

n. 1 / ottobre 2008

GiornaLinux

2.0



In questo numero:

Scrivere documenti in \LaTeX
Logical Volume Manager
Arduino
Linux nelle aziende



POuL tra tradizione e rinnovamento

Hendrix <hendrixhacks@gmail.com>

O RMAI SONO PASSATI circa 2 anni da quando per la prima volta sono entrato in contatto con il Politecnico Open unix Labs. Ho conosciuto questa associazione per caso grazie a Giacomo e Guido, membri del direttivo del POuL, contribuendo insieme a loro ad organizzare la prima conferenza “Software libero e libero pensiero” e la prima edizione dei Corsi su Linux presso l'Educafé, eventi che hanno visto la straordinaria partecipazione di centinaia di studenti. In questi 7 anni di attività il POuL ha realizzato e promosso diverse iniziative dal primo numero del Giornalinux (che vi invito a rileggere sul nostro sito), alle giornate sulle Libertà Digitali, al workshop sulla sicurezza informatica sino al seminario Libero Pensiero che ha visto la partecipazione, tra i tanti, di personaggi quali Paolo Attivissimo e Alessandro Rubini.

Da allora l'associazione ha fatto passi da gigante, è cresciuta e si è rinnovata nelle persone e nelle idee, e ha trovato una bellissima sede. Nonostante questi risultati non siamo mai cambiati, non è cambiata la nostra passione verso temi quali la condivisione della conoscenza, non è cambia-

ta la nostra sana curiosità verso l'hacking informatico. Su queste basi il gruppo si è formato e continuerà ad esistere ancora per molto tempo.

Dopo aver organizzato durante lo scorso anno accademico il secondo workshop sulla sicurezza informatica, la seconda edizione dei corsi gratuiti per Linux ed aver dato vita al progetto Polibuntu (<http://groups.google.com/group/polibuntu>), nello speciale dell'ultima pagina potete trovare i nostri prossimi progetti.




Non mi rimane quindi che augurarvi buona lettura ricordandovi che: “il software è come il sesso, meglio se libero!”



Indice

L^AT_EX	3
Logical Volume Manager	6
Linux in azienda	8
Arduino	10
Lo speciale dello zio POUl	14

Quest'opera è rilasciata sotto la licenza Creative Commons BY-NC-SA 2.5. Questo significa che sei libero di riprodurre, distribuire, comunicare al pubblico, esporre in pubblico, rappresentare, eseguire o recitare l'opera e creare opere derivate alle seguenti condizioni:

-  **Attribuzione.** Devi riconoscere il contributo dell'autore originario.
-  **Non commerciale.** Non puoi usare quest'opera per scopi commerciali.
-  **Condividi allo stesso modo.** Se alteri, trasformi o sviluppi quest'opera, puoi distribuire l'opera risultante solo per mezzo di una licenza identica a questa.

In occasione di ogni atto di riutilizzo o distribuzione, devi chiarire agli altri i termini della licenza di quest'opera. Se ottieni il permesso dal titolare del diritto d'autore, è possibile rinunciare ad ognuna di queste condizioni.

Le tue utilizzazioni libere e gli altri diritti non sono in nessun modo limitati da quanto sopra.

Questo è un riassunto in linguaggio accessibile a tutti del Codice Legale:

<http://creativecommons.org/licenses/by-nc-sa/2.5/it/legalcode>



L^AT_EX

Nicola Vitucci <nick.vitucci@yahoo.it>

L^AT_EX, questo sconosciuto

CHI HA mai sentito parlare di LaTeX? Ogni tanto all'università si sente qualcuno dire "Oh, quello ha scritto la tesi in LaTeX" e si nutre per il soggetto in questione quasi un timore reverenziale, come se LaTeX fosse chissà quale bestia da domare.

In realtà, scrivere un documento in LaTeX (che, tra l'altro, si legge *Latech*) richiede poco e rende molto: l'effetto che si ottiene è decisamente professionale, a fronte di una piccola fatica per imparare il linguaggio. Ma siamo all'università per imparare, no?

Cos'è LaTeX

LaTeX è un linguaggio usato per la composizione tipografica, cioè per rendere un documento adatto alla stampa; è strutturato in maniera tale da rendere possibile la separazione delle fasi di scrittura e di formattazione del documento.

LaTeX, creato da Leslie Lamport, è un'estensione di TeX, creato da Donald Knu-

th, ed è sia il nome del linguaggio che del software che lo compila.

Strumenti necessari

Per produrre un documento in LaTeX, tutto ciò che serve è... un editor di testo e un "compilatore", cioè un programma che processa il file sorgente. TeX produce file in formato DVI, ma esistono vari programmi che convertono il formato DVI in PostScript o PDF; esiste inoltre PDFLaTeX, che invece genera direttamente file in formato PDF, così come esistono tanti editor grafici (come LyX o Kile) che rendono più semplice la creazione del file sorgente. Sono disponibili anche versioni per Windows, come MiKTeX.

In questo tutorial, supporremo di avere un semplice editor di testo e di usare PDFLaTeX per la compilazione.

Un semplice articolo

Con questo tutorial creeremo un articolo scientifico a due colonne.

Apriamo un file vuoto nel nostro editor e inseriamo la riga:

```
\documentclass[11pt,a4paper,
twocolumn]{article}
```

che indica un documento di tipo articolo, con un font base di 11 punti, stampabile su un foglio A4 e suddiviso in due colonne. Subito dopo inseriamo le righe:

```
\usepackage[italian]{babel}
\usepackage[utf8]{inputenc}
```

che permettono rispettivamente di usare le convenzioni tipografiche italiane e di usare le lettere accentate (senza questo pacchetto, ad esempio, una “è” e una “é” dovrebbero essere inserite come \‘e o \’e).

Passiamo ora alle informazioni sull’autore e sul titolo dell’articolo, ottenute con le righe

```
\title{Articolo di prova}
\author{Nicola Vitucci}
```

Inserendo poi la riga

```
\pagestyle{headings}
```

stabiliamo che l’intestazione di ogni pagina deve contenere il numero di pagina e il capitolo o la sezione corrente. Non resta ora che iniziare a scrivere il documento vero e proprio: scriviamo perciò la riga

```
\begin{document}
```

per creare la sezione del titolo e iniziamo a scrivere il nostro documento, inserendo prima di tutto la riga

```
\maketitle
```

per creare la sezione del titolo e dell’autore, e avendo cura di ricordarci di inserire, alla fine, la riga

```
\end{document}
```

(volendo possiamo inserirla subito dopo `\begin{document}` e scrivere il nostro documento in mezzo).

Per andare a capo, si può usare `\\` oppure `\newline`.

Piccola chicca: per inserire un po’ di testo, possiamo ricorrere al fantomatico “Lorem Ipsum”, un testo misterioso che si trova ogni tanto nei modelli di presentazione o di documento e che, in realtà, è un espediente tipografico per provare il font e le impostazioni di pagina. Un generatore di testi adatti allo scopo può essere trovato al sito <http://www.blindtextgenerator.com/it>.

Dopo aver inserito il testo, salviamo il file come *prova.tex* e dalla shell lanciamo il comando

```
pdflatex prova.tex
```

ottenendo così il file *prova.pdf*, che può essere aperto con un qualsiasi lettore di file PDF.

Estensioni

Ora che sappiamo come creare un semplice articolo, possiamo sbizzarrirci con le altre possibilità offerte da LaTeX. Per esempio, possiamo decidere di dividere l’articolo in sezioni e sottosezioni, inserendo le righe

```
\section{TitoloSezione}
\subsection{TitoloSottosezione}
```

come in questo esempio:

```

...
\begin{document}
\maketitle

\section{Introduzione}
Parliamo di \LaTeX \ldots

\section{Scopo dell'articolo}
Con questo articolo \ldots
\subsection{Applicazioni}
Le possibili applicazioni \ldots

\end{document}

```

Le istruzioni `\LaTeX` e `\ldots` servono rispettivamente per generare il logo di `LaTeX` e i puntini di sospensione. Volendo possiamo aggiungere la riga

```
\tableofcontents
```

dopo `\maketitle` per creare un sommario all'inizio dell'articolo.

Se vogliamo inserire una tabella:

```

\begin{center}
\begin{tabular}{|l|c|r|}
\hline
Lorem & ipsum dolor & sit amet\\
consectetur & adipiscing & elit\\
\hline
\end{tabular}
\end{center}

```

in cui la coppia `\begin{center} ... \end{center}` serve a mettere la tabella al centro, mentre la coppia `\begin{tabular} ... \end{tabular}` serve a creare la tabella; le righe `\hline` servono a creare delle linee orizzontali, mentre la `&` serve a separare le varie colonne. I parametri `{|l|c|r|}` indicano

che il testo nella tabella sarà allineato a sinistra nella prima colonna, al centro nella seconda e a destra nella terza (`l`, `c` e `r` stanno per “left”, “center” e “right”).

Per aggiungere un'immagine, bisogna prima di tutto inserire la riga:

```

\usepackage{graphicx}

nell'intestazione, e poi inserire la riga:

\includegraphics[width=12cm]
{nomefile.jpg}

```

nel punto in cui vogliamo inserire l'immagine; il parametro `width` (che non è obbligatorio inserire) è la larghezza effettiva da dare all'immagine nella pagina, mentre `nomefile.jpg` è il file contenente l'immagine desiderata. Se si compila con *pdflatex*, i formati possibili per le immagini sono PDF, JPG e PNG.

È possibile che l'immagine non venga posta esattamente nel punto in cui la si desidera, ma il corretto posizionamento è un po' più complesso e richiede qualche sforzo in più. Una piccola nota di carattere pratico: se l'indice non rispecchia esattamente l'ordine dei capitoli o se si verificano altri “strani problemi”, a volte è sufficiente lanciare la compilazione per due volte di seguito.

E così abbiamo costruito il nostro primo articolo in `LaTeX`! Nel prossimo numero vedremo come scrivere un libro o una tesi. Alla prossima!

Logical Volume Manager

Saten <satene.r@gmail.com>

LVM... cioè?

LOGICAL VOLUME MANAGEMENT è un sistema che dona flessibilità alla gestione dello spazio su disco. Su Linux esiste un'implementazione di Logical Volume Management che si chiama LVM, ossia Logical Volume Manager.

Ho deciso di scrivere un articolo in proposito perché, nonostante le ottime ragioni per adoperare questo strumento, non sono a conoscenza di alcun amico che ne faccia uso, né ero io stesso a conoscenza dei suoi molti vantaggi fino a poco tempo fa. Eppure con LVM sia il sistemista della server farm che lo smanettone da centro di calcolo casalingo saranno contenti!

Cosa ci si può fare

Ad esempio, chi come me possiede un netbook saprà che lo spazio a disposizione sul disco allo stato solido è davvero esiguo. Tuttavia, il lettore per le memorie come le SD card è integrato e il prezzo di queste ultime è in permanente discesa.



Niente di più bello che integrare i 4 Gigabyte del disco allo stato solido con i 4, 8, 16 Gigabyte della SDHC pagata qualche deci-

na d'euro! Si pone però il problema di come sfruttare tutto lo spazio a disposizione efficientemente.

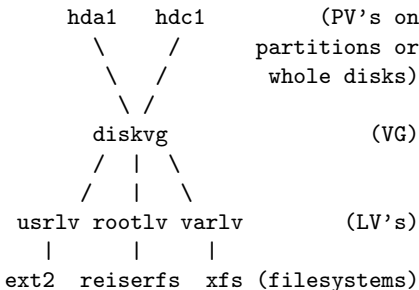
Con LVM si può tranquillamente inserire la SD card ed estendere ad essa il volume logico che si vuole senza preoccuparsi di spostare nulla o di ridimensionare partizioni preesistenti. Il sistemista, che deve gestire un centinaio di dischi con relativo accounting degli utenti, backup e ottimizzazione dello sfruttamento dello spazio disco, sarà felicissimo di limitarsi ad inserire il disco nuovo e stare a guardare mentre viene utilizzato automaticamente senza doversi curare di altro.

Chi abbia un minimo di dimestichezza con l'installazione di una distribuzione GNU/Linux sa quanto possa essere frustrante partizionare un disco senza sapere a priori quanto spazio sarà realmente necessario per `/usr` piuttosto che per `/home`. È una cosa che, personalmente, mi ha sempre dato fastidio.

Come funziona

Quello che l'utente fa in genere è mettere tutto insieme appassionatamente nella stessa partizione, che non è proprio il massimo dal punto di vista della velocità e della manutenibilità. Con LVM, invece, è sufficiente mappare i vari dispositivi che si hanno a disposizione (PV, *physical volume*) in gruppi di volumi logici (VG, (*logical volume group*)). Questo avviene suddividendo ogni volume logico (LV) in sottoparti logiche (LE, *logical extent*) che hanno una controparte fisica (PE, *physical extent*) nei vari PV.

A questo punto si può ragionare in termini di LV astruendo dall'ubicazione fisica dei PE, con i vantaggi descritti sopra. Un volume logico può essere formattato con un file system esattamente come un volume fisico e fondamentalmente si tratta allo stesso modo. Bisogna solo prestare attenzione al fatto che la partizione */boot* non venga gestita da LVM, in modo da facilitare il compito del boot loader. Per dare un senso più tangibile al funzionamento di LVM, riporto uno schema preso da una mailing list:



Please try this at home!

È tutto, spero di aver acceso l'interesse del lettore verso questo software che merita davvero tanto. Per provarlo (purtroppo) bisogna partire da una situazione pulita sui dischi, quindi non è possibile introdurlo senza avere almeno delle partizioni vuote da prendere come PV. Io consiglio di leggere la documentazione online, magari smanettare su una qualche macchina virtuale e poi tentare la sorte alla prima formattazione del proprio PC che capita!

Linux in azienda

KieZz <kiezz@qdsistemi.it>

È arrivato il momento?

QUANTE VOLTE state lavorando serenamente, siete concentrati e vi arriva una chiamata al vostro interno e una voce vi dice: “Ciao, scusa se ti disturbo, non avresti cinque minuti perchè il pc dell’amministrazione si è bloccato e tu sei l’unico che ci capisce qualcosa”? Da questo momento in poi sapete già che minimo due ore del vostro tempo saranno dedicate a fare i salti mortali per sistemare quel PC Windows infestato da virus o altre schifezze. Ricordo con particolare divertimento il PC della segretaria che ogni 20 minuti effettuava una redirect ad una pagina per l’allungamento del membro maschile, con massimo stupore della segretaria che si vedeva davanti questa scena... non propriamente da ufficio.

Linux, nelle versioni più friendly (Ubuntu, Fedora ecc.), può essere considerato ormai un rimpiazzo per il sistema operativo Microsoft nei casi in cui il PC amministrativo abbia le tipiche mansioni da ufficio, come la gestione della posta, la lettura e scrittura di file di Microsoft Office, la stampa e scansione di documenti, la condivisione dei file e la navigazione su Internet. Li-

nux non può essere un rimpiazzo quando sul PC aziendale viene caricato un applicativo, nel 99% dei casi “Win32 only”, per la gestione delle attività aziendali quali preparazione bolle, fatture, gestione contabilità, gestione magazzino, eccetera.

Questo segmento dell’IT Business, che possiamo pensare composto da piccoli software gestionali dedicati alle piccole e medie imprese, si sta lentamente riducendo grazie al Cloud Computing ed alle applicazioni Web based/SOA che stanno permeando il mondo delle grandi aziende come quello delle PMI. Una grosso incentivo a questa novità è dato anche dal fatto che lentamente ma inesorabilmente sta avvenendo un ricambio generazionale: i giovani italiani, anche se non sveltano in Europa per l’utilizzo dei PC, sono cresciuti con Internet, in un contesto tecnologicamente dinamico che ha formato le loro menti ad accettare i cambiamenti e le innovazioni tecnologiche.

Una volta era impensabile anche proporre una cosa del genere... Ricordo frasi del tipo: “Ah, non Linux... È bello ma è un sistema per smanettoni”. Ma adesso un utente aziendale spesso utilizza Windows, Firefox, Thunderbird ed OpenOffice, il passo successivo tra qualche anno potrebbe essere rimpiazzare lo stesso Windows.

Questo non avviene ancora perché la resistenza al cambiamento è una piaga che affligge molto spesso l'IT aziendale. Avviene un po' per cultura e un po' per abitudine, l'utente aziendale a differenza dell'utente consumer è molto molto meno propenso al cambiamento. Ogni novità significa un possibile rallentamento dell'attività lavorativa, la minaccia di perdere tempo dietro a beghe di natura diversa dal solito tran tran lavorativo. Un incubo. I datori di lavoro fuggono dalla possibilità di un'immediata diminuzione della produttività, mentre il dipendente, che spesso ha superato da un po' i 40, ha il terrore di dover passare più tempo in azienda per imparare nuovamente qualcosa che sostanzialmente crede di poter già fare con i vecchi sistemi.

La Microsoft fortunatamente è il maggior sponsor del cambiamento: se non facesse qualcosa di nuovo, come potrebbe rinvigorire il fatturato? Windows Vista si è abbattuto come un fulmine sulla tranquilla mediocrità introdotta da Windows XP, che è sì un sistema pesantemente buggato, ma che alla sua introduzione non creò nessuna discontinuità con il passato. Ora è diverso, Vista ha introdotto delle novità che, seppur di facciata, hanno messo in crisi i tranquilli utenti aziendali. Nonostante quello che dice Ballmer, in azienda nessuno lo vuole, e se viene trovato in qualche PC nuovo viene subito formattato/downgradato in favore di XP.

Questo forse è il momento. Noi sostenitori del Pinguino pensiamo di sì. Ecco una delle liste di pro che noi sostenitori di Linux solitamente facciamo:

1. esente da virus, malware, schifezze varie;

2. il supporto per hardware aziendale e periferiche di massa è ormai sufficientemente vasto da permettere l'acquisto di hardware delle principali marche con relativa tranquillità;
3. supporto della condivisione di file in rete con altri PC Windows;
4. Firefox, Thunderbird, OpenOffice sono strumenti che anche su piattaforma Windows sono oramai largamente utilizzati;
5. l'investimento in termini di tempo effettuato nell'apprendere l'utilizzo di una nuova piattaforma verrà ripagato con il tempo salvato nella risoluzione dei problemi periodici di Windows.

Per usufruire di questi vantaggi è necessario investire del tempo nell'apprendimento, ma se vi è all'interno dell'azienda qualcuno che già ha esperienza, l'investimento sarà sicuramente inferiore. Ogni ambiente aziendale è diverso e fare un calcolo del ROI (Return of Investment) generale è azzardato, ma il mio parere è che quello relativo alla migrazione da XP a Linux, ove possibile, potrà sicuramente insidiare quello relativo alla migrazione da XP a Vista.

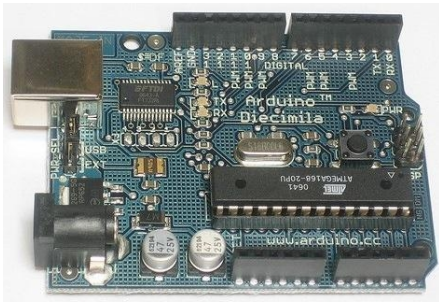


Arduino

Marco Triverio <gtrive@gmail.com>

Che cos'è Arduino?

VI PIACEREBBE comandare l'iPod via Bluetooth? Vi piacerebbe comandare la macchina del caffè dal computer? Sapete poco o niente di elettronica e di programmazione? Allora Arduino fa per voi!



Arduino è una piattaforma di sviluppo open-source e si presenta nelle fattezze di una piccola scheda delle dimensioni di un pacchetto di sigarette. Su di essa sono saldati tutti i componenti che vi servono: il "cervello elettronico" (ovvero un microcontrollore ATmega), i componenti per collegare Arduino al computer (in pratica una porta USB e un convertitore seriale-USB) e le connessioni ver-

so l'esterno (digital input/output e analog input).

Cosa si può fare con Arduino?

Dopo aver spiegato alle persone cosa sia Arduino, queste mi domandano cosa esattamente si possa fare con questa scheda: la risposta non è così facile perché Arduino è davvero molto flessibile e le possibilità sono pressoché infinite.

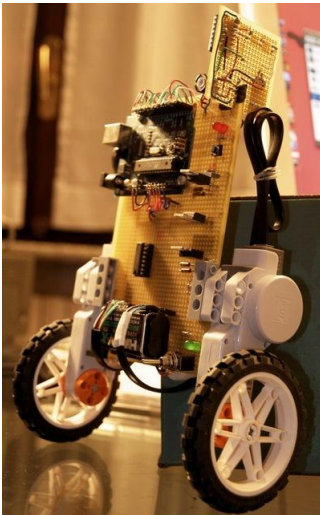
Per cercare di soddisfare la vostra curiosità vi propongo alcuni esempi di oggetti realizzati con Arduino:

- macchina del caffè Gaggia controllata da Arduino e collegabile al computer: finalmente potrete dire "Il mio PC fa anche il caffè!":
<http://www.blog.nashlincoln.com/espesso/gaggia-espesso-pid-arduino-mod>
- 3D Modelling prendendo a calci una sacca da boxe:
<http://www.fluidforms.at/de/CassiusVideo.php>
- schermo interattivo:
<http://it.youtube.com/watch?v=a0QkrCiZdf4>

- comandare un'interfaccia fatta in Flash utilizzando Arduino e dei sensori di distanza:

<http://it.youtube.com/watch?v=4Vgqo1VK0zM>

- Arduino fa anche parte del mio progetto di tesi di primo livello; ho infatti costruito "Arduway", un robot con due ruote poste sullo stesso asse (proprio come un Segway) in grado di rimanere in equilibrio.



Come vedete le possibilità sono più o meno sterminate... (se invece non ne avete ancora abbastanza, date un occhio a <http://www.arduino.cc/playground/Projects/ArduinoUsers>).

Ma è davvero possibile creare tutto questo con conoscenze di elettronica e informatica basilari? La risposta è: sì! Come avete forse notato dagli esempi precedenti, uno degli utilizzi principali di Ardui-

no è quello che ne fanno artisti e disegner all'interno di esposizioni animate.

Incominciamo!

La prima cosa da fare è acquistare un Arduino (andate su www.arduino.cc e cliccate su "Buy") o farvene prestare uno da qualche vostro amico: dovrete installare tutto il software necessario, solitamente l'IDE (disponibile anche per Linux su <http://www.arduino.cc/playground/Learning/Linux>) e un driver. A questo punto siete pronti a fare il vostro primo esperimento: far lampeggiare un led. Se possedete un Arduino poco recente (come "Arduino NG" o precedenti) dovrete acquistare un LED in un qualunque negozio di elettronica; se invece avete un Arduino più recente (come ad esempio "Arduino Diecimila") potete saltare questo passo e passare alla creazione del software.

A questo punto aprite il programma e scrivete il seguente codice:

```
// il LED è collegato al digital
// pin 13
int ledPin = 13;
```

```
// eseguito una sola volta
void setup()
{
  // imposta il 13esimo pin
  // come uscita
  pinMode(ledPin, OUTPUT);
}
```

```
// ripetuto all'infinito
void loop()
```

```
{
  // accende il LED
  digitalWrite(ledPin, HIGH);

  // attende un secondo
  delay(1000);

  // spegne il LED
  digitalWrite(ledPin, LOW);

  // attende un secondo
  delay(1000);
}
```

(ulteriori informazioni su <http://www.arduino.cc/en/Tutorial/Blink>)

Non scappate! Se conoscete il C queste istruzioni dovrebbero esservi familiari; in caso contrario non preoccupatevi: imparerete in fretta. Tutti i programmi per Arduino sono divisi in due sezioni, chiamate `setup()` e `loop()`: la prima contiene tutte le istruzioni che devono essere eseguite appena Arduino viene acceso; la seconda invece contiene le istruzioni che verranno eseguite dopo la prima fase e verranno ripetute all'infinito (a meno che non si stacchi la corrente).

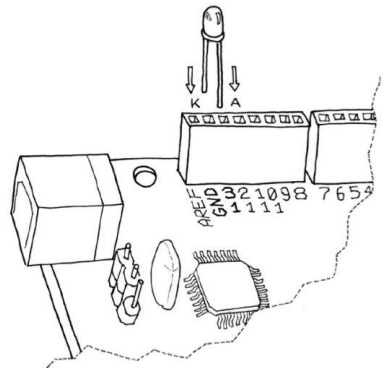
Nel caso in esame la funzione `setup()` include una singola istruzione che imposta il tredicesimo I/O digitale (quello su cui verrà attaccato il led) come uscita. Se invece avessi voluto leggere il valore presente (magari perché impostato da qualche altro dispositivo) su quella porta avrei dovuto digitare "`pinMode(ledPin, INPUT);`". Le istruzioni presenti nella funzione `loop()` invece servono a far lampeggiare il led:

- la funzione `digitalWrite()` imposta un determinato valore di tensione, che

può essere alto (circa 5 Volt) o basso (circa 0 Volt): il valore HIGH farà dunque accendere il led, un valore basso lo manterrà invece spento; chi conosce un pochino di elettronica potrebbe obiettare che 5 V potrebbero bruciare il LED, ma in realtà la tredicesima porta digitale ha in serie una resistenza che limita la corrente e permette di non danneggiarlo;

- l'istruzione `delay()` invece mette il processore di Arduino in attesa per un numero di millisecondi specificato (in questo caso 1000, pari dunque a un secondo): siccome il LED viene acceso e spento con pause di un secondo, lo si vedrà lampeggiare.

Installate a questo punto il led nella tredicesima porta, collegate Arduino alla porta USB (dovrebbe accendersi un piccolo LED) e selezionate dai menu "Tools" > "Serial Port" > "/dev/tty.usbserial" e "Tools" > "Board" > "Nome del vostro Arduino".



Compilate a questo punto il programma premendo "Verify" (il tasto che assomiglia ad un play) e caricatelo cliccando su "Upload to I/O Board"; solo nel caso in

cui abbiate un Arduino meno recente non appena avete cliccato resettate l'Arduino premendo il pulsante presente a fianco del microcontrollore. Se non avete fatto errori dopo pochi istanti il LED lampeggerà.

Comunicazione con il computer

Se vi state chiedendo come comunicare con il computer basta imparare la "procedura standard":

- nella sezione `setup()` include l'istruzione `Serial.begin(9600)`;
- usate `Serial.print("Stringa...");` o `Serial.println("Stringa...");` tutte le volte che volete stampare andando eventualmente a capo;
- per leggere i valori inviati da Arduino avviate il programma e cliccate sul pulsante della toolbar "Serial Monitor".

In basso compariranno tutti i caratteri ricevuti; potrete anche inviarne di vostri ad Arduino: perchè però questi vengano gestiti correttamente il programma dovrà includere l'istruzione `incomingByte = Serial.read()`; che memorizza all'interno della variabile `incomingByte` il valore ricevuto.

Ecco un piccolo programma di esempio:

```
// memorizza valore in ingresso
int incomingByte = 0;
```

```
void setup() {
  // apre la comunicazione
  // seriale impostando un data
```

```
  // rate di 9600 bps
  Serial.begin(9600);
}

void loop() {
  // manda dati solo quando
  // vengono ricevuti dati
  if (Serial.available() > 0)
  {
    // legge il byte inviato
    incomingByte = Serial.read();

    // rispedisce il byte letto
    Serial.print("Ho ricevuto: ");
    Serial.println(incomingByte,
                   DEC);
  }
}
```

E ora?

Ora date sfogo alla vostra fantasia e attendete con ansia il prossimo numero del *Giornalinux*, in cui spiegherò una tematica leggermente più avanzata. Buon divertimento!

Lo speciale dello zio POuL

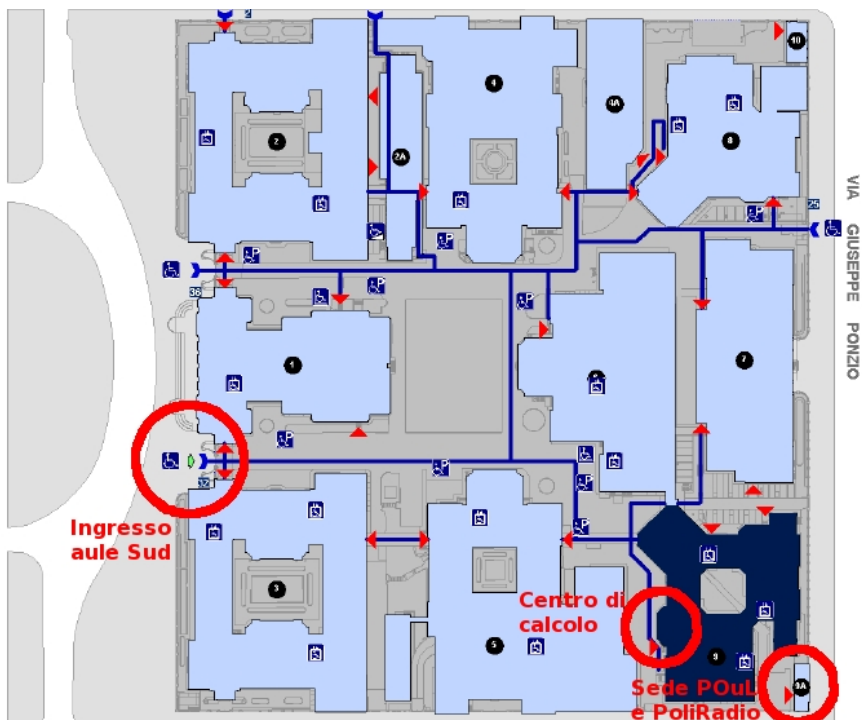
... la sezione delle comunicazioni da e verso lo zio POuL!

Agenda

- Il 25 e il 26 Ottobre saremo impegnati nell'ormai famoso evento nazionale "Linux Day" ed "Open Day" a Cinisello Balsamo presso Villa Ghirlanda, due giornate da vivere appieno nel vero spirito della cultura aperta, assieme a numerose associazioni affini tra cui Lifos (www.lifos.org) e Milug (www.milug.org). Potrete conoscerci di persona allo stand che allestiremo o alla grande install fest di Ubuntu, nel corso della quale vi assisteremo nell'installazione e configurazione del sistema operativo Linux... quindi portate i vostri portatili!
- A fine ottobre, in collaborazione con l'Area Servizi Informatici d'ateneo (www.asi.polimi.it), attiveremo uno sportello di assistenza al servizio Wireless offerto dall'università. Assisteremo passo passo nella configurazione chiunque ne abbia bisogno, qualunque sia il portatile e qualunque sia il sistema operativo usato, Windows, MacOS o Linux.



- Dopo il Linux Day organizzeremo al Politecnico di Milano la prima DrupalCon in Italia, una conferenza sul noto CMS Drupal, alla quale parteciperanno importanti realtà del settore che si susseguiranno nel corso di una giornata in vari talks con tanto di buffet finale gratuito.



Vi è venuta voglia di conoscere il mondo di Linux? Volete partecipare più da vicino alle nostre attività? Volete scrivere un articolo su questa rivista?
Iscrivetevi alla nostra mailing list oppure venite a trovarci presso la nostra sede!
E non dimenticate di controllare la Webmail del Poli: pubblicizziamo sempre con questo mezzo le nostre attività!

sito Internet: www.poul.org
numero sede: +39 02 2399 2477



La stampa della rivista è interamente finanziata dal Politecnico di Milano, che non si assume alcuna responsabilità sul contenuto.

Stampa a cura di Acheias di S. Siragusa, Milano 2008.