

# PENGENALAN ARDUINO

Juli 2011

Oleh : Feri Djuandi

Tingkat:



Pemula



Menengah



Mahir

## DAFTAR ISI:

1. MENGENAL ARDUINO.....	1
2. JENIS-JENIS PAPAN ARDUINO.....	5
3. BAGIAN-BAGIAN PAPAN ARDUINO.....	9
4. SOFTWARE ARDUINO .....	12
4.1. MENGINSTALL SOFTWARE ARDUINO .....	13
4.2. MENGINSTALL DRIVER USB PADA WINDOWS XP .....	13
4.3. MENGINSTALL DRIVER USB PADA WINDOWS 7.....	16
4.4. MENGUJI KONEKSI KOMPUTER DAN PAPAN ARDUINO .....	21

## 1. MENGENAL ARDUINO

Untuk memahami Arduino, terlebih dahulu kita harus memahami terlebih dahulu apa yang dimaksud dengan *physical computing*. **Physical computing** adalah membuat sebuah sistem atau perangkat fisik dengan menggunakan software dan hardware yang sifatnya interaktif yaitu dapat menerima rangsangan dari lingkungan dan merespon balik. Physical computing adalah sebuah konsep untuk memahami hubungan yang manusiawi antara lingkungan yang sifat alaminya adalah analog dengan dunia digital. Pada prakteknya konsep ini diaplikasikan dalam desain-desain alat atau proyek-proyek yang menggunakan sensor dan *microcontroller* untuk menerjemahkan input analog ke dalam sistem software untuk mengontrol gerakan alat-alat elektro-mekanik seperti lampu, motor dan sebagainya.

Pembuatan *prototype* atau *prototyping* adalah kegiatan yang sangat penting di dalam proses physical computing karena pada tahap inilah seorang perancang melakukan eksperimen dan uji coba dari berbagai jenis komponen, ukuran, parameter, program komputer dan sebagainya berulang-ulang kali sampai diperoleh kombinasi yang paling tepat. Dalam hal ini perhitungan angka-angka dan rumus yang akurat bukanlah satu-satunya faktor yang menjadi kunci sukses di dalam mendesain sebuah alat karena ada banyak faktor eksternal yang turut berperan, sehingga proses mencoba dan menemukan/mengoreksi kesalahan perlu melibatkan hal-hal yang sifatnya non-eksakta. Prototyping adalah gabungan antara akurasi perhitungan dan seni.

Proses prototyping bisa menjadi sebuah kegiatan yang menyenangkan atau menyebalkan, itu tergantung bagaimana kita melakukannya. Misalnya jika untuk mengganti sebuah komponen, merubah ukurannya atau merombak kerja sebuah prototype dibutuhkan usaha yang besar dan waktu yang lama, mungkin prototyping akan sangat melelahkan karena pekerjaan ini dapat dilakukan berulang-ulang sampai puluhan kali – bayangkan betapa frustasinya perancang yang harus melakukan itu. Idealnya sebuah prototype adalah sebuah sistem yang fleksibel dimana perancang bisa dengan mudah dan cepat melakukan perubahan-perubahan dan mencobanya lagi sehingga tenaga dan waktu tidak menjadi kendala berarti. Dengan demikian harus ada sebuah alat pengembangan yang membuat proses prototyping menjadi mudah.

Pada masa lalu (dan masih terjadi hingga hari ini) bekerja dengan hardware berarti membuat rangkaian menggunakan berbagai komponen elektronik seperti resistor, kapasitor, transistor dan sebagainya. Setiap komponen disambungkan secara fisik dengan kabel atau jalur tembaga yang disebut dengan istilah “*hard wired*” sehingga untuk merubah rangkaian maka sambungan-sambungan itu harus diputuskan dan disambung kembali. Dengan hadirnya teknologi digital dan microprocessor fungsi yang sebelumnya dilakukan dengan *hard wired* digantikan dengan program-program software. Ini adalah sebuah revolusi di dalam proses prototyping. Software lebih mudah diubah dibandingkan hardware, dengan beberapa penekanan tombol kita dapat merubah logika alat secara radikal dan mencoba versi ke-dua, ke-tiga dan seterusnya dengan cepat tanpa harus mengubah pengkabelan dari rangkaian.

Saat ini ada beberapa alat pengembangan prototype berbasis microcontroller yang cukup populer, misalnya:

- Arduino → <http://www.arduino.cc>
- I-CubeX → <http://www.infusionsystems.com>
- Arie Robotics Project Junior → <http://www.arobotineveryhome.com>
- Dwengo → <http://www.dwengo.org>
- EmbeddedLab → <http://www.embedded.arch.ethz.ch>
- GP3 → <http://www.awce.com/gp3.htm>

Di antara sekian banyak alat pengembangan prototype, Arduino adalah salah satunya yang paling banyak digunakan.

**Arduino** dikatakan sebagai sebuah *platform* dari *physical computing* yang bersifat *open source*. Pertama-tama perlu dipahami bahwa kata “platform” di sini adalah sebuah pilihan kata yang tepat. Arduino tidak hanya sekedar sebuah alat pengembangan, tetapi ia adalah kombinasi dari hardware, bahasa pemrograman dan **Integrated Development Environment (IDE)** yang canggih. IDE adalah sebuah software yang sangat berperan untuk menulis program, meng-*compile* menjadi kode biner dan meng-*upload* ke dalam *memory* microcontroller. Ada banyak projek dan alat-alat dikembangkan oleh akademisi dan profesional dengan menggunakan Arduino, selain itu juga ada banyak modul-modul pendukung (sensor, tampilan, penggerak dan sebagainya) yang dibuat oleh pihak lain untuk bisa disambungkan dengan Arduino. Arduino berevolusi menjadi sebuah platform karena ia menjadi pilihan dan acuan bagi banyak praktisi.

Salah satu yang membuat Arduino memikat hati banyak orang adalah karena sifatnya yang open source, baik untuk hardware maupun software-nya. Diagram rangkaian elektronik Arduino digratiskan kepada semua orang. Anda bisa bebas men-download gambarnya, membeli komponen-komponennya, membuat PCB-nya dan merangkainya sendiri tanpa harus membayar kepada para pembuat Arduino. Sama halnya dengan IDE Arduino yang bisa di-download dan diinstal pada komputer secara gratis. Kita patut berterima kasih kepada tim Arduino yang sangat dermawan membagi-bagikan kemewahan hasil kerja keras mereka kepada semua orang. Saya pribadi betul-betul kagum dengan desain hardware, bahasa pemrograman dan IDE Arduino yang berkualitas tinggi dan sangat berkelas.

Arduino dikembangkan oleh sebuah tim yang beranggotakan orang-orang dari berbagai belahan dunia. Anggota inti dari tim ini adalah:

- Massimo Banzi Milano, Italy

- David Cuartielles Malmoe, Sweden
- Tom Igoe New York, US
- Gianluca Martino Torino, Italy
- David A. Mellis Boston, MA, USA

Profil mengenai anggota tim tersebut dan kontribusinya bisa diakses pada situs web <http://www.arduino.cc/playground/Main/People>.

Saat ini komunitas Arduino berkembang dengan pesat dan dinamis di berbagai belahan dunia. Berbagai macam kegiatan yang berkaitan dengan proyek-proyek Arduino bermunculan dimana-mana, termasuk di Indonesia. Yang membuat Arduino dengan cepat diterima oleh orang-orang adalah karena:

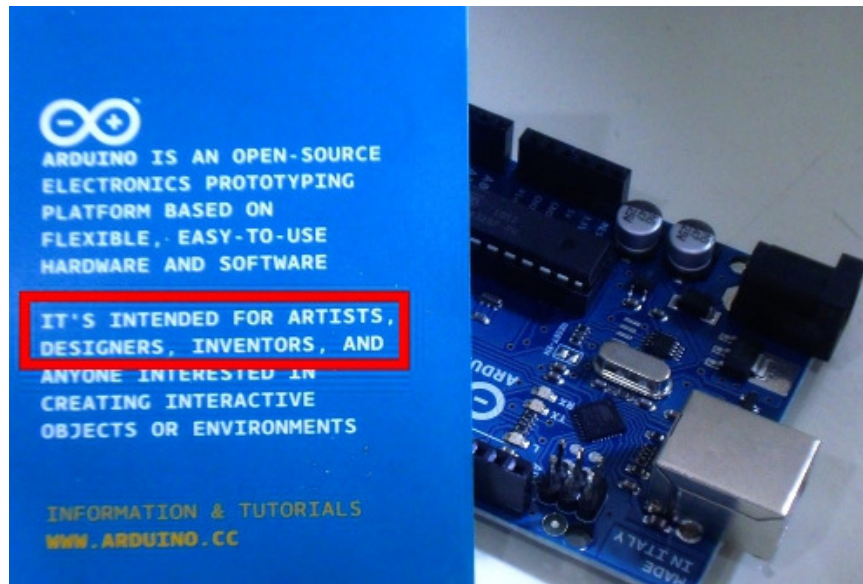
- Murah, dibandingkan platform yang lain. Harga sebuah papan Arduino tipe Uno asli buatan Italia yang saya beli di tahun 2011 seharga Rp 290.000,-. Sebuah investasi yang sangat murah untuk berbagai keperluan proyek. Harganya akan lebih murah lagi jika pengguna membuat papan sendiri dan merangkai komponen-komponennya satu per satu.



- Lintas platform, software Arduino dapat dijalankan pada system operasi Windows, Macintosh OSX dan Linux, sementara platform lain umumnya terbatas hanya pada Windows.
- Sangat mudah dipelajari dan digunakan. **Processing** adalah bahasa pemrograman yang digunakan untuk menulis program di dalam Arduino. Processing adalah bahasa pemrograman tingkat tinggi yang dialektanya sangat mirip dengan C++ dan Java, sehingga pengguna yang sudah terbiasa dengan kedua bahasa tersebut tidak akan menemui kesulitan dengan Processing. Bahasa pemrograman Processing sungguh-sungguh sangat memudahkan dan mempercepat pembuatan sebuah program karena bahasa ini sangat mudah dipelajari dan diaplikasikan dibandingkan bahasa pemrograman tingkat rendah seperti Assembler yang umum digunakan pada platform lain namun cukup sulit. Untuk mengenal Processing lebih lanjut, silakan mengunjungi situs web-nya di <http://www.processing.org>.

- Sistem yang terbuka, baik dari sisi hardware maupun software-nya.

Sangat menarik ketika membuka kotak pembungkus papan Arduino terdapat tulisan bahwa Arduino diperuntukan bagi seniman, perancang dan penemu. Sungguh membesarkan hati dan membangkitkan semangat bahwa penggunaanya tidak harus teknisi berpengalaman atau ilmuwan berotak jenius. Anda tertarik untuk menjadi seniman digital?



Secara umum Arduino terdiri dari dua bagian, yaitu:

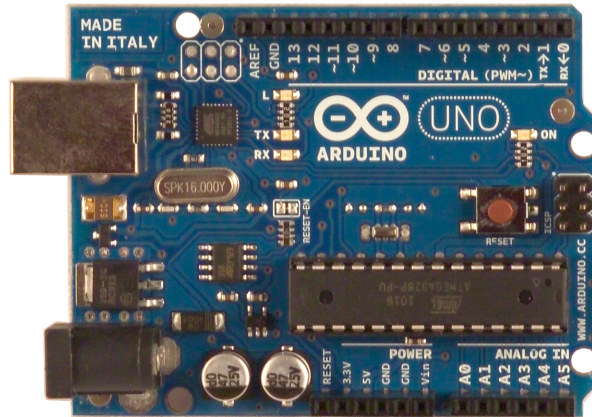
1. Hardware → papan input/output (I/O)
2. Software → Software Arduino meliputi IDE untuk menulis program, *driver* untuk koneksi dengan komputer, contoh program dan *library* untuk pengembangan program.

Selanjutnya kita akan mengenal masing-masing bagian ini lebih jauh.

## 2. JENIS-JENIS PAPAN ARDUINO

Saat ini ada bermacam-macam bentuk papan Arduino yang disesuaikan dengan peruntukannya seperti diperlihatkan berikut ini:

### ARDUINO USB

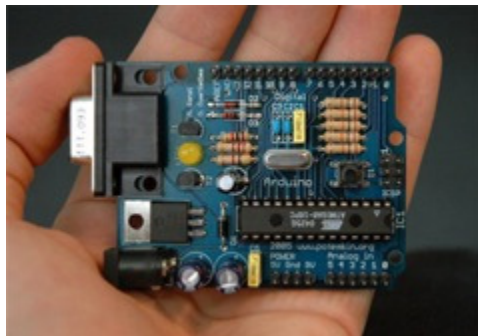


Menggunakan USB sebagai antar muka pemrograman atau komunikasi komputer. Contoh:

- Arduino Uno
- Arduino Duemilanove
- Arduino Diecimila
- Arduino NG Rev. C
- Arduino NG (Nuova Generazione)
- Arduino Extreme dan Arduino Extreme v2
- Arduino USB dan Arduino USB v2.0

### ARDUINO SERIAL

Menggunakan RS232 sebagai antar muka pemrograman atau komunikasi komputer.



Contoh: Arduino Serial dan Arduino Serial v2.0

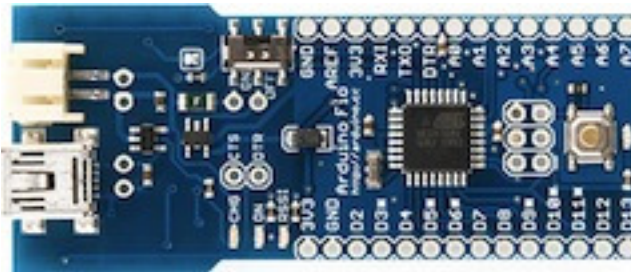
## ARDUINO MEGA



Papan Arduino dengan spesifikasi yang lebih tinggi, dilengkapi tambahan pin digital, pin analog, port serial dan sebagainya. Contoh:

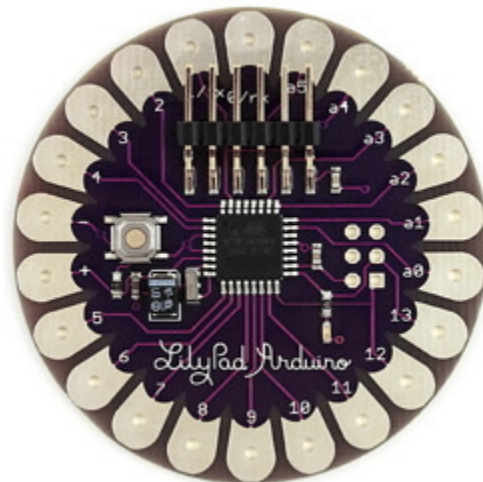
- Arduino Mega
- Arduino Mega 2560

## ARDUINO FIO



Ditujukan untuk penggunaan nirkabel.

## ARDUINO LILYPAD



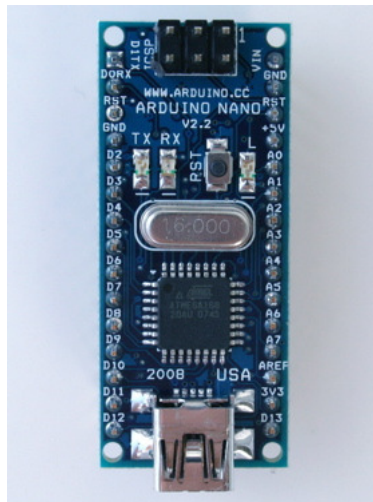
Papan dengan bentuk yang melingkar. Contoh: LilyPad Arduino 00, LilyPad Arduino 01, LilyPad Arduino 02, LilyPad Arduino 03, LilyPad Arduino 04

### ARDUINO BT



Mengandung modul *bluetooth* untuk komunikasi nirkabel.

### ARDUINO NANO DAN ARDUINO MINI



Papan berbentuk kompak dan digunakan bersama *breadboard*. Contoh:

- Arduino Nano 3.0, Arduino Nano 2.x
- Arduino Mini 04, Arduino Mini 03, Arduino Stamp 02

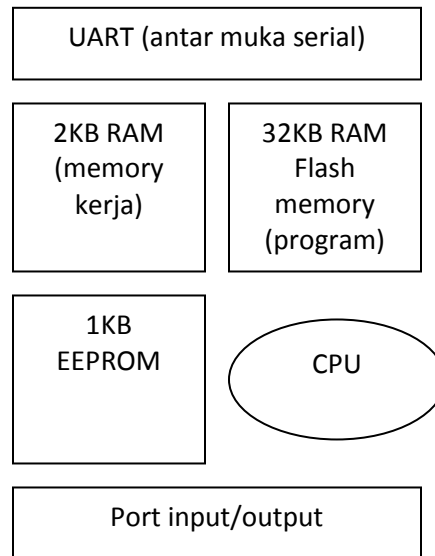


**CATATAN:**

Dengan begitu beragamnya papan Arduino yang ada di pasaran wajar jika seorang pemula akan kebingungan untuk menentukan tipe papan apa yang sebaiknya digunakan. Sebagai sama-sama pemula yang ingin berbagi pengalaman, saya akan menganjurkan untuk memulai dengan tipe *Duemilanove* atau *Uno* mengingat kedua tipe papan ini yang paling banyak digunakan oleh para aktivis Arduino saat ini. Arduino Uno adalah generasi yang terakhir setelah *Duemilanove* dan dari sisi harganya sedikit lebih mahal karena memiliki spesifikasi yang lebih tinggi (microcontroller: Atmega328 dan flash memory: 32 KB).

Komponen utama di dalam papan Arduino adalah sebuah microcontroller 8 bit dengan merk **ATmega** yang dibuat oleh perusahaan **Atmel Corporation**. Berbagai papan Arduino menggunakan tipe ATmega yang berbeda-beda tergantung dari spesifikasinya, sebagai contoh Arduino Uno menggunakan ATmega328 sedangkan Arduino Mega 2560 yang lebih canggih menggunakan ATmega2560.

Untuk memberikan gambaran mengenai apa saja yang terdapat di dalam sebuah microcontroller, pada gambar berikut ini diperlihatkan contoh diagram blok sederhana dari microcontroller ATmega328 (dipakai pada Arduino Uno).



Blok-blok di atas dijelaskan sebagai berikut:

- *Universal Asynchronous Receiver/Transmitter (UART)* adalah antar muka yang digunakan untuk komunikasi serial seperti pada RS-232, RS-422 dan RS-485.
- 2KB RAM pada memory kerja bersifat *volatile* (hilang saat daya dimatikan), digunakan oleh variable-variabel di dalam program.
- 32KB RAM flash memory bersifat *non-volatile*, digunakan untuk menyimpan program yang dimuat dari komputer. Selain program, flash memory juga menyimpan *bootloader*.



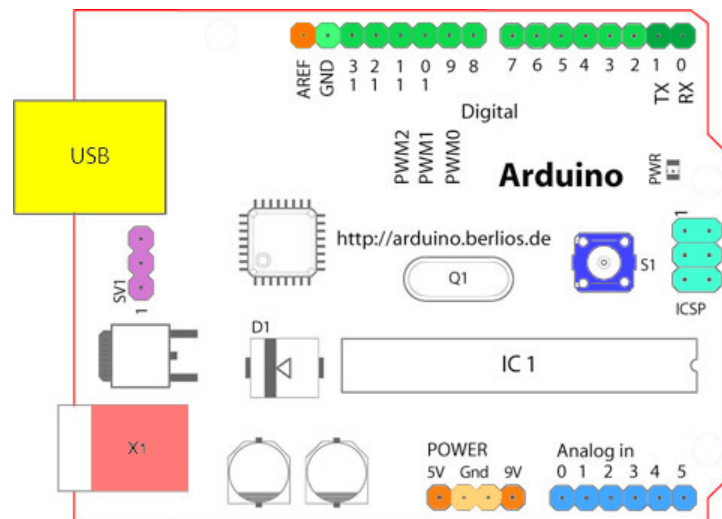
**Bootloader** adalah program inisiasi yang ukurannya kecil, dijalankan oleh CPU saat daya dihidupkan. Setelah bootloader selesai dijalankan, berikutnya program di dalam RAM akan dieksekusi.

- 1KB EEPROM bersifat non-volatile, digunakan untuk menyimpan data yang tidak boleh hilang saat daya dimatikan. Tidak digunakan pada papan Arduino.
- *Central Processing Unit (CPU)*, bagian dari microcontroller untuk menjalankan setiap instruksi dari program.
- Port input/output, pin-pin untuk menerima data (input) digital atau analog, dan mengeluarkan data (output) digital atau analog.

Setelah mengenal bagian-bagian utama dari microcontroller ATmega sebagai komponen utama, selanjutnya kita akan mengenal bagian-bagian dari papan Arduino itu sendiri.

### 3. BAGIAN-BAGIAN PAPAN ARDUINO

Dengan mengambil contoh sebuah papan Arduino tipe USB, bagian-bagiannya dapat dijelaskan sebagai berikut.



#### 14 pin input/output digital (0-13)

Berfungsi sebagai input atau output, dapat diatur oleh program.

Khusus untuk 6 buah pin 3, 5, 6, 9, 10 dan 11, dapat juga berfungsi sebagai pin analog output dimana tegangan output-nya dapat diatur. Nilai sebuah pin output analog dapat diprogram antara 0 – 255, dimana hal itu mewakili nilai tegangan 0 – 5V.

#### USB

Berfungsi untuk:

- Memuat program dari komputer ke dalam papan
- Komunikasi serial antara papan dan komputer

- Memberi daya listrik kepada papan

### Sambungan SV1

Sambungan atau *jumper* untuk memilih sumber daya papan, apakah dari sumber eksternal atau menggunakan USB. Sambungan ini tidak diperlukan lagi pada papan Arduino versi terakhir karena pemilihan sumber daya eksternal atau USB dilakukan secara otomatis.

### Q1 – Kristal (*quartz crystal oscillator*)

Jika microcontroller dianggap sebagai sebuah otak, maka kristal adalah jantung-nya karena komponen ini menghasilkan detak-detak yang dikirim kepada microcontroller agar melakukan sebuah operasi untuk setiap detak-nya. Kristal ini dipilih yang berdetak 16 juta kali per detik (16MHz).

### Tombol Reset S1

Untuk me-reset papan sehingga program akan mulai lagi dari awal. Perhatikan bahwa tombol reset ini bukan untuk menghapus program atau mengosongkan microcontroller.

### In-Circuit Serial Programming (ICSP)

Port ICSP memungkinkan pengguna untuk memprogram microcontroller secara langsung, tanpa melalui bootloader. Umumnya pengguna Arduino tidak melakukan ini sehingga ICSP tidak terlalu dipakai walaupun disediakan.

### IC 1 – Microcontroller Atmega

Komponen utama dari papan Arduino, di dalamnya terdapat CPU, ROM dan RAM.

### X1 – sumber daya eksternal

Jika hendak disuplai dengan sumber daya eksternal, papan Arduino dapat diberikan tegangan DC antara 9-12V.

### 6 pin input analog (0-5)

Pin ini sangat berguna untuk membaca tegangan yang dihasilkan oleh sensor analog, seperti sensor suhu. Program dapat membaca nilai sebuah pin input antara 0 – 1023, dimana hal itu mewakili nilai tegangan 0 – 5V.

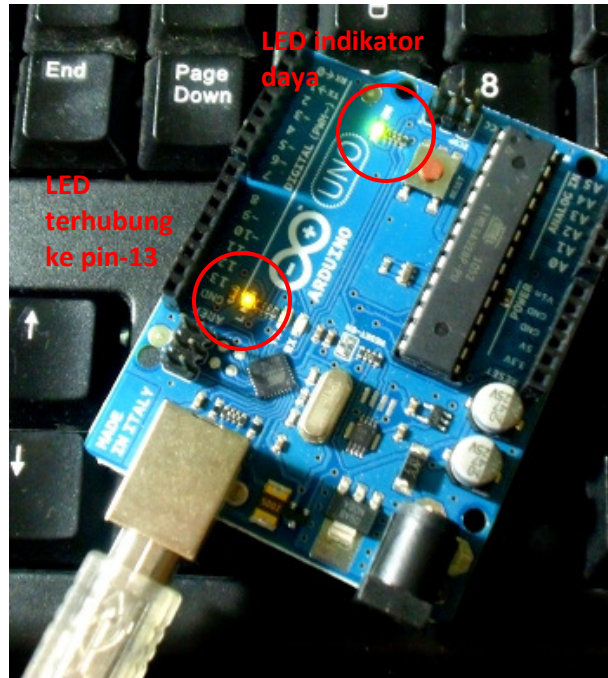


#### CATATAN:

Untuk selanjutnya pembahasan pada dokumen ini akan digunakan papan Arduino yang berbasiskan USB dan papan yang akan dijadikan contoh adalah Arduino Uno.

Tanpa melakukan konfigurasi apapun, begitu sebuah papan Arduino dikeluarkan dari kotak pembungkusnya ia dapat langsung disambungkan ke sebuah komputer melalui kabel USB. Selain

berfungsi sebagai penghubung untuk pertukaran data, kabel USB ini juga akan mengalirkan arus DC 5 Volt kepada papan Arduino sehingga praktis tidak diperlukan sumber daya dari luar. Saat mendapat suplai daya, lampu LED indikator daya pada papan Arduino akan menyala menandakan bahwa ia siap bekerja.



Pada papan Arduino Uno terdapat sebuah LED kecil yang terhubung ke pin digital no 13. LED ini dapat digunakan sebagai output saat seorang pengguna membuat sebuah program dan ia membutuhkan sebuah penanda dari jalannya program tersebut. Ini adalah cara yang praktis saat pengguna melakukan uji coba. Umumnya microcontroller pada papan Arduino telah memuat sebuah program kecil yang akan menyalakan LED tersebut berkedip-kedip dalam jeda satu detik. Jadi sangat mudah untuk menguji apakah sebuah papan Arduino baru dalam kondisi baik atau tidak, cukup sambungkan papan itu dengan sebuah komputer dan perhatikan apakah LED indikator daya menyala konstan dan LED dengan pin-13 itu menyala berkedip-kedip.



**CATATAN:**

Setelah mengeluarkan papan Arduino dari kotaknya, harap berhati-hati dengan listrik statis dan hubungan singkat karena bagian bawah papan Arduino tidak ditutup dengan lapisan pelindung. Dianjurkan untuk tidak menyentuh bagian bawah atau kaki-kaki komponennya dengan tangan untuk menghindari bahaya listrik statis dari tubuh Anda.

Hati-hati juga meletakkan papan Arduino pada meja. Pastikan tidak ada logam atau cairan yang bisa mengakibatkan hubungan pendek yang bisa merusak komponen. Usahakan meletakkan papan Arduino pada alas berbahan plastik yang aman.

Kita akan melakukan pengujian papan Arduino lebih jauh dengan merubah program dan memuatnya ke dalam papan setelah, namun setelah melewati beberapa pembahasan berikut ini terlebih dahulu.

## 4. SOFTWARE ARDUINO

Sehubungan dengan pembahasan untuk saat ini software Arduino yang akan digunakan adalah driver dan IDE, walaupun masih ada beberapa software lain yang sangat berguna selama pengembangan Arduino.

IDE Arduino adalah software yang sangat canggih ditulis dengan menggunakan Java. IDE Arduino terdiri dari:

- *Editor program*, sebuah window yang memungkinkan pengguna menulis dan mengedit program dalam bahasa Processing.
- *Compiler*, sebuah modul yang mengubah kode program (bahasa Processing) menjadi kode biner. Bagaimanapun sebuah microcontroller tidak akan bisa memahami bahasa Processing. Yang bisa dipahami oleh microcontroller adalah kode biner. Itulah sebabnya compiler diperlukan dalam hal ini.
- *Uploader*, sebuah modul yang memuat kode biner dari komputer ke dalam memory di dalam papan Arduino.



### CATATAN:

Sebuah kode program Arduino umumnya disebut dengan istilah **sketch**. Kata “sketch” digunakan secara bergantian dengan “kode program” dimana keduanya memiliki arti yang sama.

Berikut ini adalah contoh tampilan IDE Arduino dengan sebuah sketch yang sedang diedit.

```
Blink | Arduino 0022
File Edit Sketch Tools Help
Blink
/*
  Blink
  Turns on an LED on for one second, then off for one second...

  This example code is in the public domain.
  */

void setup() {
  // initialize the digital pin as an output.
  // Pin 13 has an LED connected on most Arduino boards:
  pinMode(13, OUTPUT);
}

void loop() {
  digitalWrite(13, HIGH); // set the LED on
```

Pembahasan berikutnya akan dijelaskan langkah-langkah untuk menginstal IDE Arduino:

- Mendapatkan software Arduino

- Menginstall driver Arduino
- Menguji koneksi komputer dan papan Arduino

#### 4.1. Menginstall Software Arduino

File instalasi software Arduino dapat diperoleh pada alamat situs web di bawah ini yang tersedia untuk sistem operasi Windows, Mac dan Linux:

<http://arduino.cc/en/Main/Software>

File instalasi ini berbentuk kompresi. Untuk menjalankan software-software Arduino maka file tersebut harus diekstrak ke dalam sebuah direktori. Beberapa software Arduino ditulis menggunakan bahasa pemrograman Java termasuk IDE-nya, sehingga ia tidak perlu diinstal seperti software pada umumnya tapi dapat langsung dijalankan selama komputer Anda telah terinstall Java *runtime*. IDE ini bisa langsung digunakan untuk membuat program namun untuk saat ini belum bisa dipakai untuk berkomunikasi dengan papan Arduino karena driver harus diinstal terlebih dahulu.

#### 4.2. Menginstall Driver USB Pada Windows XP

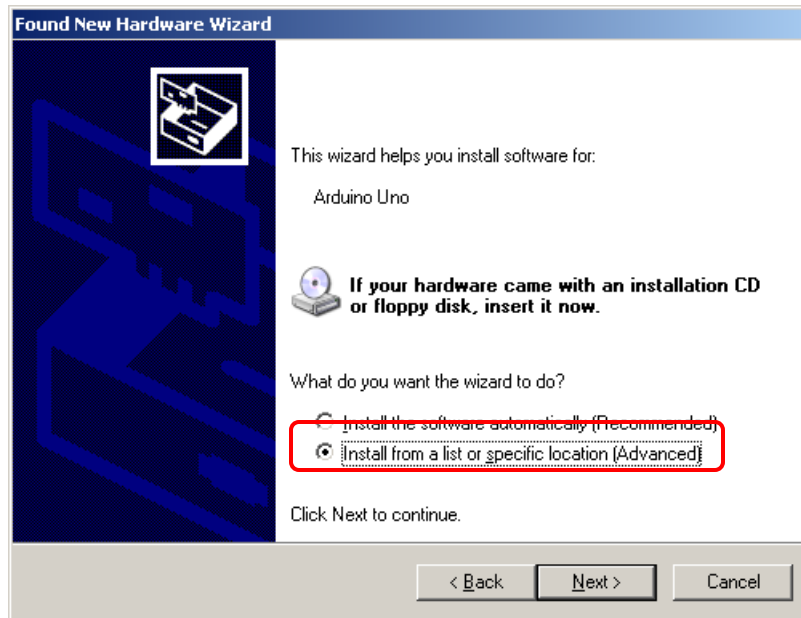
Pada topik ini akan dijelaskan langkah-langkah instalasi driver USB pada Windows XP.

1. Sambungkan papan Arduino dengan sebuah komputer melalui kabel USB.
2. Dengan segera komputer akan mendeteksi kehadiran sebuah perangkat baru yang belum ia kenal dan Windows akan menampilkan sebuah window wizard seperti berikut ini.

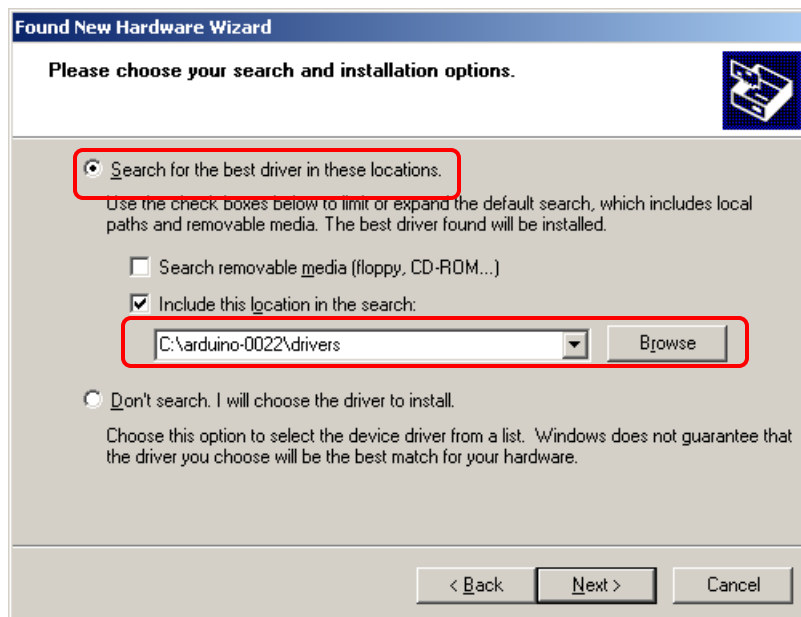


Jawab dengan **“No, not this time”** dan tekan **Next**.

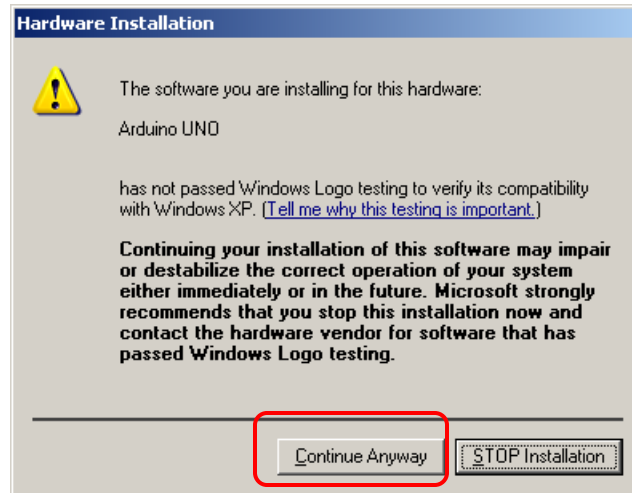
3. Wizard akan mencari software driver untuk perangkat tersebut. Silakan menjawab dengan “**Install from a list or specific location (Advance)**”. Lanjutkan dengan **Next**.



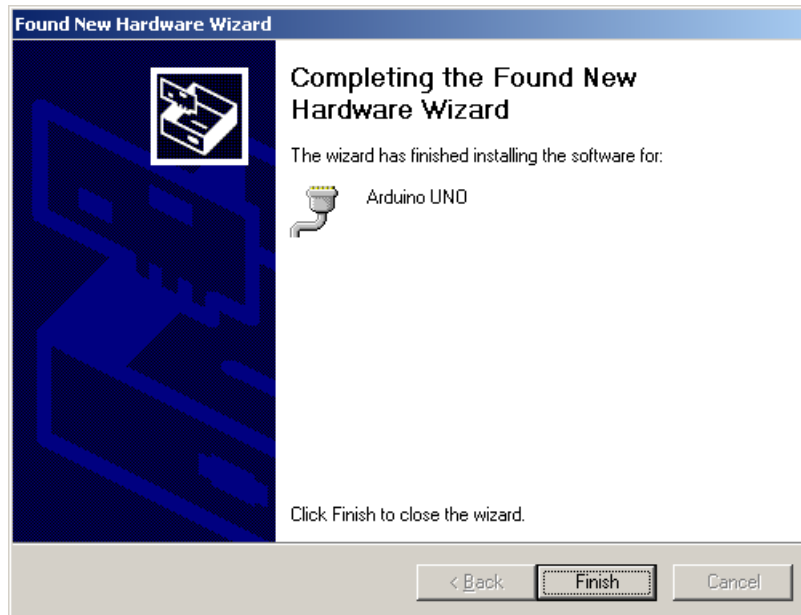
4. Tentukan lokasi dimana software Arduino ditempatkan pada komputer, pada contoh gambar di bawah ini adalah C:\arduino-0022. Silakan sesuaikan lokasinya sesuai dengan hasil ekstrak software Arduino pada komputer Anda. Di dalam lokasi tersebut terdapat sebuah direktori bernama **drivers**, arahkan wizard untuk mencari driver di dalam direktori tersebut.



Klik **Next** untuk melanjutkan. Jika muncul sebuah window peringatan seperti di bawah ini, jawab dengan “**Continue Anyway**”.



5. Jika driver Arduino selesai diinstal pada komputer maka pada akhir proses akan tampil sebuah pesan berhasil seperti berikut ini.



Tekan **Finish** untuk menutup wizard. Driver telah berhasil diinstal.

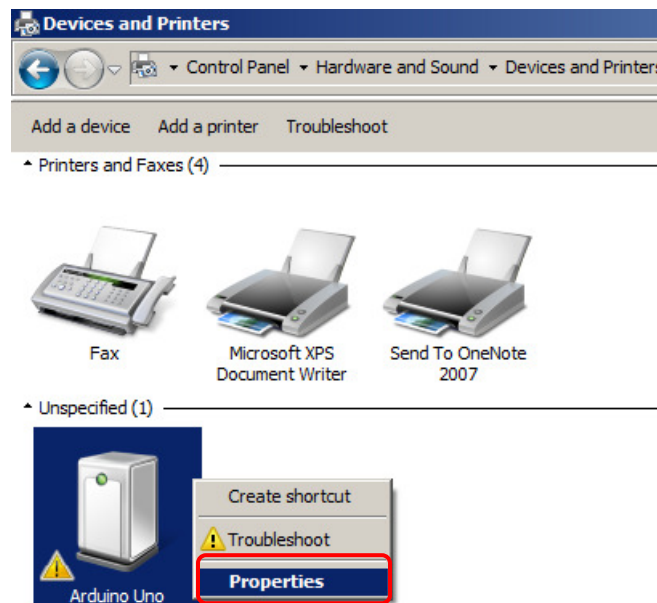
### 4.3. Menginstall Driver USB Pada Windows 7

Pada topik ini akan dijelaskan langkah-langkah instalasi driver USB pada Windows 7.

1. Sambungkan papan Arduino dengan sebuah komputer melalui kabel USB. Umumnya Windows tidak bereaksi apa-apa saat papan Arduino telah terhubung walaupun sebetulnya Windows telah mendeteksi kehadiran sebuah perangkat baru. Untuk memulai instalasi driver, silakan menjalankan program **Control Panel** kemudian memilih **“View devices and printers”**.



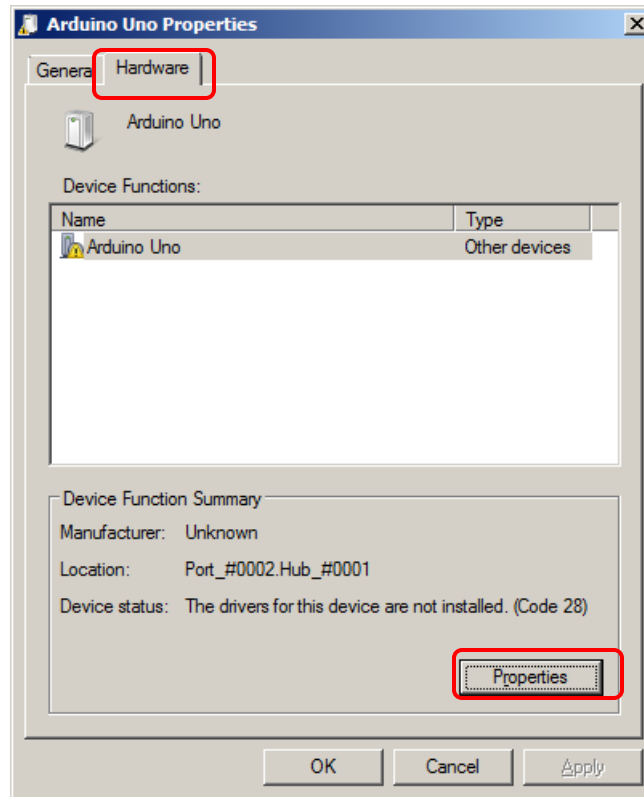
2. Papan Arduino akan tampak pada daftar perangkat namun karena ia belum dikonfigurasi dengan benar maka ia akan muncul di daftar *unspecified* dan di dekatnya tampak sebuah lambang peringatan (segitiga kuning dengan tanda seru) yang artinya perangkat ini belum bekerja dengan benar.



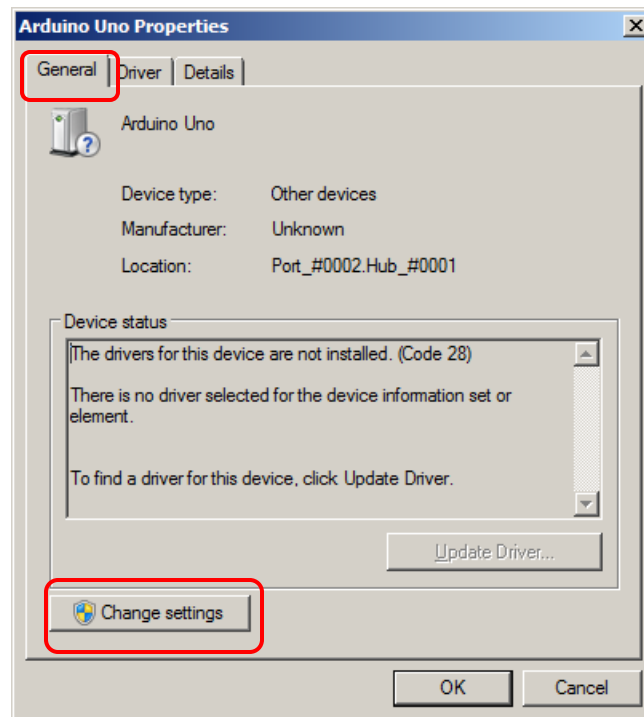
Klik kanan pada icon Arduino kemudian pilih menu **Properties**.



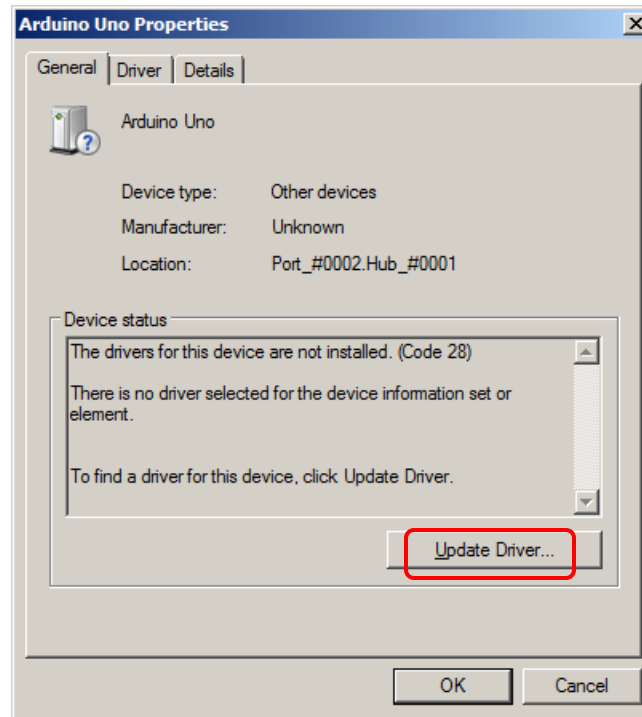
3. Pada tab **Hardware** klik tombol **Properties**.



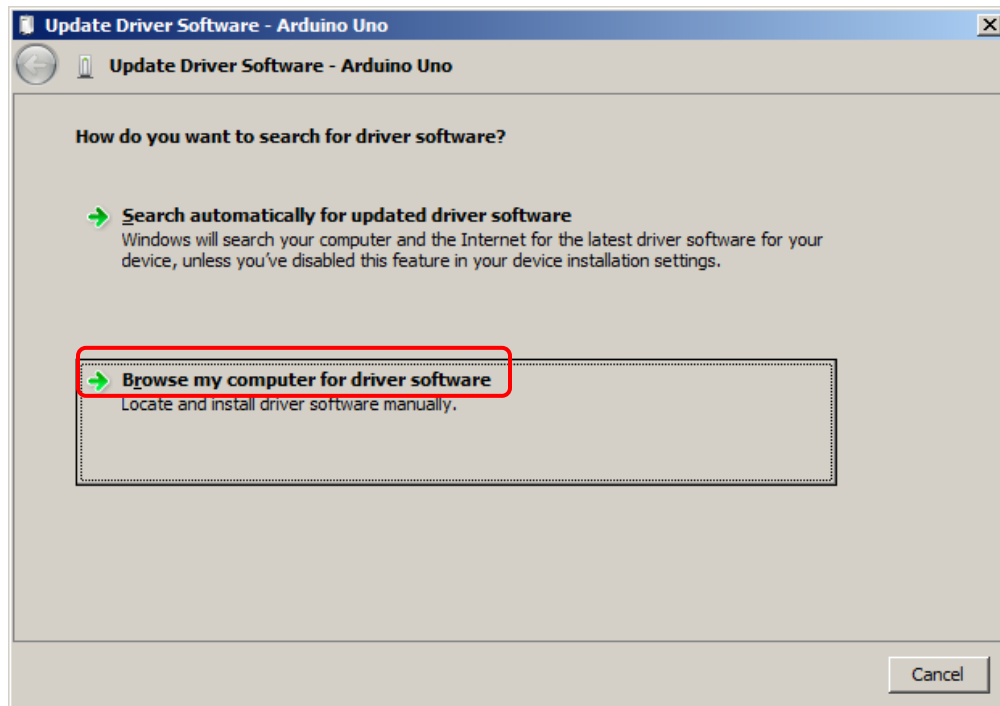
4. Muncul sebuah window baru. Pada tab **General** klik tombol **"Change settings"**.



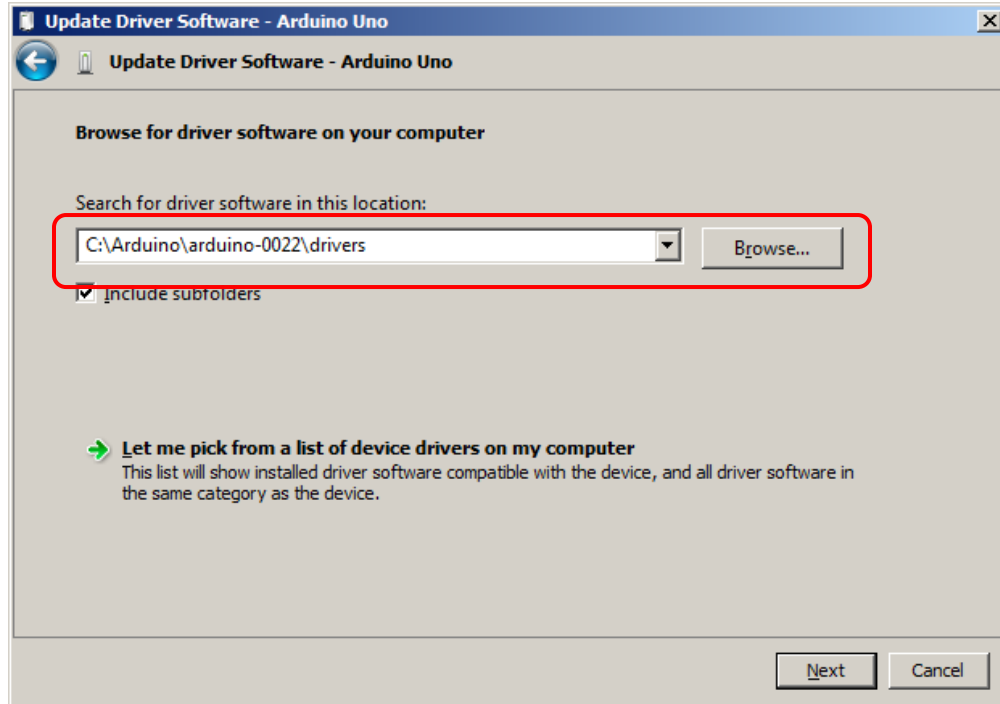
5. Pada tab yang sama, klik tombol **“Update Driver”**.



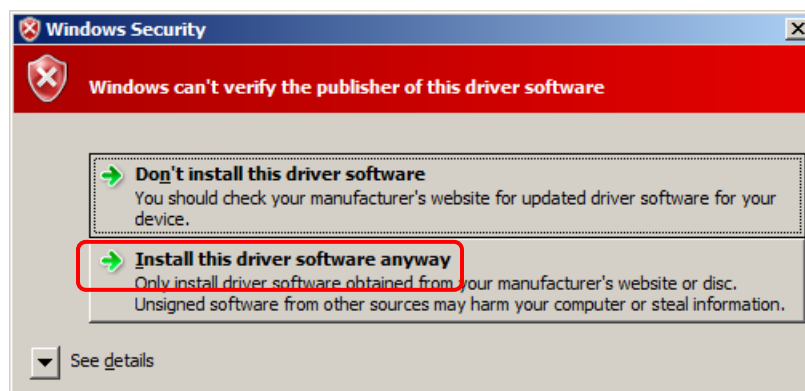
6. Klik **“Browse my computer for driver software”** untuk menentukan sendiri lokasi driver Arduino.



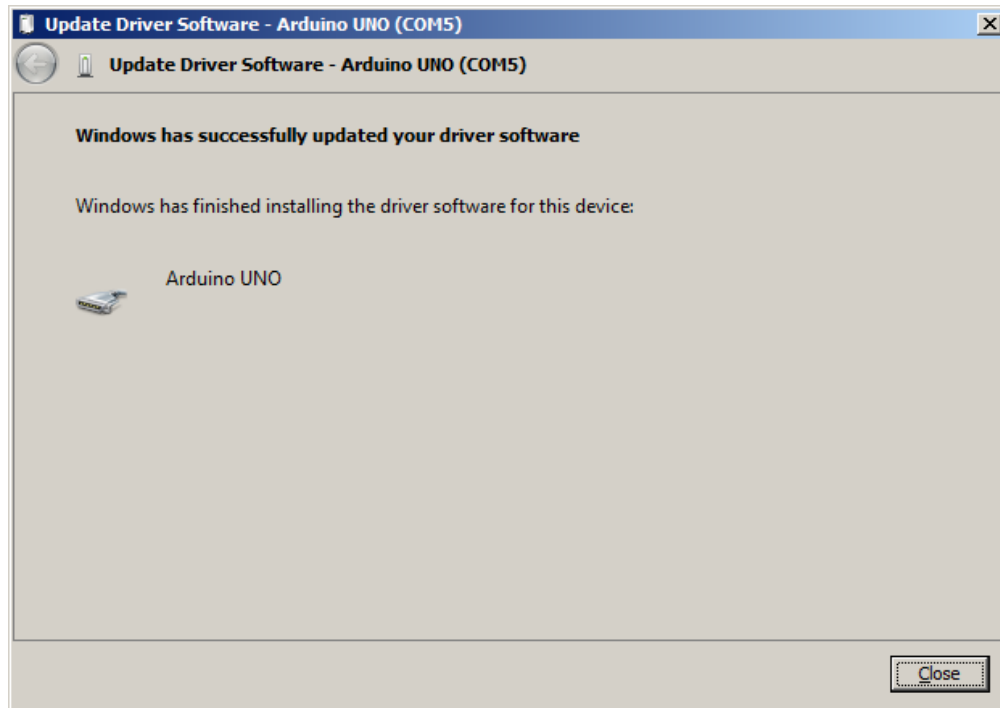
7. Tentukan lokasi dimana software Arduino ditempatkan pada komputer, pada contoh gambar di bawah ini adalah C:\Arduino\arduino-0022. Silakan sesuaikan lokasinya sesuai dengan hasil ekstrak software Arduino pada komputer Anda. Di dalam lokasi tersebut terdapat sebuah direktori bernama **drivers**, arahkan untuk mencari driver di dalam direktori tersebut.



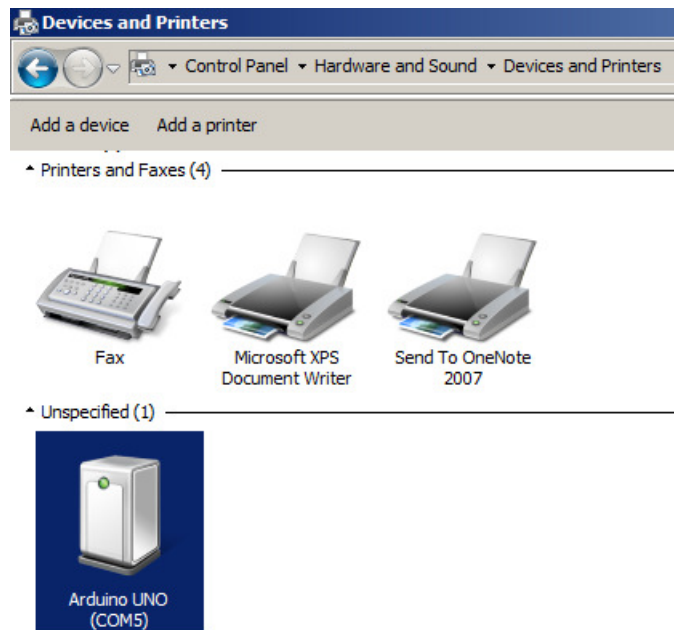
Klik **Next** untuk melanjutkan. Jika muncul sebuah window peringatan seperti di bawah ini, jawab dengan **“Install this driver software anyway”**.



8. Jika driver Arduino selesai diinstal pada komputer maka pada akhir proses akan tampil sebuah pesan berhasil seperti berikut ini.



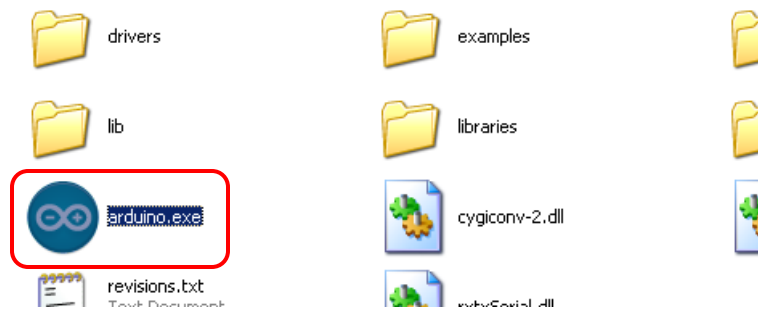
9. Jika Anda kembali ke Control Panel maka tampak gambar segita kuning telah hilang dan Windows telah dapat mengenal papan Arduino.



## 4.4. Menguji Koneksi Komputer dan Papan Arduino

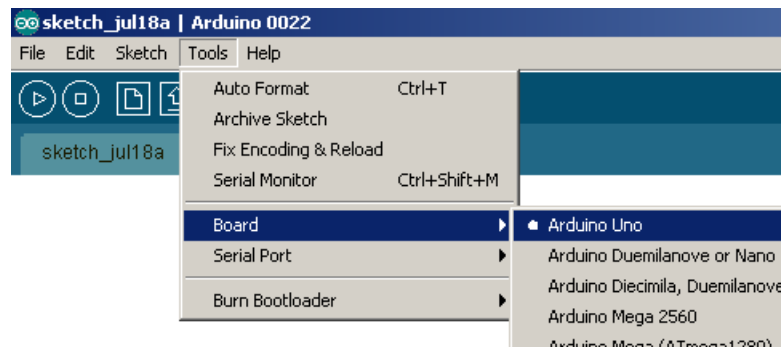
Sekalipun sebuah papan Arduino dapat bekerja dengan mendapat asupan daya dari sebuah komputer, namun hal itu tidak berarti ia dapat berkomunikasi dengan komputer tersebut. Untuk memastikan Arduino telah terpasang dengan benar dan dapat berkomunikasi dengan interaktif maka ia perlu diuji.

1. Jalankan IDE Arduino dengan menjalankan sebuah file bernama **arduino.exe** pada lokasi software Arduino.

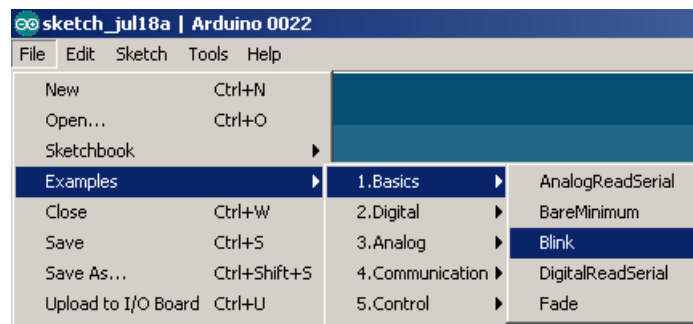


Walaupun tampak seperti program Windows pada umumnya, namun sebetulnya program ini adalah sebuah program Java. Jika Anda menemukan sebuah pesan kesalahan kemungkinan besar pada komputer belum terinstal *Java Runtime Environment (JRE)* atau *Java Development Kit (JDK)*. Untuk mendapatkan salah satu software tersebut, silakan men-download-nya dari situs web <http://www.oracle.com>.

2. Jalankan menu **Tools** → **Board** kemudian pilih tipe papan yang sesuai.



3. Jalankan menu **File** → **Examples** → **1.Basic** → **Blink**. Ini adalah program sederhana yang fungsinya adalah membuat lampu LED menyala berkedip-kedip seperti yang telah dijelaskan sebelumnya.

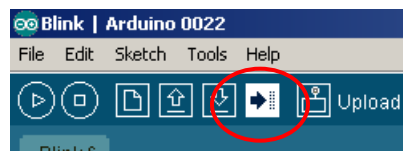


```
/*  
  Blink  
  Turns on an LED on for one second, then off for one second,  
  repeatedly.  
  
  This example code is in the public domain.  
  */  
  
void setup() {  
  // initialize the digital pin as an output.  
  // Pin 13 has an LED connected on most Arduino boards:  
  pinMode(13, OUTPUT);  
}  
  
void loop() {  
  digitalWrite(13, HIGH); // set the LED on  
  delay(1000);           // wait for a second  
  digitalWrite(13, LOW); // set the LED off  
  delay(1000);           // wait for a second  
}
```

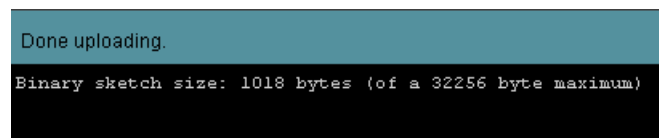
Tidak perlu khawatir jika Anda kurang memahami cara kerja sketch di atas karena kita akan belajar bahasa pemrograman Processing pada pembahasan tersebut nanti. Untuk saat ini cukup perhatikan baris-baris yang ditandai. Bagian itu adalah perintah untuk menunda aliran program selama satu detik (1000 mili detik). Jadi bila lampu LED diperintahkan menyala pada baris sebelumnya, maka dengan perintah `delay()` lampu itu akan bertahan menyala selama satu detik sebelum ia diperintahkan untuk padam pada baris berikutnya.

Silakan mengubah kedua angka 1000 itu menjadi 200 agar interval nyala-padam menjadi lebih pendek.

4. Pada toolbar klik tombol **Upload** untuk memuat sketch tersebut ke dalam papan Arduino.



Jika Anda cukup beruntung maka sketch akan dimuat, ditandai dengan pesan berhasil seperti di bawah ini.

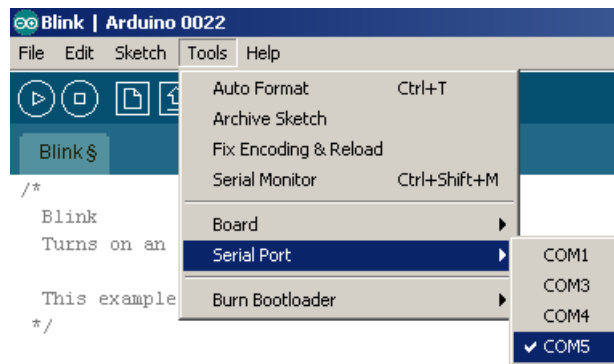


Namun jika kurang beruntung (seperti pada kebanyakan kasus umumnya) maka akan muncul pesan kesalahan seperti berikut.

```
Problem uploading to board. See http://www.arduino.cc/en/Guide/Troubleshooting#upload for suggestions.  
Binary sketch size: 1018 bytes (of a 32256 byte maximum)  
avrdude: stk500_getsync(): not in sync: resp=0x00  
avrdude: stk500_disable(): protocol error, expect=0x14, resp=0x51
```

Walaupun pesan itu tidak cukup jelas menjelaskan apa masalahnya, tapi umumnya karena IDE belum dikonfigur dengan benar sehingga komputer dan papan Arduino tidak dapat berkomunikasi.

Solusinya cukup mudah, yaitu cukup mengganti pilihan serial port melalui menu **Tools** → **Serial Port**. Jika Anda tidak yakin pada port nomor berapa papan Arduino itu terhubung, coba pilih sebuah nomor port lalu jalankan upload seperti langkah sebelumnya. Jika pesan kesalahan masih muncul, ganti nomor port-nya dan lakukan berulang-ulang sampai upload berhasil.



Saat sketch yang sudah dimodifikasi tersebut berhasil dimuat ke dalam papan Arduino maka tampak lampu LED menyala dan padam dengan frekuensi yang lebih cepat. Silakan lakukan eksperimen sendiri misalnya menambah delay dan lihat apa yang terjadi.

Bagian ini menutup pembahasan tentang pengenalan Arduino. Walaupun cukup pendek namun saya berharap artikel ini memberi pengertian yang jelas kepada para pembaca, membuka wawasan dan visi mengenai potensi besar dari platform ini serta membangkitkan rasa antusias untuk memulai perjalanan panjang dan berpetualang bersama Arduino.

**REFERENSI:**

- Arduino, <http://www.arduino.cc>
- Banzi, Massimo. "Gettting Started with Arduino". O'Reilly. 2008
- Physical computing , [http://en.wikipedia.org/wiki/Physical\\_computing](http://en.wikipedia.org/wiki/Physical_computing)

*Penulis adalah lulusan dari Tenik Elektro Universitas Trisakti, Jakarta pada tahun 1995. Menggemari elektronika dan pemrograman komputer dari dahulu hingga sekarang. Beberapa buku tentang bahasa pemrograman dan database telah diterbitkan oleh Elexmedia, sementara artikel-artikel lainnya dipublikasikan secara gratis di [www.tobuku.com](http://www.tobuku.com).*

*Penulis dapat dihubungi melalui email dengan alamat [feri.djuandi \[at\] gmail \[dot\] com](mailto:feri.djuandi@gmail.com)*



Mat 10:8 "...Kamu telah memperolehnya dengan cuma-cuma, karena itu berikanlah pula dengan cuma-cuma."