

## ৬/ রেগুলেটর দিয়ে ভিমিং লাইট কন্ট্রোল:

প্রয়োজনীয় উপকরণ:

১/ পটেনশিওমিটার

২/ ব্রেডবোর্ড

৩/ জাম্পার ওয়্যার

৪/ JRC বোর্ড

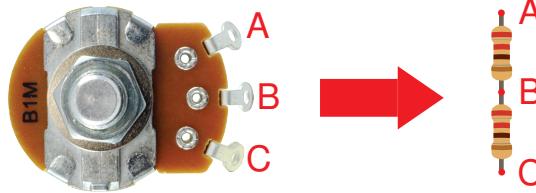
৫/ এলইডি লাইট

৬/ একটি ২২০ ওহম রেজিস্টর

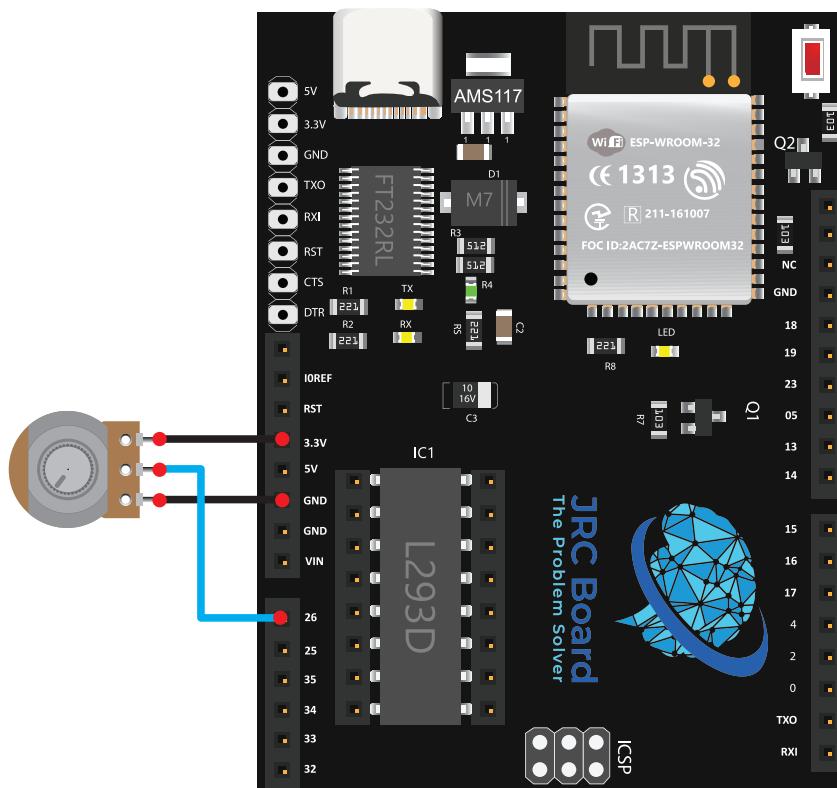
বর্ণনা: আগের প্রজেক্টে আমরা শিখে কিভাবে একটি এলইডি লাইটের উজ্জ্বলতা PWM সিগ্নাল এর মাধ্যমে নিয়ন্ত্রণ করা যায়। কিন্তু এখানে কাজ টা তো কোডে হচ্ছে তাইনা? মানে কোডে লিখে দিতে হয় যে লাইট টা কতটুকু উজ্জ্বলতায় জ্বলবে এবং আপলোড দেবার পরে লাইটের উজ্জ্বলতায় আমাদের কোন হাত থাকেনা। কেমন হবে যদি আমরা এমন সিস্টেম বানাই যেখানে আমরা পটেনশিওমিটার এর মাধ্যমে বাইরে থেকে JRC বোর্ডে ইনপুট দিয়ে এরপর সে অনুযায়ী লাইটের উজ্জ্বলতা নিয়ন্ত্রণ করি? এই কাজটিই আমরা এই প্রজেক্টে করে দেখবো। তার আগে আমরা জেনে নিই এই যে রেগুলেটর ব্যবহার করবো বা যেটিকে আমরা পটেনশিওমিটার হিসেবেও চিনি, সেটি আসলে কি এবং কিভাবে ব্যবহার করতে হবে। এটি দেখতে নিচের চিত্রের মতো হয়:



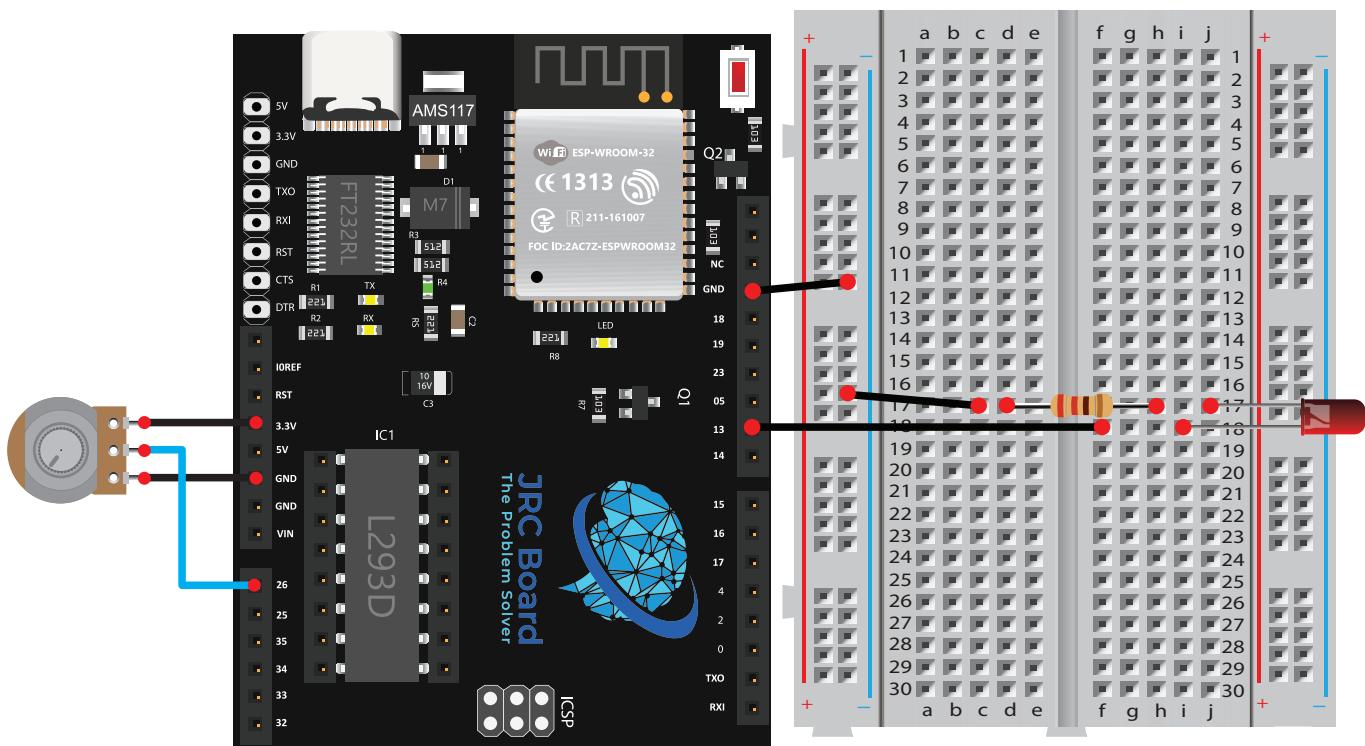
এটি দেখতে অনেকটাই আমাদের ফ্যানের রেগুলেটর এর মতো যেখানে মাথায় ঘুরানোর মত একটি নব থাকে এবং নিচে তিনটি পাঁ বের করা থাকে। এটি প্রকৃতপক্ষে একটি পরিবর্তনশীল রোধ যেখানে একদম কিনারের দুই পায়ের মধ্যকার রোধ একই থাকে এবং বাকিদের মধ্যে রোধের মান পরিবর্তন হতে থাকে নবের ঘোরানোর উপর নির্ভর করে।



উপরের ছবিতে দেখতে পাচ্ছো যে এই পটেনশিওমিটার এর তিনটা পাঁ আসলে কিরকম সাকিট নির্দেশ করে। এখানে নব একদিকে ঘুরালে A ও B এর মধ্যকার রোধের নাম করতে থাকে এবং B ও C এর মধ্যকার রোধ বাড়তে থাকে। আবার নব উল্টোদিকে ঘুরালে A ও B এর মধ্যকার রোধের মান কমে যায় এবং B ও C এর মধ্যকার রোধের মান বেড়ে যায়। যেটিই হোক না কেন, A ও C প্রান্তের মধ্যকার তুল্যরোধ সবসময় একই থাকে। এর ফলে এটাকে ভোল্টেজ ডিভাইডার হিসেবে ব্যবহার করা যায় যা সম্পর্কে আমরা পূর্বেই জেনে এসেছি। সুতরাং বুবাতেই পারছি যে এখানে কিনারের দুই প্রান্তের যেকোন এক প্রান্ত ৩.৩ ভোল্টের পিনে এবং আরেকটি গ্রাউন্ড পিনে বসিয়ে দিয়ে মাঝখানের পিনটির সাথে JRC বোর্ডের যেকোন এনালগ পিনের সংযোগ দিয়ে তা থেকে এনালগ রিডিং নেয়া সম্ভব। নিচে এরকম একটি উদাহরণ সাকিট দেখে নিইঃ



এইভাবে সংযোগ দিয়ে আমরা এনালগ রিডিং নিতে পারি। এবং এনালগ রিডিং দেখবার কোড পূর্বেই ব্যাখ্যা করে দেয়া হয়েছে। তাই সে কথা আবার না বলি। এখন দেখবো যে কিভাবে এই এনালগ রিডিং কে কাজে লাগিয়ে আমরা লাইটের উজ্জ্বলতা কে নিয়ন্ত্রণ করতে পারি। প্রথমেই এই পটেনশিওমিটার এবং এলইডি লাইট কে একত্র করে একটি সাকিঁটি তৈরী করিঃ



এবার এর কোডিং এর পালা। এখানে প্রথমেই আমাদের কে পটেনশিওমিটার থেকে এনালগ রিডিং নিতে হবে এবং সেটি একটা ভ্যারিয়েবলে সংরক্ষণ করতে হবে।

```
void setup(){
}
void loop(){
    int p = analogRead(26);
}
```

আমাদের যেহেতু একটি লাইটে PWM সিগ্নাল দিতে হবে এর উজ্জ্বলতা নিয়ন্ত্রন এর জন্য, সেক্ষেত্রে আমরা শুরুতেই এর জন্য দরকারী সেটাপ করে নিই যা আমরা পূর্বেই শিখে এসেছি।

```
void setup(){
    ledcSetup(0,1000,8);
    ledcAttachPin(13,0);
}
void loop(){
    int p = analogRead(26);
}
```

এখানে আমরা 0 নাম্বার টাইমার চ্যানেল কে 8বিটে সেটাপ করে 13 নাম্বার পিনের সাথে যুক্ত করেছি। এখন আমরা যে এনালগ ইনপুট পাছি সে অনুযায়ী যদি আউটপুট করতে চাই সেক্ষেত্রে কোডের সিস্টেম এরকম হবার কথা:

```
void setup(){
    ledcSetup(0,1000,8);
    ledcAttachPin(13,0);
}
void loop(){
    int p = analogRead(26);
    ledcWrite(0, p);
}
```

এখানে কোড অনুযায়ী p ভ্যারিয়েবল এ যে ভ্যালু সেভ হচ্ছে সেটই চ্যানেলের ভোল্টেজ নিয়ন্ত্রনের মান হিসেবে ব্যবহার করা হচ্ছে, ফলে পটেনশিওমিটার ঘূরালে সেটি এলইডি লাইটের উজ্জ্বলতা নিয়ন্ত্রন করার কথা। কিন্তু এখানে একটি ছোট সমস্যা রয়ে গেছে। আমরা জানি যে JRC বোর্ডে এনালগ ইনপুট নেয়া হয় 12 বিট রেজিস্টারে কিন্তু আমরা PWM আউটপুট দিচ্ছি 8 বিট রেজিস্টারে। এক্ষেত্রে পটেনশিওমিটার ঘূরালে সর্বোচ্চ ইনপুট রিডিং নেবার কথা 8095 যা কিনা 8বিট PWM কন্ট্রোলে সর্বোচ্চ মান 255 কে ছাড়িয়ে যায়। এক্ষেত্রে দুটি সমাধান আছে।

```

void setup(){
    ledcSetup(0,1000,12);
    ledcAttachPin(13,0);
}
void loop(){
    int p = analogRead(26);
    ledcWrite(0, p);
}

```

প্রথম সমাধান এটাই যে আমরা যদি PWM আউটপুটকেও ১২ বিটে সেটাপ করে ফেলি, সেক্ষেত্রে এনালগ ইনপুট এবং আউটপুট একই ভ্যালু রেঞ্জে থাকবে এবং এখানে আর কিছুই করা লাগবেনা! এটা কেবল JRC বোর্ডেই করা সম্ভব। আরডুইনো তে এনালগ আউটপুট রেজিস্ট্রেশন বাড়ানোর সুযোগ নেই। সেক্ষেত্রে আমরা একটা বিশেষ সমাধান ব্যবহার করে থাকি যেটা কিনা JRC বোর্ডেও প্রয়োগ সম্ভব। সেটা হলো এনালগ ইনপুট এ যে মান আসছে, তাকে আনুপাতিক হারে কমিয়ে এনে ৮বিটের ভ্যালু রেঞ্জের মধ্যে রেখে এরপর সেই ভ্যালু ব্যবহার করা। নিচের চিত্রের কোডে তা দেখতে পারো:

```

void setup(){
    ledcSetup(0,1000,8);
    ledcAttachPin(13,0);
}
void loop(){
    int p = analogRead(26);
    p=(p/4096)*256;
    ledcWrite(0, p);
}

```

এখানে ৮ বিট এবং ১২ বিটের অনুপাতকে ব্যবহার করে সেটা দিয়ে p ভ্যারিয়েবলের ভ্যালু কনভার্ট করে আবার p এর মধ্যেই সেভ করা হয়েছে। এবং সে ভ্যালু টা কাজে লাগালেই প্রজেক্টে আর কোনপ্রকার ঝামেলা হবেনা!

এই আনুপাতিক হারে রূপান্তর এর কাজ টা চাইলে আরেকভাবেও করা যায়। সেটা হলো একটি বিশেষ ফাংশনের ব্যবহার করে, যেটার নাম হলো map() ফাংশন। নিচের কোডে লক্ষ্য করে দেখো:

```

void setup(){
    ledcSetup(0,1000,8);
    ledcAttachPin(13,0);
}
void loop(){
    int p = analogRead(26);
    p = map(p,0,4095,0,255);
    ledcWrite(0, p);
}

```

এখানে ম্যাপ ফাংশনের ভেতরে ৫ টি প্যারামিটার আছে, একে একে সব বলছি। প্রথমে যে ভ্যারিয়েবল এর ভ্যালু নিয়ে কাজ করতে চাও, সেটা লিখতে হবে। আমরা যেহেতু p তে যে ইনপুট এসেছে সেটা নিয়ে কাজ করবো তাই এখানে আমি p লিখেছি। এর পরের দুইটা প্যারামিটার হলো আমরা এই ভ্যারিয়েবল এর ভেতরে সর্বনিম্ন এবং সর্বোচ্চ কতো ভ্যালু আশা করছি। যেহেতু আমরা ১২ বিট রেজিস্ট্রেশনে এনালগ রিডিং নিচ্ছি, সেক্ষেত্রে এখানে সর্বনিম্ন মান আসতে পারে ০ এবং সর্বোচ্চ ৪০৯৫...সেই মানগুলিই লিখে দিয়েছি। এরপরের দুইটায় বসাতে হবে রূপান্তরের পরে আমরা সর্বনিম্ন এবং সর্বোচ্চ যে দুটি মান আশা করছি সেই দুটি মান। যেহেতু আমরা ৮ বিট সিস্টেমে রূপান্তর করতে চাচ্ছি, সেক্ষেত্রে ৮ বিটে সর্বনিম্ন মান ০ এবং সর্বোচ্চ মান ২৫৫ লিখে দিয়েছি সবার শেষে। এবং এই map ফাংশনের যে আউটপুট আসবে সেটা আবার p ভ্যারিয়েবলের মধ্যেই সেভ করে নিচ্ছি। এতে করেই কাজ হয়ে যাবে!

এখন কথা হলো কেন আমরা এতো কষ্ট করে map() ফাংশনের ব্যবহার করতে যাবো? যখন এডভান্সড রোবটিক্স করতে যাবে, তখন দেখতে পাবে ভ্যালু কনভার্শনের চেয়ে সর্বনিম্ন এবং সর্বোচ্চ মান নিয়ে কাজ করলে কনফিউশন কর হয়। এবং আরেকটা বড় সুবিধা হলো এখানে চাইলে আমরা নির্দিষ্ট ভ্যালু রেঞ্জ নিয়ে কাজ করতে পারি।

**map(p,100,4095,0,255);**

আরেকটা বড় সুবিধা হলো এখানে আমরা কার্যকরী ভ্যালু রেঞ্জ নির্ধারণ করে দিতে পারি। যেমন আমি এখানে ২য় প্যারামিটার এ ০ এর জায়গায় 100 বসানোর ফলে p ভ্যারিয়েবলে যদি 100 এর চেয়ে ছোট কোন ইনপুট আসে, সেটা সে গ্রাহ্য করবেনা এবং আউটপুটে রূপান্তর এর পরে সর্বনিম্ন মান যেটা আসার কথা (০ লেখা), সেটিই আসবে। এভাবে আমরা এনালগ সেন্সর এর ইরর কারেকশনও চাইলে করতে পারি। এখানেই আমাদের এই প্রজেক্টে এর কাজ শেষ!