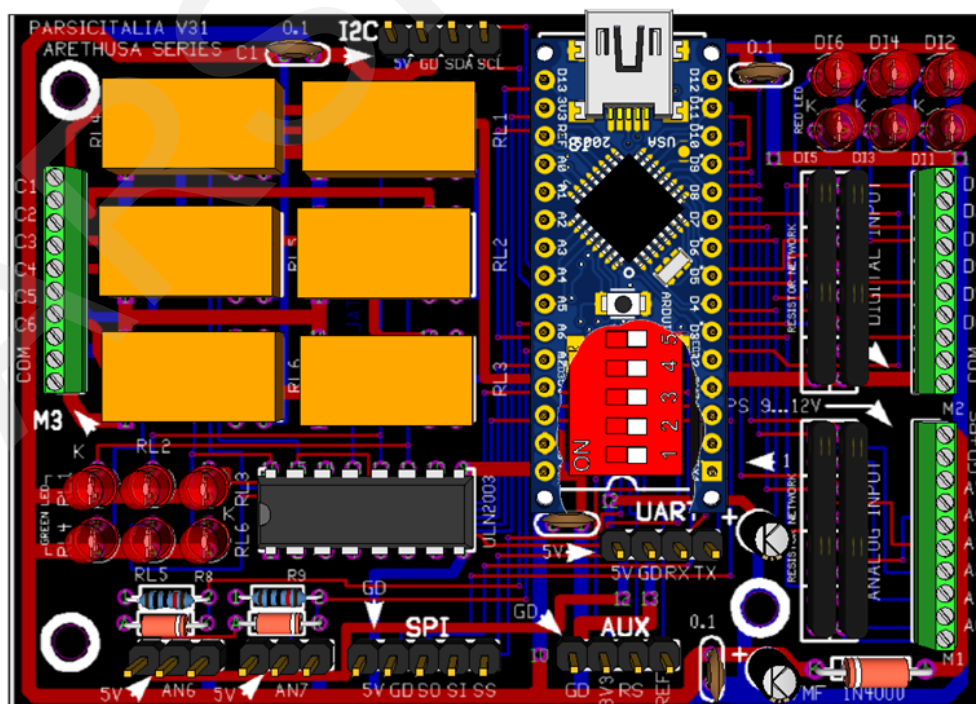
Procedere con l'inserimento dei componenti nelle apposite piazzuole, rispettando i valori e le polarità.

Inserire prima i componenti a basso profilo, resistenze, condensatori ceramici, diodi e reti resistive. Poi quelli a profilo più alto come trimmer, condensatori elettrolitici e regolatore. Infine, connettori, morsettiere e relè.

Si richiede un minimo di attenzione nella posa dei led di segnalazione. **Rispettate assolutamente** il colore **Rosso** (led 5Vcc) per gli ingressi, il colore **Verde** (led a 12Vcc) per le uscite. L'inversione dei colori potrebbe causare la distruzione del led. A fine lavoro lavare il circuito, per rimuovere le incrostazioni, impiegando detergente liquido alcalino (pH11), oppure solvente chimico non tossico. Asciugare con un getto d'aria calda. Il risultato sarà un circuito con i componenti perfettamente allineati e saldature lucide, esenti da opacità.

Predisposizioni

All'interno del connettore di supporto di Arduino Nano, è alloggiato un **dip-switch** che, opportunamente selezionato, permette di abilitare le funzioni specializzate della scheda. Lasciando gli interruttori in posizione **ON**, i pin **PB2/B3/B4/B5** e **AD4/AD5** saranno collegati ai rispettivi terminali dell'Arduino Nano e svolgeranno le funzioni digitali di **OUT** i primi, ed **Input Analogici** i secondi.



DS POS	1	2	3	4	5	6
ON	PB5	PB3	PB4	PB2	AD5	AD4
ON	RL5	RL3	RL4	RL2	ATTIVAZIONE RELE'	
ON	INGRESSI ANALOGICI				AD5	AD4
OFF(1)	PB5	PB3	PB4	PB2	FUNZIONE SPI	
OFF(2)	FUNZIONE I2C				AD5	AD4

- ponendo a **OFF** gli switch **1-4** le uscite relè **2/3/4/5** saranno **disabilitate** ed abilitata la funzione **SPI**
- ponendo a **OFF** gli switch **5-6** gli ingressi analogici **AD4** e **AD5** saranno **disabilitati** ed abilitata la funzione **I2C**.

Ingressi/Uscite digitali. Ingressi Analogici

Ciascuno dei **14 pin** digitali di Arduino Nano può essere impiegato come ingresso o uscita. Questi pin sono di tipo bidirezionale, funzionano con un livello di **5V** e possono fornire massimo **40mA**. Sono dotati di resistenza di pull-up interna, disconnessa di default, del valore di 20-50KOhm. Alcuni di questi terminali svolgono funzioni specializzate. Nel progettare la scheda e dovendo operare delle scelte hardware, la V31AN è stata impostata per svolgere funzioni standard di I/O, senza trascurare la possibilità di impiego delle linee seriali **I2C ed SPI**, per eventuali espansioni circuitali. Sono prodotti numerosi chip in grado di svolgere funzioni di I/O expander ed ADC, sia su linea **I2C** che **SPI**. Tra questi indichiamo i più conosciuti quali : MCP23017, MCP23S17, MCP3008, PCF8573, MAX7219 e molti altri ancora. Tutti con numerose librerie disponibili sul circuito internet.

POS	1	2	3	4	5	6	7	8
M1	PC0	PC1	PC2	PC3	PC4	PC5	GND	+12V
A. Nano ANL	A0	A1	A2	A3	A4/SDA	A5/SCL	ANALOGICI *	
M2	GND	GND	PD7	PD6	PD5	PD4	PD3	PD2
A. Nano I/O	DIGITALI In-Out*		D7	D6	D5	D4	D3	D2
M3 (Relè)	C1	C2	C3	C4	C5	C6	COM	COM
A. Nano I/O	PB0	PB1	PB2/SS	PB3/SI	PB4/SO	PB5/SCK	USCITE Relé *	
ADC AUX	Ingressi ADC disponibili ai PIN AN6 e AN7						ADC6	ADC7

Note** : SDA/SCL/SS/SI/SO/SCK sono I/O condivisi
 Digitali In-Out intesi come I/O bidirezionali

Connettori ausiliari

L'hardware della scheda comprende una serie di connessioni ausiliarie che permettono di implementare il comportamento interattivo di Arduino. Una caratteristica molto interessante di questa scheda è quella di poter espandere Arduino tramite l'uso delle linee **I2C ed SPI**. Questi protocolli seriali permettono di demandare ad una scheda esterna funzioni di elaborazione gravose, che non si vogliono eseguire sul master Arduino Nano. Il bus **I2C** permette collegamenti con dispositivi a due fili, il bus **SPI** permette collegamenti con dispositivi a tre fili. Si ricorda che, per evitare conflitti hardware, è necessario predisporre gli **switch** di selezione collocati all'interno del supporto a 32 pin.

Connettori	1	2	3	4	5
CN1	5V	ADC6	GND		
CN2	5V	ADC7	GND		
CN3	GND	SI	SO	SCK	SS
CN4	GND	3V3	RES	REF	
CN5	NC	5V	GND	RX	TX
CN6	5V	GND	SDA	SCL	

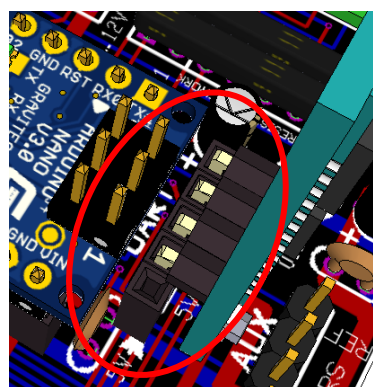
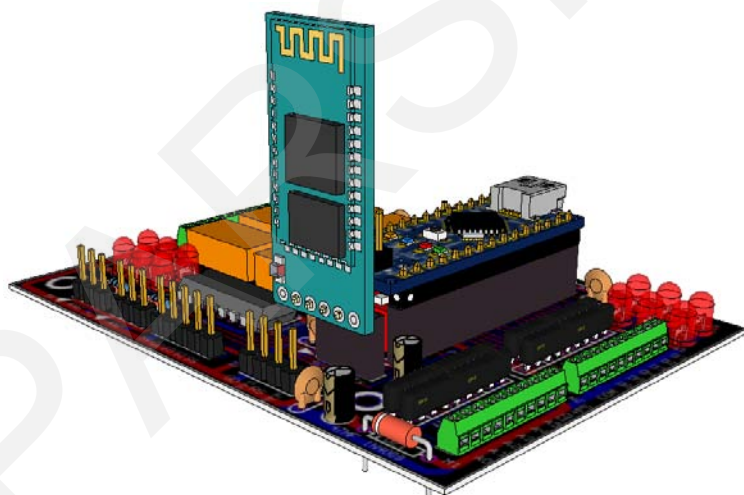


Impiegando le APP appositamente progettate per Arduino è possibile il controllo a distanza di questa scheda.

Connessione Bluetooth (UART RX-TX)

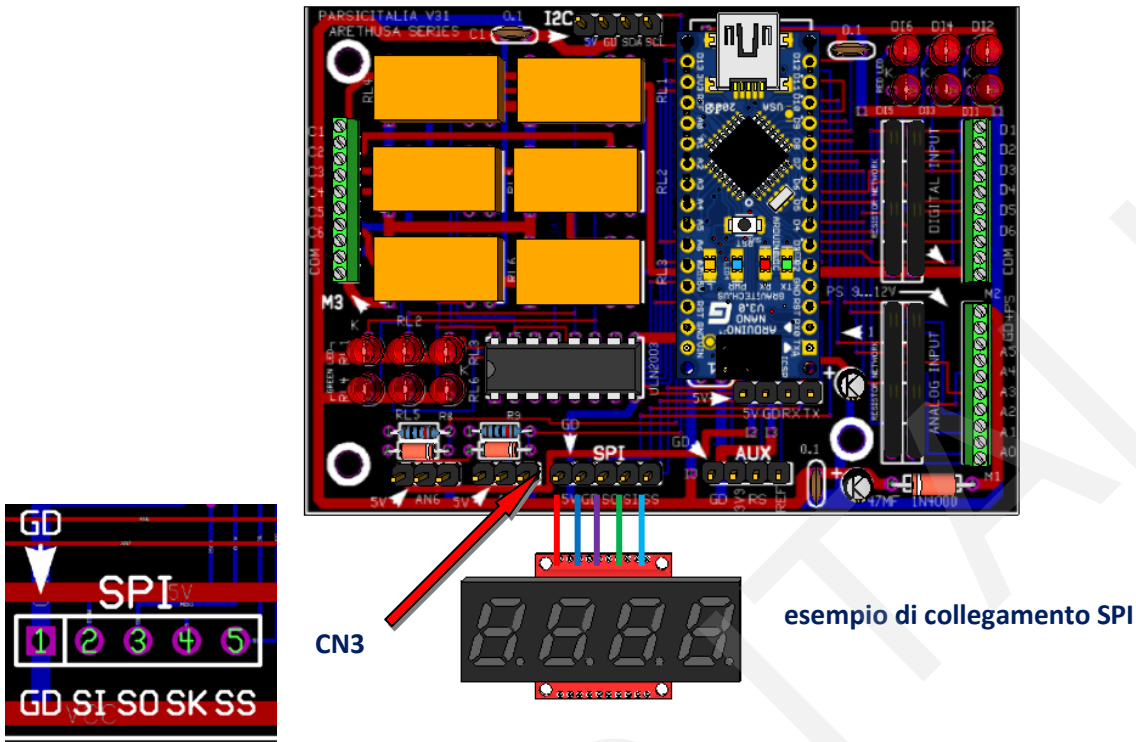
La scheda V31 è predisposta per connettere direttamente al connettore **CN5** una scheda Bluetooth, tipo **HCS**, posizionandola come in figura.

N.B. Il pin n.1 di CN5 è disconnesso dal PCB . Vedi particolare in figura



Connessione SPI - I2C

Per collegare un dispositivo esterno, al bus **SPI**, impiegare il connettore **CN3**.



Connettore CN3 SPI

- **GND**
- **MOSI**
- **MISO**
- **SCK**
- **SS**

Ricordarsi di aprire gli switch corrispondenti prima di effettuare un collegamento

Connettore CN6 I2C

- **1 +5V**
- **2 GND**
- **3 SDA**
- **4 SCL**

Ricordarsi di aprire gli switch corrispondenti prima di effettuare un collegamento

E ormai accertato che la maggiorparte delle architetture disponibili sul mercato, come gli 8051, x86, ARM, PIC, AVR, MSP, STM, ecc. sono corredate di periferiche **I2C ed SPI**. Questo permette ampia portabilità dei sistemi, potendo collegare insieme due architetture differenti e dispositivi periferici come memorie, RTC, expander digitali, WiFi, ecc. La scheda V31AN si connette alle periferiche, in modalità **SPI** per mezzo di 4 fili, oltre il collegamento comune di massa. Il collegamento è di tipo **Master - Slave** ed ogni periferica, secondo il suo costruttore, adotta un sistema di gestione diverso dei propri registri. Conviene sempre consultare il datasheet relativo alla periferica impiegata per avere tutti i dettagli tecnici. **SPI**, è un sistema di trasmissione dati ideato da **Motorola** e poi, col seguire degli anni, è stato adottato praticamente da gran parte dei costruttori di chip. L'interfaccia fisica è composta essenzialmente da una linea seriale su cui sono inviati blocchi dati sincronizzati da un segnale di clock.

I collegamenti SPI sono così caratterizzati :

- **SCLK o SCK** è una linea di clock serve a sincronizzare i dati ;
- **MOSI o SDO** è una linea dati e serve a trasferire i dati dal Master allo Slave;
- **MISO o SDI** è una linea che è impiegata dal Master per ricevere i dati dallo Slave;
- **SS o CS** è la linea che abilita uno più dispositivi Slave, uno alla volta.

Mentre le prime tre linee sono quelle comuni del bus SPI, la linea **CS** è quella che permette di selezionare, uno alla volta, i dispositivi Slave. Se la linea CS non è abilitata, i dati che sono inviati sul bus sono ignorati dal dispositivo Slave. Dato che parliamo di un sistema di comunicazione full-duplex, è previsto che i dati siano inviati dal Master sulla linea MOSI o SDO mentre lo Slave invia i dati sulla linea MISO o SDI.

Il funzionamento del protocollo SPI è basato su poche linee di comunicazione che, impiegando appositi registri a scorrimento, shift register, trasformano ciascun byte da trasmettere o ricevere, in una sequenza ordinata di bit, passando dalla modalità parallelo a quella seriale e viceversa. Non essendoci bit di parità e stop la sincronizzazione dei dati è affidata al clock.

Oltre ai collegamenti di clock e dati, esiste il collegamento di selezione del dispositivo Slave : **SS Slave Select** oppure **CS Chip Select**. Questo terminale permette la comunicazione con più Slave tutti connessi con il medesimo Master.

Quando il CS è a livello logico alto, è inattivo e qualunque dato presente all'ingresso viene ignorato dallo Slave. Dato che ogni collegamento Slave richiede un controllo CS, si devono predisporre, al Master, altrettanti pin di controllo CS tanti quanto sono le periferiche da gestire.

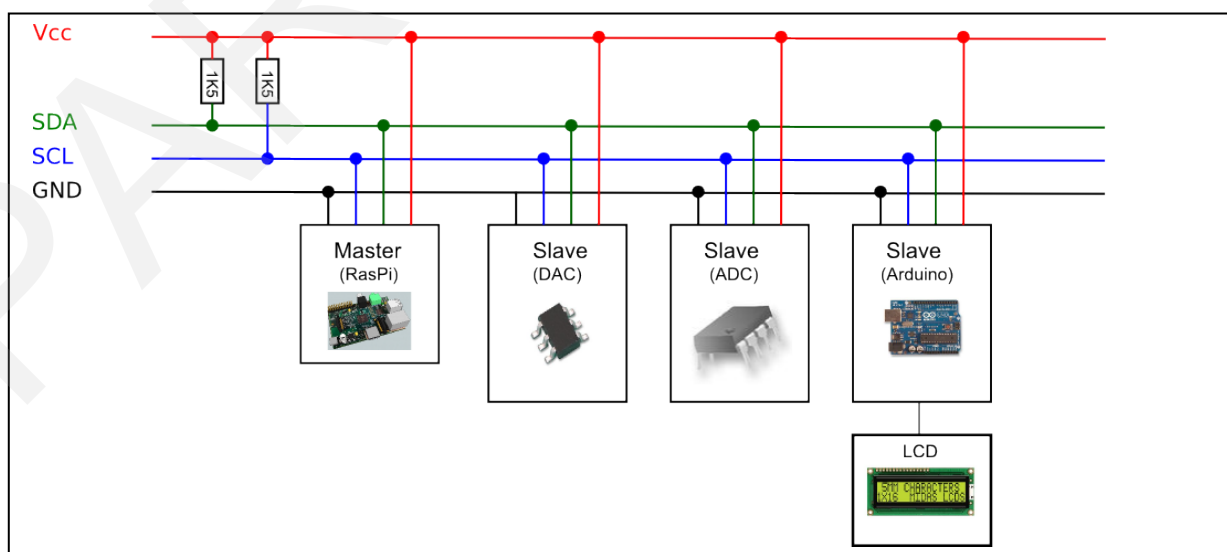
Abbiamo accennato che ogni costruttore struttura i propri chip in modo diverso, pertanto i registri delle MCU assumeranno una propria configurazione interna, secondo le regole del protocollo SPI.

Link : [Arduino SPI reference](#)

Connessione I2C

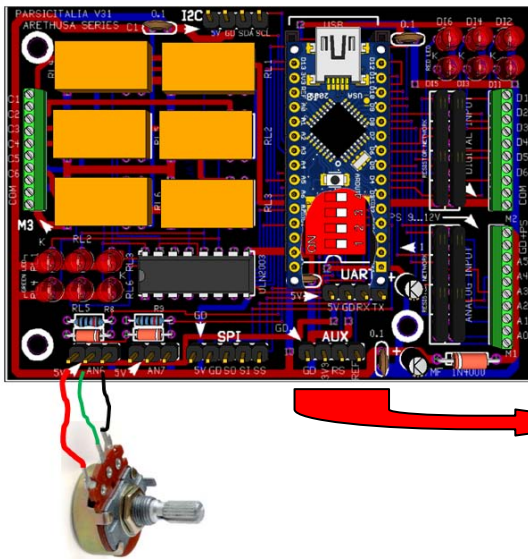
Il bus I2C consiste di due linee bidirezionali che sono tenute alte a Vdd per mezzo di resistenze di pull-up. La linea SDA corrisponde ai dati seriali e SCL è il clock seriale. I moduli standard I2C supportano indirizzi slave a 7 bit e supportano fino a 112/128 nodi, alcuni moduli estesi supportano indirizzamenti slave a 10 bit. Le velocità standard si aggirano intorno ai 10kbps per la modalità Low-speed, 100kbps nella modalità Fast-plus e 3.4Mbps nella modalità Hi-speed. I tre tipi di messaggi definiti dal **protocollo I2C** sono un singolo messaggio che il master scrive ad uno slave, un singolo messaggio che il master legge da uno slave e un messaggio combinato dove il master dispone almeno due letture e/o scritture ad uno o più slave. La sequenza di comunicazione inizia con il master che invia un bit di start seguito dall'indirizzo dello slave a 7 o 10 bit e infine un bit che seleziona l'operazione: 1 per leggere e 0 per scrivere. A questo punto, se l'indirizzo dello slave esiste sul bus, lo slave invierà un bit di acknowledgment al master. I dati sono dunque trasmessi sulla linea SDA nella direzione specificata dal master. Un bit di **acknowledgment** viene mandato alla fine della trasmissione di ogni byte fino al ricevimento della fine della trasmissione. L'unica eccezione si ha quando il master è in modalità ricezione e lo slave in modalità trasmissione il master non invierà nessun bit di acknowledgment dopo l'invio dell'ultimo bit ricevuto. In ultimo la comunicazione viene terminata con il master che invia un comando di stop. I comandi di **start e stop** altro non sono che semplici transizioni dall'alto al basso (start) sulla linea SDA con SCL alta, o dal livello basso al livello alto (stop) sulla linea SDA con SCL a livello alto. Le transizioni per i bit di dati sono sempre eseguite quando la linea SCL è bassa; lo stato alto è solo per i comandi di start e stop.

Il collegamento I2C fa capo al connettore **CN6** [Arduino wire library](#)



Connessioni ADC C6 - C7 CN1-CN2. Connettore riferimenti e servizio CN5

La scheda dispone di due ingressi analogici ausiliari **ADC6 - ADC7**, disponibili sui terminali **CN1 e CN2**. A questi ingressi può essere collegata qualunque sorgente analogica esterna, con range di variazione contenuta entro 5V.
 Nota : gli ingressi **ADC6 e ADC7** possono funzionare soltanto come ingressi analogici.



Connettore riferimenti ausiliari

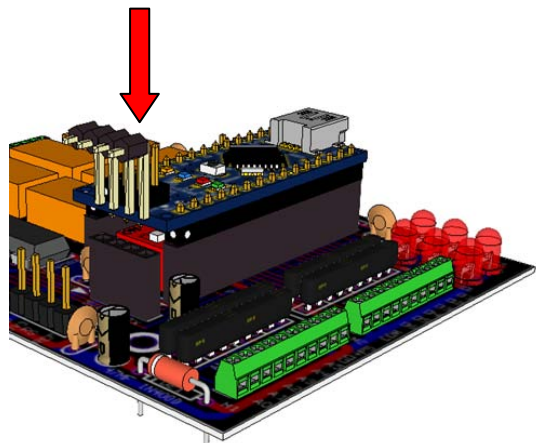
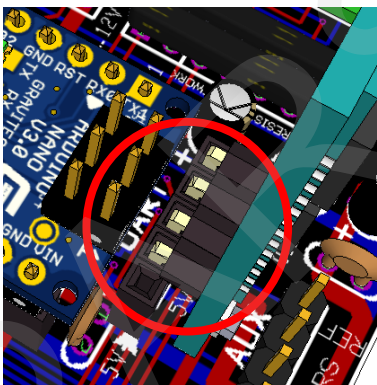
Il connettore **CN5**, fornisce i seguenti segnali di servizio e riferimento :

- 1 GND
- 2 3,3V (derivato dal FT232TL)
- 3 Reset
- 4 ADC REF

Connettore CN5 UART

Al connettore **CN5** possono essere collegati dispositivi di comunicazione seriale. Il connettore è un 5 poli, ma solo quattro di essi sono realmente collegati al circuito. Il pin n.1 è disconnesso (vedi foto precedente) :

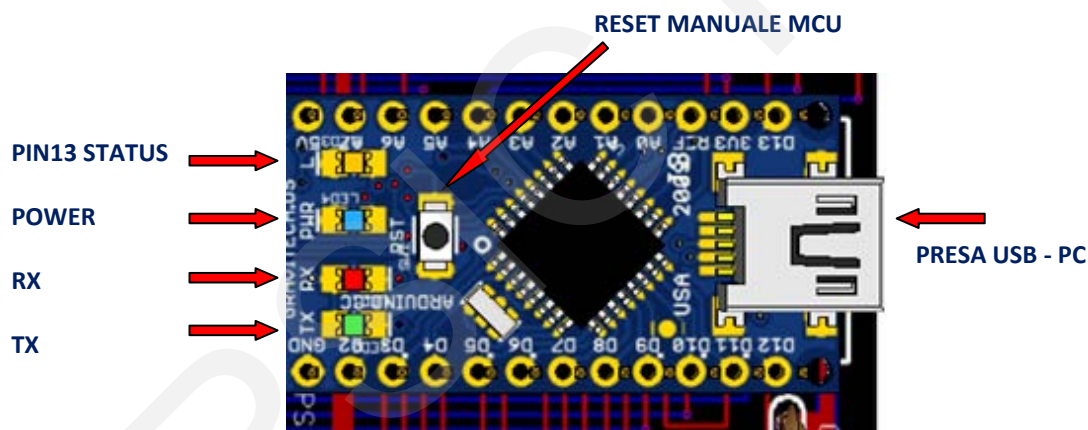
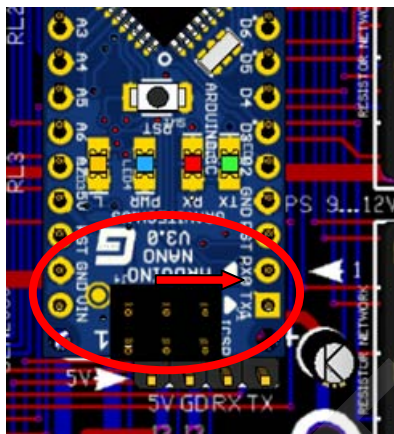
- 1 non connesso
- 2 +5V
- 3 RX
- 4 TX



Il software Arduino include una libreria **Wire** per semplificare l'uso dell' **I2C bus ed SPI**. Per l'impiego di queste librerie si consiglia di consultare il sito <http://arduino.cc/en/Reference/Wire>

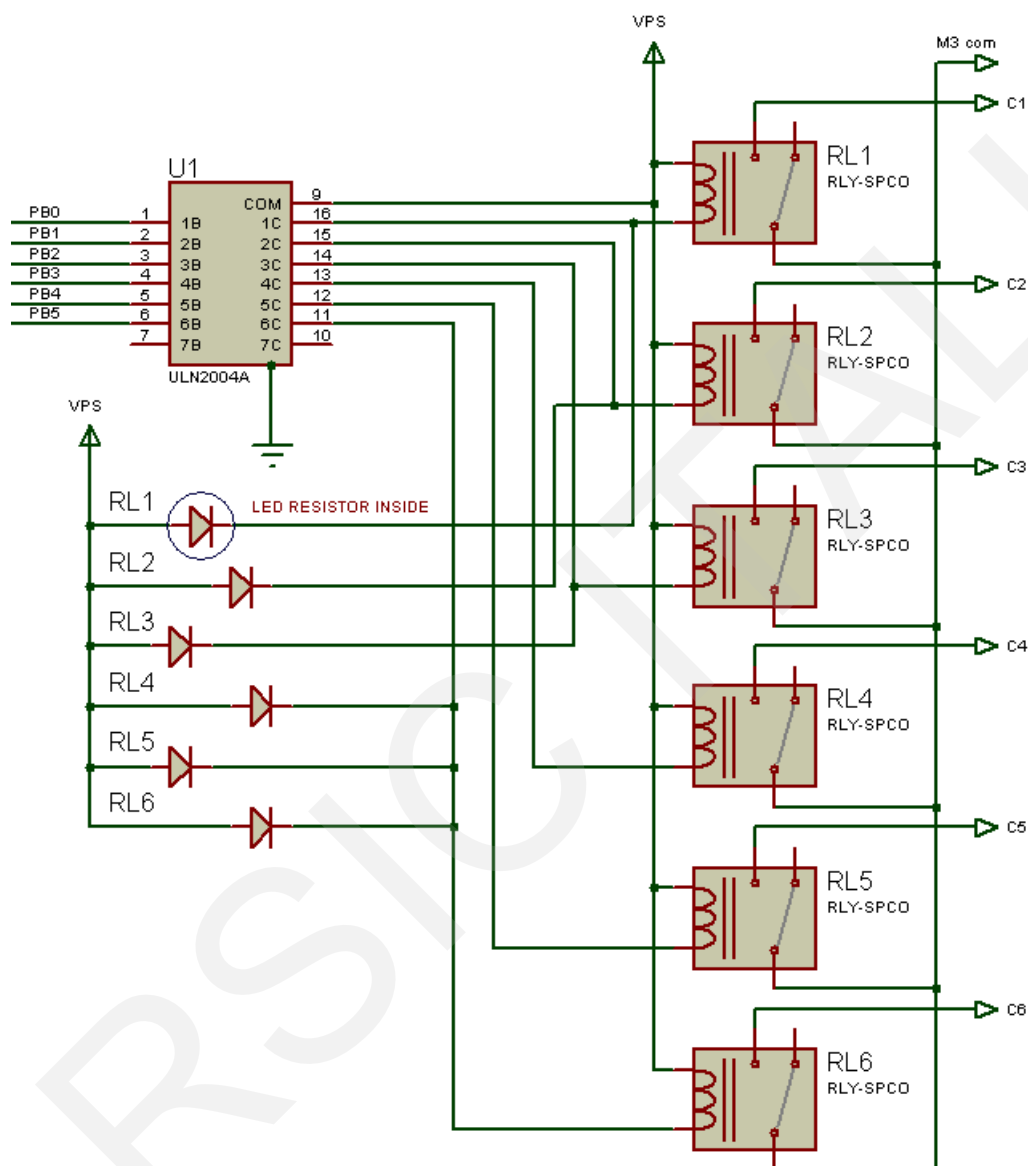
Connettore ICSP

E' localizzato sulla scheda Arduino Nano. L' **ATmega328** sulla Arduino Nano è dotato di un bootloader che permette di caricare il codice programma, senza l'uso di un programmatore hardware esterno. E' possibile ignorare il bootloader, caricando l'eseguibile attraverso il connettore **ICSP**, collegandolo ad un programmatore esterno. Il caricamento del codice da host, è del tutto automatico quando si lavora con l'IDE di Arduino. Un circuito, a bordo dell'Arduino Nano, permette di resettare il microcontrollore automaticamente, prima dell'invio dei dati alla memoria programma.



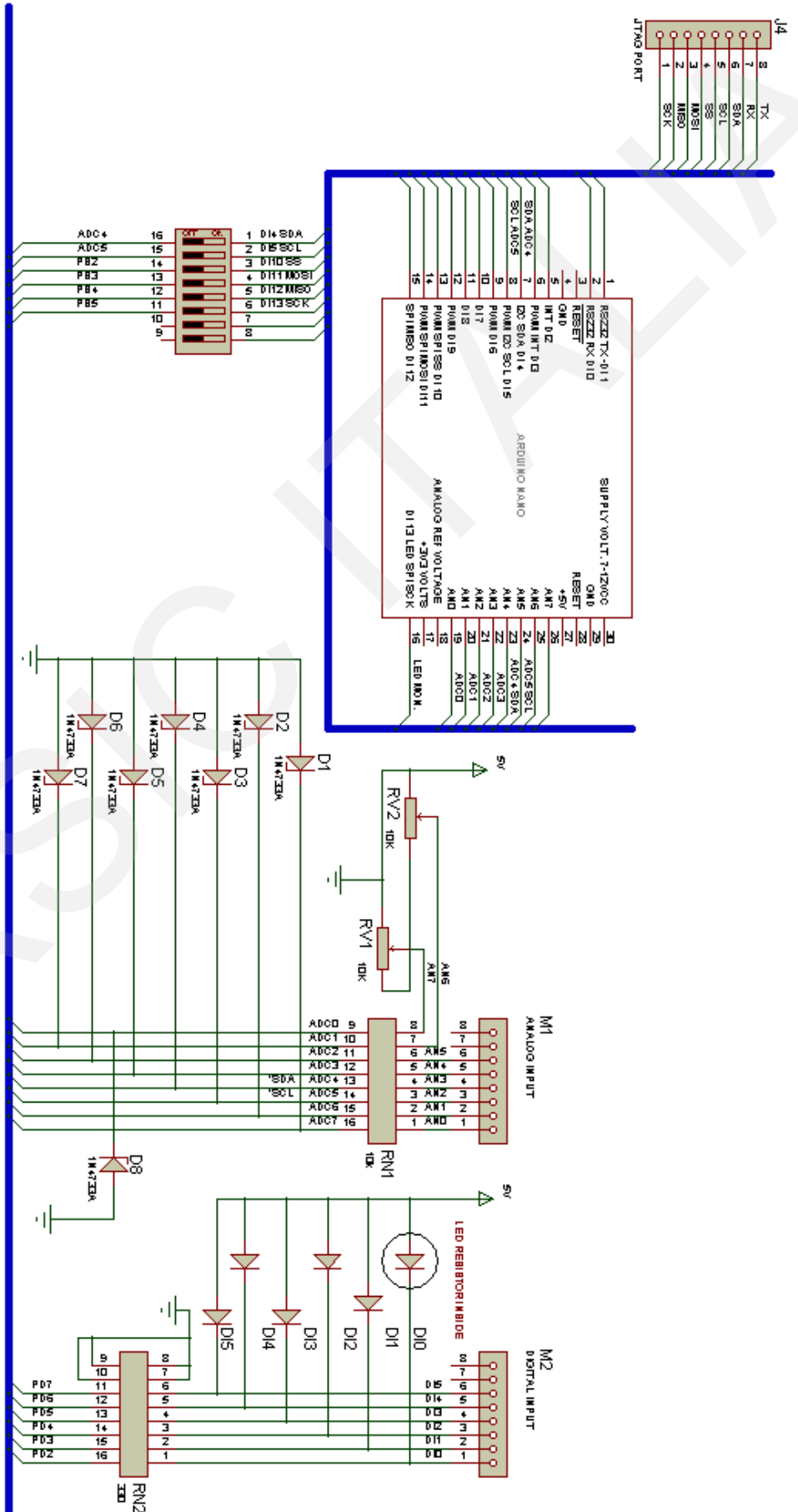
Uscite digitali

Le uscite digitali da **RL1 a RL6**, sono collegate ai **PORT PB0-PB5** dell'Arduino Nano. I contatti dei relè possono commutare carichi sia in **DC** che in **AC** fino ad un massimo di **1 Ampere**. La corrente indicata, è riferita a carichi DC resistivi oppure induttivi in **AC1**. Per carichi differenti o superiori collegare all'esterno opportuni relè di potenza. Si consideri che i relè sono collegati tutti con il contatto centrale in **comune** ai morsetti **7-8** della connettore **M3**.

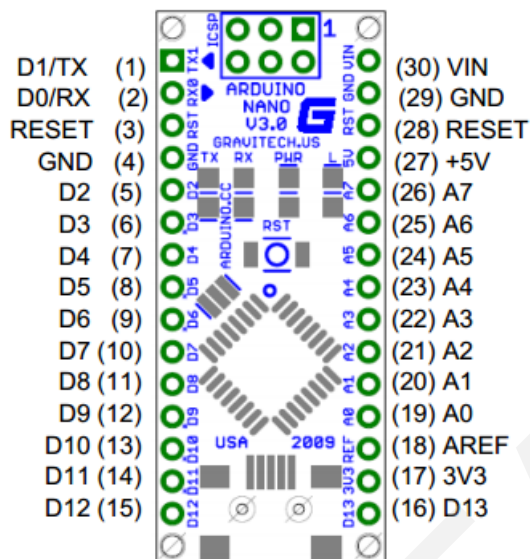


Uscite digitali a relè

Ingressi Analogici e digitali . JTAG porte seriali



Arduino Nano Pin Layout



Pin No.	Name	Type	Description
1-2, 5-16	D0-D13	I/O	Digital input/output port 0 to 13
3, 28	RESET	Input	Reset (active low)
4, 29	GND	PWR	Supply ground
17	3V3	Output	+3.3V output (from FTDI)
18	AREF	Input	ADC reference
19-26	A0-A7	Input	Analog input channel 0 to 7
27	+5V	Output or Input	+5V output (from on-board regulator) or +5V (input from external power supply)
30	VIN	PWR	Supply voltage

Indirizzi internet utili :

- Microchip® MCP23S17 <http://ww1.microchip.com/downloads/en/DeviceDoc/21952b.pdf>
- Microchip MCP3008 <http://ww1.microchip.com/downloads/en/DeviceDoc/21295C.pdf>
- Atmel® ATMEGA328 <http://www.atmel.com/Images/doc8161.pdf>
- Atmel® Setup and use of the SPI <http://www.atmel.com/images/doc2585.pdf>
- Arduino® SPI library <http://arduino.cc/en/Reference/SPI>
- Arduino® Reference Wire <http://arduino.cc/en/Reference/Wire>
- Arduino® Arduino Nano <http://arduino.cc/en/Main/arduinoBoardNano>
- Gravitech Bluetooth <http://arduino.cc/en/Main/ArduinoBoardBT?from=Main.ArduinoBoardBluetooth>

- Bibliografia
- Wikipedia Arduino Hardware
 - Arduino.cc sito ufficiale Arduino
 - ITemcelettronica Comunicazione seriale I2C SPI
 - Gravitech Arduino Nano 3.0