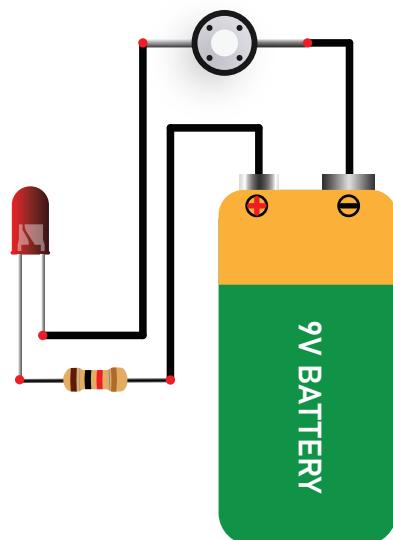


৩/ পুশ বাটন এবং ডিজিটাল রিডিং:

প্রয়োজনীয় উপকরণ:

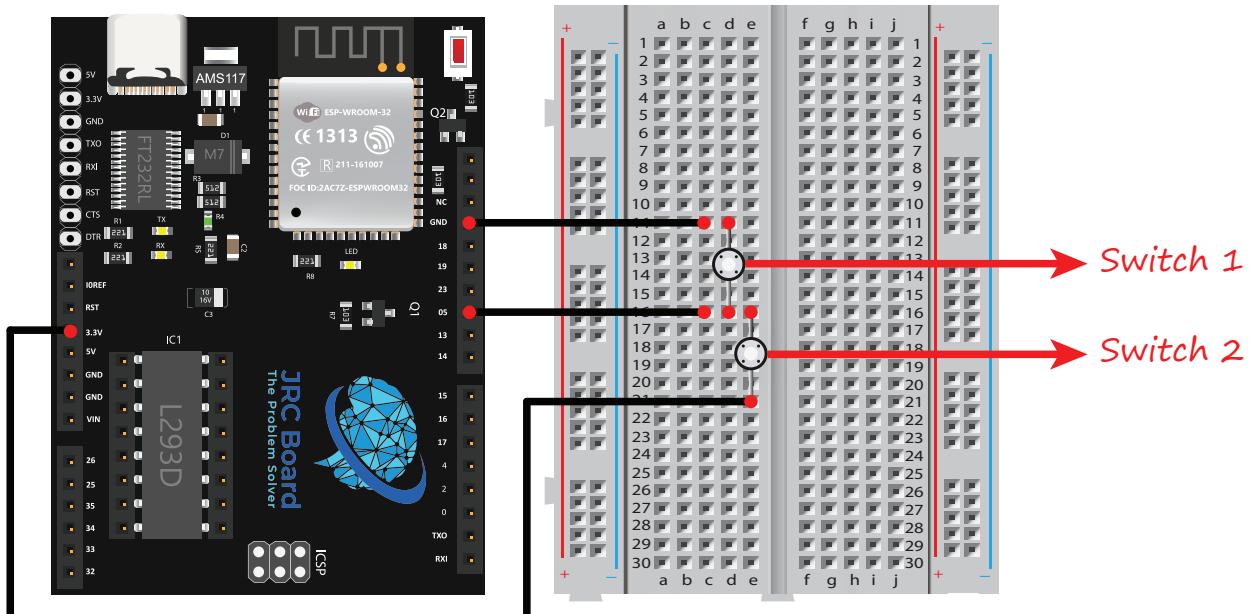
- ১/ পুশ বাটন
- ২/ ৱেডবোর্ড
- ৩/ জাম্পার ওয়্যার
- ৪/ JRC বোর্ড
- ৫/ এলইডি লাইট
- ৬/ ৯ভার্টেন্স

বর্ণনা: পুশ বাটন কে অনেকে মোমেন্টারি সুইচও বলে থাকে। বাজারে অনেক ধরণের পুশ বাটনই পাওয়া যায়। কিন্তু সবগুলোর কাজ কিন্তু একই ধরনের, চাপ দিলে এর যেকোন দুই প্রান্তের বর্তনী পূর্ণ হয়ে যায় এবং ছেড়ে দিলে প্রান্ত দুটির মধ্যে আর কোন প্রকার সংযোগ থাকেনা। আমরা যে সুইচ দিয়েছি সেখানে কেবল দুটি মাথা থাকে যা বাটন প্রেস করলে নিজেদের মধ্যে তড়িৎ বর্তনী সম্পূর্ণ করে ফেলে। নিচে একটা উদাহরণ দেখানো হচ্ছে:



এখানে যে সার্কিট টি দেখতে পাচ্ছো, তার সাথে কিন্তু চির ১.৮ এর প্রায় মিল রয়েছে। কেবল এলইডির খাগড়াক প্রান্তের সাথে ব্যাটারির খাগড়াক প্রান্তের সরাসরি সংযোগ না হয়ে মাঝে একটি পুশ বাটন সিরিজ সংযোগে বসেছে। যখন পুশ বাটনে কোনপ্রকার চাপ দেয়া হয়না, তখন এর দুই প্রান্তের বর্তনীর কোনপ্রকার সংযোগ না থাকায় লাইট টি জ্বলেনা। কিন্তু যেই মুহূর্তে চাপ দেয়া হয়, তখন বর্তনী পূর্ণ হয়ে যায় এবং লাইট টি জ্বলতে শুরু করে। ব্যাপারটা অনেকটা ব্রিজের মতো। ব্রিজ তুলে দিলে তড়িৎ আসা-যাওয়া করতে পারেনা, চাপ দিলে ব্রিজের সংযোগ স্থাপন হয় বলে তড়িৎ আসা-যাওয়া করতে পারে।

JRC বোর্ডে যেকোন GPIO পিন থেকে ডিজিটাল রিডিং নিতে এই পুশ বাটন ব্যবহার করবো। ডিজিটাল রিডিং পরীক্ষা করতে নিচের সার্কিট টি তৈরী করিঃ



এই সার্কিটে আমরা 05 পিনে ডিজিটাল রিডীং নেবার ব্যবহৃত করছি। এর সাথে একটি পুশ বাটনের দ্বারা GND পিনের সংযোগ দেয়া আছে এবং আরেকটি পুশ বাটনের দ্বারা 3.3V পিনের সাথে সংযোগ দেয়া আছে। আমরা যদি প্রথম পুশ বাটনে চাপ দিই, তবে দেখা যাবে যে এর সাথে বোর্ডের প্রাউফ্রেডের সরাসরি সংযোগ হয়ে গিয়েছে যার ফলে এর মধ্যে 0 ভোল্ট পাবে। আবার ২য় পুশ বাটনটিতে চাপ দিলে দেখা যাবে যে এর সাথে 3.3V পিনের সরাসরি সংযোগ হয়ে গেছে। এর ফলে 05 নাম্বার পিনে 3.3 ভোল্ট পাবে।

ডিজিটাল রিডিং মানে কি? এখানে কেবল দুই ধরনের অবস্থা রিডিং নিতে পারবে - ভোল্টেজ আছে, অথবা ভোল্টেজ নেই। যদি বোর্ডের যেকোন পিনে 1.5 ভোল্ট বা তার থেকে বেশি ভোল্টেজ দেয়া হয়, তবে সে ধরে নিবে যে ঐ পিনে ডিজিটাল রিডিং স্টেট HIGH। 1.5 ভোল্ট থেকে কম ভোল্টেজ পেলে ধরে নিবে যে তার মধ্যে কোনওকার ভোল্টেজ আসেইনি। অর্থাৎ তার মধ্যে ডিজিটাল রিডিং স্টেট LOW ধরে নিবে। চলো আমরা এর কোডিং এর দিকে একটু নজর দিইঃ

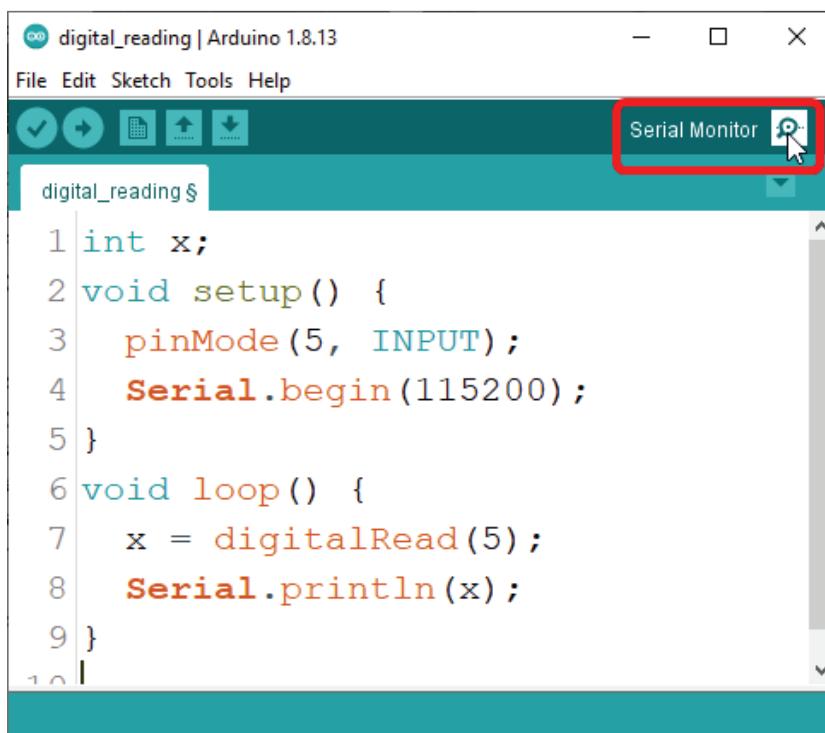
```
int x;
void setup(){
    pinMode(5, INPUT);
    Serial.begin(115200);
}

void loop(){
    x= digitalRead(5);
    Serial.println(x);
}
```

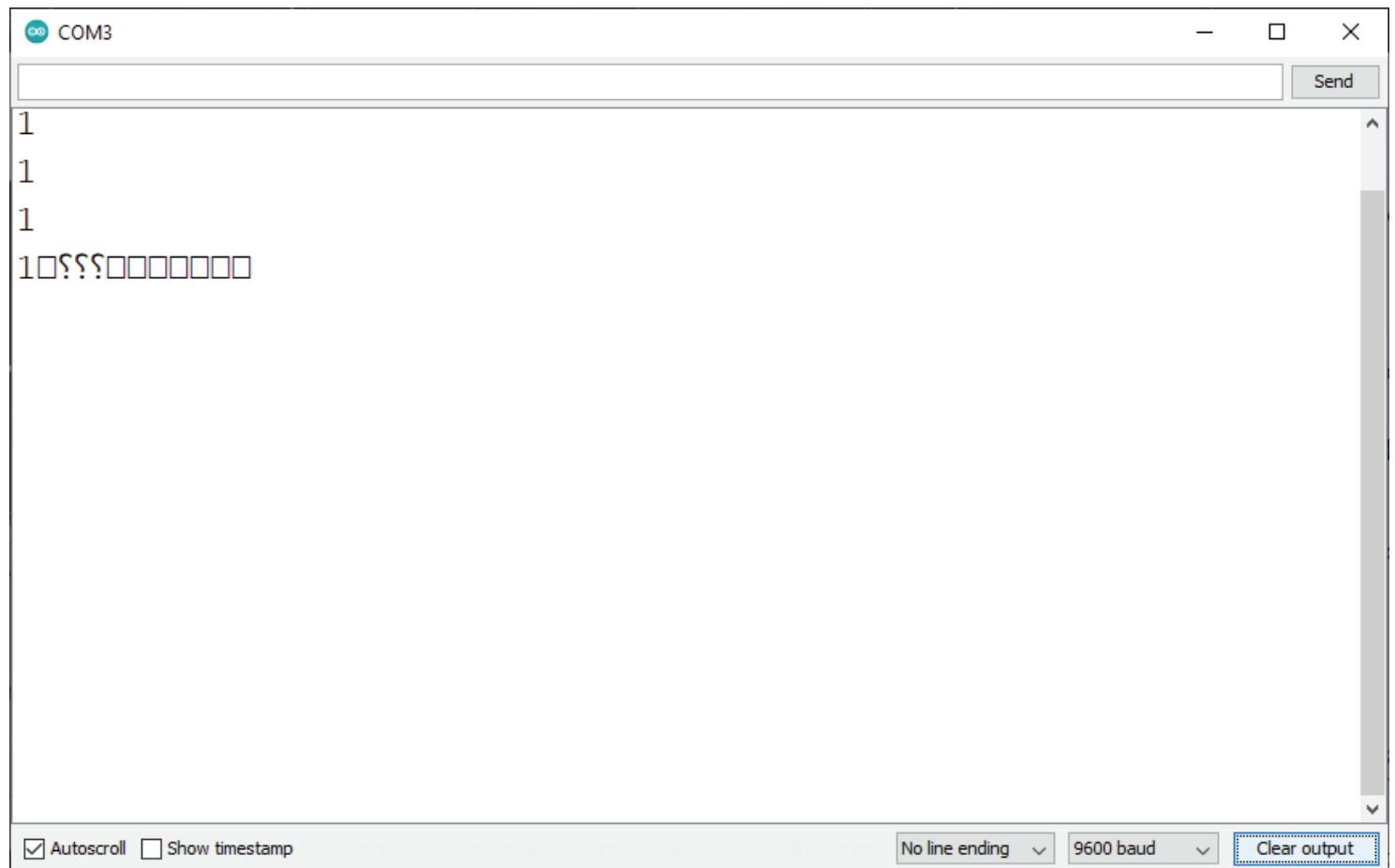
এখানে যেহেতু 5 নাম্বার পিনে ডিজিটাল রিডিং নেয়া হচ্ছে, তাই **setup** ফাংশনের ভেতরে অবশ্যই এর ভূমিকা নির্ধারণ করে দিতে হবে। যেহেতু এতে ডিজিটাল রিডিং এর জন্য ইনপুট নিতে হবে, তাই আমরা শুরুতে **pinMode(5, INPUT)** লিখে দিয়েছি। চাইলে আমরা এই রিডিং এর মান কম্পিউটার মনিটর এ দেখাতে পারি। এর জন্য সিরিয়াল কমিউনিকেশন এর দরকার পড়ে। এই কমিউনিকেশন টা শুরু করার জন্য আমরা **setup** ফাংশনের ভেতরে লিখে রাখি যে **Serial.begin()**। এখানে ব্র্যাকেটের ভেতরে একটি মান লিখব যেটি কিনা সে সোজা কথায় চ্যানেল হিসেবে ধরে নেয়। কম্পিউটার থেকে এক্সেন্ট এ চ্যানেলে রিডিং না নিলে কোন কিছুই শো করবেন। **ESP32** সাধারণত 115200 বডরেটে কিছু ডাটা পাঠায় বলে আমরা আপাতত **Serial.begin(115200)** লিখে দিয়েছি।

এবার **loop()** ফাংশনের ভেতরে কি লেখা আছে সেদিকে নজর দিই। প্রথমেই ডিজিটাল রিডিং এর জন্য আমরা **digitalRead()** ফাংশনটি ব্যবহার করি যেটির ব্র্যাকেটের ভেতরে যেই পিন থেকে রিডিং নিতে চাচ্ছ সেই পিন নাম্বার লিখে দিতে হয়। এরপর এর থেকে আউটপুট পাবো যে পিন এর রিডিং হাই পাচ্ছে নাকি লো পাচ্ছে। এই আউটপুট স্টোর করে রাখার জন্য আমরা একটি গ্লোবাল ভ্যারিয়েবল ডিক্লেয়ার করে রেখেছি যার নাম দেয়া হয়েছে X। এখন X এর ভেতরে যে মান সেভ হয়েছে, তা যদি আমরা কম্পিউটারফের স্ক্রিনে দেখতে চাই, সেক্ষেত্রে সিরিয়াল আউটপুট পেতে হবে বোর্ডের কাজ থেকে। এর জন্য আমরা **Serial.print()** নামক একটি ফাংশন ব্যবহার করেছি যেটির ভেতরে যে ভ্যারিয়েবল এর মান দেখতে চাই, সে ভ্যারিয়েবল এর নাম লিখে দিই। যদি আমরা লেখা টা পাশাপাশি না দেখে প্রতিবার নতুন লাইনে দেখতে চাই, সেক্ষেত্রে **Serial.println()** ফাংশনটি ব্যবহার করা যায়।

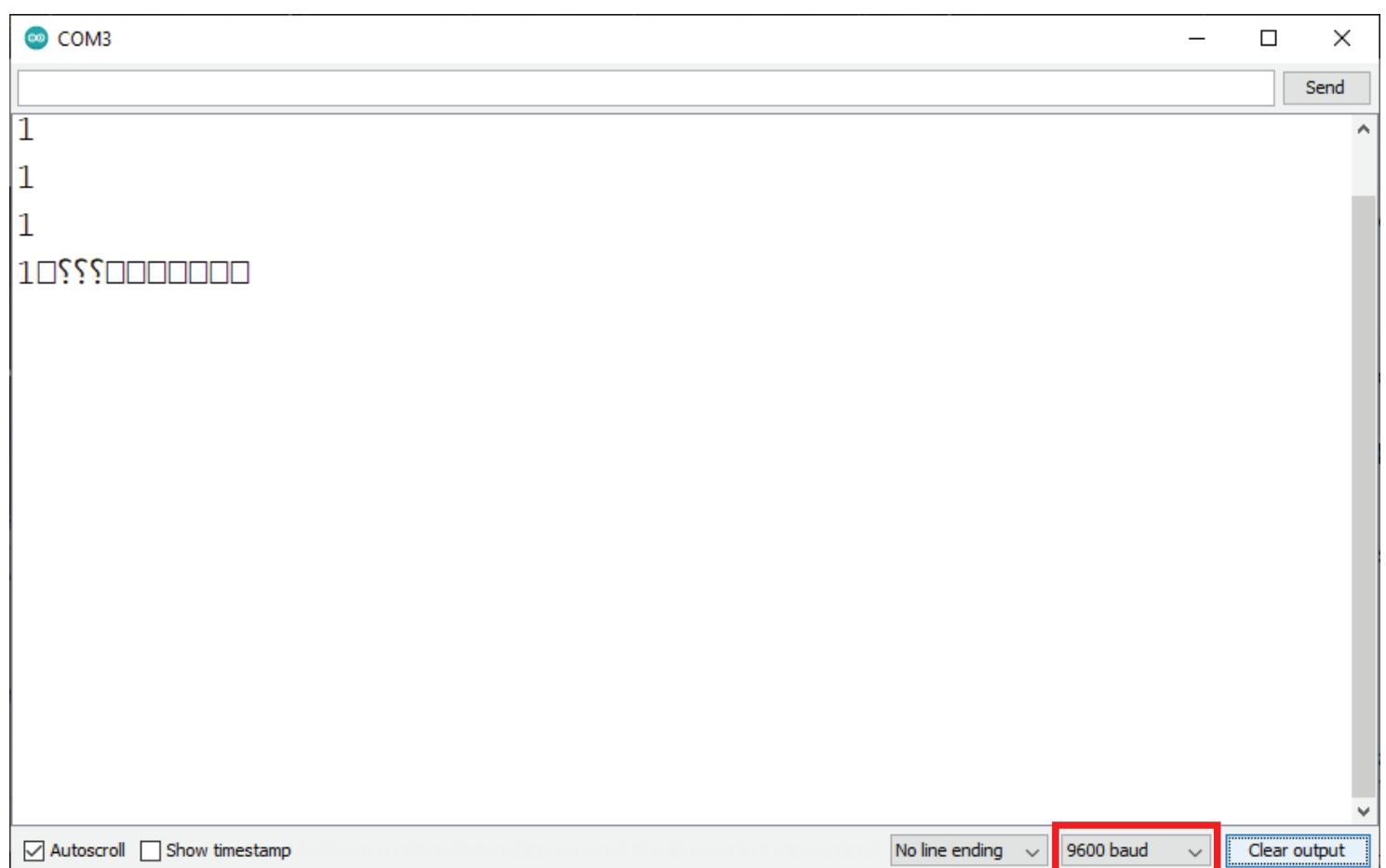
কোড টি আপলোড দিয়ে সিরিয়াল মনিটর ওপেন করতে নিচের ছবি অনুযায়ী বাটনে ক্লিক করি অথবা **ctrl+shift+m** প্রেস করি।



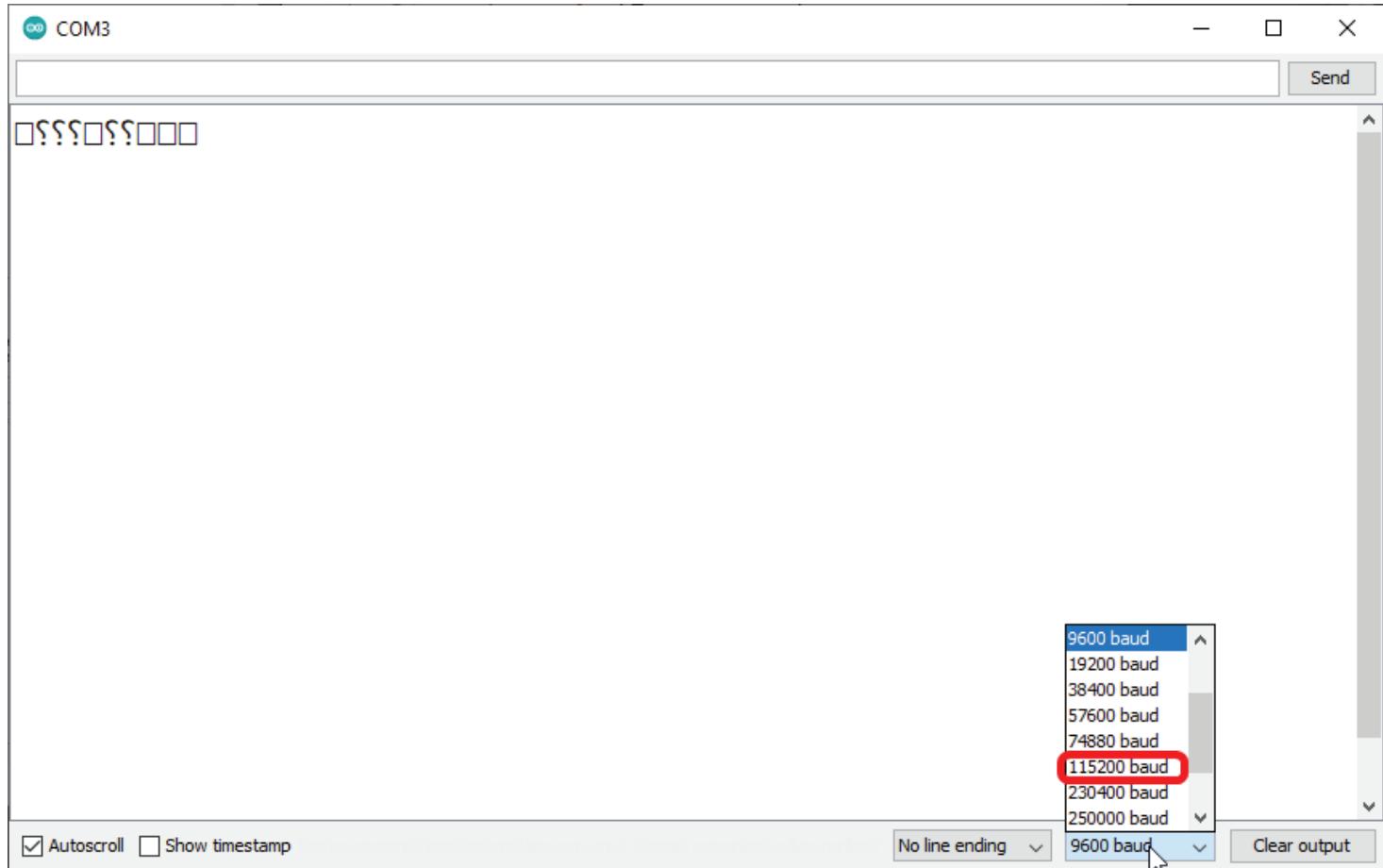
ক্লিক করার পর নিচের মতো একটি উইন্ডো ওপেন হবে।



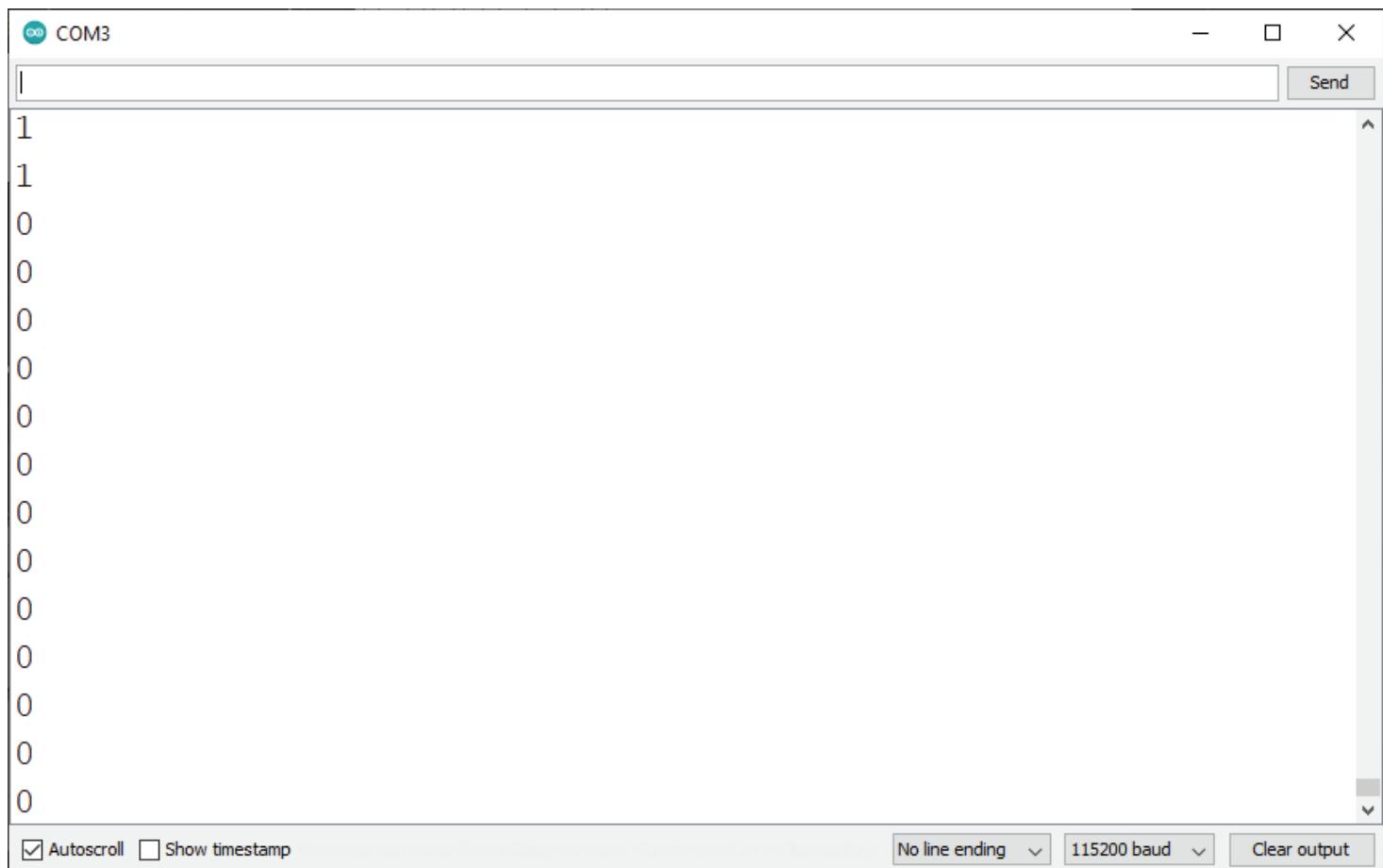
এখানে দেখতে পাচ্ছো যে মনিটরে দুর্বোধ্য কিছু চিহ্ন দেখাচ্ছে। এর কারণ হচ্ছে আমরা শুরুতে `Serial.begin()` ফাংশনের ব্র্যাকেটের ভেতরে বডরেট হিসেবে 115200 দিয়েছিলাম। কিন্তু যখন আমরা সিরিয়াল মনিটরে আউটপুট দেখতে চাচ্ছি সেখানে বডরেট 9600 দিয়ে রাখা।



এই ব্যাপার টি সমাধা করতে তুমি সেখানে ক্লিক করে সঠিক বডরেট টি সিলেক্ট করে নিতে পারো।



এখন তুমি অনবরত ডাটা পেতে থাকবে সুন্দরমতো।



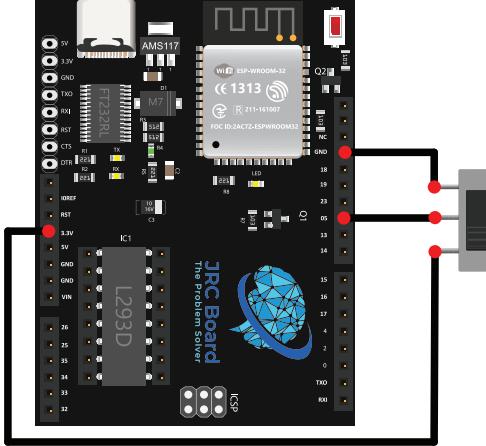
এখানে যদি প্রথম সুইচ টি চেপে ধরে রাখো তবে চিন্তা করে দেখো কি হতে পারে। বোর্ডের ৫ নং পিনের সাথে আমরা সুইচের মাধ্যমের গ্রাউন্ড পিনের সরাসরি সংযোগ দিচ্ছি। এর ফলে এখানে সবসময় ০ ভোল্ট পাচ্ছে। এখন I0OP() ফাংশনের ভেতরে প্রথমে আমরা সেই 5 নং পিন থেকেই ডিজিটাল রিড নিয়ে সেই ভ্যালু X নামক ভ্যারিয়েবলে স্টোর করছি। এখন উক্ত পিনে যদি সবসময় ০ ভোল্ট পায়, তবে X ভ্যারিয়েবল এর ভেতরে সমসময় 0-ই সেভ হতে থাকবে। এবং সিরিয়াল মনিটরে আমরা যতোবারই সেই ভ্যারিয়েবল এর মান দেখাতে যাই, ততবারই ০ প্রদর্শন করবে। এর ফলে আমরা সিরিয়াল মনিটরে উপর থেকে নিচে ০ গেতে থাকবো। যদি দেখো যে শুরুতে আবারও সেই দুর্বোধ্য লেখা আসছে, তবে বোর্ডের রিসেট বাটনে একবার প্রেস করলেই হয়ে যাবে।।

একই ভাবে যদি তুমি ১ নং সুইচ ছেড়ে দিয়ে ২ নং সুইচে চাপ দাও, সেক্ষেত্রে উক্ত পিনের সাথে ৩.৩ ভোল্ট এর সরাসরি সংযোগ স্থাপন হবে। তখন রিডিং এর মান সবসময় ১ দেখতে পাবে।

কিন্তু তুমি যদি কোন সুইচই চাপ না দাও, সেক্ষেত্রে কি হবে? পিন টি তো গ্রাউন্ডের সাথেও সংযোগ নেই আবার ৩.৩ ভোল্টের সাথেও সংযোগ নেই। সেক্ষেত্রে এর মান ০ ও ১ মধ্যে উঠানামা করতে থাকবে এবং তোমরা চিত্র ৩.৮ এর মতো গারবেজ রিডিং পেতে থাকবে।

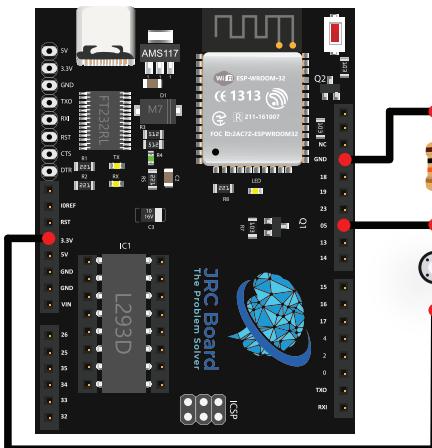
এখন উপায়?

এখানে এমন ব্যবস্থা করতে হবে, যাতে করে এটা চাপ দিলে এক প্রান্তে, এবং ছেড়ে দিলে অপর প্রান্তে যুক্ত হয়। এর জন্য কয়েকটি বর্তনী নিচে উদাহরণস্বরূপ দেখাচ্ছি:



এই বর্তনীতে ঢাক্কি পিন বিশিষ্ট একটি স্লাইড সুইচ ব্যবহার করা হয়েছে যেখানে মাঝের পিন টি ডিজিটাল পিনে ঢুকেছে এবং এক প্রান্ত গ্রাউন্ড পিনে, এবং অপর প্রান্ত ৩.৩ ভোল্টের পিনে ঢুকেছে। এখানে সুইচটি কেবল দুই দিকে রাখা যায়। এক দিকে রাখলে গ্রাউন্ডের সাথে ডিজিটাল পিনের সংযোগ হয়, এবং অন্য দিকে সরিয়ে দিলে ৩.৩ ভোল্টের সাথে ডিজিটাল পিনের সংযোগ স্থাপন হয়। এর ফলে দেখা যাচ্ছে এই সুইচ পিনটিকে যেকোন একটির সাথে সংযোগ রাখতে বাধ্য করছে। এভাবে চাইলে সমস্যার সমাধান সম্ভব। কিন্তু এই সুইচ সহজে ব্রেডবোর্ডে বসানো সম্ভব নয় বলে এখানে ব্যবহার করা হচ্ছে।

এখন এই একই কাজ উপরোক্ত দুই পিনবিশিষ্ট পুশবাটন দিয়েও করা সম্ভব। সেক্ষেত্রে নিচের সার্কিট টি দেখো:

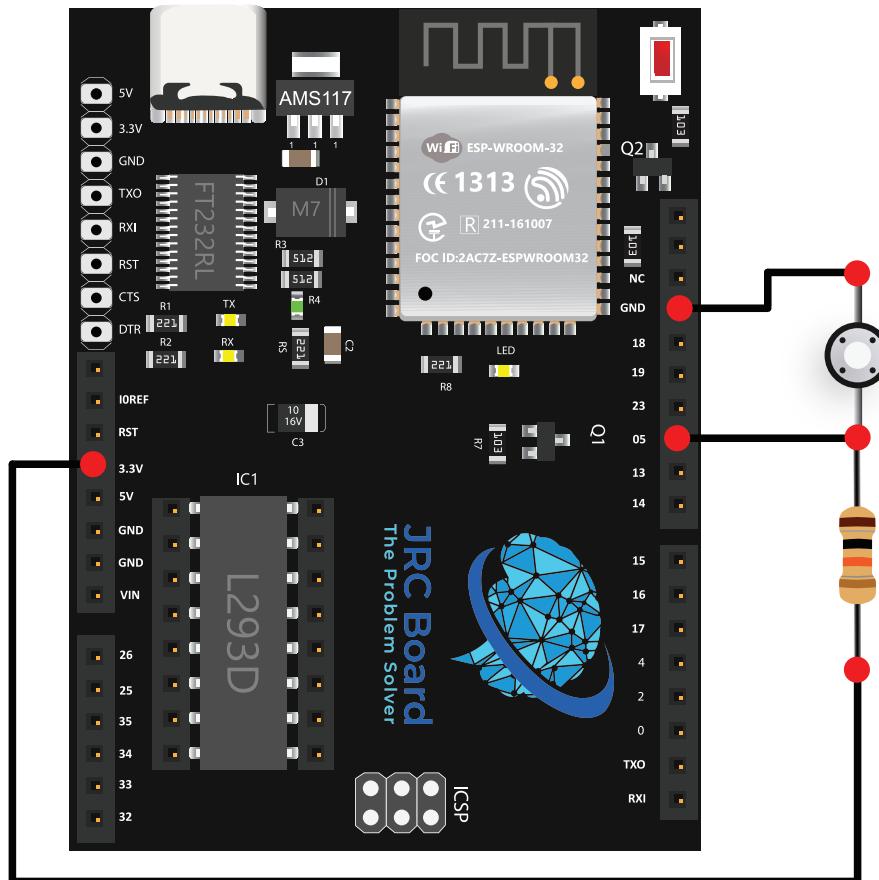


এখানে দেখতে পাচ্ছো যে ডিজিটাল পিন থেকে একটি তার বের হয়ে দুই দিকে চলে গেছে। এক দিকে সুইচ পার হয়ে ৩.৩ ভল্ট এর সাথে সংযোগ স্থাপন হয়েছে, অপর দিকে একটি ১০ কিলো-ওহম রেজিস্টর এর মাধ্যমে গ্রাউন্ডের সাথে সংযোগ স্থাপন হয়েছে। সাধারণ অবস্থায় আমরা যদি বাটনে কোনপ্রকার চাপ দিই, তবে এটি ১০ কিলোওহমের মাধ্যমে গ্রাউন্ডের সাথে যুক্ত হয়ে থাকে বলে রিডিং আর লাফালাফি না করে এখন ০ পেতে থাকে। তবে যদি তুমি বাটন প্রেস করো, সেক্ষেত্রে চিন্তা করে দেখো যে কি ঘটবে? আমরা জানি তড়িৎ সেন্সরেই প্রবাহ হতে চায় যেদিকে তার পক্ষে যাওয়া সহজ। গ্রাউন্ডের আগে একটি ১০ কিলোওহমের রোধ রয়েছে, কিন্তু বাটন প্রেস করার কারণে ৩.৩ ভোল্ট এবং ডিজিটাল পিনের মধ্যে কোনরূপ বাধা নেই। ফলে পিনে সুন্দরমতো ৩.৩ ভোল্ট রিডিং পাবে বা ডিজিটাল রিডিং এর ভ্যালু ১ আসবে। মোদা কথায়, সুইচ প্রেস না করলে গ্রাউন্ডের সাথে সংযোগের কারণে ডিজিটাল রিডিং স্টেট লো এবং বাটন প্রেস করলে ডিজিটাল রিডিং স্টেট হাই থাকবে। এখানে আর আগের মতো গারব্যাজ ভ্যালু আসবেন।

এই সার্কিটের কিন্তু একটি বিশেষ নাম রয়েছে। নরমালি বাটন প্রেস না করলে এর ফ্লোটিং ভ্যালু কে টেনে নামিয়ে ০ বানিয়ে দেয় বলে এই সার্কিটের নাম "পুলডাউন সার্কিট (pulldown circuit)"।

এই সার্কিটে যদি ডিজিটাল রিডিং নই, তবে বাটন প্রেস করা মানে রিডিং স্টেট হাই, এবং বাটন ছেড়ে দেয়া মানে রিডিং স্টেট লো।

ঠিক এই সেটাপে কিন্তু বীপরীতটাও করা সম্ভব। সেটার জন্য এই বর্তনী লক্ষ্য করে দেখো:



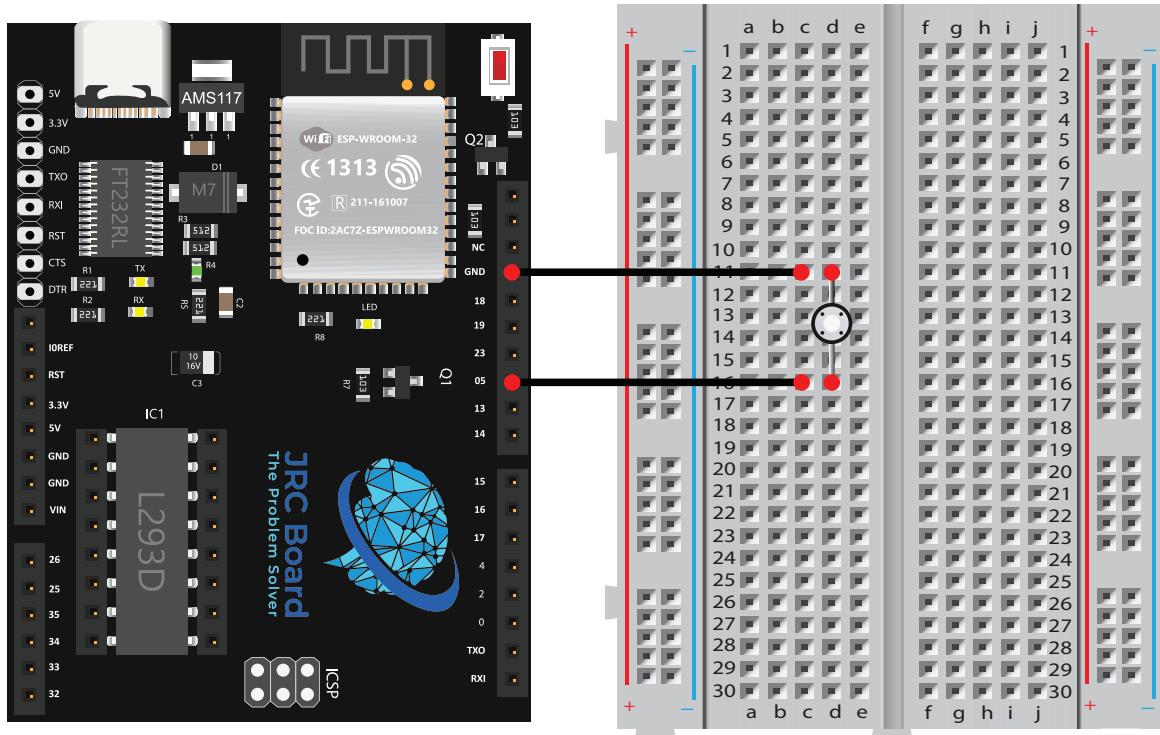
এখানে কম্পোনেন্ট এবং কানেকশন প্রায় আগের সার্কিটের মতোই, কিন্তু ছোট দুটি পার্থক্য রয়েছে। এখানে রেজিস্ট্র পার হয়ে গ্রাউন্ডের বদলে ৩.৩ভোল্ট এর পিনে চুকেছে, এবং সুইচ পেরিয়ে ৩.৩ ভোল্টের বদলে গ্রাউন্ডে সংযোগ স্থাপন করা হয়েছে। এর ফলে সুইচের আসল কার্যক্রম অপরিবর্তীত থাকলেও এর রিডিং উলটে যাবে। এখন সুইচ এ কোনপ্রকার চাপ না দিলে সাধারণত ডিজিটাল রিডিং স্টেট হাই হয়ে থাকবে, এবং চাপ দিলে ডিজিটাল স্টেট লো হয়ে যাবে। এখানে সাধারণ অবস্থায় সবসময় ডিজিটাল রিডিং হাই পাচ্ছে বলে এই সার্কিটের নাম "পুলআপ সার্কিট (pullup circuit)"।

এই অবস্থায় যদি বাটন প্রেস করো, তবে ডিজিটাল রিডিং স্টেট লো পাবে। আর বাটন প্রেস না করলে ডিজিটাল রিডিং হাই পাবে।

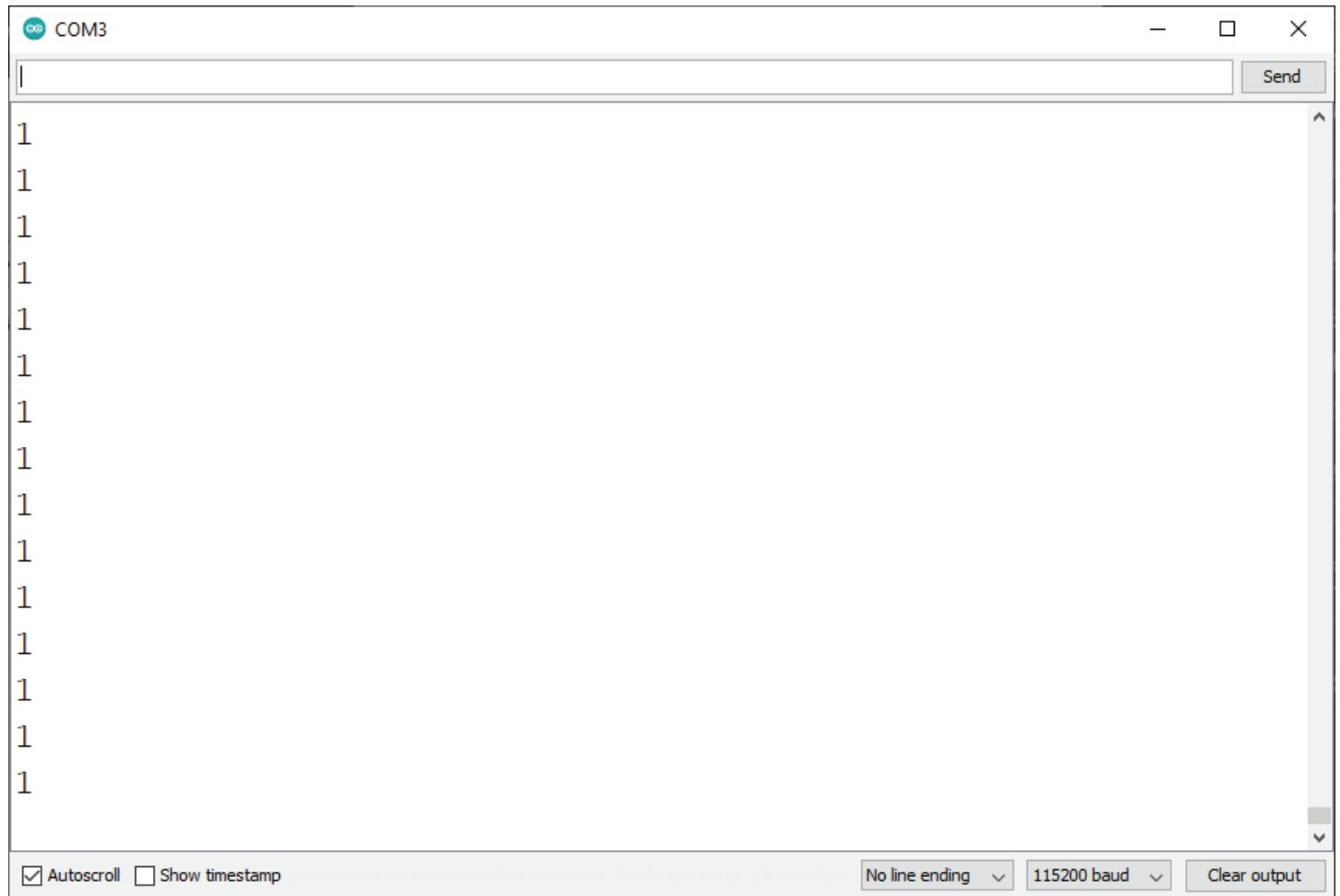
চাইলে উপরের সার্কিটের মতো আলাদা রেজিস্ট্র ব্যবহার না করে ESP32 এর ভেতরের রেজিস্ট্র ব্যবহার করেও কাজ চালিয়ে নেয়া যায়। এই ধরনের ফিচার কে আমরা বলি ইনপুট পুল-আপ এবং ইনপুট পুল-ডাউন ফিচার। যদি কোন ডিজিটাল পিনে কোনোকার তারের সংযোগ না দেয়া হলে সে যদি নিজে নিজে হাই হয়ে থাকে, তবে সেটাকে ইনপুট পুল-আপ বলে এবং লো হয়ে থাকলে সেটাকে ইনপুট পুল-ডাউন ফিচার বলে। এগুলো ফাংশন এর মাধ্যমে এস্টিভেট করতে হয়। চলো আমরা নিচের কোড টি ট্রাই করে দেখি:

```
int x;  
void setup(){  
    pinMode(5, INPUT_PULLUP);  
    Serial.begin(115200);  
}  
  
void loop(){  
    x= digitalRead(5);  
    Serial.println(x);  
}
```

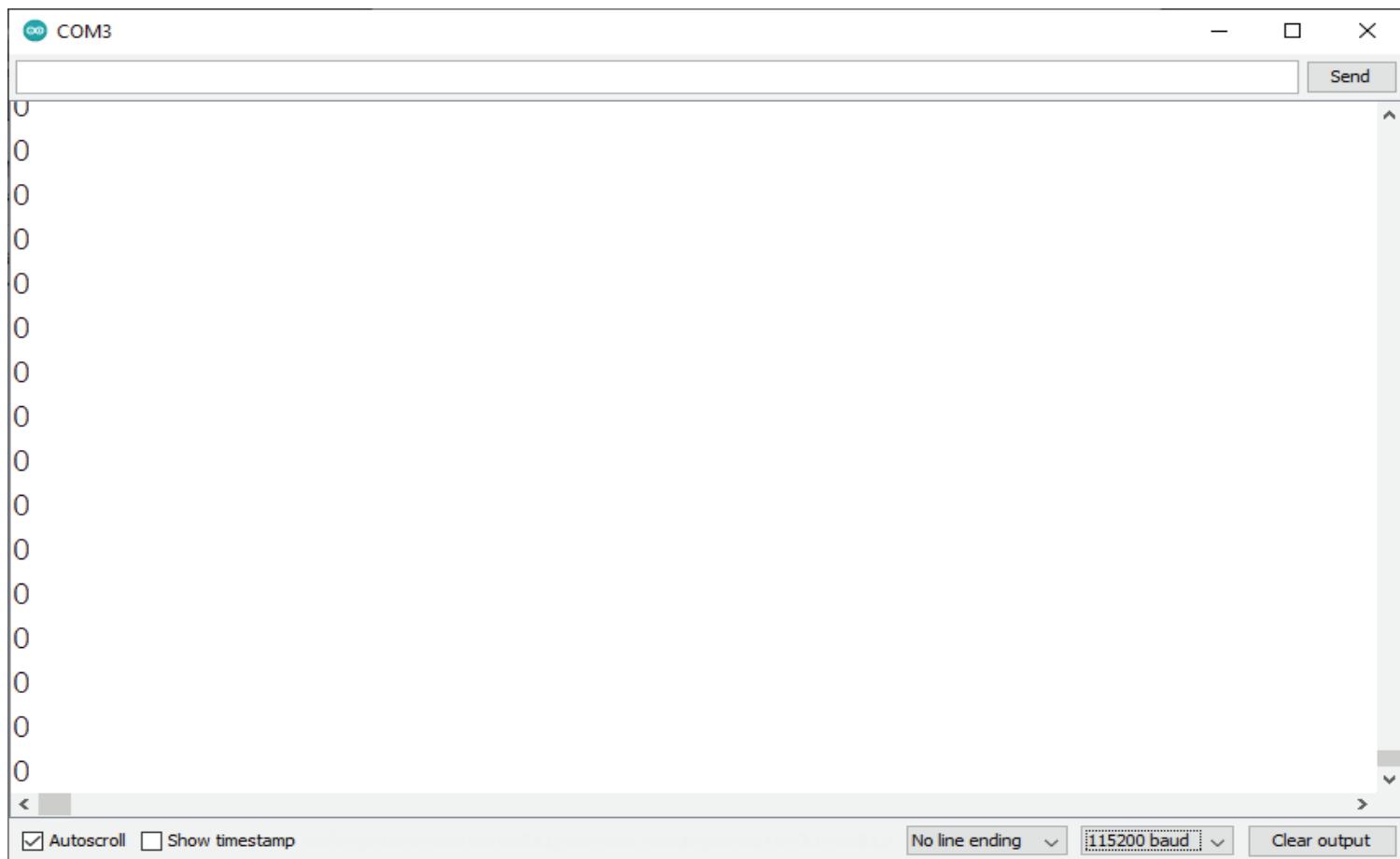
এখনে `setup()` ফাংশনের ভেতরে `pinMode` ফাংশন দ্বারা পিনের ভূমিকা নির্ধারণ করার সময় আমরা সাধারণ "INPUT" না লিখে "INPUT_PULLUP" লেখা হয়েছে। এর মাধ্যমে ডিজিটাল ৫ নং পিনের ভেতরকার ইনপুট পুল-আপ ফিচার এস্টিভেট হয়ে যায়। এখন চলো নিচের সর্কিট টা বানিয়ে দেখি:



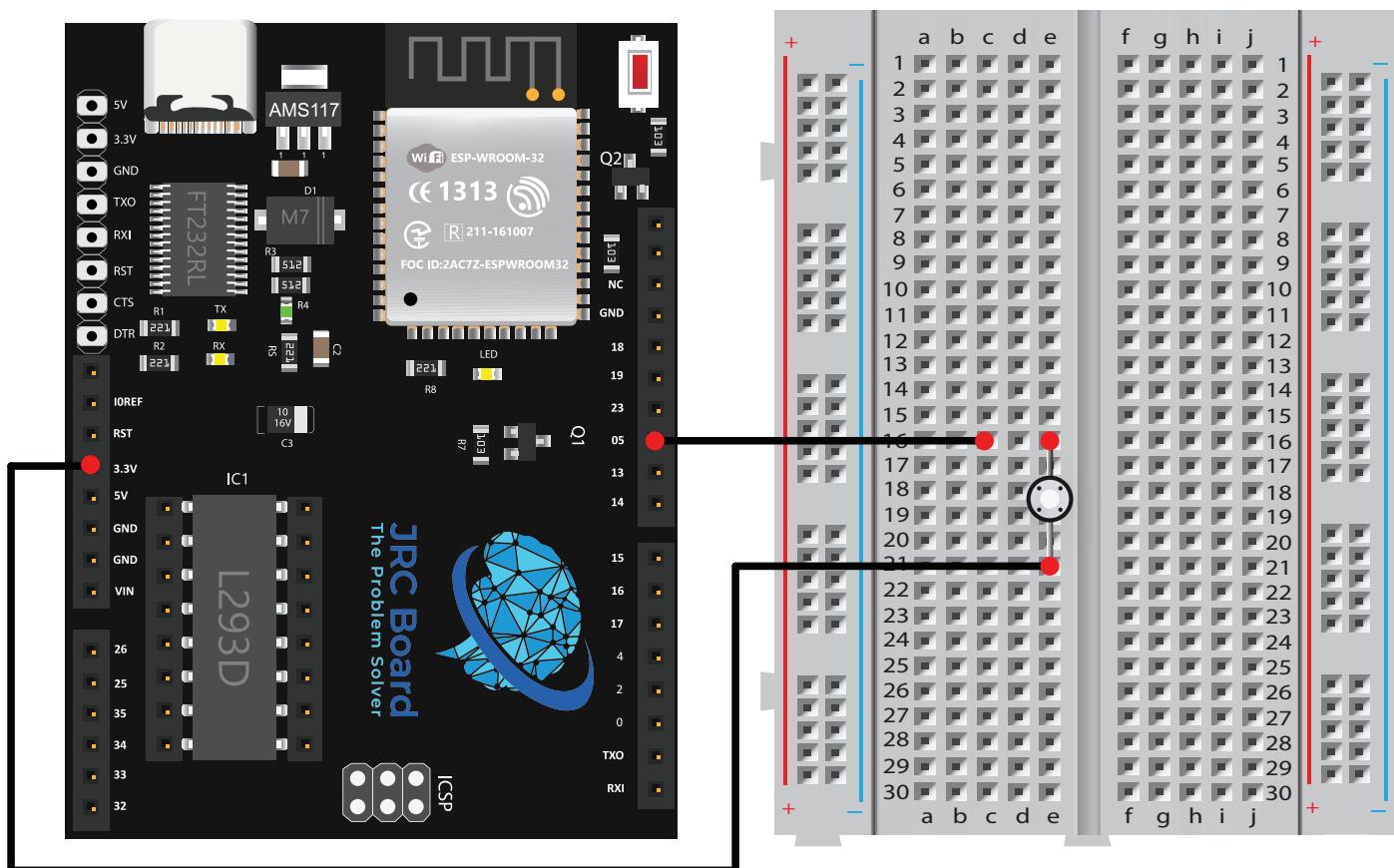
এই সার্কিটে কোড আপলোড দেবার পর যদি আমরা আগের মত সিরিয়াল মনিটর ওপেন করি, তবে দেখতে পাবো যে বাটন প্রেস না করলে আগের মতো গারবেজ ভ্যালু না দিয়ে এখন একটানা 1 দেখাচ্ছে। কারণ এই পিনের সাথে গ্রাউন্ড পিনের শর্ট না করালে অর্ধাংশ সুইচ টি চাপ না দিলে এটি তার ভেতরকার রেজিস্ট্র এর মাধ্যমে হাই হয়ে থাকছে।



এবং আমরা যদি বাটন প্রেস করি, সেক্ষেত্রে এর সাথে গ্রাউন্ড পিনের শর্ট হয়ে যাবার কারণে এর ডিজিটাল রিডিং লো দেখাবে।



একই ভাবে আমরা চাইলে সেই ডিজিটাল পিনে ইনপুট পুলডাউন ফিচারও প্রয়োগ করে দেখতে পারি! এর জন্য কোডে কিরকম পরিবর্তন আনতে হবে তা নিচে দেখে নাও এবং পরীক্ষা করে দেখার জন্য পরের ছবিতে দেয়া সার্কিট টি তৈরী করে নাও:



```

int x;
void setup(){
    pinMode(5, INPUT_PULLDOWN);
    Serial.begin(115200);
}

void loop(){
    x= digitalRead(5);
    Serial.println(x);
}

```

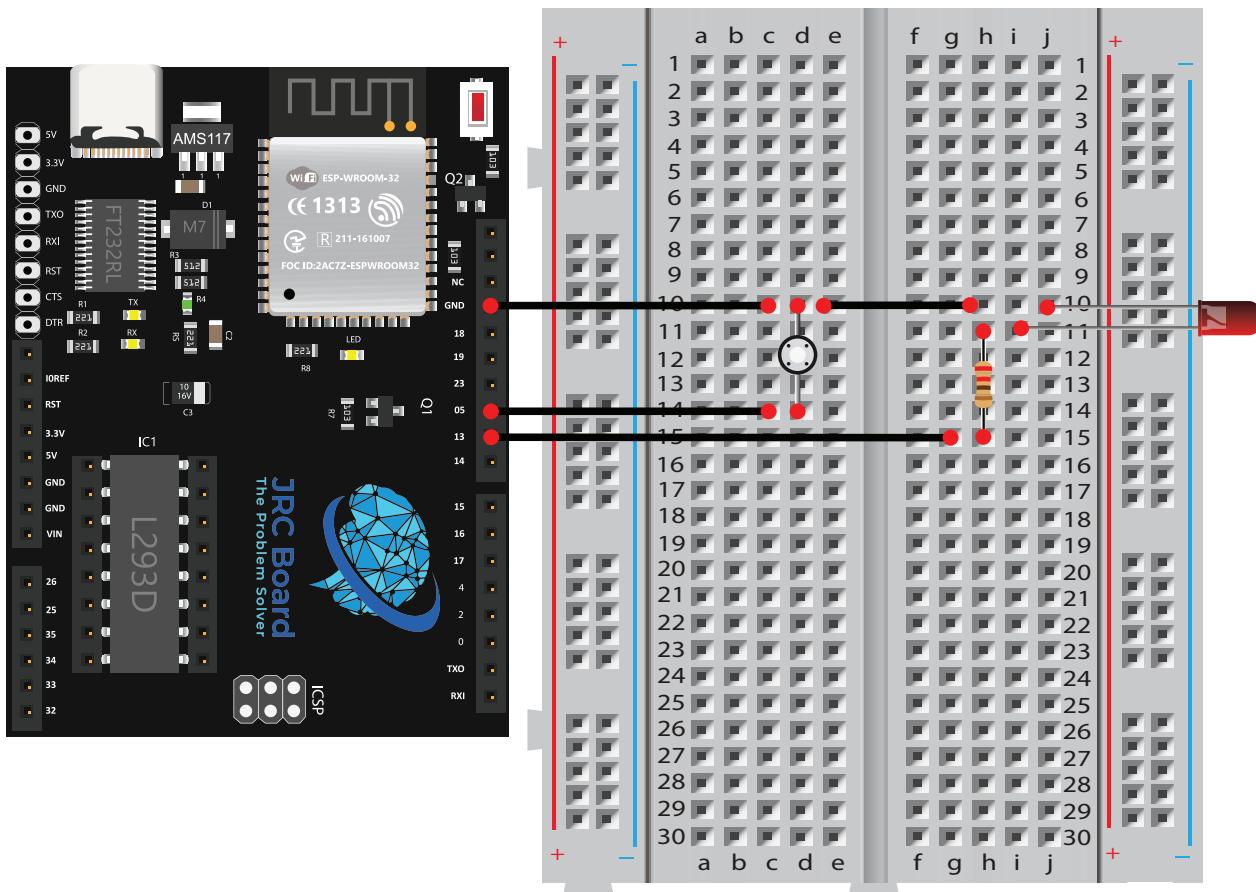
এই সার্কিটে কোড আপলোড দেবার পর যদি আমরা আগের মত সিরিয়াল মনিটর ওপেন করি, তবে দেখতে পাবো যে বাটন প্রেস না করলে এখানে এখন ১ এর বদলে ০ দেখাচ্ছে। কারণ কোডের মাধ্যমে আমরা ইনপুট পুলডাউন ফিচার এস্টিভেট করেছি যেটার ফলে ঐ পিনের সাথে ৩.৩ ভোল্টের কোনপ্রকার সংযোগ না হলে তা সাধারণ অবস্থায় লো থাকবে।

The screenshot shows a serial monitor window titled "COM3". The main text area displays a series of "0" characters, representing the digital value read from pin 5. The bottom status bar shows "Autoscroll" is checked, and the baud rate is set to "115200 baud".

এখানে বাটন প্রেস করলে দেখা যাচ্ছে যে ডিজিটাল ৫ নং পিনের সাথে ৩.৩ ভোল্ট পিনের সরাসরি সংযোগ হচ্ছে, এর ফলে এই পিনে ডিজিটাল রিডিং হাই স্টেটে পাবে।

এভাবে আমরা মাত্র ২ পিনের সুইচ দিয়েই অতিরিক্ত কোন সার্কিটের ব্যবহার না করেই সরাসরি কোডের বিশেষ ফিচারের মাধ্যমে নির্ভুলভাবে ডিজিটাল রিডিং নিতে পারি।

এবার চলো এই যে বাটন রিডিং নিচ্ছি, সেটি দিয়ে মজার কোন কাজ করা যাক। নিচের ডিজাইন মোতাবেক সার্কিট তৈরী করে ফেলো:



এখানে সুইচের রিডিং এর জন্য ডিজিটাল ৫ নং পিন ব্যবহার করছি এবং এই সুইচ অনুযায়ী আউটপুট দেখানোর জন্য একটি এলইডি লাইট স্থাপন করেছি যার খণ্ডাত্মক প্রান্ত বোর্ডের গ্রাউন্ডের সাথে এবং এর ধ্বনাত্মক প্রান্ত একটি ২২০ ওহম রেজিস্টর পেরিয়ে ডিজিটাল ১৩ নাম্বার পিনে লাগিয়েছি। ডিজিটাল ৫ নং পিন থেকে একটি সুইচের মাধ্যমে গ্রাউন্ড পিনের সংযোগ দিয়েছি। সুতরাং এখানে আমরা JRC বোর্ডের ইন্টার্নাল পুল-আপ ফিচার টি ব্যবহার করছি। এবার নিচের কোডের দিকে লক্ষ্য করো:

```

int x;
void setup(){
    pinMode(5, INPUT_PULLUP);
    pinMode(13, OUTPUT);
    Serial.begin(115200);
}

void loop(){
    x= digitalRead(5);
    if(x == 0){
        digitalWrite(13,1); delay(500);
        digitalWrite(13,0); delay(500);
        digitalWrite(13,1); delay(500);
        digitalWrite(13,0); delay(500);
    }
    else digitalWrite(13,0);
}

```

এখানে যেহেতু আমরা লাইটে ডিজিটাল আউটপুট দিবো, তাই `setup()` ফাংশনের ভেতরে এর ভূমিকা আউটপুট নির্ধারণ করে ফেলি। এবং সুইচের জন্য ইনপুট পুল-আপ ফিচার টি এন্টিভেট করি।

`loop()` ফাংশনের ভেতরে কোডের স্ট্রাকচার একদম সরল। যদি বাটন প্রেস করে, তবে ডিজিটাল রিডিং লো আসে তাইনা? এর মানে যখন `X` ভ্যারিয়েবল এর মান `0` হবে তখনই আমরা ধরে নিবো যে বাটন প্রেস হয়েছে। এর জন্য এলইডি লাইট ২ বার ব্লিংক করার স্টেটমেন্ট লিখেছি। ফলে সুইচ টি চাপ দিলে লাইট দুইবার ব্লিংক করে থেমে যাবে। চাপ দিয়ে ধরে রাখলে লাইট টি ব্লিংক করতেই থাকবে।

এভাবে আমরা বোর্ড থেকে ডিজিটাল রিডিং নেয়া শিখে গেলাম!