<h3 class="post-title entry-title" itemprop="name" style="caret-color: rgb(102, 102, 102); color: #666666; font-family: "Trebuchet MS", Trebuchet, Verdana, sans-serif; font-feature-settings: normal; font-kerning: auto; font-optical-sizing: auto; font-size-adjust: none; font-size: 24.48px; font-variant-alternates: normal; font-variant-east-asian: normal; font-variant-ligatures: normal; font-variant-numeric: normal; font-variant-position: normal; font-variation-settings: normal; font-weight: normal; font-width: normal; line-height: normal; margin: 0px; position: relative;">Laporan Akhir 2 Modul 2</h3><div class="post-header" style="caret-color: rgb(102, 102, 102); color: #666666; font-family: "Trebuchet MS", Trebuchet, Verdana, sans-serif; font-size: 15.84px; line-height: 1.6; margin: 0px 0px 1.5em;"> [KEMBALI KE MENU SEBELUMNYA] <br style="background-color: white; font-family: "Times New Roman", Times, FreeSerif, serif; font-size: 13.2px;" /> DAFTAR ISI
 1. Prosedur 2. Hardware dan Diagram Blok 3. Rangkaian Simulasi dan Prinsip Kerja 4. Flowchart dan Listing Program 5. Video Demo 6. Analisa 7. Download File

 <span style="background-color: white;

caret-color: rgb(102, 102, 102); color: #666666; font-family: times, "times new roman", serif; font-size: 13.2px;">
 Laporan Akhir 2 Sistem Lampu Jalan Otomatis
 1. Prosedur [Kembali]
 <div style="border: 0px; box-sizing: border-box; caret-color: rgb(102, 102, 102); color: #666666; font-family: "Times New Roman", Times, FreeSerif, serif; font-size: 13.2px; margin: 0px; padding: 0px;">1. Rangkai rangkaian di proteus sesuai dengan kondisi percobaan.
2. Buat program untuk mikrokontroler STM32 Nucleo G474RE di software STM32 CubeIDE.
3. Compile program dalam format hex, lalu upload ke dalam mikrokontroler.
4. Setelah program selesai di upload, jalankan simulasi rangkaian pada proteus. <div style="border: 0px; box-sizing: border-box; caret-color: rgb(102, 102, 102); color: #666666; font-family: "Times New Roman", Times, FreeSerif, serif; font-size: 13.2px; margin: 0px; padding: 0px;">5. Selesai.
 2. Hardware dan Diagram Blok [Kembali]
 Hardware : <br /> a) Mikrokontroler STM32 Nucleo G474RE
 <img alt="" data-original-height="487" data-original-width="426" height="240" src="https://blogger.googleusercontent.com/img/a/AVvXsEj93p4A6QCbYOTKY5hulFr59SwoJKJOZhY-USjGq9rBTIhNKpL9qqHHPt42xQi2uj-fZGzY0MgQimnYYB52_e4UYcJ3lWhD6NOd09o1eCeZ6bBqcnfT68Wffkb9mEXG-i-A9eg-B6i_p1nzCNzCFwPlPYklRPtqHhAWZiSD4xON8cG6RlpY

KgAlBiYj5Wq6" style="-webkit-box-shadow: rgba(0, 0, 0, 0.1) 1px 1px 5px; background-attachment: scroll; background-clip: border-box; background-image: none; background-origin: padding-box; background-position: 0% 0%; background-repeat: repeat; background-size: auto; border: 1px solid rgb(238, 238, 238); box-shadow: rgba(0, 0, 0, 0.1) 1px 1px 5px; box-sizing: border-box; max-width: 100%; padding: 5px; position: relative;" width="210" />

 2. LED

 3. PIR Sensor <img alt="" data-original-height="1112" data-original-width="1772" height="175" src="https://blogger.googleusercontent.com/img/a/AVvXsEhGy8dEg5TYJ0FBlOK_CqMKKjEvfkoKegNitotgpbci4dZs04EFLuJCUGI5ZDbx0_KXfyij3TofHSpq9qL1GshKngg9q10cFh59eq3Y2P8gj4uRRMXi1QtCulM9DgvGpcHfFBVoBskvdseNCQJqWjcgn2mX-GWFvoEXwEmWiYdKMDmzUe04YPbe3oCtK6-I" style="-webkit-box-shadow: rgba(0, 0, 0, 0.1) 1px 1px 5px; background-attachment: scroll; background-clip: border-box; background-image: none; background-origin: padding-box; background-position: 0% 0%; background-repeat: repeat; background-size: auto; border: 1px solid rgb(238, 238, 238); box-shadow: rgba(0, 0, 0, 0.1) 1px

1px 5px; box-sizing: border-box; max-width: 100%; padding: 5px; position: relative;" width="280" />

 4. Power Supply
   <div class="separator" style="clear: both; text-align:

justify;"> 5.Push Button

 6. Breadboard

 7. Adaptor <a

href="https://blogger.googleusercontent.com/img/a/AVvXsEgHWdzVF72Ous3_1OgDjaYbA4joL-SBMGfUtVwon_7RpIkvK7mxBHx7qWi97aggVaRMQgJF6kk278IysUPEpRTVVjtglGzLUTCBkZt4Ul8b0O3hAWLhpf0unNvnbLvAQAE1tmugilDXuFkEmPURDNgBWAHiNW_ivLH6uGimi2YkCHB6WEConoa2_4UlOmGH" style="color: #888888; margin-left: 1em; margin-right: 1em; text-decoration: none;">

 8. Resistor

 9. Jumper <img alt="" data-original-height="500" data-original-width="500" height="240" src="https://blogger.googleusercontent.com/img/a/AVvXsEg1IyR9nIgoq2LSACpchmaa0PSLYE74DTM1Cqt9cXZLfz-SttyvTGIYyn8a2YwPTRjFDdgbQGHtncU95nc-umezisFWw_eIL-OvbzZAbotUFMBthovwjVs1r3t4F0HNyH3A5QG79BuSPxKCPOKes5r9C6090khNZ4yQIoC5DjH0kw0hqGW

kBE9QVhCaE6M5" style="-webkit-box-shadow: rgba(0, 0, 0, 0.1) 1px 1px 5px; background-attachment: scroll; background-clip: border-box; background-image: none; background-origin: padding-box; background-position: 0% 0%; background-repeat: repeat; background-size: auto; border: 1px solid rgb(238, 238, 238); box-shadow: rgba(0, 0, 0, 0.1) 1px 1px 5px; box-sizing: border-box; max-width: 100%; padding: 5px; position: relative;" width="240" />

 Diagram Blok :

 3. Rangkaian Simulasi dan Prinsip Kerja [Kembali]
 Rangkaian Simulasi <span style="font-family: times, "times new roman",

serif;">

 Prinsip Kerja : <p style="-webkit-text-stroke-color: rgb(0, 0, 0); -webkit-text-stroke-width: 0px; font-family: "Times New Roman"; font-feature-settings: normal; font-kerning: auto; font-optical-sizing: auto; font-size-adjust: none; font-size: 20px; font-style: normal; font-variant-alternates: normal; font-variant-caps: normal; font-variant-east-asian: normal; font-variant-emoji: normal; font-variant-ligatures: normal; font-variant-numeric: normal; font-variant-position: normal; font-variation-settings: normal; font-width: normal; line-height: normal; margin: 0px 0px 12px;">
</p><p style="-webkit-text-stroke-color: rgb(0, 0, 0); -webkit-text-stroke-width: 0px; font-family: "Times New Roman"; font-feature-settings: normal; font-kerning: auto; font-optical-sizing: auto; font-size-adjust: none; font-size: 20px; font-style: normal; font-variant-alternates: normal; font-variant-caps: normal; font-variant-east-asian: normal; font-variant-emoji: normal; font-variant-ligatures: normal; font-variant-numeric: normal; font-variant-position: normal; font-variation-settings: normal; font-width: normal; line-height: normal; margin: 0px 0px 12px;">Prinsip kerja rangkaian ini dimulai dari pembacaan intensitas cahaya lingkungan oleh LDR melalui ADC. Apabila nilai yang terbaca menunjukkan kondisi terang (siang hari), maka LED akan dimatikan. Sebaliknya, jika terdeteksi kondisi gelap (malam hari), sistem akan mengaktifkan sensor PIR untuk mendeteksi adanya pergerakan. Ketika gerakan terdeteksi, LED menyala terang dengan nilai PWM LED_FULL. Jika tidak ada gerakan dalam selang waktu tertentu, LED tetap menyala namun

dalam kondisi redup menggunakan nilai LED_DIM. <p style="-webkit-text-stroke-color: rgb(0, 0, 0); -webkit-text-stroke-width: 0px; font-family: "Times New Roman"; font-feature-settings: normal; font-kerning: auto; font-optical-sizing: auto; font-size-adjust: none; font-size: 12px; font-style: normal; font-variant-alternates: normal; font-variant-caps: normal; font-variant-east-asian: normal; font-variant-emoji: normal; font-variant-ligatures: normal; font-variant-numeric: normal; font-variant-position: normal; font-variation-settings: normal; font-width: normal; line-height: normal; margin: 0px 0px 12px; min-height: 13.8px;">
 <p style="-webkit-text-stroke-color: rgb(0, 0, 0); -webkit-text-stroke-width: 0px; font-family: "Times New Roman"; font-feature-settings: normal; font-kerning: auto; font-optical-sizing: auto; font-size-adjust: none; font-size: 20px; font-style: normal; font-variant-alternates: normal; font-variant-caps: normal; font-variant-east-asian: normal; font-variant-emoji: normal; font-variant-ligatures: normal; font-variant-numeric: normal; font-variant-position: normal; font-variation-settings: normal; font-width: normal; line-height: normal; margin: 0px 0px 12px;">Push button difungsikan sebagai interrupt untuk mengubah status mode darurat melalui variabel emergency_mode. Saat mode darurat aktif, LED akan langsung dimatikan dan sistem tidak melakukan pembacaan baik dari LDR maupun PIR. Pengaturan tingkat kecerahan LED dilakukan menggunakan PWM dari TIM3, sehingga lampu dapat beroperasi dalam kondisi mati, redup, atau terang sesuai dengan kondisi cahaya sekitar dan hasil deteksi gerakan.

 4. Flowchart dan Listing Program [Kembali]
 Flowchart : <img alt="" data-original-height="562" data-original-width="380" height="387"

src="https://blogger.googleusercontent.com/img/a/AVvXsEgqZNRbk6zWG5djEbTXhRL_P8BBUv38eMyAPDFhC53Qpib4V30ex-v8COpqYAtPUphluBqTdPTj3Kdp1ywn7vDK8Ddt90KMpq2LqnPjY3SSoIwMC0t_lqX0GmJ_fkggK__OD7fBNC5-XbRt8LEL7Q5RdkaMX1upM6spT-Ch0uejwaVsoUE9NNF1ZCJYa2DY=w261-h387" style="-webkit-box-shadow: rgba(0, 0, 0, 0.1) 1px 1px 5px; background-attachment: scroll; background-clip: border-box; background-image: none; background-origin: padding-box; background-position: 0% 0%; background-repeat: repeat; background-size: auto; border: 1px solid rgb(238, 238, 238); box-shadow: rgba(0, 0, 0, 0.1) 1px 1px 5px; box-sizing: border-box; max-width: 100%; padding: 5px; position: relative;" width="261" />

 Listing Program : a. main.c #include "main.h" // HANDLE ADC_HandleTypeDef hadc1; TIM_HandleTypeDef htim3; // VARIABLE volatile uint8_t emergency_mode = 0; uint32_t last_motion_time = 0; // fallback tombol uint8_t last_button_state = 1; // PARAMETER #define LDR_THRESHOLD 2000 #define MOTION_TIMEOUT 5000 #define LED_OFF 0 #define LED_DIM 100 #define LED_FULL 1000 // ================= CLOCK ================= void SystemClock_Config(void) { RCC_OscInitTypeDef RCC_OscInitStruct = {0}; RCC_ClkInitTypeDef RCC_ClkInitStruct = {0}; RCC_OscInitStruct.OscillatorType = RCC_OSCILLATORTYPE_HSI; RCC_OscInitStruct.HSIState = RCC_HSI_ON; HAL_RCC_OscConfig(&RCC_OscInitStruct); RCC_ClkInitStruct.ClockType = RCC_CLOCKTYPE_HCLK | RCC_CLOCKTYPE_SYSCLK; RCC_ClkInitStruct.SYSCLKSource = RCC_SYSCLKSOURCE_HSI; RCC_ClkInitStruct.AHBCLKDivider = RCC_SYSCLK_DIV1; HAL_RCC_ClockConfig(&RCC_ClkInitStruct, FLASH_LATENCY_0); } <div style="font-size:

13.2px;">// ================= GPIO ================= <div style="font-size:
13.2px;">void MX_GPIO_Init(void) { <div
style="font-size: 13.2px;"> __HAL_RCC_GPIOA_CLK_ENABLE(); <div
style="font-size: 13.2px;"> __HAL_RCC_GPIOB_CLK_ENABLE(); <div
style="font-size: 13.2px;"> GPIO_InitTypeDef GPIO_InitStruct = {0}; <div
style="font-size: 13.2px;"> // PIR → PA1 <div style="font-size:
13.2px;"> GPIO_InitStruct.Pin = GPIO_PIN_1; <div style="font-size:
13.2px;"> GPIO_InitStruct.Mode = GPIO_MODE_INPUT; <div style="font-size:
13.2px;"> GPIO_InitStruct.Pull = GPIO_NOPULL; <div style="font-size:
13.2px;"> HAL_GPIO_Init(GPIOA, &GPIO_InitStruct); <div style="font-size:
13.2px;"> // BUTTON → PB1 (PULL-UP + INTERRUPT) <div style="font-size:
13.2px;"> GPIO_InitStruct.Pin = GPIO_PIN_1; <div style="font-size:
13.2px;"> GPIO_InitStruct.Mode = GPIO_MODE_IT_FALLING; <div
style="font-size: 13.2px;"> GPIO_InitStruct.Pull = GPIO_PULLUP; <div
style="font-size: 13.2px;"> HAL_GPIO_Init(GPIOB, &GPIO_InitStruct); <div
style="font-size: 13.2px;"> // LED PWM → PA6 <div style="font-size:
13.2px;"> GPIO_InitStruct.Pin = GPIO_PIN_6; <div style="font-size:
13.2px;"> GPIO_InitStruct.Mode = GPIO_MODE_AF_PP; <div style="font-size:
13.2px;"> GPIO_InitStruct.Pull = GPIO_NOPULL; <div style="font-size:
13.2px;"> GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_LOW; <div
style="font-size: 13.2px;"> GPIO_InitStruct.Alternate = GPIO_AF1_TIM3; <div
style="font-size: 13.2px;"> HAL_GPIO_Init(GPIOA, &GPIO_InitStruct); <div
style="font-size: 13.2px;"> // IRQ untuk PB1 (EXTI0_1) <div style="font-size:
13.2px;"> HAL_NVIC_SetPriority(EXTI0_1_IRQn, 0, 0); <div style="font-size:
13.2px;"> HAL_NVIC_EnableIRQ(EXTI0_1_IRQn); <div style="font-size:
13.2px;">} // ================= ADC
================= void MX_ADC1_Init(void) <div
style="font-size: 13.2px;">{ <div style="font-size:
13.2px;"> __HAL_RCC_ADC_CLK_ENABLE(); <div style="font-size:
13.2px;"> hadc1.Instance = ADC1; <div style="font-size:
13.2px;"> hadc1.Init.Resolution = ADC_RESOLUTION_12B; <div style="font-size:
13.2px;"> hadc1.Init.DataAlign = ADC_DATAALIGN_RIGHT; <div style="font-size:
13.2px;"> hadc1.Init.ScanConvMode = ADC_SCAN_DISABLE; <div
style="font-size: 13.2px;"> hadc1.Init.ContinuousConvMode = DISABLE; <div
style="font-size: 13.2px;"> HAL_ADC_Init(&hadc1); <div style="font-size:
13.2px;"> ADC_ChannelConfTypeDef sConfig = {0}; <div style="font-size:
13.2px;"> sConfig.Channel = ADC_CHANNEL_0; <div style="font-size:
13.2px;"> sConfig.Rank = ADC_REGULAR_RANK_1; <div style="font-size:
13.2px;"> HAL_ADC_ConfigChannel(&hadc1, &sConfig); <div
style="font-size: 13.2px;">} // ================= PWM
================= void MX_TIM3_Init(void) <div
style="font-size: 13.2px;">{ <div style="font-size:
13.2px;"> __HAL_RCC_TIM3_CLK_ENABLE(); <div style="font-size:
13.2px;"> htim3.Instance = TIM3; <div style="font-size:

13.2px;"> htim3.Init.Prescaler = 64; <div style="font-size:
13.2px;"> htim3.Init.Period = 1000; <div style="font-size:
13.2px;"> htim3.Init.CounterMode = TIM_COUNTERMODE_UP; <div
style="font-size: 13.2px;"> HAL_TIM_PWM_Init(&htim3); <div style="font-size:
13.2px;"> TIM_OC_InitTypeDef sConfigOC = {0}; <div style="font-size:
13.2px;"> sConfigOC.OCMode = TIM_OCMODE_PWM1; <div style="font-size:
13.2px;"> sConfigOC.Pulse = 0; <div style="font-size:
13.2px;"> HAL_TIM_PWM_ConfigChannel(&htim3, &sConfigOC,
TIM_CHANNEL_1); } //
================ INTERRUPT ================ <div style="font-size:
13.2px;">void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin) <div style="font-size:
13.2px;">{ if (GPIO_Pin == GPIO_PIN_1) <div
style="font-size: 13.2px;"> { emergency_mode
= !emergency_mode; } <div style="font-size:
13.2px;">} // ================ HELPER
================ uint16_t read_LDR(void) <div
style="font-size: 13.2px;">{ <div style="font-size:
13.2px;"> HAL_ADC_Start(&hadc1); <div style="font-size:
13.2px;"> HAL_ADC_PollForConversion(&hadc1, HAL_MAX_DELAY); <div
style="font-size: 13.2px;"> return HAL_ADC_GetValue(&hadc1); <div
style="font-size: 13.2px;">} void set_LED(uint16_t
value) { <div style="font-size:
13.2px;"> __HAL_TIM_SET_COMPARE(&htim3, TIM_CHANNEL_1,
value); } //
================ MAIN ================ int
main(void) { <div style="font-size:
13.2px;"> HAL_Init(); <div style="font-size:
13.2px;"> SystemClock_Config(); <div style="font-size:
13.2px;"> MX_GPIO_Init(); <div style="font-size:
13.2px;"> MX_ADC1_Init(); <div style="font-size:
13.2px;"> MX_TIM3_Init(); <div style="font-size:
13.2px;"> HAL_TIM_PWM_Start(&htim3, TIM_CHANNEL_1); <div
style="font-size: 13.2px;"> while (1) <div style="font-size:
13.2px;"> { // ===== FALLBACK BUTTON
===== uint8_t current_button =
HAL_GPIO_ReadPin(GPIOB, GPIO_PIN_1); if
(last_button_state == 1 && current_button == 0) <div style="font-size:
13.2px;"> { emergency_mode =
!emergency_mode; HAL_Delay(50); <div
style="font-size: 13.2px;"> } last_button_state
= current_button; // ===== MODE DARURAT
===== if (emergency_mode) <div
style="font-size: 13.2px;"> { <div style="font-size:
13.2px;"> set_LED(LED_OFF); <div style="font-size:

13.2px;"> continue; } uint16_t ldr = read_LDR(); uint8_t pir = HAL_GPIO_ReadPin(GPIOA, GPIO_PIN_1); // SIANG if (ldr < LDR_THRESHOLD) { set_LED(LED_OFF); } else { // MALAM if (pir == GPIO_PIN_SET) { last_motion_time = HAL_GetTick(); } if (HAL_GetTick() - last_motion_time < MOTION_TIMEOUT) { set_LED(LED_FULL); } else { set_LED(LED_DIM); } } HAL_Delay(100); } } b. main.h #ifndef __MAIN_H #define __MAIN_H #include "stm32c0xx_hal.h" // ================= PIN DEFINITIONS ================= // LDR (ADC) #define LDR_PORT GPIOA #define LDR_PIN GPIO_PIN_0
 // PA0
 // PIR SENSOR #define PIR_PORT GPIOA #define PIR_PIN GPIO_PIN_1
 // PA1
 // PUSH BUTTON (INTERRUPT) #define BUTTON_PORT GPIOB #define BUTTON_PIN GPIO_PIN_1 // LED PWM #define LED_PORT GPIOA #define LED_PIN GPIO_PIN_6
 // PB1
 // PA6 (TIM3_CH1)
 // ================= FUNCTION PROTOTYPES ================= void SystemClock_Config(void); void MX_GPIO_Init(void); void MX_ADC1_Init(void); void MX_TIM3_Init(void); #endif

 <div style="caret-color: rgb(102, 102, 102); color: #666666;

font-family: "Times New Roman", Times, FreeSerif, serif; font-size: 13.2px;"> 5. Video Demo [Kembali]
 <div class="separator" style="caret-color: rgb(102, 102, 102); clear: both; color: #666666; font-family: "Trebuchet MS", Trebuchet, Verdana, sans-serif; font-size: 15.84px; text-align: center;"><iframe allowfullscreen="allowfullscreen" class="b-hbp-video b-uploaded" frameborder="0" height="210" id="BLOGGER-video-4ca3db6fd7f2e6b1-13690" mozallowfullscreen="mozallowfullscreen" src="https://www.blogger.com/video.g?token=AD6v5dw_jWPj2ukiyTulZhbiCHx17vHqhLcKV935MlrSg5IOEL6zfcM2B7v0U-hHqC8JyZ1z4h33pwfordQG0Wc4wnF2b7aHwjhlYhPerluECa5_qHoOGnjg7jQs7Mi4XfXkfgUcshd2&origin=abigail232061.blogspot.com" webkitallowfullscreen="webkitallowfullscreen" width="280"></iframe>
<div class="separator" style="caret-color: rgb(102, 102, 102); clear: both; color: #666666; font-family: "Trebuchet MS", Trebuchet, Verdana, sans-serif; font-size: 15.84px; text-align: center;">
 <div class="separator" style="caret-color: rgb(102, 102, 102); clear: both; color: #666666; font-family: "Times New Roman", Times, FreeSerif, serif; font-size: 13.2px; text-align: center;">
 6. Analisa [Kembali]
 1.Analisa bagaimana perbedaan implementasi PWM antara STM32 serta dampaknya terhadap kontrol motor dan LED! Jawab: PWM pada STM32 dihasilkan oleh timer (misalnya TIM3) dengan mengatur nilai duty cycle melalui register compare. Perubahan duty cycle mempengaruhi lebar pulsa, sehingga: <span

class="Apple-tab-span" style="white-space: pre;"> • pada LED → mengatur tingkat kecerahan • pada motor/servo → mengatur posisi atau kecepatan Semakin besar duty cycle, semakin besar energi yang diberikan ke beban.
 2.Analisa bagaimana cara pembacaan nilai sensor analog menggunakan ADC pada STM32! Jawab: ADC membaca sinyal analog (0–3.3V) lalu mengubahnya menjadi data digital 12-bit (0–4095). Prosesnya: • konfigurasi channel ADC • start konversi (HAL_ADC_Start) • tunggu selesai (HAL_ADC_PollForConversion) • ambil data (HAL_ADC_GetValue) Nilai ini kemudian digunakan untuk pengambilan keputusan (misalnya threshold LDR).
 3.Analisa bagaimana penggunaan interrupt eksternal dalam mendeteksi input dari sensor atau tombol pada STM32! Jawab: Interrupt eksternal digunakan untuk mendeteksi perubahan input (misalnya tombol) secara langsung tanpa polling. Saat terjadi trigger (falling/rising edge), STM32 menjalankan fungsi callback seperti: HAL_GPIO_EXTI_Callback() Keuntungannya adalah respon cepat dan efisien karena tidak perlu pengecekan terus-menerus di loop.
 4.Analisa bagaimana cara kerja fungsi HAL_GetTick() pada STM32! Jawab: HAL_GetTick() mengembalikan waktu dalam satuan milidetik sejak sistem mulai berjalan. Fungsi ini berbasis timer SysTick dan digunakan untuk: • pengukuran waktu (interval) • delay non-blocking Contohnya untuk menghitung selang waktu detak atau timeout.
 5.Analisa bagaimana perbedaan konfigurasi dan kontrol pin PWM serta pemanfaatan timer internal pada STM32 dalam menghasilkan sinyal PWM! Jawab: PWM dikonfigurasi dengan: • Prescaler → mengatur resolusi waktu • Period (ARR) → menentukan frekuensi PWM • Compare (CCR) → menentukan duty cycle
 Timer internal menghasilkan sinyal periodik, dan perubahan nilai CCR mengubah lebar pulsa tanpa mengubah frekuensi.
 6.Bagaimana mengatur pergerakan motor servo pada stm 32? Jawab: Servo dikontrol dengan PWM frekuensi 50 Hz (periode 20 ms).

Posisi ditentukan oleh lebar pulsa:	• ~1 ms → posisi minimum • ~1.5 ms → posisi tengah • ~2 ms → posisi maksimum
 Di STM32 diatur dengan: __HAL_TIM_SET_COMPARE() Perubahan nilai compare mengubah sudut servo (misalnya 1200–1800 µs).

 7. Download File [Kembali]

 <p style="cursor: url("https://cur.cursors-4u.net/others/oth-6/oth542.ani"), url("https://cur.cursors-4u.net/others/oth-6/oth542.png"), auto !important; font-size: 15px;"><b style="color: #757575; cursor: url("https://cur.cursors-4u.net/others/oth-6/oth542.ani"), url("https://cur.cursors-4u.net/others/oth-6/oth542.png"), auto !important;">   Rangkaian dan Program Wokwi [tekan disini]</p><p style="cursor: url("https://cur.cursors-4u.net/others/oth-6/oth542.ani"), url("https://cur.cursors-4u.net/others/oth-6/oth542.png"), auto !important; font-size:

15px;">   Video Rangkaian [tekan disini]</p><p style="cursor: url("https://cur.cursors-4u.net/others/oth-6/oth542.ani"), url("https://cur.cursors-4u.net/others/oth-6/oth542.png"), auto !important; font-size: 15px;">   HTML [tekan disini][Kembali][Kembali][Kembali][Kembali][Kembali]</p>