* Project title: ECRL(Eye Comfort Reading Light).
* Team members: Mohammad Mahdi Moradi G.G., Morteza Nazari Sumarin.
* Institution: Tabriz University.
* Supervisor’s name: Prf Mohammadzadeh.
* Date of Submission: 2025-05-16, 1404-02-26.

### ۱. حالت) Fallback بازگشت اضطراری( چیست؟ چه مسئولیتی دارد؟

حالت **Fallback** زمانی فعال می‌شود که سیستم نتواند هیچ نوری را تشخیص دهد، یعنی شرایط **تاریکی مطلق** یا **عدم دریافت داده معتبر از سنسور** برقرار باشد. این حالت زمانی تشخیص داده می‌شود که:

currentYellowRatio == -1

**وظیفه حالت Fallback :**

اطمینان از اینکه حتی در صورت خرابی سنسور یا تاریکی کامل، سیستم بتواند نوری پایه فراهم کند.

* جلوگیری از شرایط ناراحت‌کننده یا خطرناک (مثلاً مطالعه در تاریکی مطلق).
* استفاده از مقادیر پیش‌فرض برای روشن‌سازی LEDها:

newYellowLED = default\_k \* targetYellowRatio;

newWhiteLED = default\_k \* (1.0 - targetYellowRatio);

**۲. برنامه از مقدار YOI/WOI = -1 چه برداشتی دارد؟**

این مقدار زمانی ایجاد می‌شود که:

float currentYellowRatio = (total > 0) ? YOI / total : -1;

اگر total برابر صفر باشد (یعنی هیچ نوری از LEDها یا محیط ثبت نشده)، برنامه نسبت زرد را به **عدد -1** تنظیم می‌کند.

**برداشت برنامه:**

* فعال‌سازی **حالت (Fallback) مود ۱**.
* نشان‌دهنده شرایطی مانند:
  + سنسور از کار افتاده یا قطع شده.
  + تاریکی کامل در محیط.

**۳. هیسترزیس (Hysteresis) چیست و چرا استفاده شده است؟**

**هیسترزیس** یک ناحیه بافر (حاشیه) اطراف محدوده راحتی نور است که با مقدار زیر تعریف می‌شود:

const float hysteresis = 2.0;

**هدف استفاده از هیسترزیس:**

* جلوگیری از تغییرات سریع بین حالات مختلف برنامه در اثر نویز یا نوسانات ناچیز نور محیط.
* تثبیت عملکرد سیستم و جلوگیری از چشمک‌زدن یا تغییرات مکرر.
* مشابه مفهومی به نام "Debouncing" در الکترونیک دیجیتال.

مثال استفاده:

isTooDark = ambientTotal < (L\_min - hysteresis);

isTooBright = ambientTotal > (L\_max + hysteresis);

**۴ .هر کدام از حالات Mod 1-4 چه کاری انجام می‌دهند؟**

**حالت ۱: Fallback**

if (currentYellowRatio == -1)

* در صورت عدم دریافت نور از سنسور.
* استفاده از مقادیر پیش‌فرض برای LEDها.
* فعال‌سازی حالت اضطراری برای تامین حداقل نور.

**حالت ۲ و ۳: نور بیش از حد یا کمبود نور**

else if (isTooDark || isTooBright)

* **نور کم:** کاهش تدریجی نور.
* **نور زیاد:** کاهش تدریجی نور. (رفتار دو حالت یکی در نظر گرفته شده)
* کاهش شدت LEDها با **ضریب کاهش 0.9**. (یعنی نور مرحله n، 0.9 ضریبی از مرحله قبلی خود است)
* حفظ نسبت نور زرد به سفید در محدوده مطلوب (0.58 تا 0.62).

**حالت ۴: در محدوده آسایش**

else

* تنظیم دقیق نسبت نور زرد و سفید بدون تغییر محسوس در شدت کل نور.
* اگر نسبت زرد کم باشد : افزایش تدریجیLED زرد.
* اگر نسبت زرد زیاد باشد : افزایش تدریجی LED سفید.
* میزان افزایش های تدریجی 0.1 حالات قبلی هستند.
* تنظیم بر اساس خطای موجود با نسبت هدف (0.6).

**۵. فرمول‌های ریاضی استفاده شده در کد و هدف آن‌ها:**

**الف. برآورد نور محیط:**

ambientWhite = min(r, min(g, b));

ambientYellow = (r + g) / 2.0;

* استخراج نور سفید و زرد از داده‌های RGB.
* تبدیل داده‌های خام به اطلاعات قابل استفاده برای کنترل نور.
* <https://www.analog.com/en/resources/technical-articles/map-colors-of-a-cie-plot-and-color-temperature-using-an-rgb-color-sensor.html?utm_source=chatgpt.com>
* <https://www.sciencedirect.com/science/article/abs/pii/S0039914022000406?utm_source=chatgpt.com>

**ب. محاسبه شدت خروجی نور:**

YOI = prevYellowLED + ambientYellow;

WOI = prevWhiteLED + ambientWhite;

total = YOI + WOI;

currentYellowRatio = (total > 0) ? YOI / total : -1;

* برآورد سهم نور تولیدشده توسط LED و محیط.
* کمک به تنظیم رنگ و دمای نور.
* اگر مقدار -1 شود، یعنی خطایی رخ داده و ورودی ها درست نبوده اند، یا سنسور درست کار نکرده، به طور کلی خطایی بوده که صلاح به ادامه کار نیست و برنامه اقدام به جبران این خطا میکند:
* if (currentYellowRatio == -1) {
* Serial.println("Mode: Darkness Fallback");
* newYellowLED = default\_k \* targetYellowRatio;
* newWhiteLED  = default\_k \* (1.0 - targetYellowRatio);
* inFallbackMode = true;
* }
* با در نظر گرفتن خطاهای ممکن، سیستم در حالت عادی(سالم بودن قطعات و شرایط مناسب) فقط به یک علت میتواند چنین اتفاقی بیافتد و آن تاریکی مطلق است. در نتیجه ما یک استانداری در نظر گرفته ایم که در این حالت مقادیر خروجی را به حالت دیفالت درآورد.

**پ. ضریب مقیاس برای کنترل روشنایی:**

newYellowLED = round(prevYellowLED \* scalingFactor);

* کاهش تدریجی نور برای اجتناب از تغییرات ناگهانی.

**ت. حفظ نسبت مطلوب زرد به کل:**

while ((ratio < 0.58 || ratio > 0.62) && ...)

* تضمین هماهنگی رنگ نور با نسبت هدف.
* کنترل دمای رنگ برای آسایش چشمی.

**ث. تنظیم دقیق در حالت راحتی:**

float adjust = 0.1 \* (difference\_from\_target) \* maxLEDIntensity;

* تنظیم تدریجی با ضریب کنترل برای جلوگیری از پرش و نوسان.
* دو حالت difference from target چنین هستند:
* 0.6 – currentYellowRatio
* currentYellowRatio - 0.6