

আরডউনোতে হাতেখড়ি

রঞ্জল আমীন
মুনেম শাহরিয়ার
সাদিয়া কবির দিনা

করোনা ভাইরাসের বিস্তারের ফলে স্কুল কলেজ বন্ধ হয়ে অনেক শিক্ষার্থীই বাসায় থাকছে।
তাঁদের বাড়িতে থাকার এই সময়টুকুতে সুন্দর কিছু শেখার জন্য
আরডুইনোতে হাতেখড়ি বইয়ের লেখক-গ্রন্থের পক্ষ হতে
বইটির পিডিএফ উন্মুক্ত করা হল।

ଲେଖକଦେର ଉଠିମାହ ଦିତେ ସଂଗ୍ରହ କରତେ ପାରେନ ଆରଡୁଇନୋତେ ହାତେଥିଡି ବହିୟେର ହାର୍ଡ କପି। ବହିମେଳା ୨୦୨୧ ଏ ବହିଟିର ଏକଟି ନତୁନ ସଂକ୍ଷରଣ ଆସଛେ। ନତୁନ ସଂକ୍ଷରଣେ ଥାକବେ ଆରଓ ନତୁନ ନତୁନ କିଛୁ। ସାଥେ ପୁରୋନୋ ସଂକ୍ଷରଣେର କିଛୁ ବାନାନ ଓ ପ୍ରିନ୍ଟଗତ ଭୁଲକ୍ରତ୍ତିଓ ଶୁଧରେ ଦେଯା ଥାକବେ।

ଆରଡୁଇନୋତେ ହାତେଥିଡି ପଡ଼ା ଶେଷ ହଲେ, ଆରଓ ଏଡଭାନ୍ସ କିଛୁ ଶେଖାର ଜନ୍ୟ, ଆରଓ ମଜାର କିଛୁ ବାନାନୋର ଜନ୍ୟ ପଡ଼ତେ ପାର ହୁୟେ ଓଠୋ ଏକଜନ ପ୍ରବଲେମ ସଲଭାର ବହିଟି ଯାତେ ତୁମି ଶିଖବେ ପ୍ରବଲେମ ସଲଭିଂ, ଇନ୍ଟାରନେଟ ଅବ ଥିଂସ (ଆଇଓଟି), ରୋବଟିକ୍ ଏବଂ ଆରଡୁଇନୋ !!!

ଲେଖକଦେର ସବ ବହିୟେର ରକମାରି ଲିଙ୍କଃ
www.rokomari.com/book/author/61130

আরডুইনোতে হাতেখড়ি

আরভুইনোতে হাতেখড়ি

রঞ্জল আমীন
মুনেম শাহরিয়ার
সাদিয়া কবির দিনা

আদৰ্শ



প্রকাশক : আদর্শ

প্রধান বিক্রয়কেন্দ্র : আদর্শ বই

কনকর্ড অ্যাম্পোরিয়াম, কঁটাবন, ঢাকা ১২০৫

+০২-৯৬১২৮৭৭, ০১৭১০৭৭৯০৫০

Email: info@adarshapublications.com

Website: www.adarshapublications.com

আরডুইনোতে হাতেখড়ি

প্রথম প্রকাশ : ২৯ মাঘ ১৪২৪, ১০ ফেব্রুয়ারি ২০১৮

© লেখক

সর্বস্বত্ত্ব সংরক্ষিত; লেখক ও প্রকাশকের লিখিত অনুমতি ব্যতীত
যেকোনো মাধ্যমে বইটি আংশিক বা সম্পূর্ণ প্রকাশ একেবারেই নিষিদ্ধ

প্রচ্ছদ : আদর্শ ডিজাইন

মুদ্রণ ব্যবস্থাপনা : আদর্শ প্রিন্টার্স

প্রধান পরিবেশক : আদর্শ বই

রকমারিতে আদর্শৰ বই : www.rokomari.com/adarsha

মূল্য : বাংলাদেশে ৩০০ টাকা

Arduinote Hatekhari (Published in Bengali)

by Ruhul Amin, Munem Shahriar, Sadia Kabir Dina

Published by Adarsha

Concord Emporium, Kataban, Dhaka 1205

ISBN: 978-984-9266-44-0

উৎসর্গ

আমাদের মা-বাবাদের

সূচি

ভূমিকা	৯
শুরুর আগে...	১০
কেনাকিনি	১২
আরডুইনো তুমি কে?	১৪
কম্পিউটার ও ৱণ্টির গল্ল	২৩
একটুখানি কোড লিখি	৩১
বাতি জ্বালানো দিয়ে শুরু	৪১
অনেকগুলো বাতি জ্বালাই	৫৩
বাজারের কানা	৫৯
রোবটের সুইচ	৬৭
চলো বানাই চোর ধরার অ্যালার্ম	৮২
টিভির ভলিউম ঘোরানোর সেই দিনগুলি	৯১
ডষ্টের আরডুইনো	৯৭
বাদুড়ের চোখ	১০৩
ডানে-বাঁয়ে ঘুরে সেই পুতুলাটি	১১০
এলসিডি স্ক্রিনে কেমন মজা	১১৫
ঘোরাব এবার একটা চাকা	১২১
আরডুইনোতে পাওয়ার সাপ্লাই	১৩১
এবং তারপর?	১৪১
পরিশিষ্ট	১৪৬
তথ্যসূত্র	১৪৭
নির্ঘন্ট	১৪৯

ভূমিকা

বিশ্বব্যাপী ইন্টারনেট অব থিংস বা আইওটি এখন একটি বড় ও সম্ভাবনাময় প্রযুক্তি। আইওটিকে সহজ ভাষায় বলতে গেলে বলা যায় আমরা যা করি সবকিছুকে ইন্টারনেটের সাথে কানেক্ট করে একধরনের স্বয়ংক্রিয় ব্যবস্থার সৃষ্টি করা। অদূর ভবিষ্যতে চতুর্থ শিল্প বিপ্লবের উত্থান এই আইওটির জন্য হতে পারে বলেও অনেকে ধারণা করছেন। উন্নত বিশ্বের সাথে তাল মিলিয়ে চলতে আমাদেরও এক পা করে এগোনো উচিত। আর সে ক্ষেত্রে শুরু করা প্রয়োজন একেবারে গোড়া থেকে।

আইওটির প্রথম পাঠ হওয়া উচিত আরডুইনো থেকে এবং সে পাঠ দিব্য স্কুলেই সম্ভব। আর এ কারণে সহজ ভাষায় আরডুইনো উপস্থাপনের তাগিদ থেকে আমাদের এ বই।

আমরা কেউই লেখালেখিতে নিয়মিত নই, কিন্তু একটা সুন্দর বই উপহার দিতে আমাদের প্রচেষ্টার কোনো ঘাটতি ছিল না।

বই সম্পর্কে যেকোনো মন্তব্য, সমালোচনা বা পরামর্শ জানানোর অনুরোধ রইল।

রঞ্জল আমিন

মুনেম শাহরিয়ার

সাদিয়া কবির দিনা

শুরুর আগে...

প্লেটো বলে গেছেন, যেকোনো কাজের সবচেয়ে গুরুত্বপূর্ণ অংশ হলো শুরুটা। তাই তুমি যত দিন শুরু করবে না, তত দিন জানবে না তোমার দ্বারা কত কী করা সম্ভব ছিল!

আমরা আরডুইনো দিয়ে আমাদের কাজ শুরু করব। কিন্তু কেন আরডুইনো? কারণ, আরডুইনো সহজ। শেখা, কোড করা, নতুন প্রজেক্ট বানানো সবই খুব সহজ। আরডুইনো দিয়ে তুমি প্রোগ্রামিং চর্চাও করতে পারবে মজার ছলে। সবচেয়ে বড় কথা হলো, তুমি যতটা চমৎকার কল্পনা করতে পারবে, ততটাই উত্তাবনী প্রজেক্ট বানিয়ে ফেলতে পারবে আরডুইনো দিয়ে।

তুমি কি তোমার স্কুলের বিজ্ঞানমেলায় সেরা একটি প্রজেক্ট বানাতে চাও? তুমি কি একটা ছোট গাড়ি বানাতে চাও, যা নিজে থেকেই পথ খুঁজে চলবে? নাকি এমন একটা রোবট যে তোমার ফোন-নিয়ন্ত্রিত?

তুমি কি জানো অটোমেশন কী? আমাদের জীবনব্যবস্থার কোনো একটা বিষয়কে অটোমেটিক বা কম্পিউটারাইজড করে ফেলা হলো অটোমেশন। যেমন ধরো, তোমার বাসায় পানির মেশিন আছে। ট্যাংকি খালি হলে পানির মেশিন চালু করতে হয়। আবার পানি ভর্তি হলে মেশিনের সুইচ হাতে বন্ধ করতে হয়। এ কাজটা যদি কম্পিউটারকে ঘটনা অনুযায়ী সিদ্ধান্ত নেওয়ার কাজ শিখিয়ে বলে দেওয়া হতো এবং সে নিজে নিজেই করতে পারত, তা হতো অটোমেশন। তোমার পুরো বাড়িকে অটোমেশন করে স্মার্ট হোম বানিয়ে ফেলতে চাও?

আজকে তুমি যদি এই বইটি হাতে নিয়ে আরডুইনো শেখা শুরু করো,
তবে আজকেই শুরু হবে ইলেক্ট্রনিকস— রোবটিকস— ইন্টারনেট অব
থিংস— অটোমেশনের জগতে তোমার পথচলা...

যখন তোমার করা কোড চোখের সামনে একটা বাতি জ্বালাবে, যখন
তোমার লেখা কোড অনুযায়ী একটা রোবটের চাকা ঘুরবে, চলবে ডানে-
বাঁয়ে... সেই মুহূর্তের অনূভূতিটুকু অসামান্য।

এবার তুমি নিজেকে প্রশ্ন করো, শুরুর ‘আর কত দেরি, পাঞ্জেরী?’

বইটি কাদের জন্য?

বলা হয় যে বই পড়ার কোনো বয়স লাগে না। যেকোনো বয়সে নতুন
কিছু শিখতে কোনো একটা বই চাইলেই মানুষ পড়তে পারে। আমাদের
এই বইটাও সবার জন্য।

তবে হ্যাঁ! আমরা বইটি এমনভাবে লিখেছি যেন নিম্নমাধ্যমিকের
যেকোনো শিক্ষার্থী বইটি পড়ে সর্বোচ্চ আনন্দ পায়।

আমাদের বিশ্বাস, এই বয়সীদের মাথায় প্রোগ্রামিংয়ের ভূত আর
আরডুইনোর পোকা ঢুকিয়ে দিলে আমরা শিগগিরই একটা দারুণ বিন্দুর
ঘটিয়ে ফেলতে পারব।

এক বালকে পুরো বই

এই বইয়ে আমরা চেষ্টা করেছি তোমাকে আরডুইনো আর বেশ কিছু
প্রয়োজনীয় সেন্সর সম্পর্কে প্রাথমিক ধারণা দিতে। এই প্রাথমিক ধারণা
নিয়ে তুমি ইন্টারনেট ঘেঁটে আরো পড়াশোনা করতে পারবে এবং মজার
সব প্রজেক্ট বানাতে পারবে।

মনে রাখবে, তুমি এই বইয়ের প্রতিটি অধ্যায় একটার পর একটা
ক্রমানুসারে পড়বে। শূন্য থেকে শেষ পর্যন্ত একটা সিড়ি বেঁয়ে ওঠার
মতো। কোনো অধ্যায় বাদ দিয়ে চলে গেলে পরের অধ্যায়ে বুঝতে কিন্তু
বেশ সমস্যা হবে!

শূন্যের আগে
কেনাকিনি

রোবটিকসের কাজ করার জন্য টুকটাক বেশ কিছু যন্ত্রপাতি কিনে ফেলতে হবে। যন্ত্রপাতিগুলোর দাম খুব বেশি নয়। তোমার এলাকার ইলেকট্রনিকস পার্টস বিক্রেতার কাছে খুঁজলেই পেয়ে যাবে জিনিসগুলো। অথবা অনলাইন শপ থেকেও কিনতে পার।

ক্রম	প্রয়োজনীয় যন্ত্রপাতি	পরিমাণ	সম্ভাব্য মূল্য (টাকা)	কেনা হলে টিক দিয়ো	যেকোনো মন্তব্য লেখো
১	Arduino UNO	১	৪০০-৬০০		
২	Bread Board	১	১০০-১৫০		
৩	Jumper Wire / Connecting Wire (male to male, male to female)	২০-২৫	৩ (প্রতিটি)		
৪	SONAR Sensor	১	১৮০		
৫	PIR Sensor	১	১২০		
৬	এলাইডি (ডিম ভিন্ন ৩ রঙের)	১০	১ (প্রতিটি)		
৭	Buzzer	১	১০-১৫		
৮	Hot glue gun + Glue Stick	১	২০০-৩০০		

৯	GEAR MOTOR DC 5V	১	৮০-১০০		
১০	LCD Display (16*2) + IC Rail (female)	১	২০০		LCD-তে IC rail- টা পাঢ়ার যেকেনো ইলেক্ট্রনিক মেকার আংকেলের কাছে তাতাল দিয়ে ঝালাই করে নিতে হবে
১১	SERVO motor (Micro)	১	২০০-২৫০		
১২	Resistance (220 Ohms, 10K Ohms)	১০	০.২৫(প্রতি)		
১৩	Transistor	২	৩ (প্রতি)		
১৪	Potentiometer (10K)	২	৬-১০		
১৫	চাকা		১০০		অ্যাডপ্টার কেনা গুরুর দিকে অপশনাল
১৬	৯ ভোল্ট ব্যাটারি (ক্লিপসহ)	২	৩৫ (প্রতি)		

শূন্য

আরডুইনো তুমি কে?

আজ থেকে প্রায় বেশ কয়েক বছর আগে ২০০৩ সালে ইতালির কয়েকজন প্রকৌশলী (Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino David Mellis) ভাবলেন, আমাদের জীবনের ছোট-বড় সমস্যাগুলো যদি কম্পিউটারের মতো যন্ত্র দিয়ে সাধারণ যেকোনো মানুষ নিজেরাই সমাধান করতে পারে? তারা এমন কিছু একটা বানাতে চাইলেন যা দিয়ে একদম ছোট বাচ্চা থেকে বড় মানুষ, কোনো রকম ইঞ্জিনিয়ারিং ডিগ্রি ছাড়াই এসব যন্ত্র বানিয়ে ফেলতে পারে। এ জন্য তারা বানালেন ওপেন সোর্স এই প্ল্যাটফর্ম আরডুইনো। সেই থেকে আমাদের এই বইয়ের গঞ্জের শুরু।

তাহলে আরডুইনো হচ্ছে এক ইতালিয়ান কোম্পানির বানানো একটা ডেভেলপমেন্ট বোর্ড (Development Board) আর এই ডেভেলপমেন্ট বোর্ডের প্রাণে আছে একটি মাইক্রোকন্ট্রোলার (Microcontroller)

এখন কথা হলো, ডেভেলপমেন্ট বোর্ড কী? আর মাইক্রোকন্ট্রোলারই-বা কী জিনিস?

সহজ করে বললে, ডেভেলপমেন্ট বলতে বোঝায় কোনো কিছু বানানো। যেমন ধরো কাদামাটি টিপে টিপে পুতুল বানানোও একধরনের ডেভেলপমেন্ট। কিন্তু এই বইয়ে আমরা ডেভেলপমেন্ট বলতে বোঝাচ্ছি ইলেক্ট্রনিকস কোনো যন্ত্র বানানো। যেমন ধরো ছোট একটা গাড়ি রোবটের প্রজেক্ট বানানোটা হলো এখানে ডেভেলপমেন্ট। এখন এই

আরডুইনো তুমি কে

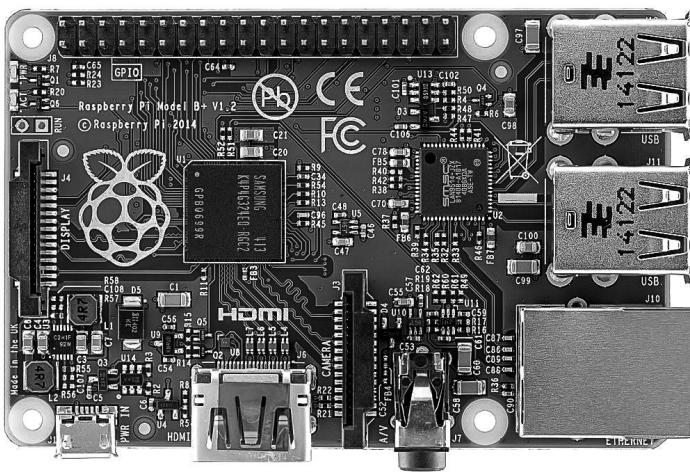
ডেভেলপমেন্টের কাজে আমরা যে যন্ত্রটি মূলে ব্যবহার করি, যার ওপর ভিত্তি করে পুরো প্রজেক্টটা চলে, তাকেই বলছি ডেভেলপমেন্ট বোর্ড। এখানে আমাদের আরডুইনো হলো ডেভেলপমেন্ট বোর্ড।

একটা প্রজেক্টের প্রাণে আছে ডেভেলপমেন্ট বোর্ড। আবার এই ডেভেলপমেন্ট বোর্ডের প্রাণে যে যন্ত্রটি আছে, সেটা হলো মাইক্রোকন্ট্রোলার।

তুমি কী ধরনের প্রজেক্ট বানাতে চাও তার ওপর নির্ভর করে বিভিন্ন ধরনের ডেভেলপমেন্ট বোর্ড পাওয়া যায়। কয়েকটি নাম যদি বলি, Raspberry Pi, Banana Pi, Intel Edison, Udoo Neo, Adafruit Flora আরো কত! তুমি এই নামগুলো ধরে একটু গুগল সার্চ করলেই আরও অনেক জানতে পারবে।

উদাহরণস্বরূপ তোমাকে একটি ডেভেলপমেন্ট বোর্ডের কথা বলি—
রাস্পবেরি পাই (Raspberry Pi)

রাস্পবেরি পাইকে বলা হয়, একটি সিঙ্গেল বোর্ড কম্পিউটার। মানে একটি বোর্ডের মাঝেই একটি মাঝারি মানের কম্পিউটারের সব ক্ষমতা দেওয়া হয়েছে। ইন্টারনেট অব থিংসের (আইওটি) কাজ করার সময় রাস্পবেরি পাই বেশ কাজে দেয়। রাস্পবেরি পাই তোমরা আরডুইনো শেখার পরে শিখবে।



চিত্র ০.১ : রাস্পবেরি পাই

এখন প্রশ্ন করতে পারো, ডেভেলপমেন্ট বোর্ড ছাড়া কি প্রজেক্ট বানানো সম্ভব নয়? হ্যাঁ, অবশ্যই সম্ভব। কিন্তু সেটা অনেক বেশি কষ্টসাধ্য ও সময়সাপেক্ষ।

এবার আসি মাইক্রোকন্ট্রোলারে। মাইক্রোকন্ট্রোলার হচ্ছে কম্পিউটারের ছোট ভাই। একটা কম্পিউটার বলতে আমরা বুঝি বড়সড় একটা বাক্স (বা সিস্টেম ইউনিট), এর মধ্যে থাকে প্রসেসর, র্যাম, মাদারবোর্ড, রম, হার্ডিডিক্স, পাওয়ার সাপ্লাই লাইন। এবং এর বাইরে এর সাথে থাকে মনিটর, কিবোর্ড, মাউস ইত্যাদি। সবগুলো উপাদান একসাথে মিলে তৈরি করে একটা পূর্ণাঙ্গ কম্পিউটার।

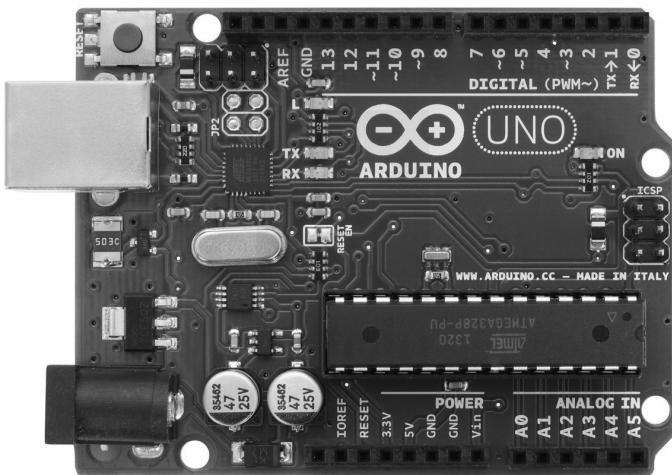
কিন্তু শুরুর দিকে আমাদের রোবটিক্স বা আইওটির ছোট ছোট প্রজেক্ট করতে তো এত বড় বা শক্তিশালী কম্পিউটারের প্রয়োজন নেই। এ জন্য একটা কম্পিউটারের মতোই প্রসেসর, র্যাম, মেমরি সব মিলে বিজ্ঞানীরা বানালেন মাইক্রোকন্ট্রোলার। পার্থক্য হলো কম্পিউটারের চেয়ে এর সাইজ অনেক ছোট। ছোট একটা আইসি বা চিপ (Chip) মাত্র। মাইক্রোকন্ট্রোলার আবার দামেও সন্তা। কম্পিউটারের দাম ৩০-৩৫ হাজার টাকা হলে মাইক্রোকন্ট্রোলার তার প্রায় চার হাজার ভাগের এক

ভাগ। মাত্র ৮০ টাকা। তাই দৈনন্দিন ছোট ছোট অটোমেশন কাজে একে ব্যবহার করা হয়। তোমার বাসার মাইক্রোওয়েভ ওভেন, ওয়াশিং মেশিন, ডিজিটাল ঘড়ি, ক্যালকুলেটর, বিল্ডিংয়ের লিফট সবকিছুর ভেতরেই মাইক্রোকন্ট্রোলার আছে।

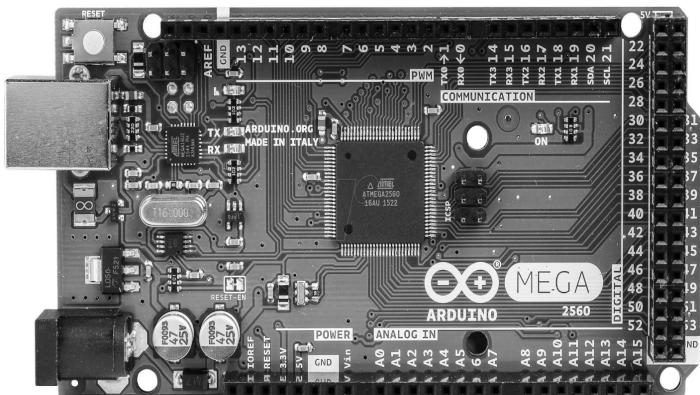
তুমি কখনো রিমোট কন্ট্রোল খেলনা গাঢ়ি খুলে দেখেছ? ওর ভেতরেও কিন্তু কালো পাতের মতো সার্কিট বোর্ডে মাইক্রোকন্ট্রোলার আছে।

সাধারণত এই মাইক্রোকন্ট্রোলার দিয়ে যন্ত্র বানাতে বালাই করে অনেক কিছু লাগাতে হয়। আমরা যেন মাইক্রোকন্ট্রোলারটিকে খুব সহজে আমাদের কাজে ব্যবহার করতে পারি, সে জন্যই সৃষ্টি বিভিন্ন ডেভেলপমেন্ট বোর্ড। সরাসরি মাইক্রোকন্ট্রোলার ব্যবহার করেও প্রজেক্ট বানানো যায়, তবে সেগুলো একটু অ্যাডভান্স লেভেলের। শুরুর পর্যায়ে আমরা আরডুইনো ব্যবহার করব।

নিচের ছবিতে একটা আরডুইনোর ছবি দেখা যাচ্ছে। আরডুইনোর অনেকগুলো মডেল আছে। সবচেয়ে জনপ্রিয় হচ্ছে আরডুইনো উনো (Arduino UNO) একটু বড় প্রজেক্ট করতে চাইলে কিনে নিতে পারো আরডুইনো মেগা (Arduino Mega)। এ ছাড়া সাইজে খুব ছোট আরডুইনো হিসেবে আছে ন্যানো (Arduino Nano)

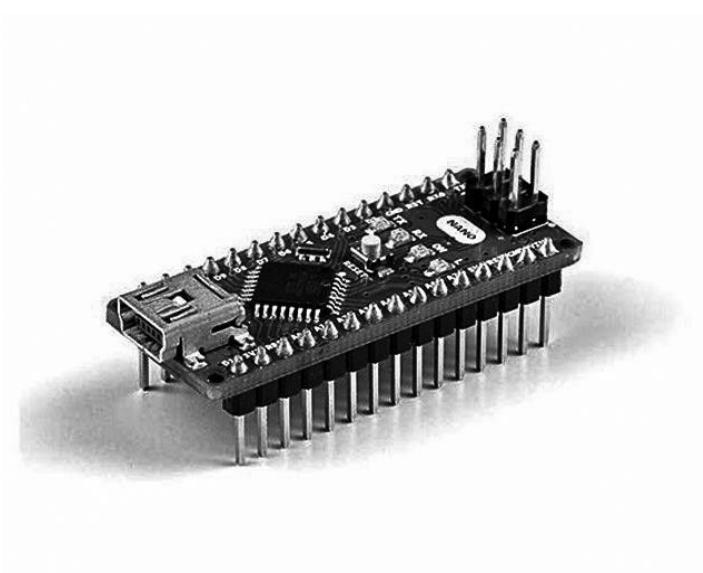


চিত্র ০.৩ : আরডুইনো উনো



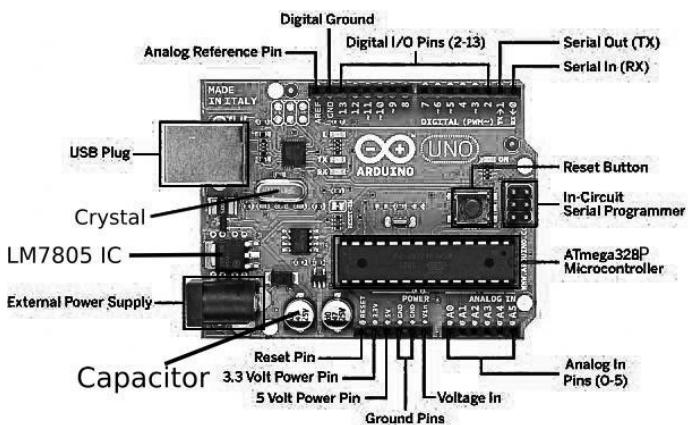
চিত্র ০.৪ : আরডুইনো মেগা

আরডুইনো তৃতীয় ক্ষেত্র



চিত্র ০.৫ : আরডুইনো ন্যানো

চলো আরডুইনোকে আমরা আরো কাছ থেকে দেখি



চিত্র ০.৬ : কাছ থেকে আরডুইনো

এখানে একটা আরডুইনোর ছবি দেখা যাচ্ছে। বাঁ দিকে আছে ইউএসবি প্লাগ। এখান থেকে একটা কেবল লাগিয়ে কম্পিউটার থেকে কোড আপলোড দেওয়া হয়। এ ছাড়া তুমি ইউএসবির ৫ ভোল্ট পরিমাণ কারেন্ট দিয়ে আরডুইনো চালাতেও পারবে।

বাঁ দিকে পাশেই আছে এক্সটারনাল পাওয়ার সাপ্লাই পোর্ট। এটা দিয়ে মূলত আরডুইনোতে পাওয়ার দেওয়া হয়। এখানে ৭-১২ ভোল্ট পর্যন্ত পাওয়ার দেওয়া যায়। এ ছাড়া তুমি ভোল্টেজ ইন (Vin) পিনের মাধ্যমেও আরডুইনোতে পাওয়ার সাপ্লাই করতে পারো।

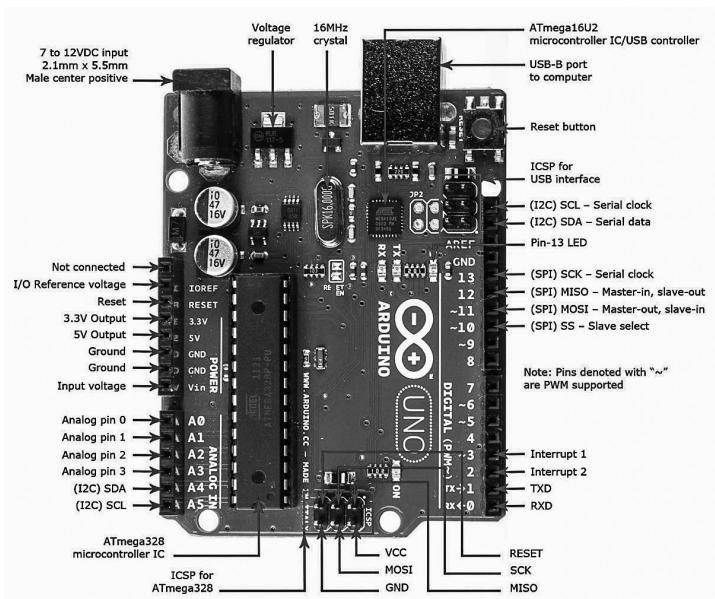
ইউএসবি প্লাগের পাশেই চিকন সাদা শার্টের বেতামের মতো একটা জিনিস দেখা যাচ্ছে, এটা হচ্ছে ক্রিস্টাল। এর কাজ আরডুইনোকে চালু করা। এটা মূলত ক্লক পালস (একধরনের সিগন্যাল বা তরঙ্গ) সাপ্লাই দিয়ে থাকে। এটার সম্পর্কে আমরা পরে জানব।

পাওয়ার পোর্টের পাশে গোল ধরনের দুইটা জিনিস দেখতে পাচ্ছ, এগুলো হচ্ছে ক্যাপাসিটর বা ধারক।

ইউএসবি প্লাগের ঠিক একদম পাশে ছোট কালো চার কোনা একটা জিনিস আছে। এটাও একটা মাইক্রোকন্ট্রোলার। এর কাজ হচ্ছে তোমার লেখা প্রোগ্রামকে এই যে বড় মাইক্রোকন্ট্রোলার রয়েছে, ‘ATMEGA 328P’-এর বোঝার উপযোগী করে তোলা। কারণ কম্পিউটার তোমার-আমার মতো বাংলা বা ইংরেজি বোঝে না, সে শুধু বোঝে শূন্য বা এক-এর উপস্থিতি।

ওপরে যে পিনগুলো দেখা যাচ্ছে, এগুলো হচ্ছে ডিজিটাল ইনপুট-আউটপুট পিন।

০ থেকে ১৩ পর্যন্ত এখানে মোট ১৪টি ডিজিটাল পিন আছে। এর মধ্যে শূন্য হচ্ছে ‘RX’ পিন আর ১ হলো ‘TX’ পিন। এই দুটো পিন দিয়ে অন্য কোনো ডিভাইসের সাথে ডেটা ট্রান্সফার করা যায়।



চিত্র ০.৭ : আরডুইনো উনো বিস্তারিত পিন পরিচিতি

এ ছাড়া আছে কয়েকটা গ্রাউন্ড পিন, এটা মূলত একটা ইলেকট্রিকাল সার্কিটের ঝগাঝুক প্রান্ত। তাহলে ধনাত্মক প্রান্ত কোনটা হতে পারে? হ্যাঁ ঠিক ধরেছ, **৫ ভোল্ট** লেখা যে পিনটা আছে, সেটা ধনাত্মক প্রান্ত। এ ছাড়া আছে **৩.৩ ভোল্ট** পিন, এটা দিয়ে তুমি একটু কম পরিমাণে ভোল্টেজ পাঠাতে পারো। মনে রাখবে, এক্সটারনাল পাওয়ার ইনপুট আর ভোল্টেজ-ইন (V_{in}) এই পিন দুটি হলো আরডুইনোতে পাওয়ার ইনপুট দেওয়ার পিন। কিন্তু **৫ ভোল্ট** বা **৩.৩ ভোল্ট** লেখা পিন দুটি হচ্ছে আরডুইনো থেকে পাওয়ার আউটপুট হিসেবে নিয়ে কোনো এলইডি বা সেপ্সেরে দেওয়ার পিন।

এ ছাড়া আছে রিসেট পিন, যেটা মূলত রিসেট বা পুনরায় শুরুর কাজে ব্যবহার করা যায়। আছে রিসেট বাটন, এটার কাজ আরডুইনো চলা অবস্থায় এর মধ্যে যে প্রোগ্রামটা আছে, সেটা শুরু থেকে আবার চলার জন্য।

এ ছাড়া অ্যানালগ কাজ করতে চাইলে তোমার জন্য আছে ‘A0-A5’
পর্যন্ত ছয়টি

অ্যানালগ পিন। পরে কোনো এক অধ্যায়ে যখন আমরা ডিজিটাল-
অ্যানালগের পার্থক্য বুবাব, তখন এর কাজ বুবো যাব।

আমরা জানি, আরডুইনোর প্রাণ মাইক্রোকন্ট্রোলার। আরডুইনোতে
উন্নতে ব্যবহার করা হয় ‘ATMEGA 328P’ মাইক্রোকন্ট্রোলার।
আরডুইনো মেগাতে আছে ‘ATMEGA 2560’ মাইক্রোকন্ট্রোলার।

আরডুইনোর আরো একটা অসাধারণ ব্যাপার হচ্ছে আরডুইনোর
কোডগুলো হচ্ছে ওপেন সোর্স(**Open Source**)। ওপেন সোর্স সফটওয়ার
বলতে আসলে বোবায় কোনো সফটওয়ারের সোর্স কোডগুলো কপিরাইট
আইনে পাবলিক লাইসেন্সে করে দেওয়া আছে। যে কেউ চাইলে এটা
প্রয়োজনমতো নিজের কাজে ব্যবহার করতে পারবে। মানে সহজ বাংলায়
এ কোডের সাথে তুমি যা ইচ্ছা করতে পারো। এগুলো ইন্টারনেটেই
পাওয়া যায়, তুমি চাইলে এগুলোকে নিজের মতো করে বানিয়ে আবার
অনলাইনে দিয়ে দিতে পারবে— তোমাকে কেউ কিছু বলবে না। এটাই
হচ্ছে ওপেন সোর্স-এর মজা!

আরডুইনো তুমি কে

এক কম্পিউটার ও রঞ্চির গল্প

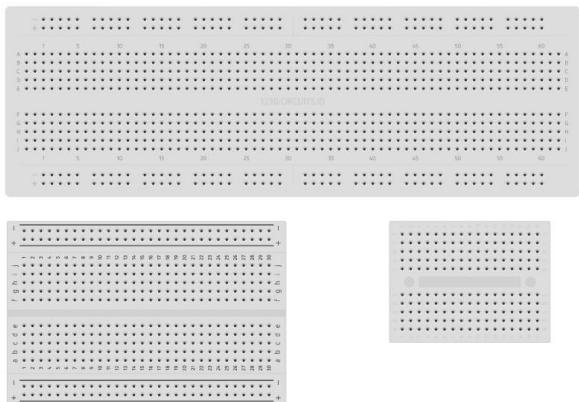
নাম শুনে মনে হচ্ছে রঞ্চির এখানে আবার কী কাজ?

আমরা এমন একটা বোর্ড নিয়ে কাজ করব যার নাম ব্রেড বোর্ড বা প্রজেক্ট বোর্ড। তোমরা ইলেক্ট্রনিক যন্ত্রপাতি দেখেছ, অনেক যন্ত্রপাতি একসাথে ঝালাই করে লাগানো থাকে। তোমরা যেহেতু ছোট আর ঝালাই করা ঝামেলার কাজ, এ জন্য সব কাজ আমরা খুব সহজে ব্রেড বোর্ডে করে ফেলব।

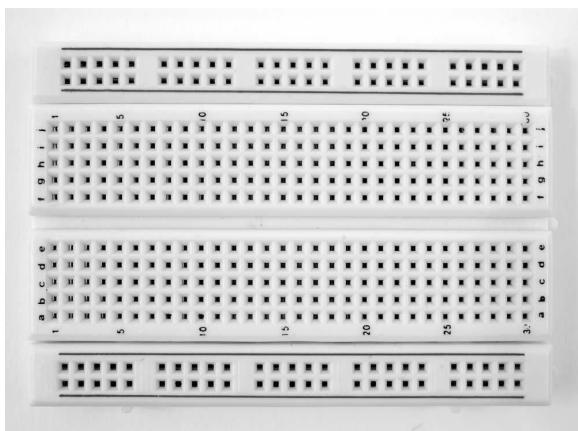
এই ব্রেড বোর্ড দেখতে পাউরঞ্চির মতো। আর এতে অনেক ফুটো আছে। আর এই ফুটোগুলোর ভেতরে তার দিয়ে কানেকশন দেওয়া আছে। এখানে খুব সহজে আমরা

কানেকশন বানিয়ে কোনো সার্কিটকে বসিয়ে দিতে পারি। ব্রেড বোর্ড বিভিন্ন রকমের হয়ে থাকে। ছোট, বড়, লম্বা বিভিন্ন সাইজ। কাজের সুবিধা অনুযায়ী আমরা এসব ব্যবহার করে থাকি।

নিচে একটা ব্রেড বোর্ডের ছবি দেওয়া আছে



চিত্র ১.১: ব্রেড বোর্ড



চিত্র ১.২: কাছে থেকে ব্রেড বোর্ড

এই ছবিতে হোট-বড় বিভিন্ন সাইজের ব্রেড বোর্ড দেখতে পাচ্ছ।
সবগুলোতে কিছু মিল আছে। ব্রেড বোর্ডে ওপরের আর নিচে দুটো
করে লাইন আছে। এখন এই লাইনগুলোর পাশে দাগও দেওয়া আছে।
এখানে লাইন যত দূর, দাগ তত দূর। মানে দাগের সমান এলাকায়
একই কানেকশন দেওয়া আছে।

আর ব্রেড বোর্ডগুলোর মাঝের অংশে ৫টা করে লম্বা লাইন আছে। এগুলো আবার দেখো ১থেকে ৩০ বা ৬০ পর্যন্ত মার্ক করা আছে। এই ১থেকে ৩০ বা ৬০ নিজেরা একেকটা লাইন, মানে এদের মাঝে ভিন্ন ভিন্ন কানেকশন আছে। অর্থাৎ তুমি ১ নম্বর লাইনের **a** থেকে **e** পর্যন্ত কোনো পিনে তার দিয়ে কানেকশন দাও তাহলে কানেকশন হবে একই।

একটু কি গুলিয়ে যাচ্ছে বিষয়টা?

তুমি এখন তোমার কেনা ব্রেড বোর্ডটি হাতে নাও আর বইয়ের সাথে মিলিয়ে পড়ো।

সহজ করে বলি, ব্রেড বোর্ডের ১ নং সারির কথা বিবেচনা করো। ১ নম্বর সারিতে যে **a b c d e** পাঁচটা ফুটো আছে, ওরা আসলে ভেতরে-ভেতরে সবাই কানেকটেড। মানে **a** ফুটোতে একটা ভোল্টেজ দিয়ে যদি তুমি **d** ফুটোতে বাতি লাগাও, তবুও জ্বলবে।

আবার এই যে ১, ২, ৩ ... নম্বর করা সারিগুলো। এরা কিন্তু আলাদা আলাদা। মানে ২ আলাদা, ৩ আলাদা, ৪, ৫, ৬ সব আলাদা। কিন্তু ৫-এর **a b c d e**-এরা আবার একই কানেকশন। ৬-এর **a b c d e** একই কানেকশন। একই ভাবে প্রতি সারির **f g h i j** এই পাঁচ ছিন্দে একই লাইন দেওয়া।

তোমার যদি ইচ্ছে করে জিনিসটার পেছনের কাহিনি জানতে, তাহলে একদিন একটা ব্রেড বোর্ডের পেছনের কাগজটা তুলে ফেলে পরীক্ষা করে দেখো।

আমার মনে হয় তুমি এবার ব্রেড বোর্ড বুঝোছ। যদি এখনো একটু সমস্যা থাকে, তবে ইউটিউবে **how to use a breadboard** লিখে সার্চ করে বাটপট কিছু টিউটোরিয়াল দেখে নাও।

তোমার সুবিধার জন্য একটি টিউটোরিয়ালের লিংক দিয়ে দিলাম : youtu.be/6WReFkfrUIk

ব্রেড বোর্ডে কানেকশন দেওয়ার ক্ষেত্রে আমরা তার (**wire**) ব্যবহার করি। এদের একটা সুন্দর নাম আছে কানেকটিং ওয়্যার বা জাম্পার ওয়্যার। এই জাম্পার ওয়্যার তিনি রকমের হয়ে থাকে

- i) মেল টু মেল (**male to male**)
- ii) মেল টু ফিমেল (**male to female**)
- iii) ফিমেল টু ফিমেল (**female to female**)



চিত্র ১.৩ : মেল টু মেল



চিত্র ১.৪ : মেল টু ফিমেল

কম্পিউটার ও রঙ্গিন গল্লা



চিত্র ১.৫ : ফিমেল টু ফিমেল

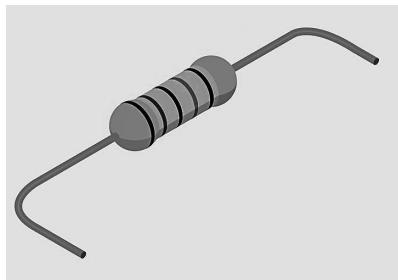
একেকটা হার্ডওয়্যারে প্রয়োজন অনুসারে আমরা এসব ব্যবহার করে থাকি।

এই বইয়ে আরো কয়েকটা বিষয় বারবার আসবে, তার মধ্যে **রোধ** (**resistance**) একটা। রোধ মানে বাধা। কারেন্ট বা ইলেক্ট্রিসিটি তো আসলে ইলেক্ট্রনের প্রবাহ। এখন এই ইলেক্ট্রন কোনো কিছুর মধ্য দিয়ে যেতে চাইলে বস্তুর ধর্ম অনুসারে বাধা পায়, এটাকে বলে রোধ। বিভিন্ন জিনিসের রোধ বিভিন্ন রকম। তামার রোধ অনেক কম, এ জন্য এটা দিয়ে আমরা বৈদ্যুতিক তার তৈরি করি, আবার প্লাস্টিকের রোধ বেশি বলে এটাকে কুপরিবাহী আবরণ দিয়ে রাখা হয়।

(আছা তোমরা কি জানো মানুষের শরীরের বৈদ্যুতিক রোধ কত? এটা একটা কুইজ থাকল। কৌতুহল থাকলে জেনে নিয়ো ইন্টারনেট ঘেঁটে।) প্রতিটা জিনিসের একটা একক থাকে, যেমন আমরা তেল কিনি লিটারে, চাল কিনি কেজিতে, এই লিটার-কেজি এরা হচ্ছে একক। রোধের একক

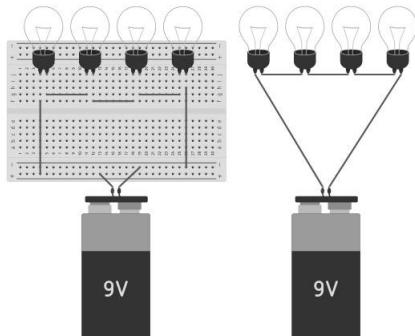
হলো ওহম (Ohm)। এর পরিমাণ বেশি হয়ে হাজার গুণ হলে আমরা বলি কিলোওহম। আমাদের এই বইয়ের প্রজেক্টগুলোতে ২২০ ওহম ও ১১০ কিলোওহম থাকলেই চলে।

আবার ধরো তোমার কোথাও ১০০ ওহম দরকার, সেই কাজ ২২০ ওহম করে দিতে পারবে— খুব একটা পরিবর্তন হয় না কার্যত।



চিত্র ১.৬ : রোধ বা রেজিস্ট্রেস

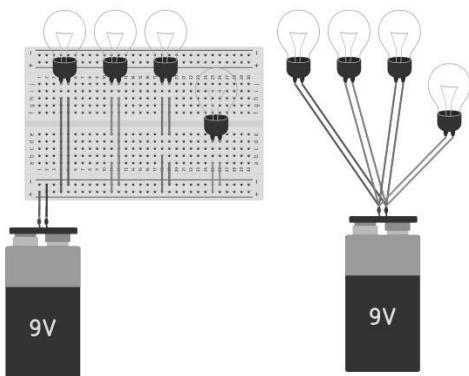
এবার আমরা শিখব কানেকশন দেওয়ার দুটো পদ্ধতি সিরিজ আর প্যারালাল। কথা না বাঢ়িয়ে আগে ছবি দেখে নিই—



চিত্র ১.৭ : সিরিজ সার্কিট

কম্পিউটার ও রঙ্গির গল্ল

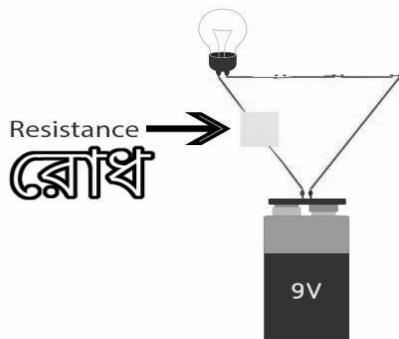
এখানে একটা ব্যাটারির সাথে অনেকগুলো লাইট লাগানো হয়েছে। এই লাইটগুলো তার দিয়ে জোড়া দেওয়া হয়েছে। একটার শুরু আরেকটার শেষ— এভাবে যে কানেকশন দেওয়া হয়, এটাকে বলে **সিরিজ কানেকশন**।
 এবার এই ছবিতে দেখো সবগুলো লাইটকে একদম ব্যাটারির সাথে লাগিয়ে দেওয়া হয়েছে। এটাকে বলে **প্যারালাল কানেকশন**।



চিত্র ১.৮ : প্যারালাল সার্কিট/ সমান্তরাল বর্তন

এবার আমরা মজার একটা কাজ করব! একটা ব্যাটারি দিয়ে একটা লাইট জ্বালাব

এ জন্য দরকার : একটা ২২০ ওহম রোধ, এলইডি বাতি, কিছু তার আর ব্রেড বোর্ড।



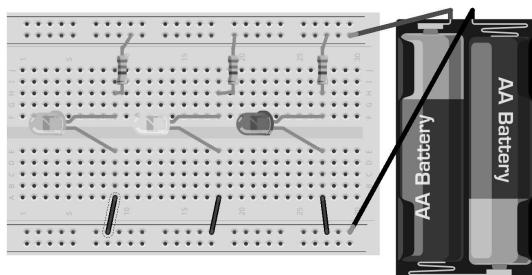
চিত্র ১.৯ : মজার পরীক্ষার বর্তনী

এবার ছবির মতো ব্যাটারির পজেটিভের সাথে এলইডির পজেটিভ আর নেগেটিভের সাথে এলইডির নেগেটিভ যুক্ত করো। যেকোনো এক প্রাণ্তে একটা রোধ দিয়ো।

এই তো! তুমি লাইটটি জ্বালিয়ে ফেলেছ।

চেষ্টা এক :

নিচের ছবি দেখে ব্রেড বোর্ডে কানেকশন দাও



চিত্র ১.১০ : ব্রেড বোর্ড ঠিকমতো শেখা হলো

আশা করি নিজেই করতে পেরেছ। আর না পারলে, গুগল সার্চ তো আছেই!

দুই একটুখানি কোড লিখি

ধরো তুমি তোমার বন্ধুকে ফোনে বললে, কাল বাজার থেকে তোমার জন্য একটা বই কিনতে। বইটা নিয়ে এসে তোমার বাসায় দিয়ে যেতে হবে। এটাও বলে দিলে, যদি সে তোমার বাসায় এসে তোমাকে না পায়, তাহলে বাসায় কলবেল চেপে কাউকে ডেকে বইটা দিয়ে যেতে এবং বাসায় তোমার আশ্মু থাকলে তার কাছে টাকা নিয়ে যেতে।

এককথায় তুমি তোমার বন্ধুকে কিছু নির্দেশনা (**instruction**) দিলে। আমরা যখন কিছু বানাব, তখন সেটাকেও কিছু নির্দেশনা দিতে হয়। এটাই প্রোগ্রাম বা কোড।

তুমি তোমার বন্ধুকে কথাগুলো জানিয়েছিলে ফোনে! আমরা এই কথাগুলো লিখব ইন্টিগ্রেটেড ডেভেলপমেন্ট এনভায়রনমেন্ট (**Integrated Development Environment — আইডিই**) বা আইডিই নামক এক জায়গায়। মানুষের ভাষা কম্পিউটার বুবাতে পারে না, এ জন্য আমাদেরকে কম্পিউটারের বোধগম্য উপযোগী মেশিন কোডে রূপান্তর করা হয়। এই প্রক্রিয়াকে বলা হয় **কম্পাইল করা** (**compile**)। আর আমরা এই ইন্ট্রাকশন লিখি টেক্সট এডিটর বা কোড এডিটরে। এই সবগুলো উপাদানকে কাজের সুবিধার জন্য একসাথে জোড়া দিয়ে যে সফটওয়্যার বানানো হয়, তাকে আইডিই বলে। এখানে লেখার ব্যাপারটা মাইক্রোসফট ওয়ার্ডে কিছু লেখার মতোই সোজা। আর যাতে লেখাটা মাইক্রোকন্ট্রোলারের বোঝার উপযোগী করা যায় এ জন্য আছে কম্পাইলার।

এবার আসি আরডুইনোর অন্যতম কাজ প্রোগ্রামিং অংশে। আমরা কোডগুলো লিখব একটা আইডিই। এর অর্থ হচ্ছে, এখানে এডিটর, কম্পাইলার আর কোড আপলোড করার সব ব্যাপার একসাথে সহজ করে দেওয়া আছে।

প্রথমে আমাদের কম্পিউটারে আইডিই ডাউনলোড করে ইনস্টল করে নিতে হবে। ‘**Arduino আইডিই**’ পাওয়া যাবে এই লিংকে

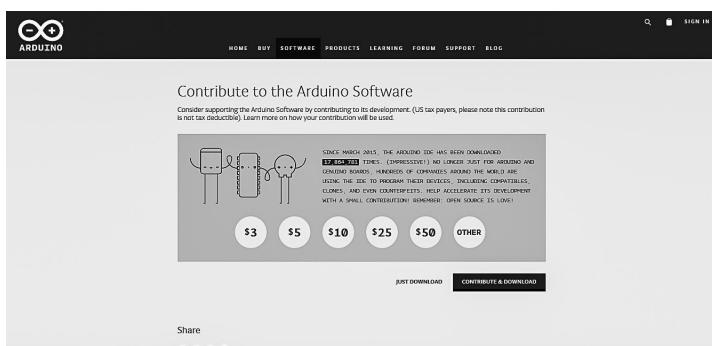
<https://www.arduino.cc/en/Main/Software>

এখানে উইডেজ ইস্টলারে ক্লিক করলে একটা পেজ আসবে।



চিত্র ২.১ : আইডিই ডাউনলোড ও ইনস্টল-প্রক্রিয়া

এখানে ডাউনলোডে ক্লিক করলে ডাউনলোড শুরু হয়ে যাবে।

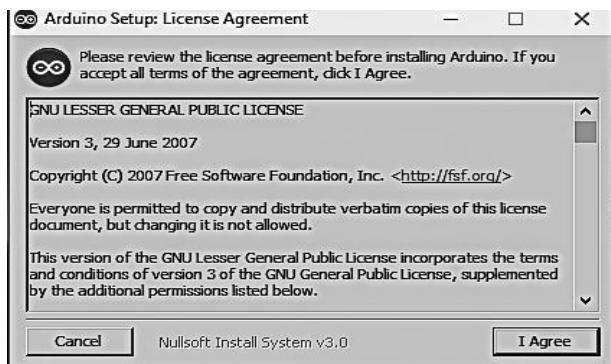


চিত্র ২.২ : আইডিই ডাউনলোড ও ইনস্টল-প্রক্রিয়া

একটুখানি কোড লিখি

অসুবিধা হলে বড় কারও সহায়তা নিয়ে ইন্টারনেট থেকে ডাউনলোড আর ইনস্টল করে নিতে পারো।

ডাউনলোডের পর স্টেআপ ফাইলটি খুলবে (**Open** করবে)। প্রথম যে পেজটা আসবে, সেখানে আই একগ্রি (**I Agree**) ক্লিক করতে হবে। মানে হচ্ছে তুমি এই সফটওয়্যারটা তোমার পিসিতে ইনস্টল (**Install**) করতে চাও।



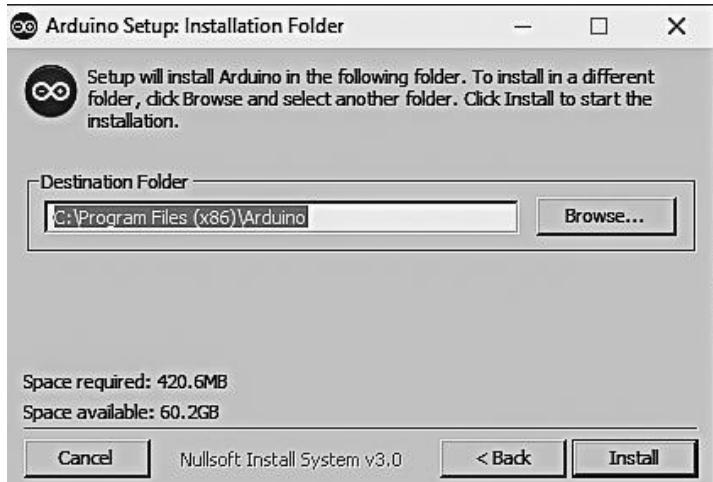
চিত্র ২.৩ : আইডিই ডাউনলোড ও ইনস্টল-প্রক্রিয়া

এরপর সবগুলো টিক চিহ্ন রেখে **next**-এ ক্লিক করতে হবে।

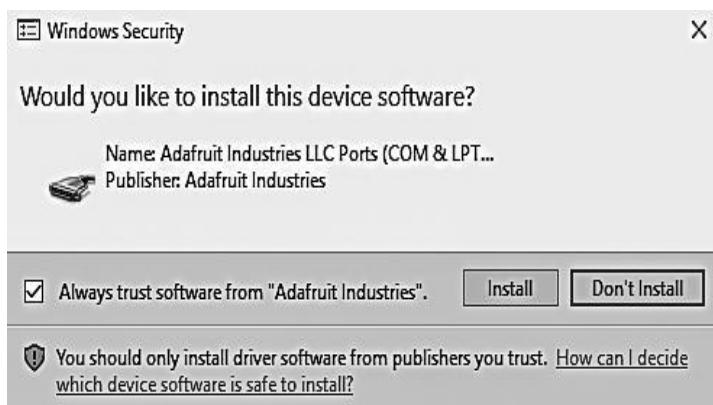


চিত্র ২.৪ : আইডিই ডাউনলোড ও ইনস্টল-প্রক্রিয়া

এরপর স্বয়ংক্রিয়ভাবে ইনস্টল হতে থাকবে। একপর্যায়ে ড্রাইভার ইনস্টল কয়েকটা প্রশ্নের বক্স/ উইন্ডো আসবে। ড্রাইভারের (**Driver**) বক্স এলে এর সবগুলোকে ইনস্টল/ নেক্সট (**install / next**)-এ ক্লিক করে ইনস্টল করে নিতে হবে। এখানে কিছু ক্রিনশট দেওয়া আছে। ইনস্টল করা শেষ হলে আইডিই সফটওয়্যার ওপেন করবে। তখন একটা মেইন পেজ আসবে।



চিত্র ২.৫ : আইডিই ডাউনলোড ও ইনস্টল-প্রক্রিয়া



চিত্র ২.৬ : ড্রাইভার ইনস্টলের সময় বেশ কয়েকবার অনুমতি চাইবে

একটুখানি কোড লিখি



চিত্র ২.৭ : ড্রাইভার ইনস্টলের সময় বেশ কয়েকবার অনুমতি চাইবে



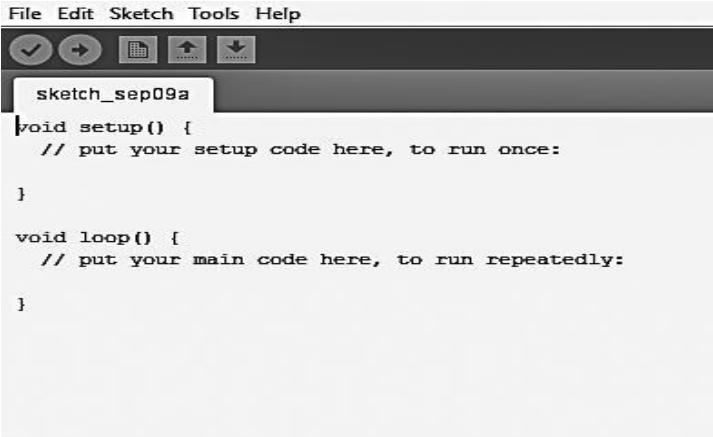
চিত্র ২.৮ : ড্রাইভার ইনস্টলের সময় বেশ কয়েকবার অনুমতি চাইবে

আরডুইনোর আইডিইতে লেখা প্রতিটি প্রোগ্রাম ফাইলকে বলা হয় এক একটি স্কেচ (sketch)

এখানে শুরুতেই সাদা অংশে প্রোগ্রাম লেখা হয়। প্রোগ্রামগুলো খুব সহজ।
ভয়েড সেটআপের (void setup) মধ্যে আরডুইনোর সাথে আমরা যে কানেকশনগুলো দিই, সেগুলো নিয়ে লিখি।

আর ভয়েড লুপের (**void loop**) মধ্যে আমরা কোনো কাজ যেটা বারবার হচ্ছে, সেটা লিখি। একটা বাস্তব উদাহরণ দিই। তোমার ঘড়িটি টিক টিক

করে চলছে, এটার মধ্যে সবচেয়ে ছোট কাজ যা বারবার হচ্ছে, কোনটা বলতে পারবে?



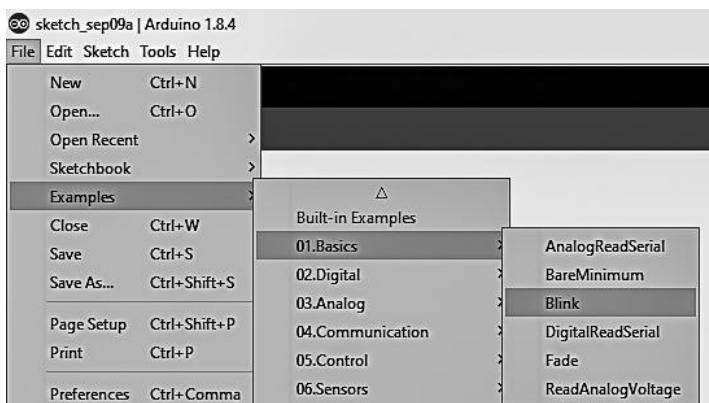
```
File Edit Sketch Tools Help
Sketch: sketch_sep09a
void setup() {
    // put your setup code here, to run once:
}

void loop() {
    // put your main code here, to run repeatedly:
}
```

চিত্র ২.৯ : আরডুইনো আইডিই চালু করার পর

একটা কঁটাওয়ালা ঘড়ির সবচেয়ে ক্ষুদ্রতম কাজ হলো সেকেন্ডের কঁটাকে প্রতি সেকেন্ড পরপর একটু করে সামনে টেনে নেওয়া। এখন চিন্তা করে দেখো তো, ব্যাটারির লাগানোর পর থেকে ঘড়ি এই একই কাজ করতে থাকে, প্রতি সেকেন্ডে একটু করে সামনে টানা। এই কাজ সে আজীবন করতে থাকে। এভাবে বারবার একই কাজ করাই হলো লুপ (**loop**)

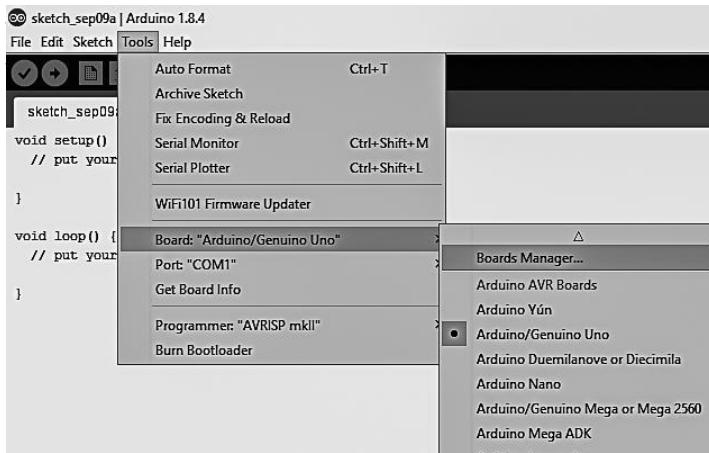
কোনো নতুন প্রোগ্রাম লিখতে চাইলে নিউ (**New**)-এ ক্লিক করলেই হবে, আর আগের লেখা কোনো ফাইল ওপেন করার জন্য ওপেন (**Open**) অপশন তো আছে। আর সংরক্ষণ করতে চাইলে সেভ (**Save**)



চিত্র ২.৯ : আরডুইনো আইডি ই চালু করার পর

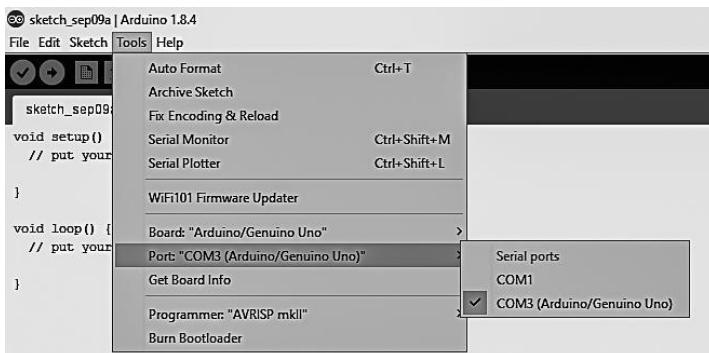
এ ছাড়া এখানে **Examples** ফাইলে অনেক প্রোগ্রাম করে দেওয়াই আছে। শুধু সার্কিটগুলো কানেকশন দিয়ে কোডগুলো আপলোড করে দিলেই হয়ে যাবে!

আরডুইনোর কোড আপলোডের জন্য কেবল (cable) দিয়ে তোমার কম্পিউটারের ইউএসবি পোর্টের সাথে আরডুইনো যুক্ত করতে



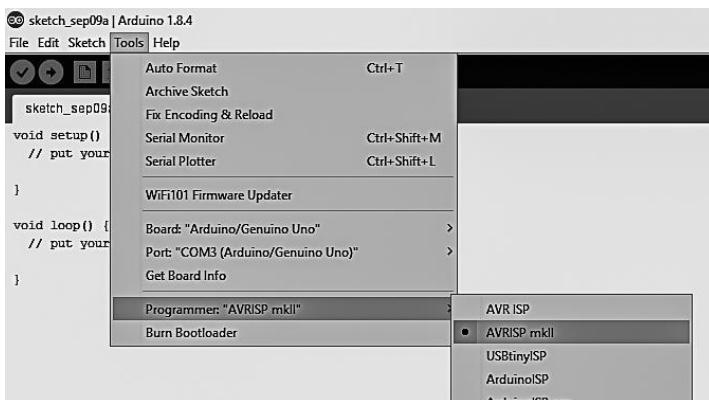
চিত্র ২.১১ : টুলসে গিয়ে সঠিক মডেলের আরডুইনো বোর্ড সিলেক্ট করে দেওয়া

এখন **tools**-এ গিয়ে **board**-এ তুমি যে মডেলের আরডুইনো চালাচ্ছ,
সেটা সিলেক্ট করতে হবে ।



চিত্র ২.১২ : আরডুইনোর জন্য সঠিক COM PORT সিলেক্ট করা ।

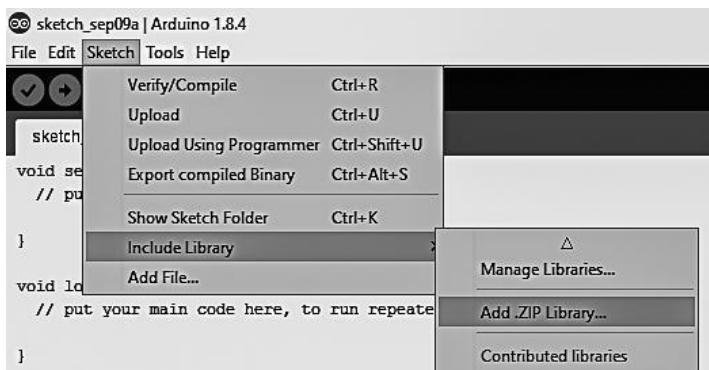
আর **Port** অপশনে গিয়ে যে পোর্টে আরডুইনো লেখা আছে, সেটা সিলেক্ট করতে হবে । অনেক সময় দেখা যায় আরডুইনোতে কোড আপলোড হয় না **ERROR** দেখায়, তখন এই টুলস অপশনটা চেক করবে ।



চিত্র ২.১৩ : টুলস অপশন থেকে প্রোগ্রামার অপশন সঠিকভাবে সিলেক্ট করা

এখানে প্রোগ্রামার হিসেবে ‘**AVRISP mkII**’ এটাই ভালো হয় ।

একটুখানি কোড লিখি



চিত্র ২.১৪ : প্রয়োজনভেদে লাইব্রেরি যোগ করার উপায়

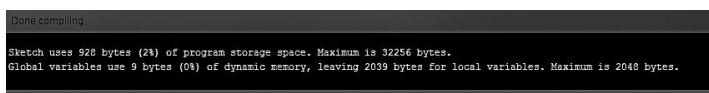
আরডুইনোতে কাজ করার জন্য অনেক লাইব্রেরি ফাইল পাওয়া যায়। এগুলো **ADD ZIP** লাইব্রেরিতে ক্লিক করে দিয়ে ফাইলটা সিলেক্ট করে দিলেই হয়ে গেল!

এখানে যে টিক চিহ্ন বাটন দেখা যাচ্ছে, এতে ক্লিক করলে কোডটা কম্পাইল হবে। মানে তোমার যন্ত্রের বোর্ডের মতো শূন্য-এক ভাষায় কু পাস্তর হয়ে যাবে।



চিত্র ২.১৫ - (বাঁ দিক থেকে কম্পাইল, আপলোড বাটন ও অন্যান্য)

কোডে তোমার যদি কোনো ভুল হয়ে থাকে নিচের ওই কালো ঘরে তোমার ওই ভুলগুলো (Error) লেখা থাকবে। ভুল ঠিক না করা পর্যন্ত তোমার কোড কম্পাইল হবে না।



চিত্র ২.১৬ : কোডে ভুল থাকলে আইডিইইর নিচ প্রান্তে পাবে এ রকম ইরর মেসেজ

আর ডান তীর চিহ্ন (চিত্র - ২.১৫) বাটনে ক্লিক করলে দেখবে তোমার কোড আপলোড হয়ে যাবে আরডুইনোতে!

তোমরা যারা মন খারাপ করছ কম্পিউটার নেই ভেবে। তোমাদের জন্য রয়েছে সুখবর। আরডুইনোর অ্যান্ড্রয়েড আইডিই পাওয়া যায়। নাম ‘**ArduinoDroid: Arduino আইডিই**’

গুগল প্লে স্টোর থেকে ইনস্টল করে নিয়ো। তুমি এটা থেকে মোবাইল ফোন দিয়েও কাজ করতে পারবে। তবে তোমার ফোনে ওটিজি (**OTG**) সুবিধা থাকতে হবে।

ওটিজি (**On the Go**) কী?

সহজ ভাষায় যদি বলি, এটা স্মার্টফোনের একটি ফিচার। ওটিজি সাপোর্ট করে এমন স্মার্টফোনগুলোতে তুমি ইউএসবি জ্যাকসংবলিত ডিভাইস সংযুক্ত করতে পারবে। যেমন ধরো পেনড্রাইভ, কার্ড রিডার, ডিজিটাল ক্যামেরা আর আরডুইনো!

বইয়ের শুরুতে আমরা বলেছি, এ বইটি পড়ে তুমি ধাপে ধাপে কাজগুলো করবে। দিন শেষে তোমার কাছে দেখানোর মতো বেশ কটি প্রজেক্ট থাকবে। তুমি যখন তোমার নিজ হাতে করা প্রজেক্ট আবু-আমুকে, শিক্ষক বা বন্ধুদের জানাবে, তখন দেখবে অনেক প্রশংসা পাবে।

তিন
বাতি জ্বালানো দিয়ে শুরু

তোমাদের মধ্যে যারা একটু চটপটে, তারা তো নাম দেখেই বুঝে গেছ আমরা কী করব! হ্যাঁ, আমরা আরডুইনো দিয়ে একটি এলইডি (LED) বাতি জ্বালাব। আমরা কয়েকটি কাজ ধাপে ধাপে করব :

প্রথমত, আরডুইনো দিয়ে একটি এলইডি জ্বালাব-নেভাব-জ্বালাব-নেভাব... অর্থাৎ লুপ (loop)

দ্বিতীয়ত, সেই এলইডি বাতির জ্বালা-নেভাব হার কমবেশি করব।

তৃতীয়ত, আমরা ওটি তিনি রঙের এলইডি বাতি একসাথে জ্বালাব। এটা অনেকটা ঝাড়বাতির মতো দেখাবে।

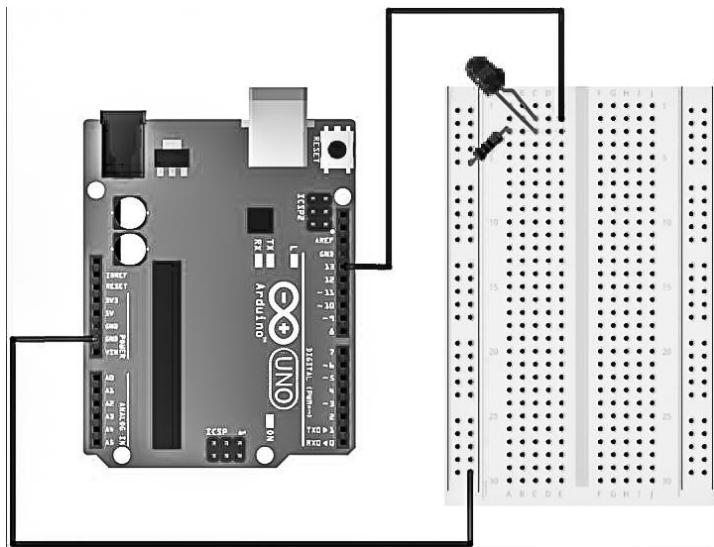
আরডুইনো তো একটু টেকনিক্যাল জিনিস, তোমাদের শুরুর দিকে একটু বুবাতে কষ্ট হতে পারে। কিন্তু ভয় পেয়ো না। পুরো বইটা একটা গল্ল বা উপন্যাসের বইয়ের মতো পড়তে থাকো। উপন্যাসে যেমন মাঝে মাঝে কাহিনির পঁচ বোঝা যায় না, কিন্তু তুমি পুরো উপন্যাস পড়া শেষ করলেই সব বোঝা যায়। তেমনি এ বইটিও তুমি একটা গল্লের মতো পুরোটা পড়বে, যে গল্লের হিরো তুমি। আর তোমার সঙ্গী হলো তোমার কল্পনার রোবট।

তুমি কি জানো, এখন আমরা যে প্রজেক্ট করব, সেটা অনেকে আগেও করেছ। আচ্ছা, তোমরা খেলাচ্ছলে কখনো ব্যাটারি আর বাতি তার

দিয়ে লাগিয়ে জ্বালাওনি? অনেকেই এভাবে বাতি জ্বালিয়েছ। আমরা সেই একই কাজ করব, শুধু আমাদের ম্যানুয়াল হাত দিয়ে করার বদলে কাজটি করবে ছোট রোবট-মস্তিষ্ক (মাইক্রোকন্ট্রোলার)

চলো, আমরা আমাদের উপকরণগুলো হাতের কাছে নিয়ে নিই। তুমি তোমার কম্পিউটারে আরডুইনো আইডিই (IDE) চালু করো। হাতের কাছে তোমার আরডুইনো উনো, কিছু এলইডি বাতি (LED- light emitting diode), কিছু জাম্পার তার (jumper wire), কিছু রোধ (220 Ohm resistor) আর অবশ্যই আমাদের প্রিয় সেই পাউরণ্টি বোর্ড (bread board) নাও।

গল্লের শুরুতে একটা ছবি দেখে নিলে কেমন হয়? আমরা আমাদের প্রজেক্টের একটা সহজ ডায়াগ্রাম দেখে নিই চলো।



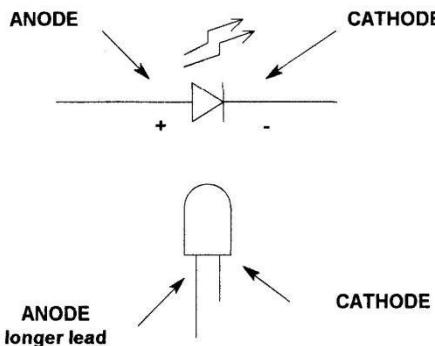
চিত্র ৩.১ : এলইডি, রোধ ও আরডুইনোর কানেকশন

এবাব ডায়াগ্রামের মতো করে তোমার নিজের আরডুইনো সার্কিট সাজিয়ে ফেলো।

চলো, সার্কিট সাজাতে আমি তোমাকে সাহায্য করছি।

বাতি জ্বালানো দিয়ে শুরু

প্রথমে ব্রেড বোর্ডে এলইডি বাতি বসাও। এলইডি বাতির বড় পা হচ্ছে অ্যানোড (Anode) আর ছোট পা হচ্ছে ক্যাথড (Cathode) বড় পা বা অ্যানোডে ব্যাটারির পজেটিভ প্রান্ত যুক্ত হয়।



চিত্র ৩.২ : এলইডির অ্যানোড ও ক্যাথড চেনা

একটি এলইডি আসলে একটি ডায়োড (Diode)। ডায়োডের দুটি প্রান্ত থাকে ধনাত্মক প্রান্ত আর ঋণাত্মক প্রান্ত। এই ধনাত্মক প্রান্তকে বলা হয় অ্যানোড আর ঋণাত্মক প্রান্তকে বলা হয় ক্যাথড। সাইজের দিক থেকে অ্যানোড একটু বড় হয়।

যখন এই ডায়োডের মধ্যে সঠিকভাবে সংযোগ দেওয়া হয়, তখন এই ডায়োড থেকে আলো নির্গত হয়। পদার্থের ধর্মের ভিত্তিতে বিভিন্ন মৌল বিভিন্ন রঙের আলো বিকিরণ করে থাকে। এ জন্য আমরা বাজারে বিভিন্ন রঙের সুন্দর সুন্দর এলইডি দেখতে পাই।

তুমি তোমার ২২০ ওহম রোধ নাও এবং এটিকেও ব্রেড বোর্ডে যুক্ত করো। রোধ দেওয়ার কারণ হচ্ছে এলইডি বাতি যেন অতিরিক্ত ভোল্টেজে পুড়ে না যায়, সে ব্যবস্থা করা।

ছবিতে দেখতে পাচ্ছ, আমরা জাম্পার ওয়্যার দিয়ে আরডুইনো ১৩ নং ডিজিটাল পিনের সাথে এলাইডির অ্যানোড বা বড় পা যুক্ত করেছি। ১৩ নং পিনে আমরা পজেটিভ ভোল্টেজ দেব।

আর ছোট পা যুক্ত করেছি আরডুইনোর গ্রাউন্ডের সাথে (**Ground / GND**)

আর যেকোনো এক পায়ের সাথে **সিরিজে** (সিরিজ কানেকশন তোমরা ব্রেড বোর্ডের অধ্যায়ে শিখেছ) রোধ যুক্ত করবে।

তুমি যখন এই সার্কিট বসাবে, আরডুইনোর পাওয়ার কেবল তখন খুলে রাখবে। ব্যাটারি / ইউএসবি পাওয়ার লাগানো অবস্থায় সার্কিট বসাতে গেলে আরডুইনো বোর্ড নষ্ট হওয়ার ভয় থাকে। পুরো সার্কিট বসানো হলে আরডুইনোর সাথে দেওয়া ইউএসবি কেবল দিয়ে তোমার কম্পিউটারের সাথে যুক্ত করবে।

এবার আরডুইনো আইডিই চালু করে ঘাটপট নিচের কোড লিখে আপলোড করো।

```
void setup() {
    pinMode(13, OUTPUT);
}

void loop() {
    digitalWrite(13, HIGH);
    delay(1000);

    digitalWrite(13, LOW);
    delay(1000);
}
```

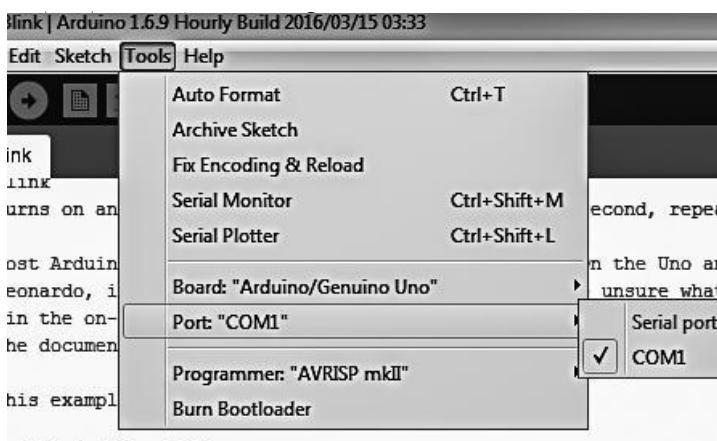
কি, কোড আপলোড হচ্ছে না?

বাতি জ্বালানো দিয়ে শুরু

দেখো তো কোথাও কোনো সেমিকোলন মিস করেছ নাকি? কিংবা বানান ভুল?

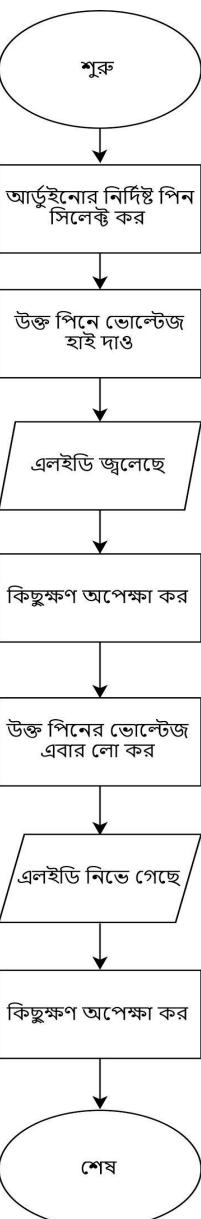
তবুও হচ্ছে না?

আচ্ছা, তুমি কি টুলস (tools) থেকে সঠিক পোর্ট (port) সিলেক্ট করেছ? আবার টুলস থেকে board অপশনে গিয়ে Arduino / Genuino UNO সিলেক্ট করেছ তো?



চিত্র ৩.৩ : কোড আপলোডে সমস্যা হলে টুলস অপশন চেক করতে হবে

এবার চলো আমরা একটা ফ্লো-চার্ট এঁকে লজিকটা বুঝি। আমরা কোড আর ফ্লো-চার্ট দুটো মিলিয়ে বুঝব।



চিত্র ৩.৪ : একটি প্রোগ্রাম চলার ফ্লো-চার্ট

বাতি জ্বালানো দিয়ে শুরু

প্রথমেই আমরা আমাদের আরডুইনোকে বলে দিই কোন পিন কোন মোডে থাকবে। তায়েড সেটআপ ফাংশন শুরুর এ কাজটি করে।

```
void setup() {  
}  
  
সেটআপ ফাংশানের ভেতরে লিখেছি  
  
pinMode(13, OUTPUT);
```

এই পিন-মোড বলে দেয় কত নম্বর পিনকে আমি কোন কাজের জন্য সেট করেছি। যেমন এখানে বলা হচ্ছে ১৩ নং পিন আউটপুট হিসেবে কাজ করবে।

তোমরা ইনপুট আর আউটপুট বোবো তো?

এই যেমন আমরা সারা বছর পড়ালেখা করে পরীক্ষার খাতায় যা লিখি তা হচ্ছে ইনপুট। আর স্যারেরা আমাদের খাতা দেখে মার্কস দিয়ে আমাদের যে রেজাল্ট দেন, সেটা হচ্ছে আমাদের এক বছরের পড়ালেখার আউটপুট।

সহজ কথায়, ইনপুট হচ্ছে কোনো কিছুতে দেওয়া। আর আউটপুট হচ্ছে কোনো কিছু থেকে বের করে নেওয়া।

এখানে আরডুইনোতে আমরা যে ইউএসবি পাওয়ার দিচ্ছি তা হলো, ইনপুট ভোল্টেজ। আর ১৩ নং পিন থেকে এলাইডির জন্য আরডুইনো থেকে আউটপুট ভোল্টেজ নিচ্ছি। উল্লেখ্য, আরডুইনোর ডিজিটাল পিন ৫ ভোল্ট বিদ্যুৎ দেয়।

আমরা আগেই জানি আরডুইনোর ০-১৩ পর্যন্ত ১৪টি ডিজিটাল পিন রয়েছে। এখন আমরা ১৩ নং ডিজিটাল পিনে ভোল্টেজ হাই দেব। এ জন্য কোডে লিখেছি

```
digitalWrite(13, HIGH);
```

ফ্লো-চার্ট (**flow chart**) দেখে বলো তো এবার আমরা কী করব?

হ্যাঁ, এবার আমরা এই ভোল্টেজ হাই অবস্থায় কিছুক্ষণ অপেক্ষা করব।
অর্থাৎ

```
delay(1000);
```

‘Delay’ শব্দের আক্ষরিক অর্থ হলো দেরি করা, থেমে থাকা, ধীরে যাওয়া বা অপেক্ষা করা।

delay(); ফাংশনের ভেতরে যত মান দেব, তত মাইক্রোসেকেন্ড পরিমাণ সময় অপেক্ষা করা হবে। আসলে এই সময়টুকু তোমার রোবটের মন্তিক্ষ নতুন কোনো কাজ করবে না। আমরা জানি, ১ second = ১০০০ microsecond

তাহলে বলো তো **delay (3000)**; লিখলে কত সেকেন্ড অপেক্ষা করবে?

শোনো, একটা বুদ্ধি দিই, ডিলে ফাংশনের ভেতরের মান ভিন্ন ভিন্ন দিয়ে পরীক্ষা করবে যে এলইডি কীভাবে জ্বলে। একবার ৫০০ দেবে, একবার ২০০০, একবার ১০ কিংবা তোমার ইচ্ছেমতো মান!

এখানে **delay()**; হচ্ছে একটা ফাংশন। ফাংশন নামটা তোমরা অনেক শুনেছ। এই যেমন এই যন্ত্রের ফাংশন কী? এ রকম অনেক বলতে শুনেছ। প্রোগ্রামিংয়ের ভাষায়ও ঠিক একই রকম।

ফাংশন একটা নির্দিষ্ট কাজ করে।

আর ওই নির্দিষ্ট কাজ করা অংশকে আমরা প্রোগ্রামের মধ্যে শুধু একবার লিখে দেওয়াকে বলে **কল** করা। কল (**call**) মানে ডাকা। তার মানে আমরা ওই ফাংশনকে ডাকছি।

আরডুইনোতে তোমরা অনেকগুলো ফাংশনের সাথে পরিচিত হবে। আরডুইনো যারা বানিয়েছেন, তারা অনেকগুলো ফাংশন বানিয়ে রেখেছেন আমাদের কাজের সুবিধার জন্য। আবার তুমি চাইলে নিজেও ফাংশন বানাতে পারো ইচ্ছেমতো।

ফাংশনের ব্যাপারটা যদি তোমার মাথায় এখনো একটু হালকা হালকা ঢুকে থাকে তাহলে একটু ভালোমতো ঢুকিয়ে নাও এবার একটা উদাহরণ থেকে। ধরো তুমি একটা বিশাল কোম্পানির মালিক! (এখন না হলেও নিশ্চয়ই একদিন হবে!)

তোমার কোম্পানি হচ্ছে শার্টের কোম্পানি। তোমরা শার্ট বানাও। তুমি মালিক হিসেবে সবকিছু দেখাশোনা করো, সব কাজ ঠিকঠাকমতো হচ্ছে কি না।

তোমার কোম্পানিতে একজন লোক আছে যে কাপড় দোকান থেকে কিনে আনে। একজন আছে কাপড়গুলো কাটে, আরেকজন আছে শার্টের বডিটা বানায়, আরেকজন আছে শুধু হাতগুলো বানিয়ে দিয়ে সেলাই করে, আরেকজন বোতাম লাগায়, আরেকজন আছে প্যাকেট করে। আর তোমার কাজ ঠিকমতো হচ্ছে কি না!

এই কাপড় আনা, কাটা, বডি বানানো, হাত সেলাই, বোতাম লাগানো, প্যাকেটিং — সবই একেকটা ফাংশন। আর তুমি হচ্ছ মেইন, মানে তুমি আছ আমাদের আরভুইনো কোডের **void loop()** মধ্যে। আর যেহেতু আরভুইনোর মধ্যে **void loop()** আমরা ছোট কাজগুলো করে থাকি আর লুপ সেই ছোট কাজগুলোই বারবার করে দেয়।

তোমার কিছুই যায়-আসে না তোমার কাপড় কাটার লোক কাজ করল না অন্য কাউকে দিয়ে করাল তুমি শুধু আউটপুট চাও।

এখন নিশ্চই তোমার মাথায় ঘুরছে আমরা একটা ফাংশন দেখে চিনব কীভাবে?

একটা ফাংশনকে লেখা হয় ঠিক এভাবে :

```
Return_type Function_Name(parameter list)
{
    // function body
}
```

প্রতিটা ফাংশনই কিছু কাজ করে। কাজ করার পর সে সেই কাজগুলো আমাদের ফেরত দেয়। কোন ফাংশন কী ফেরত দেবে, সেটা আমরা লিখে দিই **রিটার্ন টাইপে**।

ফাংশন নেম আমাদের বোঝার জন্য কার কাজ কী?

আৱ প্যারামিটাৰ হচ্ছে ফাংশনটা কী কী ইনপুট নেবে। বডিৰ মধ্যে
লিখৰ ফাংশনেৰ কাজ।

ধৰো কাপড় কাটা একটা ফাংশন। তাহলে এটা আমৱা লিখৰ এভাৱে
কাটা কাপড় কাপড় কাটা (নতুন কাপড়)

{

নতুন কাপড় একটা নাও;
জামাৰ সাইজগুলো নাও;
সাইজ অনুসাৱে কেটে ফেলো;
এবাৱ রিটাৰ্ন কৱো কাটা-কাপড়;

}

আবাৱ ধৰো এই কাটা কাপড় নেবে বডি বানানো ফাংশন এভাৱে :

সেলাই-কৱা-বডি বডি বানানো (কাটাথকাপড়, সুতা)

{

কাটাথকাপড় নাও;
সুতা নাও;
সেলাই কৱো;
রিটাৰ্ন সেলাই-কৱা-বডি;

}

আশা কৱি তুমি এতক্ষণে নিশ্চয়ই ফাংশন খুব ভালোমতো বুঝো গেছ!
এবাৱ চলো আবাৱ কোডে যাই

কিছুক্ষণ বাতি জ্বালিয়ে অপেক্ষা কৱাৱ পৰ আমৱা এবাৱ বাতিটিকে
নেভাৱ। তাহলে আমৱা কী কৱতে পাৱি বলো তো?

চলো আমৱা ডিজিটাল পিনকে রাইট কৱে এৱ ভোল্টেজ লো কৱে
দিই। অৰ্থাৎ

digitalWrite(13, LOW);

বাতি জ্বালানো দিয়ে শুৱৰ

দেখো এই digitalWrite-ও কিন্তু একটা ফাংশন! এই ফাংশন কম কারেন্ট পাঠিয়ে বাতিকে নিভিয়ে রাখে। আর লোর বদলে প্যারামিটারে হাই দিলে, বাতিটা জ্বালিয়ে রাখত।

এরপর আবার ডিলে ফাংশন দিয়ে কিছুক্ষণ অপেক্ষা করব।

আচ্ছা আমরা যদি এ কাজটি বারবার করতে থাকি। জ্বলা-নেভা-জ্বলা-নেভা-জ্বলা-নেভা...

এ জিনিসটিকে আমরা যুক্তির ভাষায় বলি ‘loop’ অর্থাৎ একই কাজ বার বার করা। এ জন্য একটি ফাংশন আছে

```
void loop() {
```

```
}
```

লুপের দুই ব্রাকেটের (**curly braces**) মাঝে আমরা যা লিখে দেব সেই কাজটা আমাদের আরডুইনো বারবার করতে থাকবে, করতেই থাকবে, করতেই থাকবে...

তোমার সুবিধার জন্য পুরো কোডটি আরেকবার দিয়ে দিচ্ছি।

```
/* the setup function runs once when you press
reset or power the board */
void setup() {
// initialize digital pin 13 as an output.
pinMode(13, OUTPUT);
}

// the loop function runs over and over again
// forever
void loop() {
    digitalWrite(13, HIGH);      // turn the LED on
    with 5V
    delay(1000);                // wait for a second
    digitalWrite(13, LOW);       // turn the LED off
    by 0V  delay(1000);          // wait for
    a second
}
```

এবার একটা বিষয় খেয়াল করেছ? আমি কোডের সাথে ডাবল স্ল্যাশ ‘//’ দিয়ে কিছু ব্যাখ্যা লিখে রেখেছি। অর্থাৎ কোন লাইনে আমি কী করেছি, তা কমেন্ট (**comment**) রেখেছি। এভাবে ডাবল স্ল্যাশ (double slash) ব্যবহার করে কমেন্টের মাধ্যমে তুমি তোমার কোডের ডকুমেন্টেশন রাখতে পারো। আর প্রথম লাইনে খেয়াল করো, কমেন্টের শুরুতে /* দিয়ে মাঝে কমেন্ট করে শেষে আবার */ দিয়েছি। এভাবেও কমেন্ট করতে পারো, /*... */

ডকুমেন্টেশন করা থাকলে কয়েক মাস পরেও তুমি তোমার কোড পড়লে বুঝতে পারবে কোথায় কী করেছিলে। গ্র্যাফ প্রজেক্ট করার সময় এমনভাবে কমেন্ট লিখে রাখা খুব জরুরি, যেন তোমার দলের বাকিরাও তোমার কোড বুঝতে পারে তাই।

আমার বিশ্বাস, তুমি ওপরের প্রজেক্টটি সুন্দরভাবে করতে পেরেছ। তুমি যদি দেখতে চাও যে তোমার মতন অন্যরা এই প্রজেক্ট কীভাবে করেছে, তাহলে ইউটিউবে (**youtube**) গিয়ে **Arduino LED blink project** লিখে সার্চ করো। আর কয়েকটা ভিডিও দেখো। অন্যদের করা প্রজেক্টগুলো দেখলে তুমি নিজের আইডিয়া আরো উন্নত করতে পারবে।

এবার, তুমি কি চেষ্টা করলে এ রকম ৩টি এলইডি একসাথে জুলাতে পারবে? চেষ্টা করে দেখো। আর তুমি যদি কাজটায় সফল হও, নিজেই নিজেকে একটা চকলেট উপহার দেবে। নিজে নিজেকে গিফট দিলে আত্মবিশ্বাস অনেক বাঢ়ে।

চার

অনেকগুলো বাতি জ্বালাই

তোমরা এতক্ষণে একটা এলইডি বাতি জ্বালাতে শিখে গেছ। এবার আমরা অনেকগুলো এলইডি জ্বালাব। এলইডিগুলো একটার পর একটা জ্বলতে থাকবে। তারপর আমরা `delay()` নিয়ে আরো কিছু মজার কাজ করব।

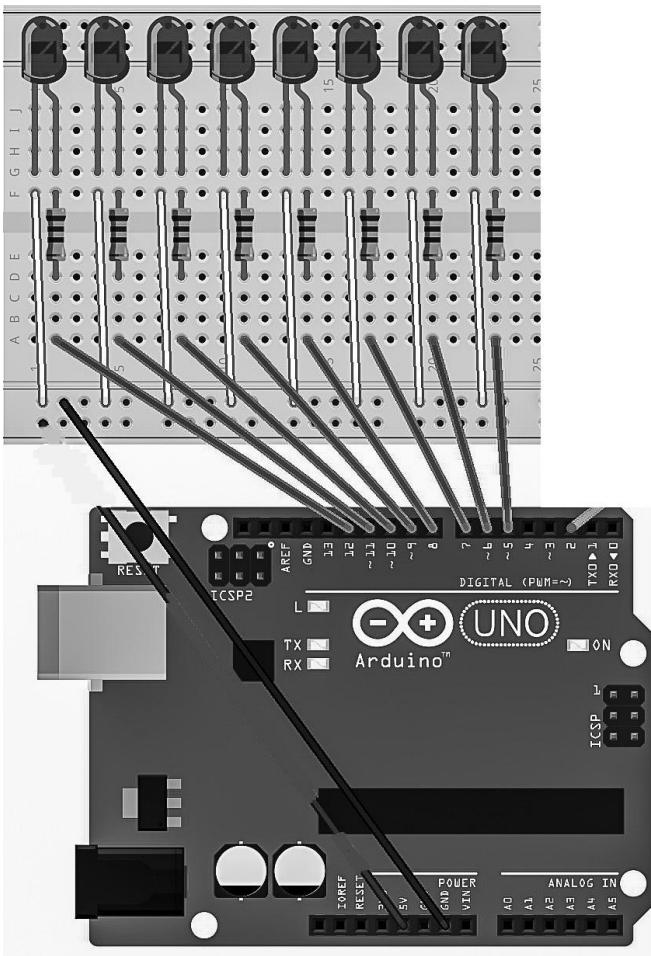
এ জন্য আমাদের দরকার অনেকগুলো এলইডি.প্রায় ৮টা। তোমরা চাইলে বিভিন্ন রঙের এলইডি নিতে পারো।

তোমাদের কি ব্রেড বোর্ডের কথা মনে আছে। সেখানে আমরা প্যারালাল কানেকশন শিখেছিলাম। এই এলইডিগুলোও আমরা এভাবে প্যারালাল কানেকশন দেব। আমরা এলইডির পজেটিভগুলো আরডুইনোর সাথে লাগাব। আর নেগেটিভগুলো সব একসাথে নিয়ে আরডুইনোর **গ্রাউন্ড (GND)** লাগিয়ে দেব।

চলো, ছবিটা দেখে নেওয়া যাক।

এখানে এলইডিগুলো ৫-১২ পর্যন্ত পিনে লাগানো হয়েছে। এবার লাগিয়ে ফেলো দেখে দেখে সব।

এবার চলো আরডুইনো আইডিইতে কোড লেখি, মানে আমাদের সিস্টেমের বুদ্ধি লিখি।



চিত্র ৪.১ : এক সারি এলইডি আরডুইনোর সাথে যুক্ত করা

অনেকগুলো বাতি জ্বালাই

```
void setup() {  
    pinMode(5, OUTPUT);  
    pinMode(6, OUTPUT);  
    pinMode(7, OUTPUT);  
    pinMode(8, OUTPUT);  
    pinMode(9, OUTPUT);  
    pinMode(10, OUTPUT);  
    pinMode(11, OUTPUT);  
    pinMode(12, OUTPUT);  
}  
  
void loop()  
{  
    digitalWrite(5, HIGH);  
    delay(1000);  
    digitalWrite(5, LOW);  
  
    digitalWrite(6, HIGH);  
    delay(1000);  
    digitalWrite(6, LOW);  
  
    digitalWrite(7, HIGH);  
    delay(1000);  
    digitalWrite(7, LOW);  
  
    digitalWrite(8, HIGH);  
    delay(1000);  
    digitalWrite(8, LOW);
```

```
digitalWrite(9, HIGH);
delay(1000);
digitalWrite(9, LOW);

digitalWrite(10, HIGH);
delay(1000);
digitalWrite(10, LOW);

digitalWrite(11, HIGH);
delay(1000);
digitalWrite(11, LOW);
```

```
digitalWrite(12, HIGH);
delay(1000);
digitalWrite(12, LOW);
}
```

আরড্বিনোর একটা বাতি জ্বালাতে তুমি যদি শিখে থাকো তাহলে এই
কোডটা দেখে তোমার কাছে নতুন লাগবে না।

আবার বলে দিচ্ছি প্রথমে **void setup()**-এ আমরা সবগুলো এলাইভি যে
আউটপুট এটা ডিক্লেয়ার (**Declare**) করে দিয়েছি।

তারপর এই তিনটি লাইন একেকটা এলাইভি জ্বালাচ্ছে।

```
digitalWrite(5, HIGH);
digitalWrite(5, LOW);
delay(1000);
```

এই তিন লাইন কীভাবে কাজ করে আগের অধ্যায়ে ভালোমতো শিখে
গেছ।

অনেকগুলো বাতি জ্বালাই

এবার আরেকটা শেষ বুদ্ধি দিই,

কোডের ডিলের মান তুমি ২০০, ৫০০, ২০০০ বিভিন্ন বাতিতে বিভিন্ন দিয়ে দেখবে। অনেক সময় একটা বাতিকে হাই করার পর আর লো করবে না।

মানে কোডের লাইনগুলো নিয়ে এক্সপেরিমেন্ট করবে নিজের মতন। তাহলে দেখবে নতুন একটা প্যাটার্ন বানিয়ে ফেলেছ! আমাদের মূল কোডে একটু এডিট করলেই নতুন নতুন প্যাটার্ন পাবে তুমি—

```
.

.

digitalWrite(5, HIGH);
delay(1000);
// digitalWrite(5, LOW);
digitalWrite(4, LOW);

digitalWrite(6, HIGH);
delay(1000);
// digitalWrite(6, LOW);
digitalWrite(5, LOW);

digitalWrite(7, HIGH);
delay(1000);
// digitalWrite(7, LOW);
digitalWrite(6, LOW);

digitalWrite(8, HIGH);
delay(1000);
// digitalWrite(8, LOW);
digitalWrite(7, LOW);

.
```

তাহলে তুমি এখন অনেকগুলো এলাইডি একটার পর আরেকটা জ্বালিয়ে ফেলেছ!

অভিনন্দন তোমাকে!

এখন থেকে তুমি যতগুলো প্রজেক্ট করলে তার সবগুলো তুমি আমাদের জানিয়ে দিতে পারো। তোমার প্রজেক্টের ছবি, ভিডিও তুলে সেগুলো তোমরা আমাদের ই-মেইল করতে পারো।

আর সবচেয়ে ভালো বুদ্ধি www.youtube.com-এ একটা আইডি খুলে আপলোড দিয়ে দিতে পারো। তোমাদের ফেসবুক আইডি থাকলে ফেসবুকেও আপলোড দিতে পারো।

আমাদের ই-মেইল করার ঠিকানা বইয়ের প্রথমে আছে।

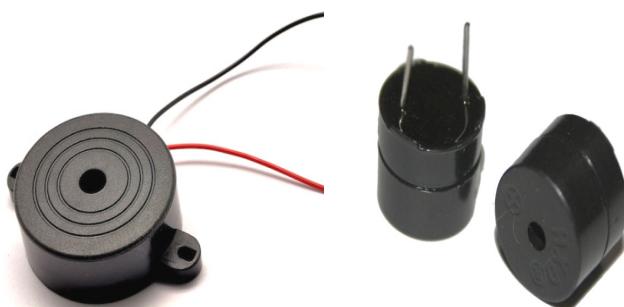
চেষ্টা ১ : কোডের মধ্যে delay ()-এর মানগুলো পরিবর্তন করে দেখো কী মজা হয়!

চেষ্টা ২: ধরো আটটা এলাইডির মধ্যে শুধু তিনটা করে জুলবে একসময়ে। অন্যগুলো সব বন্ধ থাকবে এ রকম একটা প্রোগ্রাম লিখে ফেলো!

পাঁচ

বাজারের কানা

বাজার শুনে কী মনে হচ্ছে? এ কোন বাজার? এটা কোনো শাকসবজির বাজার বা নিউমার্কেট নয়। এ হচ্ছে **Buzzer / Piezo Buzzer**, মানে যে শব্দ করে! বিপ বিপ ধরনের শব্দ দেয়!



চিত্র ৫.১ : বিভিন্ন ধরনের বাজার

আমরা সবাই ঢোল দেখেছি। ঢোলে একটা বাজানোর জন্য লাঠি বা ড্রামস্টিক থাকে। বাজার জিনিসটাও তাই! এটাকে তুমি ইলেকট্রিক ঢোল বলতে পারো।

ডোলের যেমন পর্দা আছে বাজারেও তেমনি পর্দা আছে, এই পর্দা পিজিও ইলেকট্রিক পর্দা, অর্থাৎ কারেন্টের উপস্থিতিতে

এটা বাজে। এই কারেন্টের ফ্রিকোয়েন্সির মান কমবেশি করে বাজার দিয়ে যেকোনো ফ্রিকোয়েন্সির শব্দ বাজিয়ে ফেলা যায়! আমরা বাজার দিয়ে প্রথমে কিছু সাউন্ড করব।

বাজারকে আরডুইনোর সাথে কানেক্ট করা খুব সহজ। বাজারের এক মাথায় প্লাস চিহ্ন (+) আঁকা আছে, সেই মাথার পিনটাকে আরডুইনোর যেকোনো I/O পিনে লাগাতে হবে। ধরো আমরা লাগালাম 12 নম্বর পিনটায়।

এবার আরেকটা যে মাথা আছে, সেটা হচ্ছে গ্রাউন্ড পিন। এই পিনটাকে আমাদের সংযুক্ত করতে হবে আরডুইনোর গ্রাউন্ড পিনের সাথে!

ব্যস! সার্কিট বসানো শেষ!

বাজারের সার্টিক অনেকটা চিত্র ৩.১-এ দেখা এলইডি, রোধ এবং আরডুইনোর কানেকশনের মতোই।

এবার গোর্ট সিলেক্ট করে নিচের কোডটা ধূম করে আপলোড দিয়ে দাও।

```
int buzzerPin = 12;
void setup() {
    pinMode(12,OUTPUT);
}

void loop() {
    digitalWrite(12,HIGH);
}
```

কী? কিছু শুনতে পাচ্ছ? বাজারের কানা!

আচ্ছা এবার চলো কোডটা একটু দেখি।

এখানে আমরা ১২ নম্বর পিনে বাজারকে কানেক্ট করেছি। এ জন্য মাইক্রোকন্ট্রোলারকে বুঝিয়ে দিচ্ছি এভাবে :

```
int buzzerPin = 12;
```

আগেই বলেছি সেটাপের মধ্যে সার্কিটের কথাবার্তা লেখা হয়! এখানে এই বাজারটা একটা আউটপুট ডিভাইস,

এটা আমরা কম্পিউটারকে বুঝিয়ে দিয়েছি এভাবে—
pinMode(12, OUTPUT);
এবার আসো লুপে!

এখানে আমরা ছোট কাজ যেটা বারবার হয়, সেটাকে লিখি। এখানে কাজ হচ্ছে প্রতি ক্লক পালসে একবার করে বাজারটা বাজানো!

digitalWrite(12, HIGH);
এটার মানে হচ্ছে ১২ নম্বর পিনটাতে হাই ভোল্টেজ যাচ্ছে। এ জন্য তোমার বাজার বেজেই যাচ্ছে!

এবার আসো কিছু মজার কাজ করি!

আমরা এলইডির প্রজেক্টেই রিংকিং-এ **delay** শিখেছি। এটার মধ্যে তুমি যে সংখ্যা লিখবে ততক্ষণের জন্য মাইক্রোকন্ট্রোলার থেমে থাকবে। মানে নতুন কোনো কাজ করবে না!

চলো মাঝখানে ডিলে দিয়ে কাজ করে দেখি!

```
int buzzerPin = 12;

void setup() {
    pinMode(12, OUTPUT);

}

void loop() {
    digitalWrite(12, HIGH);
    delay(1000);
    digitalWrite(12, LOW);
    delay(1000);
}
```

এবার দেখো!

এলইডি রিংকিংয়ের মতো শব্দটাও রিংক করছে!

মানে পিক-পিক পিক-পিক এ রকম একটা সাউন্ড দিচ্ছে :-ও
চলো আরো কিছু মজার কাজ করি!

আরডুইনোতে **tone()** নামে একটা মজার জিনিস আছে। তুমি এর মধ্যে এভাবে ফিক্রোয়েন্সি লিখে দিলে সেই ফ্রিকোয়েন্সিতে বাজবে!

যেমন :

tone(12, 256);

এর মানে ১২ নম্বর পিনে লাগানো বাজার ২৫৬ হার্জ ফ্রিকোয়েন্সিতে সাউন্ড দেবে!

তুমি চাইলে সময় (**duration**) ঠিক করে দিতে পারো- এভাবে!

tone(12, 256, 1000);

এক সেকেন্ড বা এক হাজার মাইক্রো সেকেন্ড ধরে বাজবে!

আরেকটা করে দেখো, **tone(12, 128, 500);**

এই টোন ফাংশনটা এ রকম :

tone(pin number, frequency, time delay);

তাহলে কি আমরা আরডুইনো দিয়ে গান বাজাতে পারি?

অবশ্যই! আমরা জানি, শব্দ হচ্ছে এই ফ্রিকোয়েন্সির চেঙ্গ।

এবার তোমার নিজের চেষ্টা, এখানে সারেগামাপার ফ্রিয়োয়েন্সি দেওয়া আছে।

বাংলার সা রে গা মা পা ধা নি সা ইংলিশে যথাক্রমে **Do Re Mi Fa Sol La Si Do**

টোনগুলো পরপর বাজাও দেখি!

তুমি অবশ্যই ওপরের কোডটুকু নিজ হাতে টাইপ করবে, তারপর কম্পাইল করবে, **compile error** দেখলে আবার বইয়ের সাথে মিলিয়ে কোড ঠিক করবে। শুরু দিকে সেমিকোলন ‘;’ আর ব্রাকেট {**curly braces**} খুব বেশি ভুল হয় আমাদের!

এবার তুমি **noTone();** ফাংশনটা নিয়ে একটু চিন্তা করো, এর কাজ আসলে কী?

আমার বিশ্বাস তুমি অবশ্যই পারবে, যদি একটু চেষ্টা করো। তবুও বুঝতে অসুবিধা হলে ইউটিউব থেকে **play with arduino buzzer** সার্চ করে ভিডিও দেখে এসো তোমার মতোই অন্যান্য দেশের ছেলেমেয়েরা এই প্রজেক্ট কীভাবে করেছে।

মজার পরীক্ষা! এই কোডটা ধূমধাম চালিয়ে দাও! দেখো কী হয়!

```
int speakerPin = 9;
```

```
int length = 28; // the number of notes
```

```
char notes[] = "GGAGCkB GGAGdc GGxecBA yyecdc";
```

```
int beats[] = { 2, 2, 8, 8, 8, 16, 1, 2, 2, 8, 8, 8, 16, 1,  
2, 2, 8, 8, 8, 8, 16, 1, 2, 2, 8, 8, 8, 16 };
```

```
int tempo = 150;
```

```
void playTone(int tone, int duration) {
```

```
for (long i = 0; i < duration * 1000L; i += tone * 2) {
```

```
    digitalWrite(speakerPin, HIGH);
```

```
    delayMicroseconds(tone);
```

```
    digitalWrite(speakerPin, LOW);
```

```
    delayMicroseconds(tone);
```

```
}
```

```
}
```

```
void playNote(char note, int duration) {
```

```
char names[] = {'C', 'D', 'E', 'F', 'G', 'A', 'B',
```

```
'c', 'd', 'e', 'f', 'g', 'a', 'b',
```

```
'x', 'y' };
```

```
int tones[] = { 1915, 1700, 1519, 1432, 1275, 1136,
```

```
1014,
```

```
956, 834, 765, 593, 468, 346, 224,
```

```
655 , 715 };
```

```
int SPEE = 5;
```

```
// play the tone corresponding to the note name
```

```
for (int i = 0; i < 17; i++) {
```

```
if (names[i] == note) {
```

```
int newduration = duration/SPEE;
```

```
playTone(tones[i], newduration);
```

```
}
```

```
}
```

```
}

void setup() {

pinMode(speakerPin, OUTPUT);

}

void loop() {

for (int i = 0; i < length; i++) {

if (notes[i] == ' ') {

delay(beats[i] * tempo); // rest

} else {

playNote(notes[i], beats[i] * tempo);

}

// pause between notes

delay(tempo);

}

}
```

কী এত বড় প্রোগ্রাম দেখে ভয় পেয়ে গেলে?

এই লিংকে গিয়ে কপি করে পেস্ট করে দাও :

www.tiny.cc/buzzer1

তুমি যদি কোডটা রান করে থাকো, তুমি দেখবে সেখানে একটা টিউন
বাজছে। তোমার কোনো বন্ধুর জন্মদিন এ মাসে থাকলে তুমি ওর জন্য
এই প্রজেক্টটা বানাতে পারো। আর প্রজেক্টের একটা ভিডিও বানিয়ে ওকে
দিতে পারো।

ছয় রোবটের সুইচ

সুইচ আমরা সবাই দেখেছি। আমাদের সবার বাসাতেই আছে। সুইচ দিয়ে আমরা কোনো কিছু চালু-বন্ধ করি।

তোমরা একটু ভেবে দেখো তোমার বাসায় পানির কলও কিন্তু একধরনের সুইচ।

পাইপলাইনে থাকে পানি। এখন একটা সুইচ দিয়ে আমরা পানির আসাযাওয়া কন্ট্রোল করি। চিন্তা করো বাসায় যদি শুধু পাইপ থাকত আর সুইচ না থাকত তাহলে কী হতো!

ট্যাংকের সব পানি পড়ে ট্যাংক খালি হয়ে যেত!

বাসার কারেটের লাইনের সুইচেরও ঠিক একই রকম কাজ। সুইচ দিয়ে আমরা আমাদের ঠিক যে সময়ে দরকার, সেই সময়ে ফ্যান লাইট জ্বালাই। আর অন্য সময় বন্ধ রাখি।

এভাবে আমরা কারেন্টকে নিয়ন্ত্রণ করছি।

এখানে আমরা সার্কিটকে নিয়ন্ত্রণ করতে একটা সুইচ ব্যবহার করব। তবে এই সুইচ হচ্ছে সাইজে অনেক ছোট। এই সুইচ ঠিক ততক্ষণ কাজ করে যতক্ষণ একে চেপে রাখা হয়।

আমাদের ছোট সেই বাটন হচ্ছে পুশবাটন! মানে যাকে পুশ (**push**) করা যায়। আরডুইনো এবং পুশবাটন (**push-button**) দিয়ে অনেক মজার কাজ করা যায়!



চিত্র ৬.১ : ছোট-বড় হরেক রকম পুশবাটন

চলো আমরা পুশবাটনের ভেতরটা একটু দেখে নিই। এটা আসলে একটা সুইচ। পুশবাটন অনেক রকমের আর অনেক সাইজের হয়ে থাকে। ওপরের ছবি দেখে তোমরা নিশ্চয়ই তা বুঝেছ। কাজের প্রয়োগ অনুসারে এটাকে ব্যবহার করা হয়।

এই সুইচটা মূলত আমরা যতক্ষণ চেপে থাকি ততক্ষণ কাজ করে।

এবার বলো দেখি, আমাদের জীবনে এ রকম চেপে থাকা সুইচের উদাহরণ কী কী?

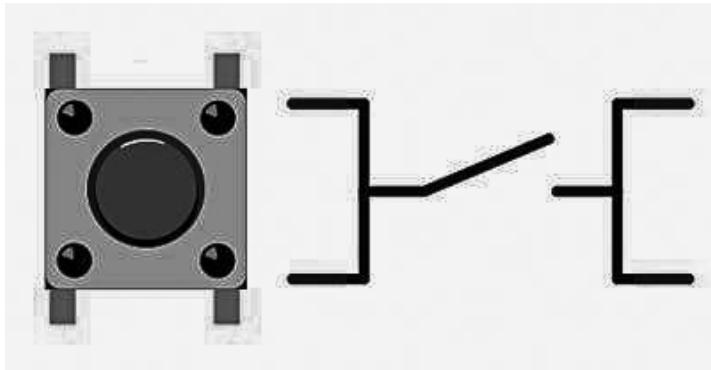


চিত্র ৬.২ : পুশবাটন কাছ থেকে দেখি

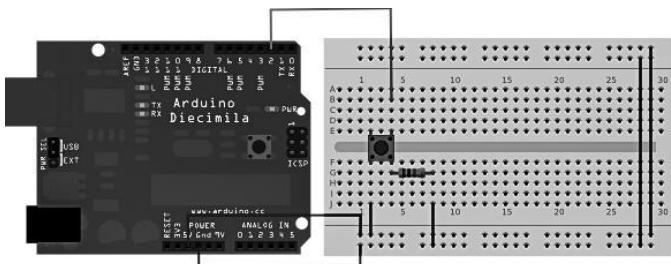
কলেজে, কম্পিউটারের কিবোর্ড, মোবাইলের পাওয়ার বাটন- এগুলোও কিন্তু একধরনের পুশবাটন।

রোবটের সুইচ

এই পুশবাটনে আসলে দুই পাশের দুই পিন সংযুক্ত করা থাকে আর দুই পিনের মাঝে কানেকশন থাকে না। পুশবাটনে চাপ দিলে ওই কানেকশনটা চালু হয়।

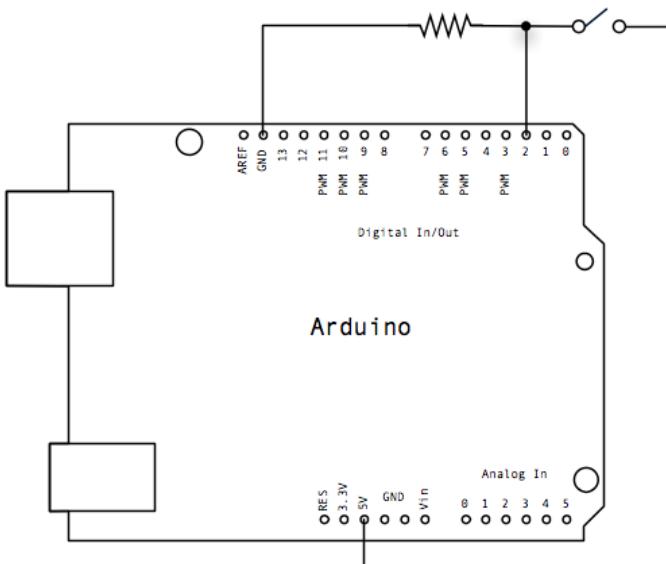


চিত্র ৬.৩ : পুশবাটনের ভেতরে যেভাবে কাজ হয়



চিত্র ৬.৪ : আরডুইনোর সাথে পুশবাটন পরীক্ষা

ছবির মতো আমরা আরডুইনোর সাথে পুশবাটন লাগিয়ে কানেকশন দেব। এখানে আমরা আরডুইনোর ২ নম্বর পিনের সাথে পুশবাটনের এক অংশ জোড়া দিয়েছি। পুশবাটনের অন্য অংশ একটা ২২০ ওহমের রোধের সাহায্যে গ্রাউন্ডে আর আরেক অংশ ৫ ভোল্ট অংশে দিয়েছি।



চিত্র ৬.৫ : আরডুইনো-পুশবাটন কানেকশনের একটি সরল চিত্র

```

int buttonPin = 2;
int buttonState = 0;

void setup() {
    Serial.begin(9600);
    pinMode(buttonPin, INPUT);
}

void loop() {
    buttonState = digitalRead(buttonPin);
    Serial.print(buttonState);
}

```

এই কোডে প্রথম দুই লাইন দিয়ে বুবিয়ে দেওয়া হয়েছে কোনটা বাটন পিন আর আরেকটা ভেরিয়েবলে বাটনের অবস্থা রাখা হয়েছে। (state মানে অবস্থা বা দশা)

```

int buttonPin = 2;
int buttonState = 0;

```

রোবটের সুইচ

তারপর আমরা সবাই জানি সেটিওপের মধ্যে এই কানেকশনগুলো গুচ্ছিয়ে লেখা হয় যাতে কম্পিউটার বুঝতে পারে। এখানে

```
Serial.begin(9600);  
pinMode(buttonPin, INPUT);
```

Serial.begin-এর দ্বারা আমরা আরডুইনোকে কম্পিউটারের সাথে যোগাযোগের অনুমতি দিচ্ছি। আমরা যে ইউএসবি (USB) কেবল দিয়ে কম্পিউটারের সাথে আরডুইনো লাগিয়ে রেখেছি, এখন এদের মধ্যে কমিউনিকেশন বা যোগাযোগ হচ্ছে।

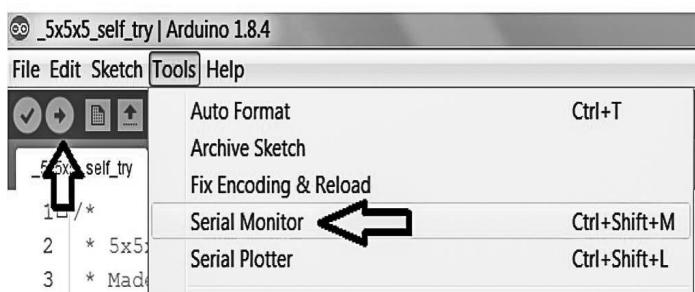
আর ৯৬০০ মানে প্রতি সেকেন্ডে ৯৬০০ বার যোগাযোগ হচ্ছে এদের মধ্যে।

pinMode দিয়ে আমরা কম্পিউটারকে বোঝাচ্ছি যে এটা একটা ইনপুট হিসেবে পাঠানো হচ্ছে।

তোমরা নিশ্চয় ইনপুট-আউটপুট বুঝে গেছ? তবুও আবার বলি। কম্পিউটারের কিবোর্ড মাউস হলো ইনপুট ডিভাইস আর মনিটর বা সাউন্ড বক্স হচ্ছে আউটপুট ডিভাইস।

কোডটা আপলোডের পর তোমার কাজ হবে টুলস -> সিরিয়াল মনিটর ওপেন করা।

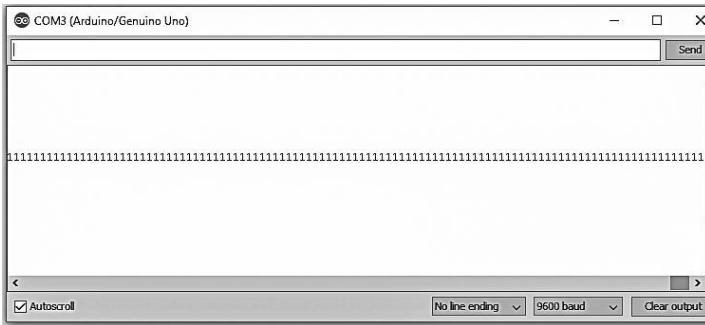
তাহলে তুমি দেখবে সেখানে একটা উপাত্ত (**Data**) যাচ্ছে শূন্য বা এক।



চিত্র ৬.৬ : টুলস অপশনে গিয়ে সিরিয়াল মনিটর চালু করা

এবার বাটন চেপে ধরো। দেখো তোমার উপাত্ত (**Data**) পরিবর্তন হয়েছে!

ডিজিটাল রিড `digitalRead()` আসলে ডেটার মধ্যে মান কত আছে পড়ছে, আর প্রিন্ট `Serial.print()` সেটা মনিটরে প্রিন্ট করছে বা দেখাচ্ছে।



চিত্র ৬.৭ : সিরিয়াল মনিটরে আরডউইনো থেকে প্রাপ্ত ডেটা সরাসরি দেখা যায়

যেহেতু আমরা প্রতিটা অবস্থার (state) মান জানি, তাই সহজে আমরা ওই অবস্থার জন্য কোড লিখে প্রয়োজনমতো কাজ করাতে পারি।

ଆମରା ଏଥିନ ବାଟନ ଚେପେ ସତକ୍ଷଣ ଥାକବ, ତତକ୍ଷଣ ଏଲଇଡ଼ି ଜୁଲବେ ଏ ରକମ କିଛୁ ଏକଟା ବାନାବ ।

নিচের কোডটা দেখো। পাশে লাইনে কমেন্ট লেখা আছে কোনটা দিয়ে
কি হচ্ছে।

```
int buttonPin = 2;      //the number of the  
pushbutton pin  
  
int LEDPin = 13;       //the number of the LED pin  
  
int buttonState = 0;    //this variable will  
change value  
  
  
void setup() {  
    // initialize the LED pin as an output  
    pinMode(LEDPin, OUTPUT);  
    // initialize the pushbutton pin as an input  
    pinMode(buttonPin, INPUT);  
}  
  
  
void loop() {  
    // read the state of the pushbutton value:  
    buttonState = digitalRead(buttonPin);  
  
    // check if the pushbutton is pressed.  
    // If it is, the buttonState is HIGH:  
    if (buttonState == HIGH) {  
        // turn LED on:  
        digitalWrite(LEDPin, HIGH);  
    }  
    else {  
        // turn LED off:  
        digitalWrite(LEDPin, LOW);  
    }  
}
```

এখানে ‘if’-এর মধ্যে আমরা চেক করছি কখন সুইচের মান ১।

‘if’ দিয়ে কম্পিউটার বোঝে যদি আর

‘else’ দিয়ে বোঝে যদি এটা না হয় তবে।

ধরো তুমি একটা রোবটকে বললে, তুমি এখান থেকে একটা যায়গায় যাবে। ধরো রংপুর। তারপর তোমার রোবটকে বলে দিলে, যদি সেখানে যাওয়ার পর তুমি একটা খাবারের দোকান পাও সেখানে যাবে। সেই দোকানে যাওয়ার পর তুমি যদি সেখানে শিঙাড়া পাও তাহলে শিঙাড়া থাবে, আর যদি না পাও তাহলে আলুর চপ থাবে।

এখানে যদি দিয়ে রোবটকে লজিক বোঝানো হচ্ছে।

```
void setup() {
    রংপুর যাও;
    if(খাবারের দোকান == হ্যাঁ) {
        দোকানের ভেতর যাও;
    }
}

void loop() {
    if(খাবার == শিঙাড়া) {
        শিঙাড়া খাও;
    }
    else {
        আলুর চপ খাও; // এলস - মানে যদি শিঙাড়া না পাও তবে আলুর
        চপ খাও
    }
}
```

লুপের ভেতর ফেলে দেওয়ায় বেচারা রোবট খেতেই থাকবে খেতেই থাকবে

এবার বলো তো আরভুইনো কোডে ‘if’ কীভাবে কাজ করছে?

ঠিক ধরতে পেরেছ আশা করি!

যদি আমরা সুইচ চেপে থাকি, তবে বাতি জুলে থাকবে, নয়তো জুলবে না।

রোবটের সুইচ

‘তুমি অবশ্যই পুরো কোড নিজ হাতে লিখবে। শুরুর দিকে কোডের কিছু অংশ তুমি বুবাবে, কিছু অংশ বুবাবে না। এখানে হতাশ হয়ে ছেড়ে দিয়ো না কখনোই।

আমরা ধীরে ধীরে একসময় সব বুবো যাব।’

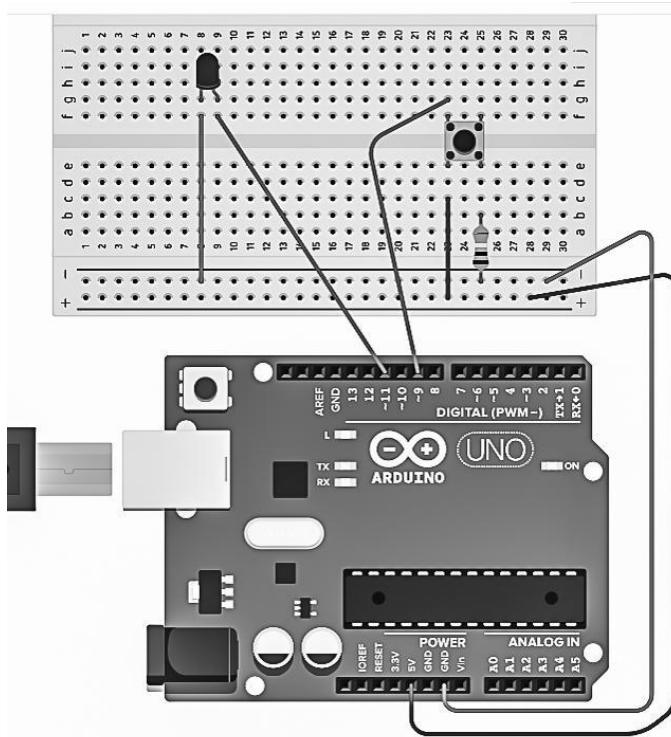
চেষ্টা ১ : একই পদ্ধতিতে পুশবাটন চেপে বাজার বাজাও

চেষ্টা ২ : আটটি পুশবাটন দিয়ে **tone()** ফাংশন ব্যবহার করে পিয়ানো বা হারমোনিয়াম বানাও

এবাবে আমরা আরও একটা মজার প্রজেক্ট করব
একে আমরা বলব এলইডি ফেডিং বা এলইডি ডিমিং (**LED fade/dim with Push Button**)

কাজটা খুব সহজ - পুশবাটন একবার চাপলে এলইডি মৃদু একটু জ্বলবে, আরেকবার চাপলে আরেকটু বেশি উজ্জ্বল জ্বলবে, আরেকবার চাপলে আরেকটু বেশি এভাবে একবার সর্বোচ্চ আলোতে জ্বলবে।
আরেকবার পুশবাটন চাপলে নিভে যাবে।

এই প্রজেক্টে তোমার প্রয়োজন হবে— আরডুইনো, ৱেড বোর্ড, জাম্পার ওয়্যার, এলইডি, পুশবাটন এবং ২২০ ওহ্ম রেজিস্ট্রার। এবাব ঝাটপট ডায়াগ্রাম দেখে তোমার আরডুইনোতে সবকিছু যুক্ত করো।



চিত্র ৬.৮ : এলইডির উজ্জ্বলতা কমবেশি করার আরডুইনো সার্কিট ডায়গ্রাম

পুশবাটনের সাথে ৫ ভোল্ট যুক্ত করো। পুশবাটনের সাথে একটি ২২০ ওহম রেজিস্টর গ্রাউন্ড করে যুক্ত করো। পুশবাটনের অপর প্রান্ত থেকে একটি তার আরডুইনোর ১১ নম্বর পিনে ইনপুট দাও।

আরডুইনো থেকে ১১ নম্বর পিন আউটপুট নিয়ে এলইডির অ্যানোডে যুক্ত করো। এলইডির ক্যাথডে গ্রাউন্ড যুক্ত করো।

এবার নিচের কোড আইডিইতে লিখে ফেলো—

রোবটের সুইচ

```
int led = 11;
int button = 9;

int buttonState = 0;
int brightness = 0;
int brightup = 20;

void setup() {
pinMode(led, OUTPUT);
pinMode(button, INPUT);
}

void loop() {
analogWrite(led, brightness);

buttonState = digitalRead(button);

if ( buttonState == 1 ) {
brightness = brightness + brightup;
}
if ( brightness >= 255 ) {
brightness = 0;
}
delay(30);
}
```

কোডের অনেক অংশেই হয়তো তোমার বুবাতে অসুবিধা হচ্ছে। তোমার সুবিধার জন্য পুরো কোডের সাথে কমেন্ট জুড়ে দিচ্ছি-

```

int led = 11; // পিন ১১ তে এলইডি যুক্ত
int button = 9; //পিন ৯-এ পুশবাটন

int buttonState = 0; //বাটনের স্টেট বা অবস্থা, মানে বাটন চাপা হলে ১
হয়ে যাবে, না চাপলে ০ থাকবে

int brightness = 0; // প্রথমে ধরে নিলাম, উজ্জ্বলতা শূন্য
int brightup = 20; // ধীরে ধীরে ২০ করে উজ্জ্বলতা বাড়াব। তুমি এখানে
২, ৫, ১০ বিভিন্ন মান দিয়ে একাপেরিমেন্ট করবে
void setup() {
pinMode(led, OUTPUT); // পিন ১১ হলো আউটপুট
pinMode(button, INPUT); // পিন ৯ হলো ইনপুট
}

void loop() {
analogWrite(led, brightness); // অ্যানালগ রাইট ফাংশনকে
ডেকে বলা হচ্ছে এলইডিকে ব্রাইটনেসের বর্তমান মান অনুযায়ী জ্বালাতে

buttonState = digitalRead(button); // পুশবাটন চাপা
হয়েছে কি না, সেই স্টেট/অবস্থা জানতে ডিজিটাল রিডকে কল করা হয়েছে

if ( buttonState == 1 ) {
brightness = brightness + brightup; // যদি বাটন চাপা
হয়, মানে স্টেট ১ - তখন ব্রাইটনেসের মান একটু বাড়িয়ে দাও। ব্রাইটআপ হলো ব্রাইটনেস
আগের চেয়ে কতটুকু বাঢ়বে।
}

if ( brightness >= 255 ) {
brightness = 0; // যখন ব্রাইটনেস সর্বোচ্চ লেভেলে চলে যাবে, মানে ২৫৫-
এর সমান বেশি হয়ে যাবে, তখন ব্রাইটনেস আবার জিরো করে বাতিটা নিভিয়ে দাও
}
delay(30); // কিছুক্ষণ ডিলে দিলে আলো ধীরে ধীরে বাড়বে। ডিলে না দিয়ে
দেখো তো কী হয়!
}

```

খেয়াল করো, **digitalRead(button);**
পুশবাটন চাপা হলে ডিজিটাল রিড থেকে ১ রিটার্ন করা হয়। তখন ‘buttonState’-
এর মান হয়ে যায় ১। আর চাপা না হলে বাটন স্টেট শূন্য থাকে।

আবার,

brightness = brightness + brightup;

এই লাইনটা তোমাদের কি একটু বিদ্যুটে লাগছে?

সহজভাবে লিখলে,

বর্তমান ব্রাইটনেস = আগের ব্রাইটনেস + যতটুক বাড়াব;

আমরা অ্যানালগ রাইট নামের একটি ফাংশন ব্যবহার করেছি।

analogWrite(led, brightness);

সহজ করে লিখলে, অ্যানালগথরাইট (এলইডির পিন নম্বর, ব্রাইটনেসের মান);

ডিজিটাল রাইট ব্যবহার করে আসলে এই ফেডিংয়ের কাজটি করে উজ্জ্বলতা
কমবেশি করা যেত না। আরডুইনোতে পিডব্লিউএম (**PWM - Pulse Width
Modulation**) নামের একটা বিষয় আছে। ডিজিটাল রাইট দিয়ে কেবলমাত্র
জিরো ভোল্ট অথবা ৫ ভোল্ট বিদ্যুৎ দেওয়া যায়। কিন্তু পিডব্লিউএম ব্যবহার করে
০ ভোল্ট, ২ ভোল্ট, ৩.৫ ভোল্ট, ৪.২ ভোল্ট এ রকম মানের বিদ্যুৎ প্রবাহ দেওয়ার
মতন একটা সিমুলেশন করা যায়। মানে, পালস উইড মড্যুলেশন পদ্ধতিতে
ডিজিটাল সিগনালকে অ্যানালগ সিগনালের মতন কাজ করানো যায়। আমরা এত
জটিলে যাব না। তুমি শুধু মনে রাখবে, যখন আলো কমবেশি করতে চাই, মোটরের
স্পিড কমবেশি করতে চাই, সার্ভো চালাতে চাই, সেই কাজগুলোয় পিডব্লিউএম পিন
দরকার হয়।

আরডুইনো উনো মডেলে ৩, ৫, ৬, ৯, ১০, ১১ পিনে দেখবে পাশে একটা ঢেউ
চিহ্ন () দেওয়া। এই চিহ্ন দিয়ে বুঝিয়েছে যে এই পিনগুলোতে পিডব্লিউএম সুবিধা
দেওয়া আছে।

পিডব্লিউএমের রেঞ্জ ০ - ২৫৫। তাই আমরা অ্যানালগ রাইট ফাংশনে ব্রাইটনেসের
মান শূন্য থেকে ২৫৫-এর ভেতর বসাতে পারব।

পিডব্লিউএম নিয়ে আসলে অনেক কিছু জানার আছে। সবকিছু এই একটা বইয়ের
আওতায় লিখে ফেলা দুরহ। আমরা তোমাকে এই বইয়ের পাশাপাশি নিয়মিত গুগল
করে শেখার উৎসাহ দেব। তাহলে গুগল করে ফেলো **PWM - Pulse Width
Modulation** লিখে।

চেষ্টা : ৩

তুমি কি এখন এমন টা সিস্টেম বানাতে পারবে যে - পুশবাটন একবার চাপলে লাল এলইডি, আরেকবার চাপলে নীল এলইডি, আবার চাপতে সাদা এলইডি জ্বলবে! তুমি এভাবে একটা ট্রাফিক রোড লাইট কন্ট্রোল বানাতে চেষ্টা করো। বানিয়ে ফেলতে পারলে আমাদের ই-মেইল করবে।

তোমার সুবিধার জন্য একটা সার্কিটা ডায়াগ্রাম দিয়ে দিলাম। কোডটুকু তুমি প্রথমে চিন্তা করবে। একটা খাতা-কলম নিয়ে ইফ-এলস লজিকে পুশবাটনের সাথে এলইডির জ্বলা-নেভা লিখবে।

আরেকটু হিনটস দেব?

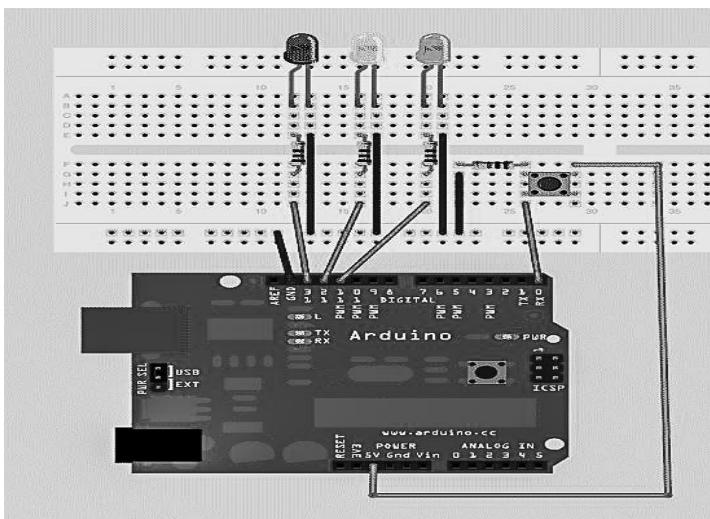
যদি পুশবাটন একবার প্রেস করি,
এলইডি-১-এ ডিজিটাল রাইট হাই করবে,
এলইডি-৩-এ ডিজিটাল রাইট লো করবে,

পুশবাটন আবার প্রেস করলে,
এলইডি-২-এ ডিজিটাল রাইট হাই করবে,
এলইডি-১-এ ডিজিটাল রাইট লো করবে,

আবার প্রেস করলে,
এলইডি-৩-এ ডিজিটাল রাইট হাই করবে,
এলইডি-২-এ ডিজিটাল রাইট লো করবে,

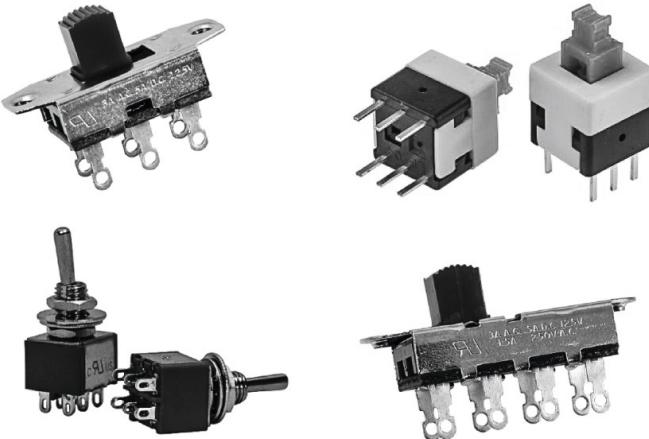
তোমার থেজেস্টি নিজে করা হলে, এই লিংকে গিয়ে এর চেয়ে আরেকটু ডেভেলপ করার বুদ্ধি নিয়ে আসবে -

www.makeuseof.com/tag/arduino-traffic-light-controller/



চিত্র ৬.৯ : আরডুইনো দিয়ে ট্রাফিক লাইট কন্ট্রোল নিজে বানাও

এই পুরো অধ্যায়ে আমরা পুশবাটনের বহুবিধি ব্যবহার দেখেছি। কিন্তু আমরা শুরুতে তোমাকে শুধু এক প্রকারের পুশবাটন দেখিয়েছি। তুমি কি জানো, এই বাটন বা সুইচ আরও অনেক প্রকারের আছে?



চিত্র ৬.১০ : হয়েক রকম বাটন ও সুইচ

সাত

চলো বানাই চোর ধরার অ্যালার্ম

তোমরা যারা খুদে বিজ্ঞানী, তারা কখনো না কখনো অবশ্যই তোমার বাড়িতে একটা অ্যালার্ম সিস্টেম বানানোর কথা কল্পনা করেছ। আজকে আমরা তোমার সেই চিন্তাটিকে বাস্তবে বানাব। আমরা এমন একটা অ্যালার্ম সিস্টেম বানাব যা ঘরের মধ্যে কোনো কিছু নড়াচড়া করলেই অ্যালার্ম দেবে। ফলে ঘরে কেউ প্রবেশ করলেই তুমি বুঝতে পারবে।

সবচেয়ে দারণ কথা কি জানো, এই সিস্টেম শুধু দিনের আলোতে নয়, অন্ধকারের ভেতরেও ঠিকঠাক বুঝতে পারে ঘরে কেউ ঢুকেছে কি না!!

তাহলে এবার তুমি তোমার প্রজেক্টের বাস্তু নিয়ে বসে পড়ো প্রজেক্ট বানাতে!

আমরা এমন একটা ডিভাইস বানাতে চাই যা কোনো কিছু নড়াচড়া করলে বুঝতে পারবে অর্থাৎ '**motion detect**' করতে পারবে। এ জন্য একধরনের সেন্সর পাওয়া যায়, নাম পিআইআর মোশন সেন্সর (**PIR motion sensor**)। **PIR** কথাটার পূর্ণরূপ হয় **Passive Infrared**.

তুমি অবলোহিত আলো (Infrared Light বা IR) সম্পর্কে তো পড়েছ?

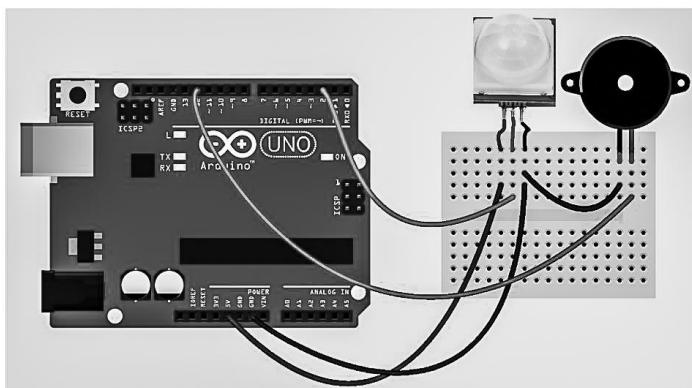
ইনফ্রারেড হলো আমাদের দৃশ্যমান আলোর চেয়েও কম তরঙ্গ-দৈর্ঘ্যের আলো। এটাকে আমরা খালি চোখে দেখতে পারি না। যেকোনো উষ্ণ বা গরম বস্তু থেকে এই ইনফ্রারেড তরঙ্গ নির্গত হয়। তার মানে আমাদের শরীর থেকেও কি প্রতিনিয়ত ইনফ্রারেড তরঙ্গ বের হচ্ছে? উত্তর- হ্যাঁ।

ইনফ্রারেড নিয়ে আমি আর কিছু বলব না। কারণ আমি বিশ্বাস করি তোমার মাঝে আগ্রহ থাকলে তুমি সেটা গুগল করে জেনে নেবে।

এবার তোমাকে সবচেয়ে মজার কথা বলি! আমরা তো আইআর খালি চোখে দেখি না। কিন্তু আমাদের এই পিআইয়ার সেন্সর ইনফ্রারেড তরঙ্গের পরিবর্তন বুঝতে পারে। এ জন্য ঘরে কেউ প্রবেশ করলেই তার শরীর নির্গত ইনফ্রারেড দেখে এই সেন্সর আরডুইনোকে সংকেত পাঠায়।

আরডুইনো তখন এই তথ্যের ভিত্তিতে বাজার অ্যালার্ম বাজিয়ে দেয়।

অনেক তো হলো তত্ত্বীয় কথাবার্তা। চলো আমরা আগে আমাদের জিনিসটা প্র্যাকটিক্যালি বানাই। তারপর নাহয় আমরা এসব নিয়ে আরেকটু পড়াশোনা করব।



চিত্র ৭.১ : আরডুইনোর সাথে পিআইআর মোশন সেন্সর এবং বাজার যুক্ত করা

এই ছবিটির মাধ্যমে খুব সহজেই আমরা বুঝে যাই এই প্রজেক্টে আমরা কীভাবে বানাব। চলো তাহলে তুমি আর আমি মিলে আরডুইনোতে কানেকশনগুলো দিয়ে ফেলি।

প্রথমে আমরা **Arduino, PIR Sensor, Piezo Buzzer, Jumper wires, bread board** সবকিছু হাতের কাছে নিয়ে বসব।

ব্রেড বোর্ডে সেঙ্গরটি বসাব। খেয়াল করো, পিআইআর সেঙ্গরের তিটি পা, একটি ‘VCC’, একটি ‘Ground’ আর একটি ‘Output pin’

যেকোনো সার্কিট বসানোর প্রথম কাজ হলো, যদি আমি পাওয়ার সাপ্লাই দিয়ে থাকি সেটি বন্ধ করা। তুমি যদি আরডুইনোতে পাওয়ার দিয়ে থাকো, তবে পাওয়ার ক্লিব খোলো।

তুমি আগেও দেখেছ, আরডুইনোতে জাম্পার ওয়্যার লাগানোর জন্য কিছু পিন থাকে, সেগুলোতে আবার ছোট করে লেখাও থাকে কোনটি কোন পিন। আমরা প্রথমেই আমাদের PIR Sensor ব্রেড বোর্ডে বসাব। বসানোর সময় দেখব, এর তিনটি পা রয়েছে - **VCC, GROUND, OUTPUT**. আমরা সাবধানে একে ব্রেড বোর্ডে বসাব। অথবা ফিমেল টু মেল জাম্পার ওয়্যার ব্যবহার করব। একটু বুদ্ধি করে বসাতে হবে, জানি তুমি পারবে।

ব্রেড বোর্ডে কম্পোনেন্ট বসানো নিয়ে এখনো অসুবিধা হচ্ছে কি? তাহলে বাটপট এই বইয়ের ব্রেড বোর্ড নিয়ে লেখা চ্যাপ্টারে আরেকবার চোখ বুলিয়ে এসো!

পিআইআর সেঙ্গরের ‘Vcc’-কে আমরা আরডুইনোর ৫ ভোল্ট পিনে কানেক্ট করব, আর ‘GND’কে আরডুইনোর গ্রাউন্ড পিনের সাথে। এই দুই পিনের মাধ্যমে আমরা সেঙ্গরকে চলার মতো বিদ্যুৎ দিচ্ছি। ‘VCC’ পিনকে অনেকে পজেটিভ ভোল্টেজ আর গ্রাউন্ডকে নেগেটিভ বলে থাকে।

এবার আমরা সেঙ্গরের আউটপুট (**OUT**) পিন থেকে জাম্পার ওয়্যারের মাধ্যমে আরডুইনোর ২ নং পিনের সাথে যুক্ত করব।

ব্যস, তুমি আমাদের প্রজেক্টের অর্ধেক কাজ করে ফেলেছ! আমরা চোর ধরার অ্যালার্মের সেন্সর বিসিয়ে ফেলেছি। এবার আমরা শব্দসংকেত দেওয়ার জন্য বাজার বসাব।

তুমি বাজারটা হাতে নাও! দেখো তো একটা প্লাস (+) আরেকটা মাইনাস (-) চিহ্ন আছে কি না? প্লাস (+ve) চিহ্নটা হলো ‘Vcc’ যেখানে আমরা কারেন্ট দেব। তাহলে বাটপাট প্লাস (+) চিহ্নযুক্ত পা (lead / leg)-এর সাথে আরডুইনোর ১০ নং পিন জাম্পার ওয়্যারের মাধ্যমে যুক্ত করে দাও।

কারো কারো বাজারে হয়তো + বা - চিহ্ন নেই। তাহলে উপায়?

একটু বুদ্ধি করে দেখো, একটা পা ছোট, আরেকটা বড়! এই বড় পা বা অ্যানোডে ব্যাটারির পজেচিভ প্রান্ত যুক্ত হয়, তার মানে, বড় পা হচ্ছে পজেচিভ বা ‘Vcc’

এই পা তবে আরডুইনোর ১০ নং পিনে যুক্ত করব।

আমি যদি বলে নাও দিই, তবুও তুমি নেগেচিভ বা ছোট পা ‘GND / Ground’-এর সাথে যোগ করে দেবে।

আমাদের প্রজেক্টের দেহ প্রস্তুত! এবার শুধু আমরা কোডের মাধ্যমে আজ্ঞাটা এর ভেতর দুকিয়ে দেব। তুমি এখন তোমার কম্পিউটারে আরডুইনো আইডিই ওপেন করবে এবং আমার সাথে সাথে নিচের কোডটুকু টাইপ করবে।

প্রথমে আমরা তিনটি নাম সেট করে নেব। এই তিনটি নাম ধরেই আমরা পুরো কোডের লজিকটা বানাব।

তাহলে আমরা নাম তিনটি ধরব-

```
int PIRsensor = 2;  
int buzzer = 10;  
int thief = 0;
```

‘int’ মানে হলো ‘integer’ বা পূর্ণ সংখ্যা। এই যেমন ২, ১০ কিংবা ০ হলো পূর্ণ সংখ্যা। এটা কেন ব্যবহার করছি এটা নিয়ে আপাতত চিন্তা করার দরকার নেই।

আমরা তিনটি নাম নিয়েছি, সেপরের আউটপুট থেকে যেই তার আরডুইনোতে ইনপুট হিসেবে আসবে, সেটার নাম দিয়েছি ‘PIRsensor’ আর একে আমরা ২ নং পিনে লাগাব, তাই এর মান ধরেছি ২।

যে তারটি আরডুইনো থেকে আউটপুট হিসেবে বের হয়ে বাজারের বড় পায়ের সাথে যুক্ত হবে, সেটার নাম দিয়েছি ‘buzzer = 10’, এর মান ১০ ধরার মানে হলো আমরা আরডুইনোর ১০ নং পিনে buzzer লাগাব।

এবার আমরা আরেকটা নাম দেব ‘thief’, মানে চোর আছে কি না, এটা জানার জন্য।

```
thief = 0;
```

চোরের মান শূন্য ধরার মানে হলো চোর নেই, আর যখন চোর আসবে এই মান সেপরের মাধ্যমে thief = 1; করে দেওয়া হবে।

আরেকটু গল্প বলি,

কম্পিউটার জগতে এই ০ আর ১ দিয়েই সব কাজ হয়। ০ এর মানে হলো ভোল্টেজ ‘LOW’, আর ১-এর মানে হলো ভোল্টেজ ‘HIGH’

আরেকভাবে বলা যায়, ০ হলো ‘false’ আর ১ হলো ‘true’

আগের সব প্রজেক্টে তুমি অনেক বার setup() ফাংশন লিখেছ। এখন আরেকবার ঘটপট লিখে ফেলো-

```
void setup()
{
    pinMode(PIRsensor, INPUT);      // declare sensor
    as input
    pinMode(buzzer, OUTPUT);       // declare buzzer
}
```

সেটআপ ফাংশনের ২য় বর্কনীর (**curly braces**) মাঝে আমরা বলে দিয়েছি কোন নামের পিন কী হিসেবে কাজ করবে।

pinMode() হলো আরেকটা ফাংশন, যে বলে দেয়- কে ইনপুট হবে আর কে আউটপুট।

চলো বানাই চোর ধরার অ্যালার্ম

পিআইআর সেন্সরের আউটপুট থেকে যে তার আসছে, সেটা হবে আরডুইনোর জন্য ইনপুট । .

আর আরডুইনো থেকে বাজারে আউটপুট হিসেবে তার যাবে ।

একটা জিনিস খেয়াল করেছ কি?

কোড লেখার সময় বন্ধনী চিহ্ন, সেমিকোলন, ক্যাপিটাল লেটার, অ্যাল লেটার এসব খেয়াল রাখতে হয় । **pinMode()**-এর জায়গায় **pinmode()** লিখলে কাজ করবে না ।

এবার আমরা লুপের ভেতরের কোডটুকু বাটপট লিখে ফেলব-

```
void loop()
{
    thief = digitalRead(PIRsensor); // read input
    value

    if (thief == 1)
    {
        digitalWrite(buzzer, HIGH);
        delay(2000);
        digitalWrite(buzzer, LOW);
        delay(1000);
    }
    else
    {
        digitalWrite(buzzer, LOW);
    }
}
```

কী সব হিজিবিজি লেখা তাই না?

আমাদের বিশ্বাস, একটু চিন্তা করলেই তুমি এই কোডের লজিকটুকু বের করে ফেলবে । চলো আমরা কোডটিকে ব্যবচ্ছেদ করি ।

```
void loop()

//আমরা লুপের ভেতরে যা লিখব মাইক্রোকন্ট্রোলার তা বারবার করবে

{
    thief = digitalRead(PIRsensor);
```

// ডিজিটাল রিড ফাংশন আমাদের সেঙ্গে থেকে জেনে নেবে যে চোর
আছে কি না

```
if (thief == 1)
```

// যদি চোর থাকে তখন সেঙ্গে ১ পাঠায়। আর তখন আমরা বাজার
বাজাই।

খেয়াল করেছ কি, এখানে সমান == চিহ্ন দুবার দেওয়া আছে == অর্থাৎ
সমান সমান।

```
{
```

```
    digitalWrite(buzzer, HIGH);
```

// যদি চোরের মান == ১ হয় আমরা বাজারে ভোল্টেজ ছাই পাঠাব।
এ জন্য আমরা ডিজিটাল রাইট ফাংশন ব্যবহার করলাম।

```
    delay(2000);
```

```
}
```

```
else
```

```
{
```

```
    digitalWrite(buzzer, LOW);
```

// যদি চোরের মান == ১ না হয়, তখন আমরা আবার ডিজিটাল রাইট
করে বাজারের ভোল্টেজ লো করে দেব।

```
}
```

```
}
```

তোমার সুবিধার জন্য আমি পুরো কোড পরিষ্কারভাবে আরও একবার
লিখে দিচ্ছি। বন্ধনী, সেমিকোলন, ক্যাপিটাল/ স্মল লেটার ঠিকমতো
দেখে বাটপট টাইপ করো। আবার খেয়াল রেখো, কোথায় = আর
কোথায় == দেওয়া!!!

চলো বানাই চোর ধরার অ্যালার্ম

```

int PIRsensor = 2;
int thief = 0;
int buzzer = 10;

void setup()
{
    pinMode(PIRsensor, INPUT);
    pinMode(buzzer, OUTPUT);
}

void loop()
{
    thief = digitalRead(PIRsensor);
    if (thief == 1)
    {
        digitalWrite(buzzer, HIGH);
        delay(2000);
    }
    else
    {
        digitalWrite(buzzer, LOW);
    }
}

```

শেষের দিকের বন্ধনীগুলো খুব মিস হয়, একটু খেয়াল রেখো।

কোড কম্পাইল ও আপলোডের পর একটু অপেক্ষা করবে। এই সেন্সর কাজ শুরু করতে ৩০-৫০ সেকেন্ডের মতো সময় নেয়। শুরূর এই সময়টুকু সেন্সরের সামনে কোনো কিছু নড়াচড়া করাবে না।

আমাদের প্রজেক্ট তো শেষ হলো, এবার চলো আমরা দেখে আসি পৃথিবীর অন্যান্য দেশে কে এই প্রজেক্টটা কীভাবে করেছে। ইউটিউবে সার্চ করো **PIR Sensor Arduino Alarm** লিখে। অথবা ভিজিট করো নিচের লিংক :

www.tiny.cc/PIRalarm

তোমাকে আরেকটি বুদ্ধি দিই, যে পিনে বাজার (**Buzzer**) লাগিয়েছ, সেখানেই বাজার বদলে একটা এলইডি বাতি বসিয়ে দিয়ে দেখো তো কেমন মজা হয়! তুমি চাইলে বাজার ও এলইডি একসাথে বসাতে পারো।

প্রজেক্ট কাজ করার পরেও তোমার একটু ত্রুটা থেকে গেছে জানি। কী এই পিআইআর সেন্সর মডিউল এবং কীভাবে এটি গতিবিধি বুঝতে পারে বা ‘motion detect’ করে?

পিআইআর সেন্সর কীভাবে কাজ করে এটা তোমার জানা উচিত। কিন্তু আমরা এই বইতে তোমাকে সেটা বলে দেব না। আমরা চাই তোমার মাঝে একটা নতুন অভ্যাস গড়ে উঠুক। আর সেই অভ্যাসটা হলো - ‘কোনো কিছু জানতে ইচ্ছে করতে তা গুগল করে খুব দ্রুত শিখে ফেলা’

ইন্টারনেটের এই যুগে কেউ যদি বলে, আমি একটা জিনিস শিখতে চাই, কিন্তু কার কাছে কোথা থেকে শিখব? — তাহলে ওর মতো বোকা কেউ নেই!

যা-ই শিখতে ইচ্ছে করছে গুগল করবা। এমনকি কীভাবে ইফেক্টিভলি সার্চ করতে হয় এটা জানতেও **how to google effectively** লিখে গুগল করতে পারো তুমি!!!

তাহলে এখন বটপট **how PIR sensor works** লিখে গুগল বা ইউটিউব ঘেঁটে নাও।

আট

টিভির ভলিউম ঘোরানোর সেই দিনগুলি

তোমরা যদি তোমাদের বাসায় বা দাদা-নানার বাসায় পুরোনো টিভি দেখে থাকো সেখানে খেয়াল করলে দেখবা যে টিভির ভলিউম কমবেশি করা হয় একটা গোল সুইচের ডানে-বামে ঘুরিয়ে।

আচ্ছা যদি না দেখে থাকো হতাশ হওয়ার কিছুই নেই। তুমি নিশ্চয় তোমার বাসার ফ্যানের রেগুলেটর দেখেছ! এবং হাত দিয়ে ঘুরিয়ে স্পিড কমবেশি করেছ। ঘোরানোর এই যন্ত্রটা হচ্ছে পটেনশিওমিটার (**Potentiometer**) আগের এক অধ্যায়ে তোমরা রোধ কী জেনেছ। এই পটেনশিওমিটার হচ্ছে রোধ কমবেশি করার যন্ত্র। আমরা এই পটেনশিওমিটার দিয়ে ঘুরিয়ে রোধের মানের কমবেশি করে বিভিন্ন মান বের করব।



চিত্র ৮.১ : একটি সাধারণ পটেনশিওমিটার

একটা পটেনশিওমিটার দেখতে ঠিক এ রকম।

ছোট সাইজেরও পটেনশিওমিটার থাকে। নিচের ছবিতে দেখতে পারবে। এদের গায়ে সর্বোচ্চ রোধ কর পর্যন্ত হবে তা লেখা থাকে। যদি ‘**10K**’ লেখা থাকে তার মানে বুবাতে হবে এই পটেনশিওমিটার শূন্য থেকে ১০ হাজার ওহম পর্যন্ত রোধ দিতে পারে।

পটেনশিওমিটারে রোধ কিন্তু একটা স্টাইলে লেখা থাকে। যেমন : ‘**10K**’-এর অর্থ হচ্ছে ১০-এর পর তিনটা শূন্য ওহম। মানে ১০,০০০ ওহম বা ১০ হাজার ওহম!

মজার না ব্যাপারটা?

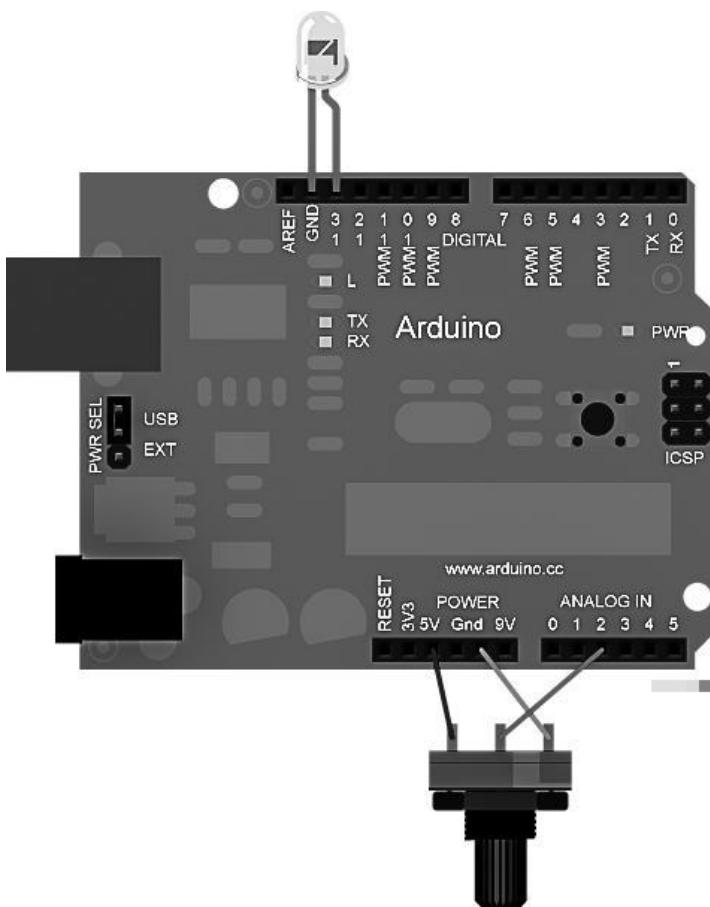


চিত্র ৮.২ : একটি হোট্ট পটেনশিওমিটার

এখন আমরা একটা পটেনশিওমিটার ঘূরিয়ে এই রোধের মানের পরিবর্তনটা দেখব আর তারপর ওটার সাথে একটা এলাইডি আলোর তীব্রতার পরিবর্তন দেখব। মানে আলো বাড়ানো-কমানো যাবে।

ছবিতে দেখাই যাচ্ছে একটা পটেনশিওমিটারের তিনটা মাথা থাকে, একটা মাথা হচ্ছে VCC বা পাওয়ার মানে এখানে ৫ ভোল্ট দিতে হবে।
আরেকটা মাথা GND

আরেক মাথা দিয়ে পটেনশিওমিটারের মান রিড করবে কম্পিউটার।



চিত্র ৮.৩ : আরডুইনোর সাথে পটেনশিওমিটার সংযুক্ত করা

ছবির মতো কানেকশন দিয়ে নিচের কোডটা আপলোড দিয়ে দাও। পটের দুই দিকের এক পা যাবে ৫ ভোল্ট পিনে, আরেক পা **GND** পিনে।

পটের মাঝের পা **Analog Input 2 (A2)** পিনের সাথে যুক্ত করবে।

চিত্রিত ভলিউম ঘোরানোর সেই দিনগুলি

```

int potPin = A2;      // select the input pin for
the      potentiometer
int LEDPin = 13;     // select the pin for the
LED

int value = 0;         // variable to store the
value coming from the sensor

void setup() {
    pinMode(LEDPin, OUTPUT); // declare the
LEDPin as an OUTPUT
}

void loop() {
    value = analogRead(potPin); // read the
value from the sensor
    analogWrite(LEDPin,value);

}

```

পোর্ট সিলেক্ট করে কোডটা আপলোড করলে তুমি দেখতে পারবে ঘোরানোর সাথে সাথে আলো ধীরে ধীরে বাড়ছে বা কমছে।

এখানে,

```

int potPin = A2;
int LEDPin = 13;
int value = 0;

```

প্রথমে আমরা বুঝিয়ে দিয়েছি যে A2 পটেনশিওমিটার আর ১৩ পিনে এলইডি বাতি। আর আমরা ভ্যালু রিড (Read) করার জন্য একটা ভেরিয়েবল/ চলক নিয়েছি যার মধ্যে ওই লুপ ঘোরাঘুরির সময় ভ্যালুগুলো চেঙ্গ হবে আর থাকবে। চলক (**Variable**) মানে ধরে নাও একটা বাক্স, যেখানে আমি ইচ্ছামতো মানগুলো জমা রাখব, আবার নেব। তোমার ধীজগণিত বইয়ে যে x, y, z ধরে অঙ্ক করো, ওগুলোও কিন্তু ভেরিয়েবল। ভেরিয়েবলের শাব্দিক অর্থ হলো, যার মান পরিবর্তন হতে পারে। একটু কি গুগল করবে এটা নিয়ে? করে দেখো!

`pinMode(LEDPin, OUTPUT);` দিয়ে আমরা বুবিয়েছি এলইডিটা আউটপুট হিসেবে কাজ করছে।

আর `analogRead(potPin);` (অ্যানালগ রিড) দিয়ে আমরা অ্যানালগভাবে এই ঘোরানোর ফলে রোধের মানের পরিবর্তন পড়ছি। এই যে রোধ পরিবর্তন এর মান, সেটাকে আমরা `value` নামের ভেরিয়েবলে রাখছি।

এবার `analogWrite(LEDPin, val);` (অ্যানালগ রাইট) দিয়ে ঠিক সেই `value`-গুলোই ব্যবহার করেছি এলইডি জ্বালাতে।

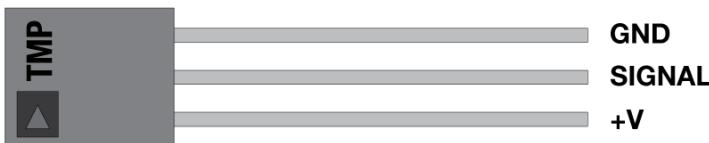
তাহলে আমরা পটেনশিওমিটারকে কোনো কিছু কমবেশি করতে ব্যবহার করেছি। এরপর তোমরা যখন এলসিডি স্ক্রিনের অধ্যায়ে যাবে, সেখানে দেখবে কীভাবে পটেনশিওমিটার দিয়ে আমরা ডিসপ্লের (Display) আলোর উজ্জ্বলতা/ব্রাইটনেস (brightness) কমবেশি করি।

চেষ্টা ১ : একটা বাজারের সাউন্ড কমবেশি করো পটেনশিওমিটার দিয়ে

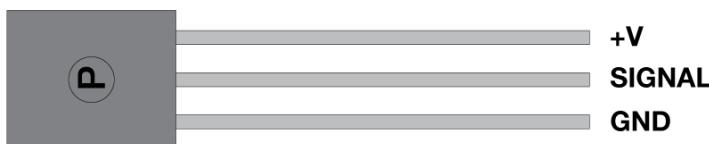
নয়

ডষ্টের আরডুইনো

তাপমাত্রা বললেই আমাদের জ্বর হওয়ার কথা মনে পড়ে যায়! আবার মনে পড়ে যায় সেই গরমকালের দিনগুলোর কথা। আজ আমরা আমাদের দেহের তাপমাত্রা আরডুইনো দিয়ে পরিমাপ করব। কথা হচ্ছে, তাপমাত্রা আমরা পরিমাপ করব কী দিয়ে?



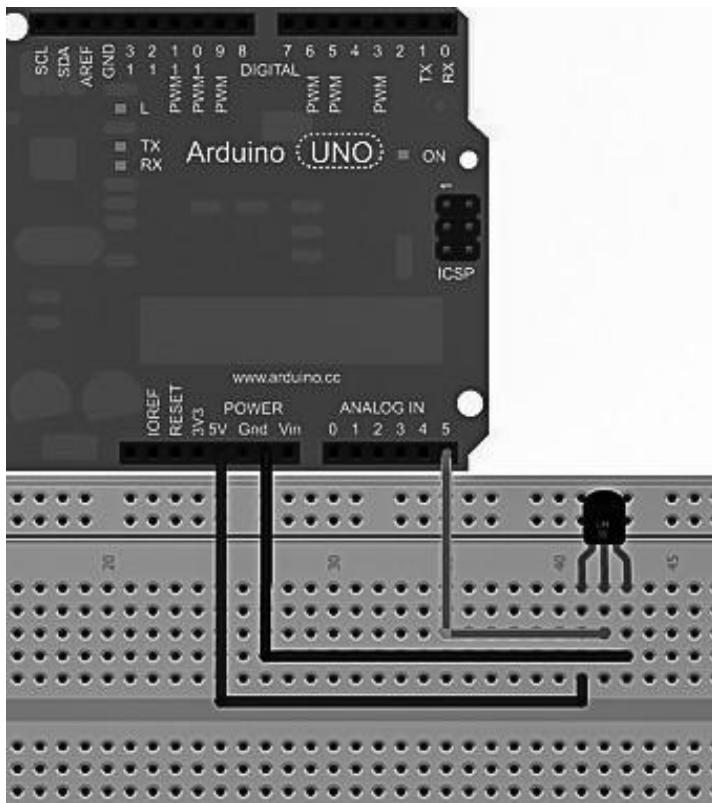
FRONT



BACK

চিত্র ৯.১ : LM35 সেন্সর

আমরা তাপমাত্রা পরিমাপে ব্যবহার করব টেম্পারেচার সেন্সর LM35 বা LM36। ছবিতে একটা LM35 সেন্সর দেখা যাচ্ছে। এটা আমাদের দেহের তাপমাত্রা মেপে দেখাবে।



চিত্র ৯.২ : আরডুইনোর সাথে LM35 সেন্সর যুক্তকরণ

এর তিনটা পিন আছে। একটা **VCC**, একটা **GND**, আরেকটা সিগন্যাল। তাহলে আমরা সিগন্যাল পিনকে একটা আই/ও পিনে লাগাব (এখানে **A5**) আর অন্য পিনগুলোকে ৫ ভোল্ট আর গ্রাউন্ডে লাগিয়ে দেব নিচের ছবির মতো। তোমার কি ভিসিসি আর গ্রাউন্ড চিনতে অসুবিধে হচ্ছে? চিত্র ৯.১ দেখে বুঝার চেষ্টা কর, সেন্সরের গায়ের গোল গর্ত অংশ উপরে ধরলে কোনটা পজেটিভ হয় আর কোনটা গ্রাউন্ড।

ডষ্ট্র আরডুইনো

নিচের কোড লিখে ফেলি বাটপট—

```
int sensorPin = A5;
int sensorVal =0;
float bodyTemp;
float voltage;

void setup()
{
pinMode(sensorPin, INPUT);
Serial.begin(9600);
}

void loop()
{

int sensorVal = analogRead(sensorPin);
//Read the sensor value

float voltage = (sensorVal/1024.0) * 5.0;
//Convert reading to voltage

float temperature = (voltage - .5) * 100;

//convert millivolts into temperature

bodyTemp=(temperature *1.8)+32;

Serial.print(bodyTemp);
Serial.print("\n");
}
```

চলো কোডগুলো একটু বুঝে নিই

আমরা এতদূর কাজ করে ভেরিয়েবলের অনেকটা বুঝে গেছি। নিচের লাইনগুলো হচ্ছে ভেরিয়েবল।

```
int sensorPin = A5;
int sensorVal =0;
float bodyTemp;
float voltage;
```

এগুলো অনেকটা ঝুঁড়ির মতো। এর মধ্যে আমরা কিছু রাখছি। তবে ভেরিয়েবলের আগে আমরা অনেক সময় **int**, **float**, **long**, **char** এসব লিখি। এসব হচ্ছে ডেটা টাইপ (**Data Type**)।

মানে ধরো ঝুঁড়ির মধ্যে তো তুমি অনেক কিছুই রাখতে পারো। পাথর রাখতে পারো আবার খেলনা পুতুলও রাখতে পারো। আবার পানি বা তেল রাখতে ঝুঁড়িতে হবে না, বালতি লাগবে। ডেটা টাইপ দিয়ে এটাই কম্পিউটারকে বোঝানো হচ্ছে যে কী রাখা হবে।

Int মানে হচ্ছে সাধারণ সংখ্যা। **float** মানে দশমিক সংখ্যা।

```
int sensorPin = A5;
```

এই লাইন দিয়ে বোঝানো হয়েছে টেম্পারেচার সেন্সরকে A5 পিনে লাগানো হয়েছে।

```
int sensorVal = 0;
```

এই লাইন দিয়ে বোঝানো হয়েছে টেম্পারেচার সেন্সর যে ভ্যালুটা দেবে, সেটা আমরা এখানে রাখব। প্রথমবার এখানে কিছু নেই তাই জিরো রাখা হয়েছে।

```
float bodyTemp;  
float voltage;
```

Float দেখে বুঝে গেছ এরা দশমিক সংখ্যার জন্য। শরীরের তাপমাত্রা মেপে আমরা এর মধ্যে রাখব। আর **voltage**-এর মধ্যে তোল্টেজের পরিমাণ থাকবে।

তুমি এত দিনে বুঝে গেছ **void setup()** আর **void loop()** নিজেরাও ফাংশন।

কোনো ফাংশন যখন কিছু রিটার্ন করে না তখন আমরা সেখানে **void** ব্যবহার করে থাকি।

যাহোক, `void setup()` এর মধ্যে সেন্সর যে ইনপুট এটা বোঝানো হয়েছে। আর আমরা কিন্তু মাইক্রোকন্ট্রোলারের সাথে ইউএসবি কেবল দিয়ে কম্পিউটারের সাথে লাগিয়ে দিয়েছি। এর মানে এদের মধ্যে কমিউনিকেশন হচ্ছে। এই কমিউনিকেশন বোঝানো হয়েছে

`Serial.begin(9600)` দিয়ে।

`Void loop()` এর মধ্যে আমরা কয়েকটা কাজ করেছি। প্রথমে আমরা সেন্সরের মধ্যে মান কত আসছে সেটা পড়ে নিয়েছি `analogRead()` দিয়ে।

`int sensorVal = analogRead(sensorPin);`

তারপর এই ভ্যালুটা আসলে ভোল্টেজে কত মান দিচ্ছে, এটা পড়েছি আমরা এই লাইন দিয়ে।

`float voltage = (sensorVal/1024.0) * 5.0;`

সেন্সর থেকে আরভুইনো যে মান পাচ্ছে, সেটাকে আমরা ভোল্টেজে রূপান্তর করছি এই সমীকরণ দ্বারা। এটা অ্যানালগ-ডিজিটাল রূপান্তরের একটি নিয়ম।

সূত্র= অ্যানালগ ভোল্টেজ = (প্রাপ্ত মান/ অ্যানালগ-ডিজিটাল কনভার্টারের সর্বোচ্চ ক্ষেত্র) * সিস্টেম ভোল্টেজ

বিষয়টা একটু জটিল। আপাতত এই গাণিতিক অংশটা না বুঝলে মন খারাপ করো না। আর যদি বেশি জানতে ইচ্ছে হয় তাহলে **ADC** (অ্যানালগ টু ডিজিটাল কনভার্টার) **DAC** (ডিজিটাল টু অ্যানালগ কনভার্টার)

লিখে গুগল করতে পারো।

সেই ভোল্টেজটা আসলে আমাদের টেম্পারেচার এককে কত তা আমরা হিসাব করে নিয়েছি এই লাইন দিয়ে

`float temperature = (voltage - .5) * 100;`

এই তাপমাত্রাটা কিন্তু আসলে এখন সেলসিয়াসে আছে। আমাদের শরীরের তাপমাত্রা আমরা কিন্তু মাপি ফারেনহাইটে। তোমরা নিশ্চয় সেলসিয়াস ফারেনহাইট বোঝো?

না বুবালে সাহায্য নাও গুগলের।

$$T_{(^{\circ}\text{F})} = T_{(^{\circ}\text{C})} \times 1.8 + 32$$

এটা হচ্ছে সেলসিয়াস থেকে ফারেনহাইটে তাপমাত্রা পরিবর্তনের নিয়ম।

```
bodyTemp=(temperature *1.8) + 32;
```

এই লাইন দিয়ে আমরা সেলসিয়াসকে ফারেনহাইটে বানিয়ে নিয়েছি।

এবার সেই তাপমাত্রা মনিটরে প্রিন্ট করার পালা! সে কাজটা করেছি আমরা এই দুই লাইন দিয়ে।

```
Serial.print(bodyTemp);
```

```
Serial.print("\n");
```

এখন তুমি সেঙ্গে হাতের আঙুল দিয়ে চেপে ধরে থাকো আর টুলস থেকে সিরিয়াল মনিটর চালু করো তুমি তোমার দেহের তাপমাত্রা দেখতে পারবে!

(Arduino IDE-> tools -> Serial Monitor)

তুমি থার্মোমিটার বানিয়ে ফেলেছ!

অভিনন্দন তোমাকে!!!

চেষ্টা ১ : এমন কিছু বানাও যখন পরিবেশের তাপমাত্রা ২৩ ডিগ্রি সেলসিয়াসের ওপর উঠে যাবে, তখন একটা নীল এলইডি জ্বলবে

চেষ্টা ২ : এবার তাপমাত্রা ২৫ ডিগ্রি সেলসিয়াসের ওপর উঠলে বেজে যাবে বাজার

চেষ্টা ৩ : ৩০ ডিগ্রি সেলসিয়াসের ওপর উঠলে বেজে যাবে এলইডি ও বাজার দুটোই। তুমি একটা পরিবেশ তাপমাত্রা মাপার যন্ত্র বানিয়ে ফেলেছ।

দশ

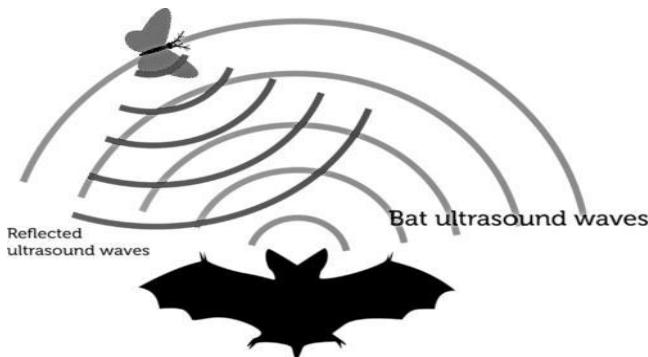
বাদুড়ের চোখ

এই অধ্যায়ে আমরা চলে যাব আমাদের ব্যাটম্যান মানে বাদুড় ভাইয়ার
কাছে!

তোমরা কি জানো যে বাদুড় চোখে দেখতে পারে না!

তাহলে বাদুড় কীভাবে চোখে দেখে?

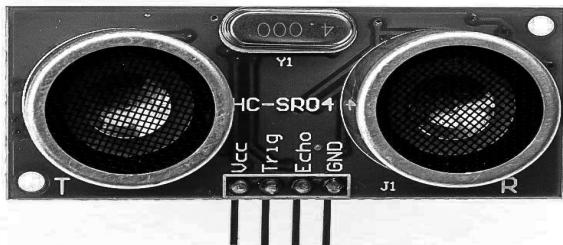
বাদুড় মুখ দিয়ে শব্দতরঙ্গ পাঠায়। সেই শব্দ আমরা শুনতে পাই না। কারণ
তা আমাদের শ্রাব্যতার সীমার বাইরে! আমরা বিশ থেকে বিশ হাজার
হার্জের মধ্যে শব্দ শুনতে পাই। বাদুড়ের করা সেই শব্দ সামনে যেতে
থাকে। সামনে যদি কিছু থাকে তাহলে শব্দ প্রতিধ্বনি হয়ে ফিরে আসে।
এতে বাদুড় বুঝতে পারে যে তার সামনে কোনো বস্তু আছে। বাদুড় যে
সব সময় ঠিক তা কিন্তু না। আমরা অনেক সময় রাস্তার বৈদ্যুতিক তারে
বাদুড়কে মরে ঝুলে থাকতে দেখি। যদি বাদুড়ের করা সেই শব্দ ঠিকমতো
ফিরে না আসে বাদুড় তখন সামনে কিছু আছে কি না, বুঝতে পারে না।



চিত্র ১০.১ : বাদুড় পথ চলে আক্ষৱাসনিক শব্দতরঙ্গ সৃষ্টি করে

এবার আসা যাক SONAR-এ। SONAR-এর পূর্ণরূপ দাঁড়ায় : Sound Navigation And Ranging

বাদুড়ের ঠিক এই ঘটনাকে কাজে লাগিয়েই রোবটিকসে SONAR নামে একটা sensor module বানানো হয়েছে। SONAR মডিউলের সামনে দুটি মাথা থাকে। দেখতে অনেকটা চোখের মতো। একটা চোখ থেকে তরঙ্গ ছড়িয়ে যায়। আরেকটা চোখ প্রতিরূপ হয়ে ফেরা তরঙ্গ গ্রহণ করে। এভাবে আমরা বুঝিতে পারি সামনে কিছু আছে কি না? আর ঠিক কত দূরত্বে আছে।



চিত্র ১০.২ : সোনার (SONAR) সেন্সর

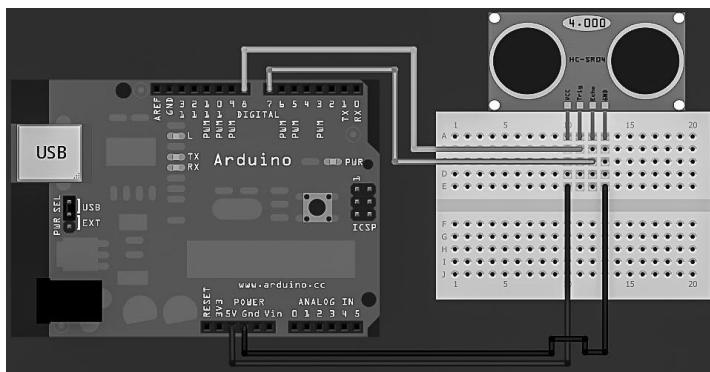
বাদুড়ের চোখ

দেখো এই সেলের রয়েছে চারটা পিন।

একটা **VCC**, আরেকটা **GND**, এরা মূলত পাওয়ার দেওয়ার পিন।

আর দুটো **TRIG** মানে ট্রিগার, মানে যে শব্দ ছড়িয়ে দেয় আর **ECHO** মানে প্রতিধ্বনি শব্দ, এর মানে হিসাবের পিন।

এবার আমরা এই SONAR-কে কাজে লাগিয়ে সামনে কিছু আছে কি না আছে, এটা হিসাব করব। আর কত দূরে আছে, এটাও হিসাব করব। চলো ছবির মতো কানেকশনটা দিয়ে নিই।



চিত্র ১০.৩ : আরডুইনো এবং SONAR সেল যুক্তকরণ

কানেকশন ঠিকমতো দেওয়া হয়ে গেলে বাটপট কোডটা আপলোড করে ফেলি!

```

// defines pins numbers
const int trigPin = 9;
const int echoPin = 10;

// defines variables
long duration; // long is a data-type
int distance;

void setup() {
pinMode(trigPin, OUTPUT); // Sets the trigPin
as an Output
pinMode(echoPin, INPUT); // Sets the echoPin as
an Input
Serial.begin(9600); // Starts the serial
communication
}

void loop() {
// Clears the trigPin
digitalWrite(trigPin, LOW);
delayMicroseconds(2);

// Sets the trigPin on HIGH state for 10 micro
seconds
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);

// Reads the echoPin, returns the sound wave
travel time in microseconds
duration = pulseIn(echoPin, HIGH);

// Calculating the distance
distance = duration*0.034/2;

// Prints the distance on the Serial Monitor
Serial.print("Distance: ");
Serial.println(distance);
}

```

তোমার যদি মনে হয় আইডিইতে কোড টাইপ করার সময় কমেন্টগুলো
বাদে শুধু কোডটুকু লিখবে তাও করতে পারো। আমরা কমেন্ট করি

বাদুড়ের চোখ

বোঝার এবং মনে রাখার সুবিধার্থে। চলো এবার এই কোডটাকে একটু দেখে নেওয়া যাক :

```
const int trigPin = 9;  
const int echoPin = 10;
```

এই দুইটা হচ্ছে ভেরিয়েবল এবং আমরা যে ৯ আর ১০ নম্বর পিনে ট্রিগার আর ইকো পিনকে লাগিয়েছি, এটা বোঝাই যাচ্ছে।

```
long duration;  
int distance;
```

লং আসলে একটু বড় আকারের ডেটা রাখার ভেরিয়েবল। এটার মধ্যে সময়ের হিসাব রাখা হচ্ছে! আর distance-এর মধ্যে রাখা হচ্ছে দূরত্ব।

void setup() এর মধ্যে লেখা আছে এই তিন লাইন।

```
pinMode(trigPin, OUTPUT);  
pinMode(echoPin, INPUT);  
Serial.begin(9600);
```

এই তিন লাইনে কী হচ্ছে, এটা আমরা এবার বলে দেব না! তুমি যদি প্রথম থেকে এই বইটা পড়ে থাকো তার মানে তুমি নিজেই বুঝে গেছ এখানে কী হচ্ছে!

void loop() এর মধ্যে digitalWrite(trigPin, LOW) লিখে trig পিনকে ক্লিয়ার করে নেওয়া হচ্ছে। delayMicroseconds(2) দিয়ে আমরা কিছুক্ষণ থেমে থাকছি।

এবার আমরা কিছুক্ষণের জন্য trigPin-এ HIGH স্টেটে রাখছি।

```
digitalWrite(trigPin, HIGH);  
delayMicroseconds(10);  
digitalWrite(trigPin, LOW);
```

এখানে ঢঁষংবওহ একটা ফাংশন যেটা বপযড়চরহ-কে পড়ে আর জানিয়ে দেয় কতক্ষণে শব্দটা ফিরে আসছে।

```
duration = pulseIn(echoPin, HIGH);
```

এই লাইন দিয়ে দূরত্বটা মেপে নেওয়া হয়।

```
distance= duration*0.034/2;
```

শব্দ উৎপন্ন হয়ে সেখান থেকে দূরের কোনো একটা বস্তুতে ধাক্কা খেয়ে ফিরে আসতে কিছু সময় লাগে। এই সময়টাই হলো ডিউরেশন। আর বায়ুতে শব্দের বেগ প্রতি সেকেন্ডে আনুমানিকভাবে 340 মিটার। সেখান থেকেই একটু অক্ষ করে বের করা হয়েছে ওপরের লাইন। খেয়াল করো, ২ দিয়ে ভাগ করা হয়েছে, কারণ শব্দ যেতে একবার সময় লাগে। আবার ধাক্কা খেয়ে ফিরে আসতে আরেকবার সময় লাগে।

আর এই দুই লাইন প্রিন্ট করে যাচ্ছে!

```
Serial.print("Distance: ");
```

```
Serial.println(distance);
```

এবার কি করবে বলো তো!

ঠিক বলেছ!

Arduino IDE->Tools ->SerialMonitor চালু করলে তুমি দেখতে পাবে সামনে কিছু আছে কি না আর থাকলে সেটা কত দূরে আছে। এবার একটা কাজ করো। তোমার বানানো বাদুড়ের চোখ মানে Sonar-এর সামনে একবার হাত দাও। সিরিয়াল মনিটরে দেখো কত প্রিন্ট করছে। এবার হাতটা দূরে সরাও। দেখো দূরত্ব পরিবর্তন হচ্ছে।

অভিনন্দন!!!

তোমার বাদুড়ের চোখ বানানো হয়ে গেছে!

বাদুড়ের চোখ

চেষ্টা : একটা SONAR নাও। তারপর আরডুইনোতে একটা বাজার লাগাও। এখন তোমাকে এমন একটা প্রোগ্রাম লিখতে হবে যাতে তোমার হাত যদি SONAR-এর ২ সেন্সormিটারের মধ্যে আনো তাহলে বাজারে বিপ বিপ করে টেন বাজতে থাকবে!



চিত্র ১০.৮ : SONAR সেন্সর আর বাজার ব্যবহার করে অঙ্কের জন্য লাঠি বানিয়েছে তোমার মতোই কিছু শিক্ষার্থী

এগারো

ডানে-বাঁয়ে ঘূরে সেই পুতুলটি

আমরা অনেক সময় অনেক খেলনা দেখেছি এ রকম যে একটা পুতুল ডান থেকে বামে, বাম থেকে ডানে একাই ঘূরে যাচ্ছে। যদি না দেখে থাকে সমস্যা নেই এ রকম একটা খেলনা আজ আমরা বানাব!

আর এই প্রজেক্টটা করতে আমাদের একটা পুতুল লাগবে। খেলনা বা হাতে বানানো হলেও হবে। আর লাগবে সার্ভো একটা মোটর।

সার্ভো মোটর অনেক রকমের হয়ে থাকে। এই মোটর মূলত কোনো কিছুকে ঘোরাতে যে শক্তি লাগে, সেই শক্তিটুকু দিতে ব্যবহার হয়ে থাকে। ক্ষমতা ও সাইজ অনুসারে অনেক রকমের সার্ভো মোটর আছে। নিচের ছবিগুলো দেখো। এগুলো সব সার্ভো মোটরের ছবি।



চিত্র ১১.১ : সার্ভো মোটর

ডানে-বাঁয়ে ঘূরে সেই পুতুলটি

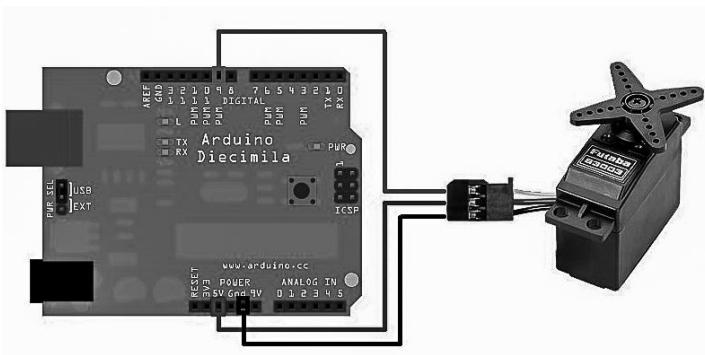


চিত্র ১১.২ - সার্ভো মোটর



এখানে সার্ভোর
মাথায় আঠা দিয়ে
পুতুলটি লাগিয়ে দাও

চিত্র ১১.৩ : সার্ভোর সাথে ছোট ছোট মজা



চিত্র ১১.৩ : সার্ভোর সাথে ছোট ছোট মজা

এবার আসো কাজে!

পুতুলটাকে ইট-গু দিয়ে দিয়ে সার্ভোর মাথায় লাগিয়ে দাও ছবির মতো করে।

তারপর চলো কানেকশন দিই সার্কিটের ছবি দেখে।

একটা সার্ভো মোটরের তিনটা তার থাকে। তিন রঙের। একটা লাল, কালো আর ছাই বা কমলা এ রকম। লালটা **VCC**, কালোটা **GND** আর অন্য যে কালারের থাকুক সেটা সিগন্যাল পিন হিসেবে কাজ করে। এবার চলো ছবির মতো করে লাগিয়ে ফেলি! আসলে মোটরটা পুতুলটাকে ঘোরাবে।

ডানে-বায়ে ঘুরে সেই পুতুলটি

এটা নিচের প্রোগ্রামটা লিখে ফেলো-

```
#include <Servo.h>

Servo myservo; // create servo object to
control a servo
Int i;

void setup() {
    myservo.attach(9); // attaches the servo on
pin 9 to the servo object
}

void loop() {

    for(i=0;i<=180;i++)
    {
        myservo.write(i);
        delay(50);
    }
}
```

তুমি যদি কোডটা ঠিকমতো আপলোড করে থাকো তুমি দেখবে পুতুলটা ডান থেকে বামে ১৮০ ডিগি ব্যবধানে ঘূরছে। এখানে কোডটা আগের মতোই তবে নতুন কিছু জিনিস আছে।

```
#include <Servo.h>
```

হচ্ছে একটা হেডার ফাইল (**Header File**), যার মধ্যে মনে করো অনেক কিছু লেখা আছে। তুমি যে রকম কোড লেখো, এই সার্ভোটাকে ঠিকমতো চালাতে বিশাল বড় একটা কোড লাগে। ওইটা ওখানে লেখা আছে। আপাতত এটা আমাদের চিন্তার বিষয় না।

তারপর **Servo myservo** দিয়ে আসলে সার্ভোটাকে কন্ট্রোল করার জন্য একটা ভেরিয়েবল নেওয়া হয়েছে। এটা দিয়ে আমরা কোণের মান ঠিক করে দেব।

myservo.attach(9) দিয়ে আমরা বুঝিয়েছি ৯ নম্বর পিনে আমরা সার্ভোটাকে লাগিয়েছি।

```
for (i=0;i<=180;i++)
```

```

{
    myservo.write(i);
    delay(50);
}

```

আর ভয়েড লুপের মধ্যে শেখা ফর আসলে একটা লুপ। ফর লুপে তুমি শুরু ও শেষ নির্ধারণ করে দিলে শর্তমতে কাজটা বারবার করতে থাকবে। এটাতে বলা আছে কীভাবে ০ থেকে ১৮০ ডিগ্রিতে পুতুলটা ঘূরবে। মসৃণভাবে ঘোরার জন্য ছোট একটা ফবষধু দেওয়া হয়েছে। তাহলে এভাবে আমরা ঘূরন্ত পুতুলের খেলনা বানিয়ে ফেললাম!

এই অধ্যায়ে আমরা প্রোগ্রামিং-বিষয়ক নতুন একটা বিষয় পেয়েছি, তা হলো ফর লুপ (for loop)

ফর লুপ সম্পর্কে জানার অনেক কিছু আছে, কিন্তু এত কিছু এই ছোট অধ্যায়ে বলা কঠিন। তোমার জন্য ফর লুপ শেখার কিছু রিসোর্স :

<http://cpbook.subeen.com/2011/08/loop.html>
<https://youtu.be/iN16QJuFNBs>

চেষ্টা : একটা SONAR নাও। ওটাকে সার্ভোর মাথায় প্লু দিয়ে লাগিয়ে একটা রাডার বানাও। যখন কোনো কিছু রাডারের আশপাশে আসবে বাজারে একটা করে বিপ দেবে!

চেষ্টা : কোনো কিছুকে শক্তি দিয়ে ঘোরাতে চাইলে আমরা সার্ভো ব্যবহার করে থাকি। সার্ভো মোটর আর আইসক্রিমের কাঠি দিয়ে তুমি কিন্তু চাইলে রোবটের হাত বানিয়ে ফেলতে পারো। এখানের ভিডিওটা দেখে চেষ্টা শুরু করে দাও :

<https://youtu.be/Q3FOCqI0IQM>

বারো

এলসিডি স্ক্রিনে কেমন মজা

তুমি কি এলসিডি স্ক্রিন দেখেছ? এখন কি মাথা চুলকে বলছ LCD Screen আবার কী জিনিস? তোমার ক্যালকুলেটরের ডিসপ্লে হলো LCD, তোমার হাতের ডিজিটাল ঘড়ির ডিসপ্লে হলো LCD, অনেকে নিশ্চয়ই LCD টিভিও দেখেছ।

LCD হচ্ছে **Liquid Crystal Display** নিয়ে আরও জানতে একটু গুগল করবে তুমি।

আজকে তুমি একটা LCD স্ক্রিনে তোমার নিজের নাম লিখে প্রিন্ট করবে।
মজা হবে তাই না?

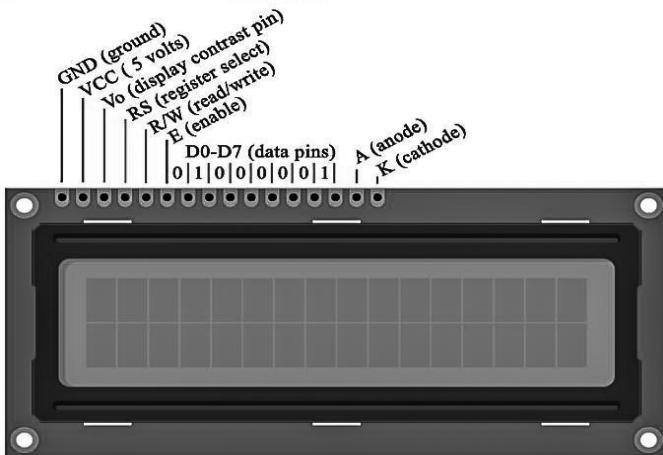
বাজারে অনেক রকমের ডিসপ্লে পাওয়া যায়। আমরা ব্যবহার করব ১৬*২ ডিসপ্লে। এর মানে একবারে তুমি এখানে ৩২টা অক্ষর প্রিন্ট করাতে পারবে। এ ছাড়া আছে একটু বড় সাইজের ১৬*৪ ডিসপ্লে। দামে একটু বেশি।

এই অধ্যায়ে তোমার প্রয়োজন হবে আমাদের আগের সেই পটেনশিওমিটার। পোটেনশিওমিটার দিয়ে আমরা ডিসপ্লের ব্রাইটনেস কমবেশি করব। আমরা এই ডিসপ্লে দিয়ে যেকোনো কিছু করতে পারি। আমাদের কোনো কিছু কাউকে দেখাতে হলে আমরা প্রিন্ট করে দেখাতে পারি। তুমি চাইলে একটা ক্যালকুলেটর বানাতে পারো, চাইলে বানাতে পারো ঘড়ি আরও কত কী!

চলো কথা না বাড়িয়ে আমরা নেমে পড়ি আরডুইনোতে কানেকশন দিতে :

এখানে প্রথম ছবিতে একটা এলসিডি স্ক্রিনের পিনের নামগুলো দেওয়া আছে। চলো পিনগুলোর কাজ কিছুটা বুঝে নেওয়ার চেষ্টা করি। পিনগুলো আমরা গোনা শুরু করব বাম দিক থেকে। সবচেয়ে বামেরটা হচ্ছে ১ নম্বর পিন আর একদম ডানেরটা ১৬।

১ নং যাচ্ছে গ্রাউন্ড। মানে কানেকশনের সেই নেগেটিভ অংশ। ২ নং যাবে 5V এ, মানে কানেকশনের পজেটিভ অংশ। ৩ নম্বর হচ্ছে ডিসপ্লের কন্ট্রাস্ট পিন। মানে এই পিন দিয়ে তুমি আলোর তীব্রতার কিছু কমবেশি করতে পারবা। রেজিস্টার সিলেক্ট পিন আসলে অনেকটা মেমরি সিলেক্টের কাজ করে। রিড-রাইট পিন থেকে ডেটা পড়ব না, ডেটা রাইট করব, এটা জানানো যায়। আর মাঝের পিনগুলো হচ্ছে ডেটা পিন। এই পিনগুলো দিয়ে তুমি যে লেখাটা প্রিন্ট করাবে, সেই লেখার তথ্যগুলো কারেন্ট আকারে যায়। শেষের দুইটা পিন, অ্যানোড-ক্যাথড অনেকটা এলইডি বাতির ধনাত্মক-ঋণাত্মক প্রাপ্তের মতো কাজ করে। এরা ব্যাকলাইট ঠিক করে।

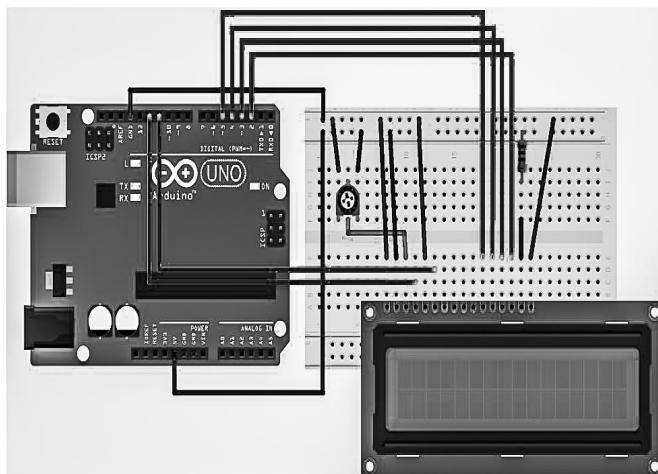


চিত্র ১২.১ : এলসিডি স্ক্রিনের পিন কনফিগারেশন

এবার চলো ছবির মতো সংযোগ দিয়ে নিই। এখানে কিছু পিনে আমরা একটু কম পরিমাণে ভোল্টেজ দেব বলে রেজিস্ট্যান্স দিয়েছি। আর কন্ট্রাস্ট

এলসিডি স্ক্রিনে কেমন মজা

(contrast) ঠিক করার জন্য পোটেনশিওমিটার।



চিত্র ১২.২ : যেভাবে আরডুইনোর সাথে এলসিডি যুক্ত করবে

নতুন কেনার পর এলসিডি ক্ষিণে জাম্পার ওয়্যার যুক্ত করার মতন ব্যবস্থা থাকে না। এ জন্য একটি মেল-টু-ফিমেল আইসি রেইল (IC rail বা pin header strip) লাগিয়ে নিতে হয়। তালাল দিইয়ে ঝালাই (Solder) করে লাগানোর কাজটি একটু কঠিন। তুমি এটা তোমার বাসার বড় কারো সাহায্য নিয়ে লাগাতে পারো। অথবা, তোমার এলাকার ইলেক্ট্রনিকস মেকার আংকেলের কাছে লাগিয়ে নিতে পারো।

এখন ছবিতে দেখো আমরা কীভাবে কানেকশন দিয়েছি। ব্রেড বোর্ডের একটা অংশ পজেটিভ থেকে এসেছে আরেক অংশ এসেছে নেগেটিভ বা গ্রাউন্ড থেকে। এখানে কয়েকবার আমরা ডিসপ্লের কয়েকটা পিনের সাথে পজেটিভের সংযোগ দিয়েছি, আর কয়েকটায় গ্রাউন্ড সংযোগ দিয়েছি। ২-৫ নম্বরের চারটা পিন দিয়ে আমরা ডিসপ্লেতে তথ্য পাঠাচ্ছি, মানে কী প্রিন্ট করতে হবে।

চিত্রে যেভাবে কানেকশন দেওয়া হয়েছে -

- LCD RS pin যুক্ত করা হয়েছে আরডুইনোর digital pin 12 এ

- LCD Enable pin যুক্ত করা হয়েছে আরডুইনোর digital pin 11 এ
- LCD D4 pin যুক্ত করা হয়েছে আরডুইনোর digital pin 5 এ
- LCD D5 pin যুক্ত করা হয়েছে আরডুইনোর digital pin 4 এ
- LCD D6 pin যুক্ত করা হয়েছে আরডুইনোর digital pin 3 এ
- LCD D7 pin যুক্ত করা হয়েছে আরডুইনোর digital pin 2 এ

এর পাশাপাশি, **10K** পটেনশিওমিটারের দুই পাশের দুই পা ৫ ভোল্ট ও গ্রাউন্ডে গিয়েছে। আর পটের মাঝের আউটপুট পিন গেছে **LCD VO pin (pin3)**-এ।

একটি ২২০ ওহম রেজিস্ট্র যুক্ত করা হয়েছে এলসিডির ১৫ অথবা ১৬ নং পিনের সাথে। এই ১৫ ও ১৬ নং অ্যানোড-ক্যাথডের ফলে স্কিনটি উজ্জ্বলভাবে জ্বলে অর্থাৎ **Backlit** জ্বালিয়ে রাখার কাজ করো।

এবার আরডুইনো আইডিই খুলে নিচের কোডটা আপলোড করে দাও।

```
const int rs = 12, en = 11, d4 = 5, d5 = 4, d6
= 3, d7 = 2;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

void setup() {
    // set up the LCD's number of columns and
    rows:
    lcd.begin(16, 2);
    // Print a message to the LCD.
    lcd.print("hello, world!");
}

void loop() {
    // set the cursor to column 0, line 1
    // (note: line 1 is the second row, since
    counting begins with 0):
    lcd.setCursor(0, 1);
    // print the number of seconds since reset:
    lcd.print(millis() / 1000);
}
```

কোডের এই অংশটুকু সাবধানে লিখবে, অনেক সময় ওলট-পালট হয়ে যায়-

```
rs = 12,
en = 11,
```

এলসিডি স্ক্রিনে কেমন মজা

```
d4 = 5,  
d5 = 4,  
d6 = 3,  
d7 = 2;
```

কোড আপলোডের পর কী দেখলে!!!

তোমার LCD তে Hello World! প্রিন্ট করছে!

এবার চলো কোডটায় একটু চোখ বুলিয়ে নিই :

প্রথমে পিনগুলো কোনটার কাজ কী, সেটা জানিয়ে দেওয়া হয়েছে।

তারপর একটা ফাংশন কল করা হচ্ছে এবং এই ফাংশনই তোমার কোডের পুরো কাজটা করে দিচ্ছে। তুমি শুধু ফাংশনটা কল করছ এখানে।

আর লুপের মধ্যে তুমি প্রিন্ট করিয়ে ডিসপ্লেটা রিফ্রেশ করাচ্ছ

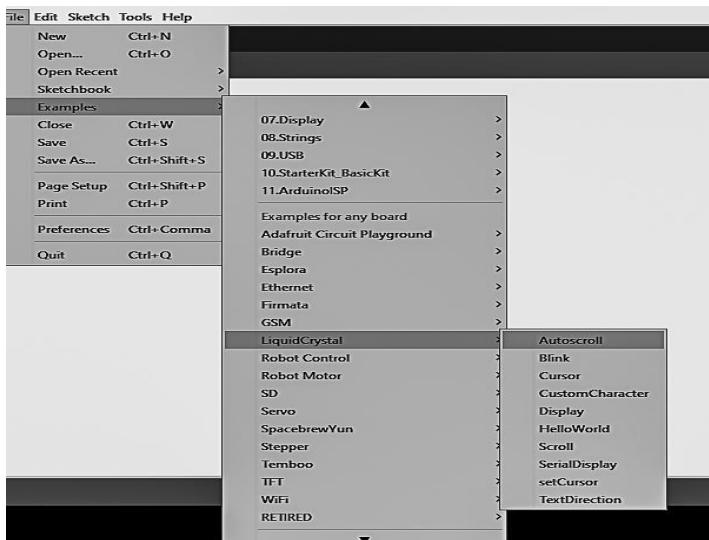
চলো এবার তোমার কাজ শুরু -

এখানে **lcd.print** একটা ফাংশন। এর মধ্যে তুমি যা লিখবে তা সেখানে প্রিন্ট হবে। এবার তুমি এই ডিসপ্লেতে তোমার নাম লিখে দেখাও তো প্রিন্ট হচ্ছে কি না?

এবার আরো মজার কাজ বাকি আছে। এই LCD-তে কাজ করার জন্য যারা আরডুইনো বানিয়েছেন তারা কিছু ফাংশন বানিয়েই রেখেছেন। আমরা সেগুলোর ব্যবহার দেখব। এ জন্য তোমাকে আরডুইনো আইডিই ওপেন করে

File->Examples->LiquidCrystal-

এ যেতে হবে। ঠিক ছবির মতো।



চিত্র ১২.৩ - এলসিডি বা লিক্যুইড ক্রিস্টাল ডিসপ্লের বিভিন্ন উদাহরণ প্রোগ্রাম

এখানে প্রায় নয়টার মতো প্রোগ্রাম আছে। একেকটা প্রোগ্রামে একেক রকম মজা। তুমি সবগুলো প্রোগ্রাম এখানে রান করে দেখো কী কী হয়।

চেষ্টা ১: একটা SONAR-এর সামনে কিছু কত দূরে আছে, সেটা কম্পিউটার মনিটরে প্রিন্ট না করে LCD ডিসপ্লেতে প্রিন্ট করাও।

চেষ্টা ২: LCD তে এমন কিছু লেখা কাউকে জানতে চাও। ধরো তুমি চাইলে তোমার ঘরে কেউ ঢুকবে না। তাই তুমি সেটা একটা ডিসপ্লেতে প্রিন্ট করবা। মানে এ রকম ‘NO ENTRY IN THIS ROOM WHILE I AM READING’ কিন্তু তোমার লেখাটা ডান থেকে বামে চলতে থাকবে।

(Hints : তুমি Examples নয়টা কোডের মধ্যে এটা নিয়ে একটা কোড পাবে।)

তেরো

ঘোরাব এবার একটা চাকা

এ অধ্যায়ে আমরা শিখব কীভাবে একটা মোটর দিয়ে চাকা ঘোরাতে হয়। তুমি যদি একটা চাকা ঘোরাতে পারো তাহলে তুমি দুইটা চাকাও ঘোরাতে পারবে। আর দুইটা চাকা ঘোরাতে জানলে তুমি কোনো কিছু চালাতে পারবে! মানে তুমি চলনশীল রোবট বানাতে পারবে! বানাতে পারবে খেলনা গাড়ি!

আমরা শুরু করব ৫ ভোল্ট ডিসি গিয়ার মোটর দিয়ে। এর সাথে আমরা একটা চাকা লাগিয়ে নেব।

দোকানে এই চাকাগুলো কিনতে পাওয়া যায়। তুমি চাইলে বানিয়ে নিতে পারো পুরোনো বোতলের কর্ক দিয়ে আবার পুরোনো খেলনা থেকে খুলে হট-গ্লু দিয়ে লাগিয়ে নিতে পারো।

নিচে মোটর ও চাকার ছবি দেওয়া আছে—



চিত্র ১৩.১ : ডিসি মোটর ও চাকা



চিত্র ১৩.২ : ডিসি মোটর

এবাব দেখো মোটরের মাথায় দুইটা পিন আছে।

এই পিন দুইটাকে দুইটা তার দিয়ে একটু তাতাল দিয়ে ঝালাই করে নিতে হবে।

সবচেয়ে ভালো হয় তোমার এলাকার ইলেকট্রনিক যন্ত্রপাতি ঠিক করার দোকান থেকে একটু ঝালাই করে নিলে।

ঝালাই করার পর তোমার মোটরটা দেখতে ১৩.২ চিত্রের মতন হবে।

এখন তুমি একটা মজার জিনিস লক্ষ করবে।

ঘোরাব এবাব একটা চাকা।

তুমি একটা ব্যাটারি নাও বা সরাসরি আরডুইনো থেকে 5V আর GND পিন এই মোটরের দুই মাথায় লাগিয়ে দাও। দেখবে চাকা একদিকে ঘূরছে। এবার এই তার দুইটা উল্টা করে লাগাও। মানে যেটায় 5V দিয়েছিলে সেখানে GND লাগাও আর যেখানে GND ছিল সেখানে 5V.

কী মজার ব্যাপারটা ঘটল দেখলে? চাকা উল্টা দিকে ঘূরছে।

তার মানে আমাদের আরডুইনো দিয়ে চাকার ঘোরার দিক ঠিক করে দিতে হবে।

আরেকটা বিষয় তোমরা হয়তো কাজ করতে করতে বুবোই গেছ যে আরডুইনোর এই পিনগুলো দিয়ে আমরা যেকোনো কিছু কন্ট্রোল করি। এবার তোমরা বলবে তাহলে এই তার দুইটাকে আরডুইনোর দুই পিনে লাগিয়ে দিয়ে কোড করে দিলেই তো হয়!

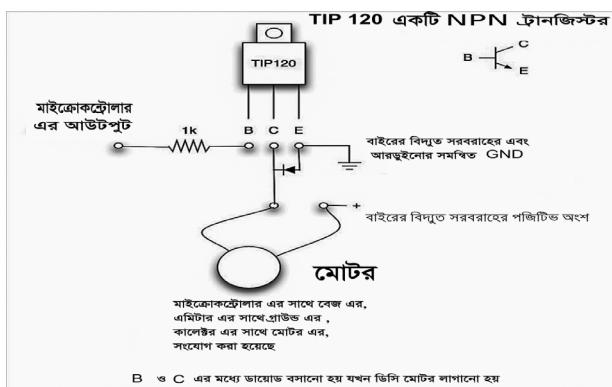
তোমরা একদম ঠিক! কিন্তু এই মোটরগুলোকে চালাতে আসলে অনেক কারেন্ট লাগে। এত কারেন্ট আরডুইনো দিতে পারে না। এ জন্য আমরা একটা ট্রানজিস্টর (**Transistor**) ব্যবহার করব। তাহলে মোটরগুলো অনেক শক্তি নিয়ে ঘূরতে থাকবে।

ট্রানজিস্টর বিভিন্ন রকমের হয়ে থাকে। বিভিন্ন কাজে এটার ব্যবহার আছে। তোমরা শুনে থাকবে রেডিওকে অনেকে ট্রানজিস্টর বলে থাকে। আসলে ট্রানজিস্টর রেডিওর একটা অংশ। এটা একধরনের সুইচ। একটা মজার ব্যাপার কি জানো, আমাদের কম্পিউটারগুলো সব অনেক ট্রানজিস্টর দিয়ে তৈরি। বিলিয়ন বিলিয়ন ট্রানজিস্টর! এমনকি আমাদের এই মাইক্রোকন্ট্রোলারটাও!



চিত্র ১৩.৪ : TIP120 ট্রানজিস্টর

চলো একটা ট্রানজিস্টরের ছবি দেখে নেওয়া যাক। এখানে আমাদের লাগবে **TIP120 Transistor**।



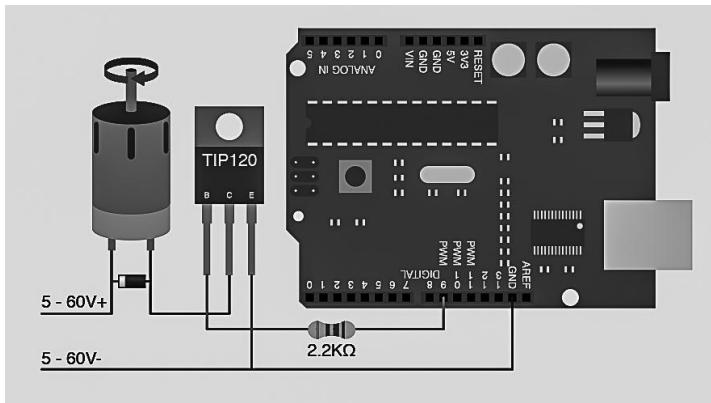
চিত্র ১৩.৫ : কানেকশনের ধাপ

ছবি দেখেই বুঝো গেছ ট্রানজিস্টরের তিনটা পা! এই তিন পায়ের কিন্তু নাম আছে। বাম থেকে ছবিতে দেখলে দেখবে **B E C** লেখা আছে।

ঘোরাব এবার একটা চাকা।

এই **B** হচ্ছে বেস, **E** হচ্ছে এমিটার আর **C** হচ্ছে কালেক্টর। তোমাকে এবার এই ট্রানজিস্টরটাকে ব্রেড বোর্ডে লাগাতে হবে। তারপর **E** পিনকে লাগাতে হবে গ্রাউন্ডে আর **C** পিন যাবে মোটরে। আর **B** পিন যাবে আরডুইনোতে।

কানেকশনের ছবিটা দেখে নিই এবার।



চিত্র ১৩.৬ : মোটর ঘোরানোর সার্কিট কানেকশন

এখানে মোটরের দুই মাথার মাঝে একটা ছোট ডায়োড (**Diode**) লাগিয়ে দেওয়া হয়েছে। কারেন্টকে একমুখী করার জন্য। ডায়োড, ট্রানজিস্টর সম্পর্কে তোমরা আরেকটু বড় ক্লাসে জানবে। আর এখন জানতে চাইলে গুগল তো আছেই।

এবার চলো যাই আইডিই অংশে এবং কোডটা লিখে ফেলি—

```

/* the transistor which controls the motor will
be attached to digital pin 9 */
int motorControl = 9;

void setup() {
    pinMode(motorControl, OUTPUT);
}

void loop() {
    // ramp up the motor speed
    for(int x = 0; x <= 255; x++){
        analogWrite(motorControl, x);
        delay(50);

    // ramp down the motor speed
    for(int x = 255; x >= 0; x--){
        analogWrite(motorControl, x);
        delay(50);
    }
    delay(1);
    // delay in between reads for stability
}

```

এবার কোডটা আপলোড করে দিলেই দেখবে চাকা ঘুরতে শুরু করে দিয়েছে।

অভিনন্দন! তুমি আরও একধাপ এগিয়ে গেলে! চলো এখন কোডের মধ্যে একটু উকিবুঁকি দিয়ে নিই।

প্রতিবারের মতোই আমরা মোটর চালাতে একটা ভেরিয়েবল নিয়ে নিয়েছি। এবং সেটা ৯ নম্বর পিনে কানেক্ট করেছি।

```

int motorControl = 9;
void setup() এর মধ্যে pinMode(motorControl, OUTPUT) |
মোটর যে আউটপুট ডিভাইস এটা আমরা বুঝিয়ে দিয়েছি রোবট মস্তিষ্ককে।

```

ঘোরাব এবার একটা চাকা।

এবার আসো void loop() অংশে। এর মধ্যে আমরা **for** নামে একটা লুপ চালিয়েছি। এটার

জন্য পুরো মোটর একবার ঘুরছে।

```
for(int x = 0; x <= 255; x++)
{
    analogWrite(motorControl, x);
    delay(50);
}
```

ঠিক একই কোড আবার উল্টা করে লিখেছি। এখানে গতির মান ২৫৫ থেকে কমে কমে ০ তে আসছে।

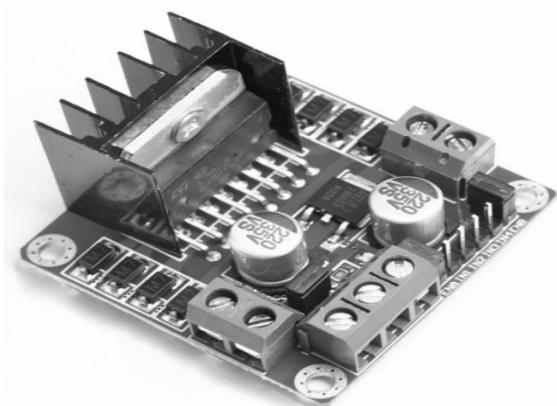
```
for(int x = 255; x >= 0; x--)
{
    analogWrite(motorControl, x);
    delay(50);
}
```

মাঝে delay() গুলো ব্যবহারে মোটরের ঘোরা অনেকটা স্থিতিশীল হয়েছে।

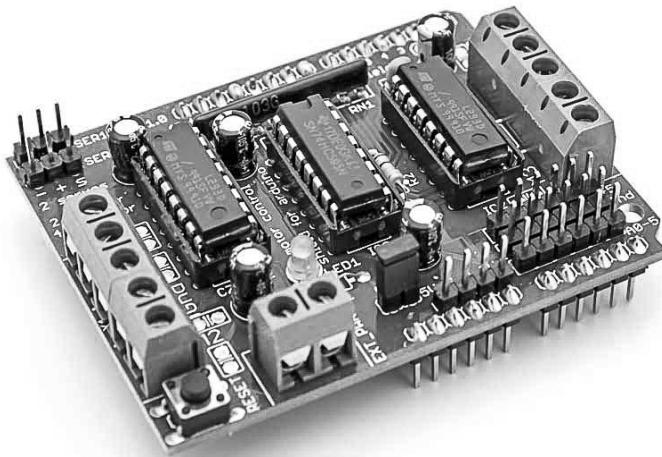
একটা চাকা ঘোরাতে পারলে তুমি দুইটা চাকা ও ঘোরাতে পারবে। চলো দেখে নিই এই চাকা দিয়ে বানানো কিছু রোবট।

বাজারে এ রকম রোবট বানানোর কিটও কিনতে পাওয়া যায়।

এ ছাড়া ট্রানজিস্টর দিয়ে আমরা মূলত একটা মোটর কন্ট্রোলার বানিয়েছি। এ রকম মোটর কন্ট্রোলার বাজারে কিনতেও পাওয়া যায়। যেমন ছবির কন্ট্রোলারের নাম **L298N** ডুয়াল এইচ ব্রিজ মোটর কন্ট্রোলার (**L298N Dual H Bridge DC Motor Controller**)



চিত্র ১৩.৭(ক) : L298N মোটর কন্ট্রোলা

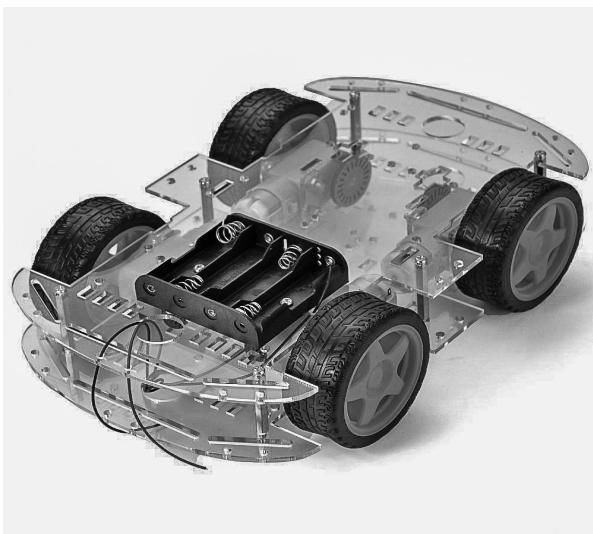


চিত্র ১৩.৭(খ) : L293d মোটর কন্ট্রোলার শিল্ড

তোমরা যখন রোবট বানাবে, তখন তোমরা মোটর কন্ট্রোলার শিল্ড বা মডিউল (**motor controller shield / motor controller module**) ব্যবহার করতে পারো। মডিউল বা শিল্ড ব্যবহারে কাজে অনেক সুবিধা পাওয়া যায়। মোটর কন্ট্রোলার শিল্ড কীভাবে কাজ করে সেই আলোচনা

ঘোরাব এবার একটা চাকা।

আমরা এই বইয়ের আওতাভুক্ত করিনি। এই বইয়ের ২য় খণ্ডে যোগ করার ইচ্ছে আছে।

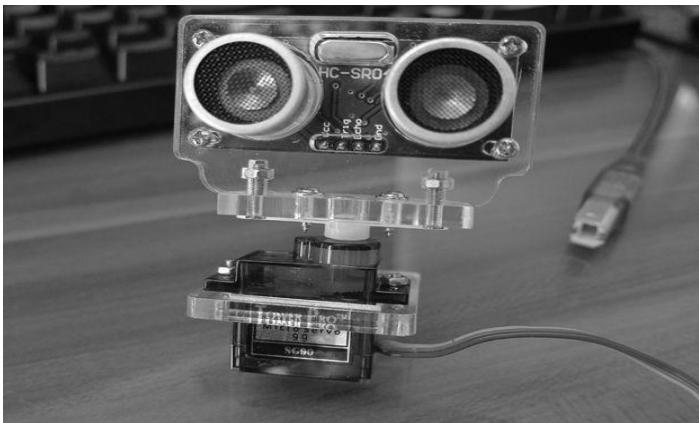


চিত্র ১৩.৮ : একটি রোবটিক কিট থেকে তৈরি চেসিস (Chassis)

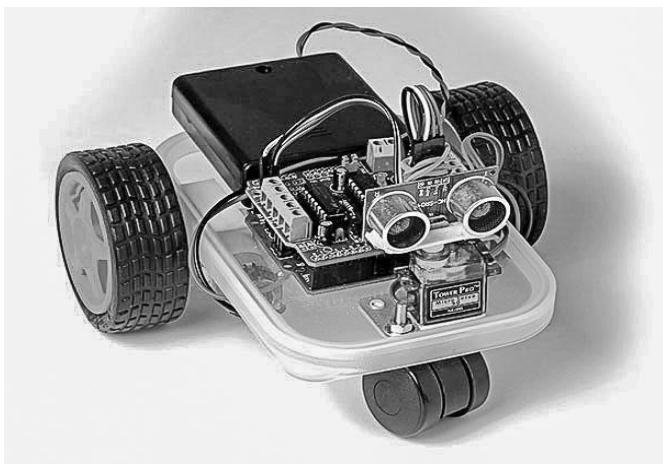
ঘটনা ১ : সার্ভো মোটর আর SONAR-এর কথা মনে আছে?

সেই সার্ভোর মাথায় লাগিয়ে দিয়েছে SONAR মডিউল। তারপর SONAR দিয়ে সামনে কিছু আছে কি না, সেটা দেখা হচ্ছে আর সার্ভো ঘাড়ের মতো মাথা ঘোরাচ্ছে।

দেখে মনে হচ্ছে রোবটের চোখ।



চিত্র ১৩.৯ : সার্ভোর সাথে সোনার (SONAR) দখতে চোখ আর ঘাড়ের মতো



চিত্র ১৩.১০ : আরডুইনো দিয়ে বানানো একটা রোবট

ঘটনা ২ : এখানে দেখো সামনের চাকা বানিয়েছে বোতলের কর্ক দিয়ে। চাইলে তোমরাও বানিয়ে ফেলতে পারো এ রকম রোবট! এই রোবট সামনে কিছু আছে কি না, সেটা প্রথমে বোঝার চেষ্টা করে। তারপর যেদিকে ফাঁকা স্থান পায়, সেদিকে এগিয়ে যায়। একে বলে **Obstacle Avoiding Robot**

যোরাব এবার একটা চাকা।

চৌদ

আরডুইনোতে পাওয়ার সাপ্লাই

আমাদের কাজ করার জন্য শক্তি প্রয়োজন। আমরা মানুষেরা সেই শক্তি পাই খাবার থেকে। তেমনি ইলেকট্রনিকস যন্ত্রের কাজ করার জন্য শক্তি প্রয়োজন। আর সেই শক্তি হলো বিদ্যুৎশক্তি।

আমাদের প্রিয় আরডুইনো চালানোর জন্যও প্রয়োজন বিদ্যুৎশক্তি। আরডুইনোর জন্য সবচেয়ে ভালো হলো ৭ থেকে ১২ ভোল্ট (Volt), আর ০.৫ থেকে ২ এম্পিয়ারের (Ampere) বিদ্যুৎপ্রবাহ।

তোমার মাথায় এখন কি একটা প্রশ্ন ঘুরছে — একবার বললাম ভোল্টেজ (Voltage), আরেকবার বললাম কারেন্ট (Current)

ভোল্টেজের একক হলো ভোল্ট,
কারেন্টের একক হলো এম্পিয়ার।

তোমরা জানো, বিদ্যুৎপ্রবাহ আসলে ইলেকট্রনের (**electron**) প্রবাহ। বৈদ্যুতিক তারের ভেতর দিয়ে ইলেকট্রন ছুটে চলে নিম্ন বিভবের দিক থেকে উচ্চ বিভবের দিকে। মানে, যেখানে ইলেকট্রন বেশি আছে সেদিক থেকে ইলেকট্রন ছুটে চলে যেদিকে ইলেকট্রন কম আছে সেই অঞ্চলে। এই যে একদিকে ইলেকট্রন বেশি পরিমাণে রয়েছে আরেক দিকে কম— একে বলে বিভব পার্থক্য, আর এই বিভব পার্থক্যের ফলে ইলেকট্রনের ছুটে চলার জন্য একটা চাপ সৃষ্টি হয়। ইলেকট্রনের ছুটে চলার জন্য এই চাপকে আমরা সহজ ভাষায় বলব ভোল্টেজ।

তোমরা দেখেছ, যেকোনো ব্যাটারির দুইটি প্রান্ত থাকে— পজেচিভ প্রান্ত, আর গ্রাউন্ড / নেগেচিভ।

এই দুই প্রান্তে কিন্তু ইলেকট্রনের পরিমাণের পার্থক্য থাকে।

আমরা যদি একটি ব্যাটারির দুই প্রান্তের সাথে একটা ছোট্ট বাতি সংযুক্ত করি তখন সেটা জুলে ওঠে। এই বাতিটা জুলার শক্তি তাহলে কোথা থেকে পায়? ইলেকট্রনের ছুটে চলা থেকে।

ব্যাটারির দুই প্রান্তের সাথে বাতি যুক্ত করলে সার্কিট পূর্ণ হয়। তখন ইলেকট্রন বেশি চাপের অঞ্চল থেকে, যে অঞ্চলে ইলেকট্রন কম সেদিকে প্রবাহিত হতে থাকে। আর ইলেকট্রনের প্রবাহ থেকেই বাতি জুলার শক্তি পায়। তারের ভেতর দিয়ে কী পরিমাণ ইলেকট্রন প্রবাহ হচ্ছে তাকে আমরা বলি **কারেন্ট (Current Flow)**।

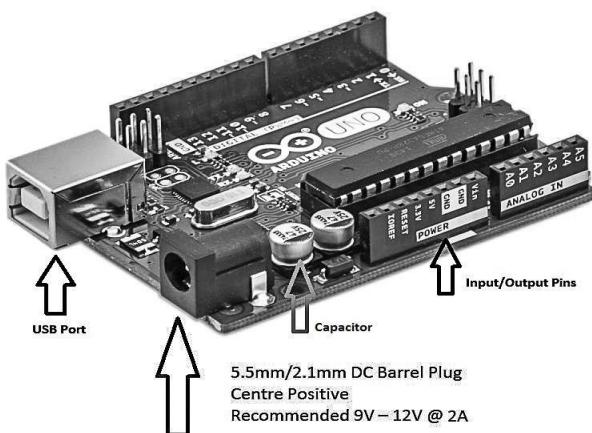
এক কথায় বলতে গেলে ইলেকট্রনকে এক স্থান থেকে অন্য স্থানে প্রেরণ করা ভোল্টেজের কাজ এবং কারেন্ট হচ্ছে ইলেকট্রনের প্রবাহ।

তোমরা ওপরে যে ব্যাখ্যাটি পড়লে, সেটিকে আমি বলব একটি '**Over Simplified Definition**' অর্থাৎ বোঝানোর সুবিধার্থে আমি অনেক কিছুকে অনেক বেশি সহজভাবে বলেছি। ভোল্টেজ এবং কারেন্ট সম্পর্কে তোমরা আরেকটু বড় ক্লাসে উঠলে তোমাদের পদার্থবিজ্ঞান বইয়ে পড়বে।



চিত্র ১৪.১ : একটি ৯ ভোল্ট ব্যাটারি

আমরা আরও সহজ ভাষায় ভোল্টেজ বলতে বুঝাব ব্যাটারির পজেটিভ প্রান্ত (+) আর নেগেটিভ/ গ্রাউন্ড প্রান্তের (-) মানের পার্থক্য। আর কারেন্ট বলতে বুঝাব তারের মধ্য দিয়ে কতটুক বিদ্যুৎ প্রবাহিত হচ্ছে তা।



চিত্র ১৪.২ : আরডুইনোতে পাওয়ার সাপ্লাই করার মতো বিভিন্ন পোর্ট

আমরা আরডুইনোকে ৩ উপায়ে পাওয়ার সাপ্লাই করতে পারি। ইউএসবি পোর্টের (USB Port) মাধ্যমে, ডিসি ব্যারেল প্লাগের (DC Barrel Plug) মাধ্যমে অথবা ইনপুট-আউটপুট পিনের (I/O Pin)

ইউএসবি পোর্টের মাধ্যমে :

তুমি যখন কম্পিউটারের ইউএসবি পোর্টের সাথে আরডুইনোকে যুক্ত করো, সেখান থেকে আরডুইনো **৫ ভেল্ট - ০.৫ মিলিএম্পিয়ার (mA)** বিদ্যুৎপ্রবাহ পায়। ছোটখাটো প্রজেক্ট করার জন্য এই পরিমাণ কারেন্ট যথেষ্ট। কিন্তু তুমি যখন মোটর চালাবে বা এমন কোনো সেন্সর চালাবে যাতে বেশি ভোল্টেজ প্রয়োজন, তখন কিছুটা সমস্যা হয়।

তা ছাড়া, সব সময় আরডুইনো কম্পিউটারের সাথে লাগিয়ে রাখাও সম্ভব নয়। তুমি যদি দেখানোর জন্য কোনো প্রজেক্ট বানাও সে ক্ষেত্রে ব্যাটারি দিয়ে বিদ্যুৎপ্রবাহ দিতে হবে।

শুরুর দিকের কাজগুলোর জন্য ইউএসবি পাওয়ার যথেষ্ট। তাই এই বইয়ের শুরুর অধ্যায়গুলোতে আমরা কম্পিউটারের ইউএসবি থেকে পাওয়ার নিয়েই আরডুইনো চালিয়েছি।

আজকাল অনেক অ্যান্ড্রয়েড ফোন চার্জারে এই ইউএসপি পোর্ট থাকে। তুমি চাইলে অ্যান্ড্রয়েড চার্জার থেকে তারটি খুলে, তোমার আরডুইনোর ইউএসবি সেখানে লাগিয়ে পাওয়ার সাপ্লাই নিতে পারো। তবে সে ক্ষেত্রে অবশ্যই খেয়াল রাখবে - আরডুইনোর ইউএসবি পোর্ট দিয়ে **৫ ভেল্ট (5V) - ০.৫ এম্পিয়ারের (0.5A)** বেশি বিদ্যুৎপ্রবাহ দেওয়া যাবে না। এই পথে অতিরিক্ত বিদ্যুৎপ্রবাহে আরডুইনো নষ্ট হয়ে যেতে পারে।

আরডুইনোতে পাওয়ার সাপ্লাই



চিত্র ১৪.৩ : ইউএসবি প্লাগ



চিত্র ১৪.৮ : ২১/৫৫ ডিসি ব্যারেল প্লাগ

ডিসি ব্যারেল প্লাগের মাধ্যমে :

বড় প্রজেক্ট করার সময় সবচেয়ে সুবিধা হয় এই প্লাগ দিয়ে কারেন্ট দিলে। তুমি চিত্রে খেয়াল করলে দেখতে পারবে ডিসি ব্যারেল প্লাগ চিহ্নিত করা রয়েছে। এই প্লাগ দিয়ে তুমি আরডুইনোকে ৯ থেকে ১২ ভোল্ট (9-12V) বিদ্যুৎপ্রবাহ দেবে। ৯-এর কম ভোল্টেজ বা ১২-এর

বেশি ভোল্টেজেও আরডুইনো চলতে সক্ষম। কিন্তু কম ভোল্টেজে বিভিন্ন সেন্সর ঠিকমতো কাজ করে না। আবার বেশি ভোল্টেজে আরডুইনো অতিরিক্ত গরম হয়ে নষ্ট হতে পারে। তাই ৯ থেকে ১২ ভোল্টের মধ্যে বিদ্যুৎপ্রবাহ দেওয়া শ্রেয়।

এই প্লাগ দিয়ে ০.৫ থেকে ২ এম্পিয়ার (**0.5-2A**) কারেন্ট দেওয়া হয়।

বাজারে বিভিন্ন ধরনের ব্যাটারি পাওয়া যায়। তুমি ছোটখাটো প্রজেক্টের জন্য ৯ ভোল্ট ব্যাটারি ব্যবহার করতে পারো। চিত্র .৭-এ একটি ৯ ভোল্ট ব্যাটারির ক্লিপের সাথে ব্যারেল প্লাগ দেখানো হয়েছে। ৯ ভোল্ট ব্যাটারি কেনার সময় দোকানদার আংকেলকে বললেই তোমাকে এই ক্লিপটা দিয়ে দেবেন।

তুমি যদি নিয়মিত আরডুইনো নিয়ে কাজ করতে চাও, সে ক্ষেত্রে তোমার জন্য পাওয়ার অ্যাডাপ্টার সবচেয়ে সুবিধার হবে। কারণ ব্যাটারি কিনে কিনে কাজ করাতে অনেক বেশি খরচ হয়।

অবশ্য তুমি চাইলে রিচার্জেবল (**Rechargeable**) ব্যাটারি ব্যবহার করতে পারো।

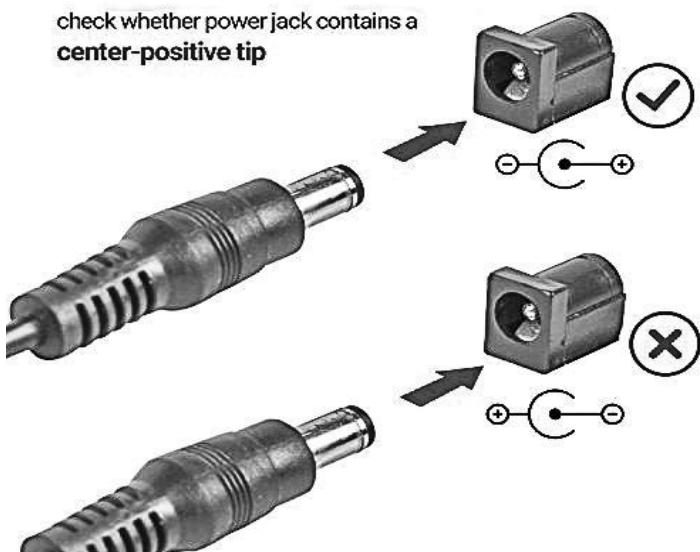
এসি টু ডিসি পাওয়ার অ্যাডাপ্টার (AC to DC Power Adapter) কেনার সময় যা খেয়াল করবে-

৯ থেকে ১২ ভোল্ট (9-12V)

১ থেকে ২ এম্পিয়ার (1-2A)

— এই রেটিংয়ের মাঝে একটি অ্যাডাপ্টার কিনবে। আর দেখে নেবে যেন প্লাগের ভেতর দিকটা পজেটিভ হয়।

check whether power jack contains a center-positive tip



চিত্র ১৪.৫ : ডিসি ব্যারেল প্লাগের ভেতর দিকটা পজেটিভ



চিত্র ১৪.৬ : ডিসি ব্যারেল প্লাগের সাথে যুক্ত একটি ৯ ভোল্ট ব্যাটারি ক্লিপ



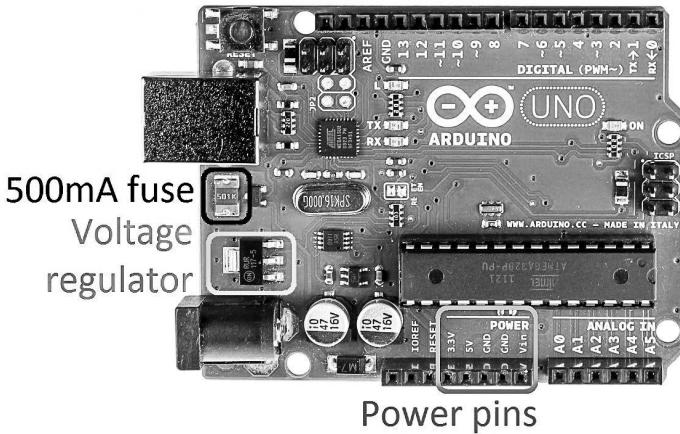
চিত্র ১৪.৭ : অ্যাডাপ্টার

আরডুইনোতে পাওয়ার সাপ্লাই

ইনপুট আউটপুট পিনের মাধ্যমে পাওয়ার সাপ্লাই
 আরডুইনোতে ‘**Vin**’ পিনের মাধ্যমেও তুমি পাওয়ার ইনপুট দিতে পারো। এই ‘**Vin Pin**’ আসলে ব্যারেল প্লাগের পজেটিভের সাথে শর্ট করা। শর্ট করা মানে, ভেতরে ভেতরে যুক্ত। সহজ কথায়, ব্যারেল প্লাগের পজেটিভ অংশ আর ‘**Vin Pin**’ আসলে একই জিনিস।

তার মানে ‘**Vin**’-এ তুমি ৯ থেকে ১২ ভোল্ট (9-12V) বিদ্যুৎপ্রবাহ দেবে। আর ০.৫ থেকে ২ এম্পিয়ার (0.5-2A) কারেন্ট প্রবাহ দেবে।

চিত্রে দেখো ‘**5V**’, ‘**3.3V**’ দুটি পিন আছে। এগুলো দিয়ে তুমি কিন্তু আরডুইনোতে পাওয়ার ইনপুট দেবে না। এখান থেকে তুমি আউটপুট নিয়ে কোনো সেন্সর চালাতে পারবে। আর ‘**GND**’ হলো গ্রাউন্ড বা নেগেটিভ প্রান্ত।



চিত্র ১৪.৮ : ফিউজ, ভোল্টেজ রেগুলেটর এবং পাওয়ার পিন

আরডুইনো বোর্ডে ভোল্টেজ রেগুলেটর (**Voltage Regulator**) নামের একটা জিনিস আছে। আমরা আরডুইনোকে ৫ থেকে ১২ বিভিন্ন ভোল্টেজে পাওয়ার সাপ্লাই দিলেও এর অভ্যন্তরের আইসি বা মাইক্রোকন্ট্রোলার একটা নির্দিষ্ট মানের বিদ্যুৎ চায়। আর এটাই হলো

ভোল্টেজ রেগুলেটরের কাজ। অর্থাৎ অতিরিক্ত ভোল্টেজকে কমিয়ে মাইক্রোকন্ট্রোলার বা আইসির জন্য পরিমিত ৫ ভোল্টে রূপান্তর করা।

আর ফিউজ (**Fuse**) নামের আরেকটি উপাদান আছে, যার কাজ হলো অতিরিক্ত কারেন্ট ফ্লো থেকে যন্ত্রপাতিকে রক্ষা করা। উদাহরণ দিই, আমরা বলেছিলাম ইউএসবি দিয়ে আরডুইনোতে ০.৫ এম্পিয়ারের বেশি প্রবাহ দেওয়া ঠিক নয়। যদি কখনো আমরা ভুলে এর চেয়ে অনেক বেশি এম্পিয়ার কারেন্ট দিয়ে ফেলি তখন? ফিউজ তখন সাময়িকভাবে আরডুইনোকে ভেতর থেকে বন্ধ করে দেবে। অতিরিক্ত কারেন্ট সরিয়ে স্বাভাবিক অবস্থায় এলে আবার ফিউজ চালু করে দেবে। আরডুইনোর এই ফিউজকে তাই বলে ‘**auto resettable fuse**’

আরডুইনোতে পাওয়ার সাপ্লাই

পনেরো
এবং তারপর?

আমাদের এই বইটা লেখার উদ্দেশ্য ছিল তোমাদের আগ্রহী মনকে আরও একধাপ এগিয়ে দেওয়া। পুরো বইয়ে আমরা চেষ্টা করেছি তোমাদের এই আরডুইনোর কাজকর্ম এবং সাথে কম্পিউটার প্রোগ্রামিংয়ের একটা ধারণা দিয়ে যেতে। আমরা চেষ্টা করেছি গল্লে গল্লে কঠিন বিষয়গুলোকে সহজভাবে তুলে ধরতে। তোমাদের সৃজনশীলতা যেন বিকাশ ঘটাতে পারো এ জন্য আমরা সম্পূর্ণ প্রজেক্টের সোর্স কোড অনেক সময় তোমাদের দিইনি। এবার তোমাদের কিছু উদাহরণ দেখাব, দেশে-বিদেশে কীভাবে এই আরডুইনো দিয়ে মানব জীবনের অনেক সমস্যা সহজভাবে সমাধান করা হয়েছে। এই অধ্যায়ের উদ্দেশ্য তোমার মস্তিষ্ককে প্রবলেম সলিভিং (Problem Solving) চিন্তাধারা বুঝতে শেখানো।

থার্স্টি ফ্লেমিঙ্গো

এই হাঁসটি হচ্ছে ত্রুফার্ট হাঁস। এর পেটের মধ্যে আছে ছোট একটা সার্কিট। আমরা অনেকেই ঘরের মধ্যে ছোটখাটো গাছ রাখি কিন্তু ঠিকমতো পানি দিতে ভুলে যাই। এই হাঁসের পেটের মধ্যে আছে একটা আর্দ্রতা মাপার সেন্সর।

গাছের যখন পানির প্রয়োজন হয়, তখন এই আর্দ্রতা সেন্সরে বিশেষ ধরনের মান পাওয়া যায়। তারপর সেখানে লাগানো বাজার (buzzer) বিপ বিপ করে শব্দ করতে থাকে, যাতে আমরা তখন গাছে পানি দিতে পারি!



চিত্র ১৫.১ : থার্স্টি ফ্লেমিঙো

ভিডিও : <https://goo.gl/zuCPzH>

বানাতে পারো : www.instructables.com/id/Thirsty-Flamingo

বিড়ালের জন্য খাবার

একটা বাসায় দুইটা বিড়াল ছিল। একটা মোটা আরেকটা পাতলা! দুটিই এক স্থান থেকে খাবার নিয়ে থেত। দিনে দিনে পাতলা বিড়াল আরও পাতলা হতে থাকল, আর মোটা বিড়াল আরও মোটা! মালিক তো চিন্তায় পড়ে গেলেন! তাই তিনি এমন একটা যন্ত্র বানালেন যার মধ্যে খাবার থাকে। আর বিড়ালের গলায় একটা **RFID (Radio Frequency IDEntification)** কার্ড ঝুলিয়ে দিলেন। এই কার্ড দিয়ে যন্ত্র বুঝতে পারে সামনে মোটা বিড়াল না পাতলা বিড়াল। এভাবে যন্ত্র তাদের খাবারের পরিমাণ ঠিক রাখে। দেখো কীভাবে আরডুইনো আমাদের জীবনের সমস্যার সমাধানে সাহায্য করছে!



চিত্র ১৫.২ : অটোমেটেড পেট ফিডার

ভিডিও : <https://youtu.be/7DwUOdVJp-8>
বানাতে পারো এখান থেকে <https://goo.gl/qet4Uq>

রাস্তার লাইট

রাতে আমরা রাস্তায় লাইট জুলতে দেখি। এখন রাস্তায় যদি কেউ নাও থাকে আমাদের লাইটগুলো কিন্তু জুলতেই থাকে। এমন যদি হতো, যখন কেউ ওই লাইটের কাছে আসবে শুধু তখনই লাইটটি জুলবে। এতে কিন্তু বিদ্যুতের অপচয় অনেক রোধ করা যেত।

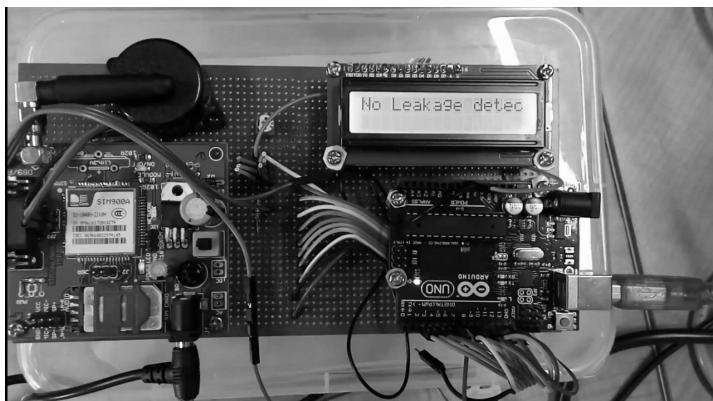


চিত্র ১৫.৩ : অটোমেটেড রাস্তার লাইট

ভিডিও <https://youtu.be/3ruXevD4Teo>
বানাতে পারো এখান থেকে <https://goo.gl/GBRd1P>

আরডুইনো গ্যাস লিকেজ ডিটেক্টর

আমাদের প্রায় সবার বাসাতেই প্রায় গ্যাসের সিলিন্ডার বা চুলা রয়েছে। অনেক সময় দেখা যায় বিভিন্ন ক্রটির জন্য গ্যাস লিক হয়ে বের হয়ে যায়। এটা কিন্তু বিশাল বড় অগ্নিকাণ্ডের মতো দুর্ঘটনা ঘটাতে পারে। আরডুইনো গ্যাস সেন্সর আর বাজার দিয়ে এ জন্য বানিয়ে ফেলা হয়েছে গ্যাস লিকেজ অ্যালার্ম। গ্যাস লিক হয়েছে বোঝামাত্রই এটা আমাদের সতর্ক করে দেবে।



চিত্র ১৫.৪ : আরডুইনো গ্যাস লিকেজ ডিটেক্টর

ভিডিও দেখো - <https://youtu.be/1pMPSws8v0U>

প্রজেক্টটি করতে চাইলে করতে পারো এখান থেকে <https://goo.gl/UdCCUi>

অথবা এখানে <https://goo.gl/9nL8P3>

দি থার্ড আই

এটা মূলত অঙ্কনের জন্য। তারা যাতে খুব সহজেই SONAR সেন্সর দিয়ে বানানো এ যন্ত্রটি দিয়ে সামনের বাধাগুলো শনাক্ত করে হাঁটা-চলা করতে পারে, সেই চেষ্টা করা হয়েছে।



চিত্র ১৫.৫ : দি থাৰ্ড আই

ভিডিও <https://goo.gl/V47bzZ>

বানাতে চাইলে <https://goo.gl/GXS32Y>

এভাবে মানবজীবনের সমস্যা সমাধানে তোমার অর্জিত জ্ঞানকে কাজে
লাগাও, এই শুভকামনাই রাইল তোমার জন্য!

ফিডব্যাক

আমাদের এই বইটা তোমাদের কাছে কেমন লেগেছে, সেটা আমাদের
জানাও নিচের অনলাইন ফরমটা পূরণ করে। এতে আমরা জানতে
পারব তোমরা কী চাও আৰ বইয়ের কোন কোন জায়গা আৱো ভালো
কৰা যায়।

আমাদের ফিডব্যাক পাঠাও এই লিংকে—

<https://goo.gl/215wDU>

পরিশিষ্ট

তোমার সবচেয়ে বড় সাহায্যকারী কে জানো? গুগল www.google.com

তারপর কিছু দরকারি ওয়েবসাইটের নাম বলে রাখি তোমাকে। তুমি যখনই ইন্টারনেটে কাজ করতে বসবে, ফেসবুক কিংবা নিছক বিনোদনে এটা-সেটা না ঘেঁটে এই সাইটগুলো �ঁাটবে। সারা দিনের একটানা বোরিং ফ্লাসের বাইরেও যে কত মজার মজার সব জিনিস আছে, তা এই সাইটগুলোতে পাবে—

<https://www.arduino.cc> <https://www.sparkfun.com/news>

<https://www.instructables.com>

<https://www.doityourself.com>

<http://fritzing.org> www.hwtomechatronics.com

www.draw.io <https://www.tinkercad.com>

এই বইয়ে মূলত আরডুইনোর বেসিক কাজকর্ম নিয়ে আলোচনা করা হয়েছে। সামনে আমরা তোমাদের জন্য অ্যাডভান্স আরডুইনোর ওপরে বই নিয়ে আসব। আপাতত আরডুইনোর চর্চা এভাবে করতে থাকো। ইন্টারনেটে আরডুইনোর ব্রগগুলোতে চোখ রাখো। বাইরের দেশের বাচ্চাকাচ্চা আরডুইনো দিয়ে কী বানাচ্ছে দেখো। নিজেও বানিয়ে ফেলো বিশ্ব কাঁপিয়ে দেওয়ার মতো নতুন কিছু! আরডুইনোকে ভালোবাসো।

হ্যাপি আরডুইনোইঁ!

তথ্যসূত্র

এই বই লিখতে আমরা প্রচুর ইন্টারনেট ঘেঁটেছি। আরডুইনোর ওপর লেখা কিছু বইও পড়েছি। আর সবচেয়ে গুরুত্বপূর্ণ হলো, এক বছর ধরে ধীরে ধীরে আরডুইনোভিত্বিক অনেকগুলো প্রজেক্ট নিজ হাতে করেছি।

Google আর Youtube ঘেঁটে অনেক কিছু নিজে নিজে শিখেছি।

আরও সাইট যেমন— Programming Electronics Academy (<https://goo.gl/Mb95Cq>)

আর আছে আরডুইনোর অফিশিয়াল ওয়েবসাইট www.arduino.cc। এই ওয়েবসাইট থেকে তুমি আরডুইনোভিত্বিক প্রচুর রিসোর্স, টিউটোরিয়াল, ফোরাম পাবে।

এই বইটা শেষ করার পর তোমার কাজ হবে, তুমিও এসব রিসোর্স ঘেঁটে নিজে নিজে শেখা শুরু করবে।

এই বইয়ের কিছু কিছু কোডের লিংক দিলাম। এখান থেকে তুমি চাইলে কপি পেস্ট করে নিয়ে কাজ করতে পারো। কিন্তু অবশ্যই বুঝে কপি পেস্ট করবে। না বুঝে কপি করলে শিখতে পারবে না।

এলইডি ব্লিংকিংয়ের কোড

<https://goo.gl/cJmjQb>

অনেকগুলো এলইডি

<https://goo.gl/WdfpGm>

বাজারের কোডগুলো

- a) <https://goo.gl/Fj2vF3>
- a) <https://goo.gl/1nQoai>
- b) <https://goo.gl/8VEUc7>

পুশবাটন নিয়ে যত কোড

- a) <https://goo.gl/yoQkqV>
- b) <https://goo.gl/sj4Egm>

পটেনশিওমিটারের কোড

<https://goo.gl/UptqqT>

টেস্পারেচার সেন্সর

<https://goo.gl/87bY94>

পুতুলটি

<https://goo.gl/t3NhjS>

চাকা

<https://goo.gl/nsNuo6>

নির্ণট

***4th Industrial Revolution-** প্রযুক্তিক্ষেত্রে ব্যাপক উত্তাবনের কারণে বিশ্বজুড়ে যে বিপ্লব ঘটে চলেছে, তাকে বিভিন্ন বোন্দারা চতুর্থ শিল্প বিপ্লব বলে আখ্যায়িত করছেন। *The Fourth Industrial Revolution* বইয়ে বিশ্ব অর্থনৈতিক ফোরামের প্রতিষ্ঠাতা ক্লাউস শোয়াব এ ব্যাপারে বিস্তারিত ব্যাখ্যা করেছেন।

২০১৬ সালে সুইজারল্যান্ডে অনুষ্ঠিত বিশ্ব অর্থনৈতিক ফোরামের বার্ষিক সম্মেলনের প্রতিপাদ্য বিষয় ছিল : চতুর্থ শিল্প বিপ্লব নিয়ন্ত্রণ (Mastering the Fourth Industrial Revolution)।

বইতে ব্যবহৃত উল্লেখযোগ্য ইংরেজি শব্দ

Analog — কম্পিউটারের ব্যবহার ছাড়া করা কাজ এমন

Anode — ধনাত্মক তড়িৎদ্বার

Arduino — ইতালিয়ান এক কোম্পানির বানানো ডেভেলপমেন্ট বোর্ড

Arduino Mega — আরডুইনোর একটি মডেল, সাইজে বড়, পিনসংখ্যা বেশি, বড় প্রজেক্ট করা যায়

Arduino Nano — খুব ছোট প্রজেক্ট করতে কাজে লাগে, আরডুইনোর একটি মডেল

Arduino Uno — আরডুইনোর একটি মডেল

Automation — স্বয়ংক্রিয়করণ

Base(Transistor) — ট্রানজিস্টরের ভিত্তি (একটি অংশ)

Bread Board — একধরনের প্রোটোটাইপ বোর্ড

Brightness — উজ্জ্বলতা

Capacitor — ধারক

Cathode — খণ্ডাত্মক তড়িৎদ্বারা

Celsius — তাপমাত্রার একক

Circuit — বর্তনী

Clock Pulse — একধরনের তরঙ্গ

Code/Source Code — কম্পিউটারের প্রোগ্রাম বা ইন্সট্রাকশন

Collector — ট্রানজিস্টরের অংশ (সংগ্রাহক)

Compilation — কম্পিউটার প্রোগ্রামকে মেশিন কোডে রূপান্তরের উপায়

Compile error — কম্পাইলেশন চলাকালে সোর্স কোডে যে ভুলগুলো ধরা পড়ে

Computer Program — কম্পিউটারের জন্য নির্দিষ্ট কিছু নির্দেশনা

Contrast — পার্থক্য

Crystal — আরডুইনোতে ব্যবহৃত অসিলেটর, যা মাইক্রোকন্ট্রোলারকে চালু করতে প্রয়োজন পড়ে

Curly braces — ‘{ }’

Current — তড়িৎশক্তি

DC Gear Motor — একধরনের চাকাবিশিষ্ট মোটর, এদের গিয়ার বলে। এরা যান্ত্রিক শক্তিকে বাড়িয়ে দেয়।

Declare — ঘোষণা করা, ধরা

Digital Pin — আরডুইনোর পিন, শূন্য অথবা এক নিয়ে কাজকর্ম

Diode — ডায়োড। দুইটা ডায়োড দিয়ে একটা ট্রানজিস্টর তৈরি। আর কম্পিউটারের একক ট্রানজিস্টর

Documentation in code — কোন কম্পিউটার প্রোগ্রাম করে কীভাবে কী জন্য লিখা হয়েছে, তা ‘//’ চিহ্ন মধ্যে লিখে রাখা। এতে সহজে পরে দেখে বোঝা যায়, অন্যরাও বুঝতে পারে।

ECHO — প্রতিধ্বনি

Electron — পরমাণুর অংশ

Emitter — ট্রানজিস্টরের অংশ, নিঃসারক

Farenheit — তাপমাত্রার একক

Frequency in Sound — শব্দতরঙ্গের কম্পাক্ষ

Function — নির্দিষ্ট কর্ম

Genuino UNO — শুধু আমেরিকায় পাওয়া যায় এমন আরডুইনো

Google — www.google.com

Google Search — গুগলে সার্চ দেওয়া

Hertz — ফিকোয়েন্সির একক

Hot Glue Gun — একধরনের প্লাস্টিকের আঠা

Infrared Light — আলোক তরঙ্গ

Install — স্থাপন করা

Instruction — নির্দেশনা

Integer — পূর্ণসংখ্যা

Internet — ইন্টারনেট

LCD — লিকুইড ক্রিস্টাল ডায়োড (Liquid Crystal Diode)

LED — লাইট এমিটিং ডায়োড (Light Emitting Diode)

Logic — যুক্তি, কম্পিউটারকে নির্দেশ যুক্তি দিয়ে বোঝাতে হয়

Open Source — মুক্ত সফটওয়্যার বিপ্লবের অংশ

OTG — একধরনের ইউএসবি পোর্ট (USB On The G)

Parallel Connection — সমান্তরাল বর্তনী

PIR motion sensor — গতি নির্দেশক সেন্সর

Raspberry Pi — ডেভেলপমেন্ট বোর্ড এবং একটি পূর্ণাঙ্গ কম্পিউটার

Reset — পুনরায় কোনো মান বসানো

Resistance — রোধ, বাধা

Return type — প্রোগ্রামিংয়ে ফাংশন যে ডেটা টাইপ রিটার্ন করে

Sensor — বিভিন্ন প্রাকৃতিক ঘটনার মান নির্ণয়ক যন্ত্র

Series Connection — শ্রেণি বর্তনী

Server — একধরনের কম্পিউটার

Servo — একধরনের বেশ শক্তিশালী মোটর

Signal — তরঙ্গ

Sketch — আরডুইনোর কোডগুলোকে বলে ক্ষেচ

Transistor — ট্রানজিস্টর একটি অর্ধপরিবাহী কৌশল, যা সাধারণত অ্যাম্পলিফায়ার এবং বৈদ্যুতিকভাবে নিয়ন্ত্রিত সুইচ হিসেবে ব্যবহৃত হয়। দ্রুত সাড়া দেওয়ার ক্ষমতা এবং সঠিক সম্পূর্ণ সঠিকভাবে কার্যসাধনের ক্ষমতার কারণে এটি আধুনিক ডিজিটাল বা অ্যানালগ যন্ত্রপাতি তৈরিতে বহুল ব্যবহৃত হচ্ছে। নির্দিষ্ট ব্যবহারগুলোর মধ্যে রয়েছে ইলেক্ট্রনিক অ্যাম্পলিফায়ার, সুইচ, ভোল্টেজ নিয়ন্ত্রক, সংকেত উপরোক্ত ও ওসিলেটর। আলাদাভাবে ট্রানজিস্টর তৈরি করা যায়। আবার সমন্বিত বর্তনীর অভ্যন্তরে একটি অতি ক্ষুদ্র স্থানে কয়েক মিলিয়ন পর্যন্ত ট্রানজিস্টর সংযুক্ত করা যায়।

TRIG SONAR — এর ট্রিগার পিন, এখান থেকে শব্দতরঙ্গ নিষ্কেপিত হয়

USB — Universal Serial Bus (কম্পিউটারের তথ্য আদান-প্রদানে ব্যবহৃত একধরনের ইন্টারফেস)

Variable — চলক

VCC — ভোল্টেজ ফর্ম কালেক্টর, ইলেক্ট্রনিকসে পাওয়ার সাপ্লাইয়ের ধনাত্মক প্রান্ত

Void — শূন্য

Voltage HIGH — HIGH দিয়ে আরডুইনোতে ৫ ভোল্ট বোর্কানো হয় বা কম্পিউটারের এক।

Volt/Voltage — বিভব

