



CENTRO PAULA SOUZA
ETEC PAULINO BOTELHO
TÉCNICO EM MECATRÔNICA

Eduardo Johansen Rodrigues
Elias Lucas Guercia Silva
Gabriel Saldanha Dos Anjos
João Guilherme Espindola

AUTOMAÇÃO DE COBERTURA RESIDENCIAL

São Carlos – São Paulo
2025

Eduardo Johansen Rodrigues

Elias Lucas Guercia Silva

Gabriel Saldanha Dos Anjos

João Guilherme Espindola

AUTOMAÇÃO DE COBERTURA RESIDENCIAL

Trabalho de Conclusão de Curso
apresentado ao Curso Técnico em
Mecatrônica da Etec Paulino Botelho,
orientado pelos Professores Luiz
Schiavone Neto e Cláudio Torres
Gonçalves, como requisito parcial para
obtenção do título de Técnico em
mecatrônica

São Carlos

2025



Curso Técnico em Mecatrônica apresentado à Etec Paulino Botelho, constituído por Eduardo Johansen Rodrigues, Elias Lucas Guercia Silva, Gabriel Saldanha Dos Anjos, João Guilherme Espindola.

Aprovado em _____ de _____ de 2025

Aprovado por:

Anderson Beluco

Evandra Maria Raimundo

Professores orientadores:

Claudio Torres Gonçalves

Luiz Schiavone Neto

DEDICATÓRIA

Dedicamos este projeto aos nossos professores, colegas e familiares, pois, por meio do apoio deles, conseguimos alcançar um grande desempenho e desenvolvimento educacional. Não há dúvidas de que foram essenciais para a nossa formação. Somos gratos por nos ajudarem a extrair o melhor de nós mesmos.

AGRADECIMENTOS

Gostaríamos de expressar nossa sincera gratidão aos nossos professores orientadores Cláudio Torres Gonçalves e Luiz Schiavone Neto por todo o apoio, instrução e dedicação, ajudando-nos a extrair o melhor do nosso aprendizado e conhecimento para tornar este projeto uma realidade possível. Nosso profundo agradecimento estende-se também aos colaboradores da ETEC Paulino e a todo o corpo docente, por nos transmitirem o conhecimento necessário para nossa formação profissional, contribuindo para nossa inserção no mercado de trabalho e na área de tecnologia e mecatrônica.

RESUMO

A cobertura automatizada tem como objetivo modernizar e facilitar os cuidados com a área de lazer das residências, protegendo o ambiente de chuvas inesperadas. O desenvolvimento baseia-se na implementação de um sensor integrado a um microcontrolador responsável pelo acionamento automático da cobertura, permitindo seu movimento de abertura e fechamento conforme as condições climáticas detectadas. Além disso, o sistema conta com um botão de acionamento manual, possibilitando ao usuário controlar a cobertura conforme a necessidade, visando ao conforto térmico em dias de sol intenso. O sistema oferece praticidade, funcionalidade e conveniência, garantindo o uso do espaço de forma eficiente e confortável em diferentes condições climáticas. Os resultados alcançados evidenciam o bom desempenho do protótipo e confirmam sua viabilidade de aplicação em residências que buscam automação e conforto.

Palavras-chave: Automação residencial; Cobertura automatizada; Sensor de chuva; Microcontrolador; Controle automático.

ABSTRACT

The automated roof aims to modernize and simplify the maintenance of residential leisure areas, protecting the environment from unexpected rainfall. The system is based on the implementation of a sensor integrated with a microcontroller responsible for the automatic operation of the roof, enabling its opening and closing according to the detected weather conditions. In addition, the system includes a manual control button that allows the user to operate the roof as needed to achieve thermal comfort on days of intense sunlight. The system provides practicality, functionality, and convenience, ensuring efficient and comfortable use of the space under different weather conditions. The results obtained demonstrate the prototype's good performance and confirm its feasibility for application in residential settings seeking automation and comfort.

Keywords: Home automation; Automated cover; Rain sensor; Microcontroller; Automatic control.

LISTA DE FIGURAS

Figura 1 – Arduino Uno	16
Figura 2 – Sensor de chuva.....	17
Figura 3 – Módulo do sensor de chuva	17
Figura 4 – Servo motor	18
Figura 5 – Sensor ultrassônico	19
Figura 6 – Chave fim de curso.....	20
Figura 7 – Botão	20
Figura 8 – Engrenagem.....	21
Figura 9 – Cremalheira.....	21
Figura 10 – LED	22
Figura 11 – Resistor	22
Figura 12 – Esquema de ligação	25
Figura 13 – Programação do Arduino.....	26
Figura 14 – Programação do Arduino.....	27
Figura 15 – Programação do Arduino.....	28
Figura 16 – Programação do Arduino.....	29
Figura 17 – Programação do Arduino.....	30
Figura 18 - Montagem	34
Figura 19 - Montagem	34
Figura 20 - Montagem e ligações dos componentes	35

LISTA DE TABELAS

Tabela 1 – Evolução da adoção de algumas tecnologias	12
Tabela 2 – Cronograma do desenvolvimento do projeto.....	31
Tabela 3 – Lista de componentes	32

ABREVIATURAS

AO – Analógico Out (Saída Analógica)

DC – Direct Current (Corrente Contínua)

IDE – Integrated Development Environment (Ambiente Integrado de Desenvolvimento)

ICSP – In-Circuit Serial Programming (Programação Serial Em Circuito)

IoT – Internet of Things (Internet das Coisas)

LED – Light Emitting Diode (Diodo Emissor de Luz)

MDF – Medium-Density Fiberboard (Fibra de Média Densidade)

MHz – Megahertz

mm – Milímetro

PLC – Power Line Carrier (Comunicação por Rede Elétrica)

TTL – Transistor-Transistor Logic (Lógica Transistor-Transistor)

USB – Universal Serial Bus

V – Volt

SUMÁRIO

1.	INTRODUÇÃO	12
1.1	OBJETIVO	13
1.2	JUSTIFICATIVA.....	14
2.	METODOLOGIA	15
3.	FUNDAMENTOS TEÓRICOS	16
3.1	ARDUINO UNO	16
3.2	SENSOR DE CHUVA	16
3.3	MÓDULO	17
3.4	SERVO MOTOR.....	18
3.5	SENSOR ULTRASSÔNICO	19
3.6	CHAVES FIM DE CURSO	19
3.7	BOTÃO.....	20
3.8	ENGRENAGEM	20
3.9	CREMALHEIRA	21
3.10	LED	21
3.11	RESISTOR	22
4.	PLANEJAMENTO DO PROJETO.....	23
5.	FUNCIONAMENTO DO SISTEMA	24
6.	ESQUEMA ELÉTRICO	25
7.	PARTE LÓGICA.....	26
8.	CRONOGRAMA DO DESENVOLVIMENTO DO PROJETO	31
9.	LEVANTAMENTO E AQUISIÇÃO DOS COMPONENTES	32
10.	CONCLUSÃO	33
11.	FOTOS DO PROJETO.....	34
	REFEÊNCIAS	36

1. INTRODUÇÃO

No final da década de 1970, surgiram os primeiros módulos “inteligentes”, onde comandos eram enviados pela própria rede elétrica da residência, com o conceito de PLC (Power Line Carrier). Eram soluções básicas, com pouca ou nenhuma integração entre si, voltadas apenas para resolver necessidades específicas, como acionar remotamente um equipamento ou iluminação.

Em 1980, surge o termo Casa Inteligente. Nessa época, itens começam a se tornar mais acessíveis em países como os Estados Unidos. Sistemas de segurança, luzes com sensores de fumaça e portões automáticos popularizaram-se nessa época.

No ano de 1990, ocorreu um grande feito considerado como a tecnologia da época: Simon Hackett e John Romkey criaram a primeira torradeira de pão controlada pela internet, inaugurando a era da Internet das Coisas (IoT – Internet of Things). Esse termo foi criado por Kevin Ashton, nove anos depois da torradeira.

Foi apresentada, no início dos anos 2000, a visão de casa inteligente, que incluía dispositivos inteligentes, como sistemas de segurança por câmeras conectadas, fechaduras smart e controle completo da iluminação. Em seguida, com os avanços dos smartphones, surgiram oportunidades únicas para a automação, que ganhou alcance global e se mantém em expansão até os dias atuais.

Tabela 1 – Evolução da adoção de algumas tecnologias

Tecnologia	2003	2004	2005	2006	2015
Cabeamento estruturado	42%	61	49%	53%	80%
Monitoramento de segurança	18%	28	29%	32%	81%
Multiroom audio	9%	12%	15%	16%	86%
Home Theater	9%	8%	11%	12%	86%
Controle de iluminação	1%	2%	6%	8%	75%
Automação integrada	0	2%	6%	6%	70%
Gerenciamento de energia	1%	5%	11%	11%	62%

Fonte: NAHB Research Centre, CEA

A cobertura automatizada não surgiu de uma invenção específica, mas como resultado natural do avanço da automação e da arquitetura. O uso de motores e sensores para movimentar estruturas retráteis é relativamente recente e ganhou força nas últimas décadas, acompanhando o crescimento das tecnologias voltadas para o conforto e a praticidade no ambiente residencial.

Este trabalho foi desenvolvido a partir da observação da necessidade e da evolução dessas soluções de automação residencial e comercial, identificando uma situação-problema comum em diversas residências e a busca por alternativas inteligentes e eficientes. Nesse cenário, a cobertura retrátil automatizada com sensor de chuva apresenta-se como uma proposta que une tecnologia, proteção e comodidade, realizando uma releitura de projetos já existentes no mercado, porém incorporando melhorias e reestruturações na parte de automação.

O sistema oferece versatilidade ao ambiente ao permitir a abertura e o fechamento automático da cobertura conforme as condições climáticas, reduzindo riscos de danos causados pela exposição à chuva e aumentando o conforto dos usuários. Além disso, o projeto busca otimizar o uso de áreas externas, tornando-as funcionais em diferentes situações, ao mesmo tempo em que agrega inovação e praticidade ao espaço.

1.1 OBJETIVO

O projeto consiste em uma cobertura automatizada capaz de avançar e recuar, programada por meio do Arduino UNO e acionada através de um sensor de chuva que identifica a presença de água. Essa automação permite o deslocamento da cobertura para proteger áreas externas contra chuvas e tempestades inesperadas, podendo ser utilizada em diversos ambientes, como residências, chácaras e áreas de lazer. O sistema conta ainda com um dispositivo de segurança, composto por um sensor ultrassônico que impede possíveis incidentes durante o funcionamento, além de um botão manual que possibilita o acionamento independente do sensor de chuva, podendo ser utilizado em situações de forte incidência solar ou em outras condições que tornem necessário o controle manual do sistema. O objetivo é proporcionar maior proteção, praticidade e melhor

aproveitamento dos espaços externos, integrando conceitos de eletrônica, mecânica e automação. Além de demonstrar a funcionalidade do sistema, o projeto visa aplicar na prática os conhecimentos adquiridos em sala de aula, laboratórios e oficinas, incentivando a inovação tecnológica e a solução de problemas do cotidiano.

1.2 JUSTIFICATIVA

Proporcionar um projeto de aplicação de automação mecatrônica residencial controlando uma cobertura retrátil automatizada oferecendo conforto e inovação práticos para proteger áreas abertas de residências e outros locais para eventos de possíveis fenômenos naturais não previstos.

2. METODOLOGIA

O desenvolvimento do projeto foi dividido em duas etapas principais. A primeira foi realizada por meio de um levantamento de pesquisas sobre automação residencial, efetuado através de sites, protótipos e referências já existentes no mercado atual. Essa fase teve como objetivo definir um sistema de sensores aplicado a coberturas de áreas residenciais abertas.

Na segunda fase, com base nas pesquisas realizadas, foram iniciadas as montagens do projeto, solucionando eventualidades e problemas que surgiram ao longo do processo. Em paralelo, iniciou-se também a programação do microcontrolador, com testes realizados tanto na montagem quanto no ambiente de programação.

3. FUNDAMENTOS TEÓRICOS

3.1 ARDUINO UNO

Arduino é uma placa microcontroladora embasada no ATmega328P. Possui 14 pinos de entrada/saída digitais 6 entradas analógicas, um ressonador cerâmico de 16 MHz, uma conexão USB, um conector de alimentação, um conector ICSP e um botão de reset. O Arduino uno recebe um código escrito na Arduino IDE (um software gratuito usado para escrever, compilar e carregar código para placas Arduino e microcontroladores compatíveis).

Figura 1–Arduino Uno



<https://www.vjcomponentes.com.br/arduino-uno-r3-com-cabo-usb> (s.d.).

3.2 SENSOR DE CHUVA

O Sensor de Chuva funciona detectando gotas de água por meio da medição na mudança da resistência elétrica entre suas trilhas de cobre expostas quando a água as cobre.

A base do sensor possui uma série de linhas de cobre expostos, em conjunto atua como um resistor variável, cuja resistência varia de acordo com a quantidade de água presente em sua superfície.

Figura 2–Sensor de chuva

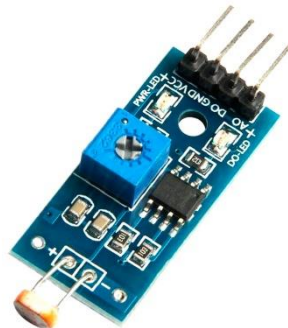


Fonte: <https://www.usinainfo.com.br/outros-sensores/sensor-de-chuva-fc37-para-arduino-modulo-de-leitura-2579.html> (s.d.).

3.3 MÓDULO

O Módulo sensor é um dispositivo eletrônico que realiza a leitura através da base sensor. Ele pode ser facilmente conectado ao Arduino ou a outros microcontroladores para fins de automação. O módulo gera uma tensão de saída que varia conforme a resistência da base sensora, a qual responde à presença de água. O mesmo gera uma tensão de saída de acordo com a resistência da base sensora e é disponibilizada em um pino de saída analógica (AO).

Figura 3–Módulo do sensor de chuva



Fonte: <https://www.casadarobotica.com/sensores-modulos/sensores/luminosidade/modulo-sensor-de-luminosidade-ldr-com-lm393-saida-digital-e-analogica> (s.d.).

3.4 SERVO MOTOR

O servo motor possui a função de converter energia elétrica em mecânica. Diferente dos motores convencionais, os servos motores são usados para ter mais controle e precisão em sua rotação, podendo controlar sua posição. O servo motor MG90D tem a rotação de 360°, ou seja, de rotação contínua.

Especificações:

Peso: 13g

Dimensões: 22,8×12,2×28,5mm

Torque de estol: 2,1kg/cm (4,8V); 2,4kg/cm (6,6V)

Velocidade de operação: 0,10s/60° (4,8V); 0,08s/60° (6,0V)

Tensão de operação: 4,8V~ 6,6V

Tipo de engrenagem: Engrenagem metálica

Carcaça do servo: Central com carcaça de liga metálica

Rolamento: Rolamento de esferas duplo

Faixa de temperatura: 0-55°C

Largura da zona morta: 1μs

Figura 4–Servo motor



Fonte: <https://www.eletrogate.com/micro-servo-motor-tower-pro-mg90d-metal-gear-360> (s.d.).

3.5 SENSOR ULTRASSÔNICO

O sensor ultrassônico permite que faça a leitura da distância de um objeto, usando ondas sonoras de alta frequência. Isso só é possível, graças a um receptor e um emissor. Após a colisão do sinal no objeto enviado pelo emissor, o sinal é refletido de volta ao receptor, o que resulta no descobrimento da distância entre o sensor e o obstáculo.

Especificações:

Tensão de Operação: 5V DC

Corrente de Operação: 15Ma

Distância de Medição: 2cm a 4m

Precisão: 3mm

Ângulo de Detecção: 15°

Frequência de Operação: 40kHz

Sinal de Entrada (Trigger): Pulso TTL de 10us

Sinal de Saída (Echo): Sinal de pulso (PWL) TTL

Figura 5—Sensor ultrassônico



Fonte: https://d229kd5ey79zj.cloudfront.net/620/images/620_1_H.png (s.d.).

3.6 CHAVES FIM DE CURSO

A chave fim de curso é um sensor de impacto, detectando a posição inicial ou final de um objeto. Cada vez que um objeto pressionar a haste, será acionado o micro switch que consequentemente avisará o microcontrolador.

Figura 6—Chave fim de curso



Fonte: <https://www.autocorerobotica.com.br/chave-fim-de-curso-micro-switch-kw10b-125v-1a> (s.d.).

3.7 BOTÃO

O botão é um dispositivo de acionamento que, quando pressionado, tem a função de abrir ou fechar o circuito, convertendo assim um comando mecânico em um sinal elétrico. As chaves geralmente são de contato de ação momentânea e podem ser do tipo NF (normalmente fechada) ou NA (normalmente aberta).

Figura 7—Botão



Fonte: <https://www.eletrogate.com/push-button-chave-tactil-6x6x6mm> (s.d.).

3.8 ENGRENAGEM

A engrenagem é uma peça mecânica composta por dentes ao seu redor, usadas para transmitir rotação e torque. Cada engrenagem possui suas características variando de acordo com o que é previsto ao uso dela, sendo essencial para diversas aplicações.

Figura 8—Engrenagem



Fonte: <https://www.cetro.com.br/engrenagem-plastica-35-dentes-de-tracao-do-tefloncorreia-sa-1000-e-sa-900/p> (s.d.).

3.9 CREMALHEIRA

A cremalheira é uma peça mecânica em forma de barra com dentes, que em conjunto com a engrenagem dentada (pinhão), transforma o movimento de rotação em movimento retilíneo.

Figura 9—Cremalheira



Fonte: <https://www.patolastore.com.br/cremalheiras-para-automacao-de-portao-modelo-guia-dentada-6-preto/p> (s.d.).

3.10 LED

Do inglês “light emitting diode”, mais conhecido como LED é um componente eletrônico que converte energia elétrica em luz com baixo consumo e longa vida útil. Seu princípio de funcionamento é a eletroluminescência, onde elétrons e “lacunas” se recombinaem em um material semicondutor, emitindo fótons no processo.

Figura 10 – LED



Fonte: <https://www.eletrogate.com/led-alto-brilho-5mm-vermelho> (s.d.).

3.11 RESISTOR

O resistor é um componente eletrônico que resiste à passagem de corrente elétrica. No circuito quando inserido o resistor, verifica-se uma diminuição na intensidade da corrente elétrica. A inserção de um resistor em série em um circuito limita a intensidade da corrente elétrica, de acordo com a Lei de Ohm. Sem o resistor no circuito dos LEDs inseridos no projeto, eles queimariam.

Figura 11 – Resistor



Fonte: <https://www.usinainfo.com.br/resistor-14-watt/resistor-10k-14w-kit-com-10-unidades-2975.html> (s.d.).

4. PLANEJAMENTO DO PROJETO

O projeto desenvolveu-se em várias etapas, iniciando pela escolha de como seria feita a montagem em termos de escala. Por se tratar do desenvolvimento de uma maquete de casa, cotamos tudo o que seria necessário para realizar sua modificação, já que decidimos trabalhar a partir de um molde de casa pré-estabelecido. Por meio de uma lista, definimos os materiais necessários para realizar a instalação e dar prosseguimento à cobertura automatizada.

Pelo fato de a casa ser feita de madeira do tipo MDF, a modificação e a estruturação do telhado foram facilitadas. O principal desafio a ser enfrentado era definir como o telhado se moveria para frente e para trás de forma suave e onde o motor seria instalado para cumprir sua função de deslocar a cobertura. Dessa forma, iniciamos com testes e medições; em seguida, cortamos hastes de madeira que serviriam como guias para o telhado automatizado. Também realizamos ajustes, como cortes para a passagem da cremalheira e adequações nos cantos para o contato do fim de curso.

Decidimos então a posição do motor e da cremalheira e, em seguida, efetuamos a colagem dos componentes em seus respectivos lugares, utilizando como principal forma de junção o adesivo termoplástico, conhecido como “cola quente”. Paralelamente à montagem, foi realizada a programação do Arduino, com testes contínuos para integrar gradualmente os componentes tanto na maquete quanto no código. Foram adicionados, nessa sequência, os seguintes elementos: sensor de chuva, servo motor, contatos de fim de curso, sensor ultrassônico, botão e, por fim, os LEDs.

5. FUNCIONAMENTO DO SISTEMA

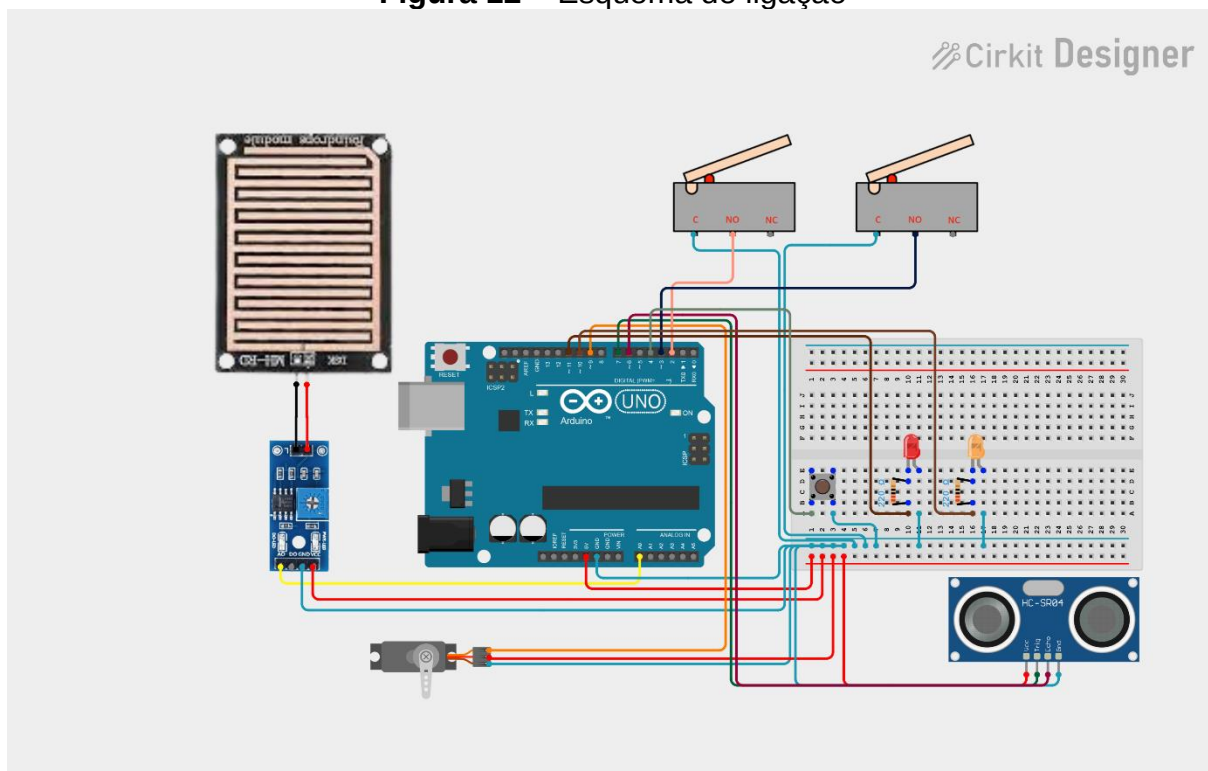
O projeto permanece energizado 24 horas por dia, operando continuamente a 100%. Todos os componentes estão conectados ao Arduino, que é alimentado por meio de um cabo USB fornecendo 5 volts, sem utilização de baterias ou outras fontes alternativas. Para garantir segurança durante manutenções ou em situações em que alguém possa, por acidente, estar sobre ou próximo ao telhado, foi utilizado um sensor ultrassônico para detectar a presença de pessoas.

LEDs indicadores foram adicionados para facilitar a visualização do estado do sistema, sendo o LED verde acionado quando o telhado está aberto e o LED vermelho quando está fechado. Ambos utilizam resistores de 220 ohms conectados à protoboard. Um botão manual também foi incorporado para situações em que seja necessário acionar a cobertura sem depender do sensor de chuva, como em períodos de sol intenso ou outras circunstâncias específicas. O sensor ultrassônico, o botão e os LEDs estão fixados na parte superior de um mini painel em caixa plástica, ficando visíveis e acessíveis para operação.

A parte mecânica do projeto é composta por uma engrenagem com 31 dentes e aproximadamente 300 mm de diâmetro, acoplada a uma cremalheira de 46 dentes com cerca de 1280 mm de comprimento, sendo ambas de polímero. O servo motor, ao ser acionado, movimenta a engrenagem, que por sua vez traciona a cremalheira, permitindo que a cobertura avance ou recue conforme a necessidade.

6. ESQUEMA ELÉTRICO

Figura 12 – Esquema de ligação



Fonte: própria

7. PARTE LÓGICA

Figura 13 – Programação do Arduino

```
1  #include <Servo.h>
2
3  Servo servo;
4
5  // pinos
6  const int pinoSensorChuva = A0;      // Sensor de chuva analógico
7  const int pinoTrig = 7;              // Trig do ultrassônico
8  const int pinoEcho = 6;              // Echo do ultrassônico
9  const int fimCursoFechado = 2;        // Fim de curso 1 (fechada)
10 const int fimCursoAberto = 3;         // Fim de curso 2 (aberta)
11 const int botaoManual = 4;            // Botão manual com INPUT_PULLUP
12 const int ledVerde = 10;              // LED verde = cobertura aberta
13 const int ledVermelho = 11;          // LED vermelho = cobertura fechada
14
15 // variáveis
16 long duracao;
17 int distancia;
18 int valorSensorChuva = 0;
19
20 bool girando = false;
21 bool modoManualAtivo = false;
22
23 // protótipos (evita problemas de "não declarado no escopo")
24 long obterDistancia();
25 void moverServo(int velocidade, int fimCursoAlvo);
26 void atualizarLeds(bool fechado, bool aberto);
27
28 void setup() {
29     servo.attach(9);
30     servo.writeMicroseconds(1500); // Motor parado
31
32     pinMode(pinoTrig, OUTPUT);
33     pinMode(pinoEcho, INPUT);
34     pinMode(fimCursoFechado, INPUT_PULLUP);
35     pinMode(fimCursoAberto, INPUT_PULLUP);
36     pinMode(botaoManual, INPUT_PULLUP);
37     pinMode(ledVerde, OUTPUT);
38     pinMode(ledVermelho, OUTPUT);
39
40     // LEDs inicialmente apagados
41     digitalWrite(ledVerde, LOW);
42     digitalWrite(ledVermelho, LOW);
43
44     Serial.begin(9600);
45 }
```

Fonte: própria

Figura 14 – Programação do Arduino

```
45 }
46
47 void loop() {
48     // Leitura de sensores
49     valorSensorChuva = analogRead(pinoSensorChuva);
50     distancia = obterDistancia();
51
52     // Leitura de estados
53     bool emFimCursoFechado = digitalRead(fimCursoFechado) == LOW;
54     bool emFimCursoAberto = digitalRead(fimCursoAberto) == LOW;
55     bool botaoPressionado = digitalRead(botaoManual) == LOW;
56     bool chovendo = (valorSensorChuva < 900); // Menor valor = molhado
57
58     // Atualiza LEDs de status
59     atualizarLeds(emFimCursoFechado, emFimCursoAberto);
60
61     Serial.print("Chuva: ");
62     Serial.print(valorSensorChuva);
63     Serial.print(" | FC Fechado: ");
64     Serial.print(emFimCursoFechado);
65     Serial.print(" | FC Aberto: ");
66     Serial.print(emFimCursoAberto);
67     Serial.print(" | Manual: ");
68     Serial.println(modoManualAtivo);
69
70     // --- BOTÃO MANUAL ---
71     static bool botaoAnterior = HIGH;
72     if (botaoPressionado && botaoAnterior == HIGH && !girando) {
73         girando = true;
74         modoManualAtivo = true; // Bloqueia o automático até chover de novo
75
76         if (emFimCursoFechado) {
77             Serial.println("Botão: abrindo manualmente...");
78             moverServo(1700, fimCursoAberto);
79         }
80         else if (emFimCursoAberto) {
81             Serial.println("Botão: fechando manualmente...");
82             moverServo(1300, fimCursoFechado);
83         }
84         else {
85             Serial.println("Botão: cobertura no meio – ignorando.");
86         }
87
88         girando = false;
```

Fonte: própria

Figura 15 – Programação do Arduino

```
89     }
90     botaoAnterior = botaoPressionado;
91
92     // --- REATIVA AUTOMÁTICO QUANDO CHOVE ---
93     if (chovendo) {
94         modoManualAtivo = false;
95     }
96
97     // --- MODO AUTOMÁTICO ---
98     if (!modoManualAtivo) {
99         if (chovendo && emFimCursoFechado && !girando) {
100             girando = true;
101             Serial.println("Chuva detectada – abrindo cobertura automaticamente...");
102             moverServo(1700, fimCursoAberto);
103             girando = false;
104         }
105
106         if (!chovendo && emFimCursoAberto && !girando) {
107             girando = true;
108             Serial.println("Sem chuva – fechando cobertura automaticamente...");
109             moverServo(1300, fimCursoFechado);
110             girando = false;
111         }
112     }
113
114     delay(100); // pequena pausa para estabilidade
115 }
116
117 // --- Movimento do servo respeitando sensor ultrassônico e fim de curso ---
118 void moverServo(int velocidade, int fimCursoAlvo) {
119     servo.writeMicroseconds(velocidade);
120
121     while (true) {
122         distancia = obterDistancia();
123
124         // Pausar se detectar obstáculo (menor que 10 cm)
125         if (distancia > 0 && distancia < 10) {
126             servo.writeMicroseconds(1500); // parar
127             Serial.println("Presença detectada. Pausando...");
128             delay(500);
129             while (obterDistancia() < 10) delay(200);
130             Serial.println("Presença saiu. Continuando...");
131             servo.writeMicroseconds(velocidade);
132         }
133     }
```

Fonte: própria

Figura 16 – Programação do Arduino

```
133
134 // Parar ao atingir fim de curso
135 if (digitalRead(fimCursoAlvo) == LOW) {
136     servo.writeMicroseconds(1500);
137     Serial.println("Fim de curso atingido. Parando...");
138     break;
139 }
140
141 delay(100);
142 }
143
144 // garante atualização dos LEDs após movimento
145 bool emFechado = digitalRead(fimCursoFechado) == LOW;
146 bool emAberto = digitalRead(fimCursoAberto) == LOW;
147 atualizarLeds(emFechado, emAberto);
148 }
149
150 // --- Função para obter distância (ultrassônico) ---
151 long obterDistancia() {
152     digitalWrite(pinoTrig, LOW);
153     delayMicroseconds(2);
154     digitalWrite(pinoTrig, HIGH);
155     delayMicroseconds(10);
156     digitalWrite(pinoTrig, LOW);
157
158     duracao = pulseIn(pinoEcho, HIGH, 30000); // Timeout 30 ms
159     if (duracao == 0) return 999;             // Nenhum objeto detectado
160     return duracao * 0.0344 / 2;             // retorna cm
161 }
162
163 // --- Função para atualizar os LEDs ---
164 void atualizarLeds(bool fechado, bool aberto) {
165     if (fechado) {
166         digitalWrite(ledVermelho, HIGH);
167         digitalWrite(ledVerde, LOW);
168     }
169     else if (aberto) {
170         digitalWrite(ledVermelho, LOW);
171         digitalWrite(ledVerde, HIGH);
172     }
173     else {
174         // Se estiver em movimento ou no meio
175         digitalWrite(ledVermelho, LOW);
176         digitalWrite(ledVerde, LOW);
177     }
178 }
```

Fonte: própria

Figura 17 – Programação do Arduino

```
176 | digitalWrite(ledVerde, LOW);  
177 | }  
178 | }  
179
```

Fonte: própria

8. CRONOGRAMA DO DESENVOLVIMENTO DO PROJETO

Tabela 2 – Cronograma do desenvolvimento do projeto

		CRONOGRAMA TCC 4º TEC MECATRONICA 2025																									
		TEMA: AUTOMAÇÃO DE COBERTURA RESIDENCIAL																									
		JULHO				AGOSTO				SETEMBRO				OUTUBRO				NOVEMBRO				DEZEMBRO					
ITENS	DESCRIÇÃO																										
1	DESENHO DO PROJETO																										
2	CRONOGRAMA DE ATIVIDADES																										
3	PESQUISA BIBLIOGRAFICA																										
4	ESQUEMA ELÉTRICO																										
5	LEITURA DA BIBLIOGRAFIA																										
6	REDIGIR OBJETIVO DO TCC																										
7	ORÇAMENTO (PLANILHA DE CUSTOS)																										
8	REDIGIR RESUMO DO TCC																										
9	DESENVOLVIMENTO TEÓRICO																										
10	REFERENCIAL TEÓRICO																										
11	RESULTADO E CONCLUSÕES																										
LEGENDA		PLANEJADO																									
		REALIZADO																									
		ATRASSO																									

Fonte: própria

9. LEVANTAMENTO E AQUISIÇÃO DOS COMPONENTES

Tabela 3 – Lista de componentes

ORÇAMENTO COBERTURA AUTOMATIZADA		
QUANTIDADE	DESCRIÇÃO	VALOR R\$
1	Arduino Uno	R\$ 38,00
1	Maquete	R\$ 54,00
1	Servo Motor	R\$ 68,00
1	Botão	R\$ 1,50
2	Led	R\$ 0,70
1	Sensor de chuva	R\$ 16,00
1	Sensor Ultrassônico	R\$ 19,00
2	Chave Fim de Curso	R\$ 6,00
1	Fiação	R\$ 28,00
1	Cobertura de MDF	R\$ 26,95
1	Protoboard	R\$ 20,00
1	Mini painel	R\$ 19,20
1	Caixa componentes	R\$ 8,95
1	Cremalheira	R\$ -
1	Engrenagem	R\$ -
	Total	R\$ 306,30

Fonte: própria

10. CONCLUSÃO

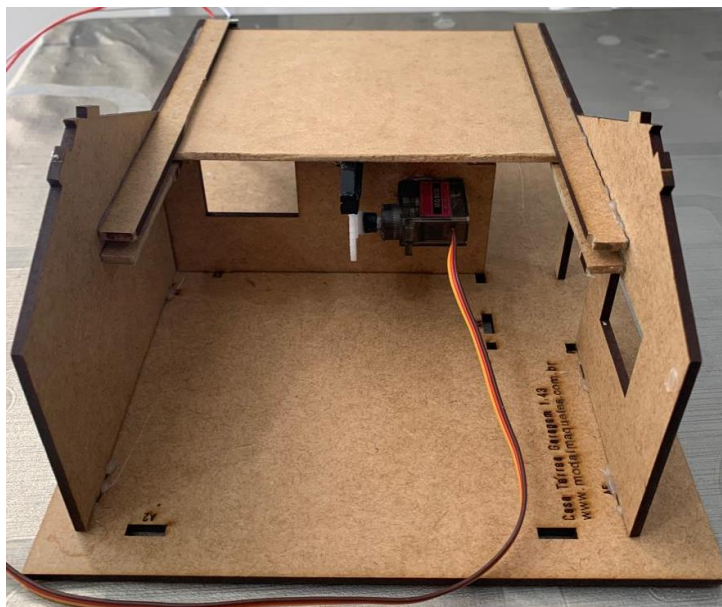
A cobertura automatizada permitiu demonstrar como pode ser aplicada de forma eficiente, solucionando problemas cotidianos e protegendo áreas externas contra condições climáticas adversas. A construção da maquete possibilitou identificar ajustes necessários, tanto na parte mecânica quanto na programação, para garantir seu pleno funcionamento.

O uso do microcontrolador Arduino UNO demonstrou ser essencial para a automação do telhado, para o controle do motor, dos sensores e dos demais componentes, garantindo, assim, o funcionamento confiável na operação do avanço e recuo da cobertura. Além disso, o trabalho prático contribuiu para o desenvolvimento de competências em automação, eletrônica e solução de problemas, destacando a importância da integração entre diferentes áreas da mecatrônica.

Com base nos resultados obtidos, o projeto demonstrou ser possível a sua aplicação na prática em uso residencial e comercial, podendo ser aprimorado futuramente, como a integração de aplicativos e o uso de materiais mais resistentes. Assim, conclui-se que a proposta trouxe soluções tecnológicas significativas, promovendo o aprendizado e favorecendo o desenvolvimento de sistemas automatizados mais eficientes e acessíveis.

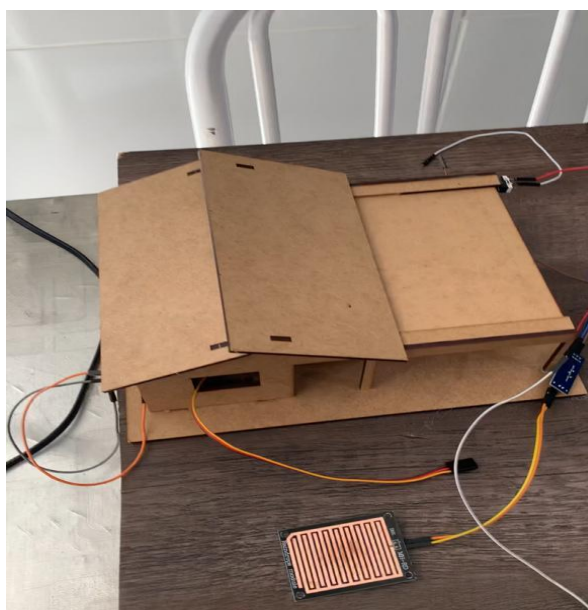
11. FOTOS DO PROJETO

Figura 18 - Montagem



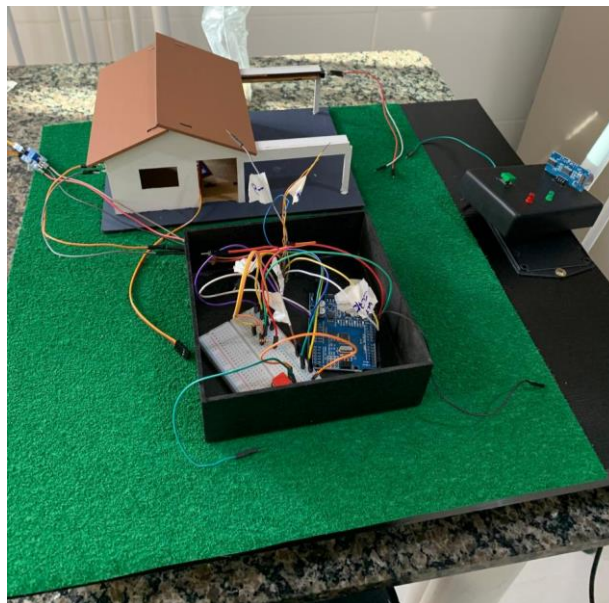
Fonte: própria

Figura 19 - Montagem



Fonte: própria

Figura 20 - Montagem e ligações dos componentes



Fonte: própria

REFEÊNCIAS

A HISTÓRIA da automação residencial: cinco décadas de evolução. [S. l.], 16 dez. 2020. Disponível em: <https://blog.positivocasainteligente.com.br/historia-automacao-residencial/#:~:text=%C2%BB%20A%20hist%C3%B3ria%20da%20automa%C3%A7%C3%A3o%20residencial:%20cinco%20d%C3%A9cadas%20de%20evolu%C3%A7%C3%A3o>. Acesso em: 10 nov. 2025.

ARDUINO Uno Rev3. [S. l.], s.d.. Disponível em: <https://store.arduino.cc/products/arduino-uno-rev3>. Acesso em: 13 out. 2025.

COMO UTILIZAR O SENSOR DE CHUVA. [S. l.], 27 set. 2022. Disponível em: <https://www.sta-eletronica.com.br/artigos/arduinos/como-utilizar-o-sensor-de-chuva>. Acesso em: 29 set. 2025.

ENGRENAGEM: o que é? Quais são os seus principais modelos? Aprenda tudo sobre esse dispositivo mecânico!. [S. l.], 8 nov. 2021. Disponível em: <https://engrenapecas.com.br/engrenagem-o-que-e-quais-sao-os-seus-principais-modelos-aprenda-tudo-sobre-esse-dispositivo-mecanico/>. Acesso em: 6 out. 2025.

MG90D. [S. l.]: TowerPro, s.d.. Disponível em: <https://towerpro.com.tw/product/mg90d-2/>. Acesso em: 6 out. 2025.

MICRO SWITCH: KW10 Series. [S. l.], s.d.. Disponível em: <https://www.javanelec.com/stfiles/getappdocument/1/true/67aae240-b63b-4251-9c29-81eefc0a2461.pdf>. Acesso em: 27 out. 2025.

MURATORI, José Roberto; DAL BÓ, Paulo Henrique. **Automação residencial: histórico, definições e conceitos**. [S. l.], 13 abr. 2011. Disponível em: <https://www.osetoreletrico.com.br/capitulo-i-automacao-residencial-historico-definicoes-e-conceitos/>. Acesso em: 25 ago. 2025.

O QUE é LED: funcionamento, tipos e exemplos. [S. l.], 4 abr. 2024. Disponível em: <https://www.makerhero.com/guia/componentes-eletronicos/led/>. Acesso em: 20 out. 2025.

RESISTORES. [S. l.], 4 ago. 2008. Disponível em: <https://brasilescola.uol.com.br/fisica/resistores.htm>. Acesso em: 20 out. 2025.

SENSOR ULTRASSÔNICO HC-SR04. [S. l.], 27 set. 2022. Disponível em: <https://www.sta-eletronica.com.br/artigos/arduinos/sensor-ultrassonico-hc-sr04>. Acesso em: 20 out. 2025.

VIANA, Carol Correia. Como utilizar o sensor de chuva com Arduino. [S. l.], 23 fev. 2024. Disponível em: <https://www.blogdarobotica.com/2024/02/23/como-utilizar-o-sensor-de-chuva-com-arduino/>. Acesso em: 29 set. 2025.