

Desenvolvimento de uma solução distribuída de controlo de tráfego temporário

David Alexandre Matos Valente a40414

Trabalho realizado sob a orientação de:

Professor Doutor José Lima

Eng. Márcio Correia.

Desenvolvimento de uma solução distribuída de controlo de tráfego temporário

Mestrado em Engenharia Industrial
(Ramo Engenharia Eletrotécnica)
Escola Superior de Tecnologia e Gestão

David Alexandre Matos Valente a40414

2023

Dedicatória

Dedico este trabalho a todos os que, de alguma forma, ajudaram-me e tornaram isso possível. Aos meus pais, Clarisse e Davide Valente, agradeço todo o apoio, encorajamento e incentivo à minha educação e ao meu pensamento crítico. Ao meu irmão, João, agradeço por todos os momentos, sejam eles bons ou maus, assim como pelos conselhos e críticas que me moldaram. Aos meus amigos, em especial ao Bruno Olavo e à Francisca Teixeira, agradeço pelos conselhos, opiniões e discussões no desenvolvimento deste trabalho.

Quero dedicar este trabalho, de forma especial, ao meu avô (in memoriam), cuja presença foi essencial na minha vida.

Dedico também ao meu afilhado, Rodrigo, e à sua irmã, Maria, para que o significado deste trabalho possa servir de incentivo ao longo caminho que terão pela frente. E também aos meus primos, em especial à Vera e ao Hugo, quero dedicar um agradecimento especial pela força e conselhos que me ajudaram a passar por estes anos de estudos. A vossa presença foi essencial e sou grato pelo apoio incondicional que sempre me deram.

Agradecimentos

Gostaria de prestar alguns agradecimentos a que aqueles, de alguma forma, me ajudaram e apoiaram na realização deste trabalho.

Em primeiro lugar gostava de agradecer ao Instituto Politécnico de Bragança (IPB), e aos professores que contribuíram para a minha formação, que me ajudaram a crescer e a ultrapassar todas as dificuldades. Também gostaria de expressar o meu agradecimento à empresa Displays & Mobility Solutions (DMS) pela oportunidade de trabalhar em conjunto e desenvolver este projeto. Agradeço à DMS pela confiança depositada em mim e pela colaboração ao longo de todo o processo.

Gostaria de expressar a minha profunda gratidão ao professor José Lima pela sua orientação durante todo o processo. Agradeço pela abertura de portas que nunca imaginei ter, pelas sugestões e correções valiosas, pelo tempo generosamente dedicado e por aceitar prontamente a orientação do meu trabalho. Ao Eng. Márcio Correia, expresse o meu sincero agradecimento pelo ensino de novas perspetivas e conceitos fundamentais para o meu desenvolvimento. Agradeço também pelo tempo dedicado à minha orientação. A sua orientação e experiência foram de imenso valor para o meu crescimento profissional e académico. Sou muito grato por todos os ensinamentos e pelo apoio constante ao longo deste trabalho. Ao meu amigo e colega de investigação, Thadeu Brito, gostaria de expressar a minha profunda gratidão por tudo o que você fez ao longo destes anos em que nos conhecemos. Agradeço imensamente pela orientação durante o projeto da licenciatura, pelos momentos em que compartilhamos conhecimentos durante os tempos de investigação e, acima de tudo, pela amizade e disponibilidade que sempre demonstrou.

O meu muito obrigado por cada palavra, cada aprendizagem, cada conversa, que sem vocês não seria possível a realização deste trabalho.

Um agradecimento aos meus amigos, por disponibilizarem-se a ajudarem de qualquer maneira, mesmo que não entendam nada do assunto, por todo o apoio nesta fase tão importante da minha vida académica e por me ajudarem sempre a dar o meu melhor.

A minha família que me ajudou a concretizar este meu sonho, que apoiou sempre e por todas as coisas boas que me têm proporcionado ao longo da minha vida, a eles dedico-lhes este meu trabalho.

A todas as pessoas que de algum modo contribuíram para a execução e realização deste trabalho e da minha vida académica, o meu muito e sincero Obrigado.

Resumo

No contexto da manutenção e construção de estradas, é essencial garantir a segurança dos utilizadores e manter um fluxo de tráfego eficiente. No entanto, muitas vezes é necessário fazer alterações temporárias na sinalização ao longo da via. Nesse sentido, os Painéis de Mensagens Variáveis (PMVs) desempenham um papel fundamental ao permitir a colocação rápida e fácil de sinais temporários, dispensando a necessidade de equipamentos volumosos e pesados.

A integração do *Internet of Things* (IoT) revolucionou como os objetos e dispositivos interagem, proporcionando uma conectividade contínua e uma troca de dados mais eficiente. Um método inovador e eficaz para comunicar informações com PMVs é apresentado neste contexto. Para isso utiliza o protocolo *Message Queuing Telemetry Transport* (MQTT) para transmitir imagens de forma segura e criptografado.

O método destaca-se na divisão da imagem em várias mensagens, cada uma contendo um segmento da imagem. Com a abordagem proposta é possível minimizar o impacto na largura de banda da rede e evitar problemas como perda de pacotes e latência. Além disso, são aplicadas técnicas avançadas de criptografia para garantir a confidencialidade e integridade dos dados da imagem (informação) durante a transmissão, protegendo contra o acesso não autorizado ou adulteração.

Um componente importante desse sistema é a central de controlo do PMV, que permite a configuração de mensagens e monitorização de possíveis problemas remotamente, sem depender de recursos adicionais ou assistência externa. Deste modo, o sistema proporciona uma gestão mais eficiente e flexível da sinalização temporária, adaptando-se rapidamente

a diferentes locais e necessidades.

O sistema foi validado em aplicações reais, utilizando painéis indicadores remotos com LEDs, no qual foi aplicado diversos testes de estresse na comunicação. Com a utilização de uma livraria de GIFs para testar o sistema, foram introduzidos erros propositados nos envios, permitindo avaliar a robustez e o desempenho do sistema diante de condições adversas, como perda de pacotes e atrasos na transmissão. A transmissão de imagens por MQTT demonstrou ser segura, confiável e capaz de manter a integridade do conteúdo da imagem. Dessa forma, essa metodologia mostra-se viável e efetiva para a transferência de imagens em aplicações de IoT.

Palavras-chave: Painéis de Mensagens Variáveis, Sinalização de trânsito temporários, *Internet das coisas*, Segmentação de Mensagens, Otimização de Carga de Mensagens.

Abstract

In the context of road maintenance and construction, ensuring user safety and maintaining efficient traffic flow is essential. However, temporary changes to road signage are often necessary. In this regard, Variable Message Signs (VMS) play a crucial role by allowing quick and easy placement of temporary signs without the need for bulky and heavy equipment.

The integration of the Internet of Things (IoT) has revolutionized the way objects and devices interact, providing continuous connectivity and more efficient data exchange. An innovative and effective method for transmitting relevant information through VMS is presented in this context. It utilizes the Message Queuing Telemetry Transport (MQTT) protocol to securely and encrypted transmit images.

The method stands out in its approach of dividing the image into multiple messages, each containing a segment of the image. This proposed approach minimizes the impact on network bandwidth and avoids issues such as packet loss and latency. Additionally, advanced encryption techniques are applied to ensure the confidentiality and integrity of image data during transmission, protecting against unauthorized access and tampering.

An important component of this system is the VMS control center, which enables message configuration and monitoring of potential issues remotely, without relying on additional resources or external assistance. This provides a more efficient and flexible management of temporary signage, quickly adapting to different locations and needs.

The system is validated in real-world applications using remote LED-based display panels, where various communication stress tests are applied. By using a dataset of GIFs

to test the system, intentional errors are introduced during transmission, allowing for the evaluation of system robustness and performance under adverse conditions such as packet loss and transmission delays. The transmission of images via MQTT has proven to be secure, reliable, and capable of maintaining image content integrity. Thus, this methodology proves to be feasible and effective for image transfer in IoT applications.

Keywords: Variable Message Signs, Temporary Traffic Signage, Internet of Things, Message Segmentation, Message Load Optimization.

Conteúdo

1	Introdução	1
1.1	Contextualização do tema	1
1.2	Objetivos	2
1.3	Organização e estrutura	3
2	A sinalização temporária	5
2.1	Objetivos da sinalização temporária	5
2.2	Tipos de situações temporárias	6
2.2.1	Perigos temporários	6
2.2.2	Trabalhos fixos	6
2.2.3	Trabalhos móveis	7
2.3	Interrupção das vias rodoviárias	7
2.3.1	Encerramento total	8
2.3.2	Encerramento parcial	10
2.3.3	Capacidade limitada de encerramento	11
2.3.4	Desvio de trânsito	11
2.4	Implementação	12
2.4.1	Sinalização de aproximação	13
2.4.2	Sinalização de posição	13
2.4.3	Sinalização final	14
2.5	Painel de Mensagem Variável (PMV) e Sinais de Mensagens Variáveis (VMS)	15

2.5.1	Painéis de mensagem variável (PMV)	15
2.5.2	Painéis luminosos de mensagem única (PLMU)	17
2.5.3	Sinais luminosos de afetação de vias (SLAV)	17
3	Tecnologias e aplicação	19
3.1	Problema e solução	19
3.2	Base dos conceitos	22
3.2.1	Tecnologias da Displays and Mobility Solutions (DMS)	22
3.2.2	Internet das Coisas (IoT)	23
3.3	Análise técnica da tecnologia	24
3.3.1	Message Queuing Telemetry Transport (MQTT)	24
3.3.2	Microcontroladores	25
3.3.3	Módulo de cartão MicroSD	29
3.3.4	Descrição do protótipo	30
4	Desenvolvimento	33
4.1	Estado do Upload	33
4.1.1	ImgToCluster	34
4.1.2	Cálculo do Cyclic Redundancy Check (CRC)	36
4.1.3	Divisão em partes e formatação JSON	37
4.1.4	Receção dos payload e armazenamento dos dados	38
4.2	Estado do GetCRC	40
4.3	Estado do SetState e GetState	41
5	Resultados do sistema	43
5.1	Implementação de testes	43
5.1.1	Envio de 100 <i>Graphics Interchange Format (GIF)s</i>	44
5.1.2	Introdução de erros	45
5.1.3	Implementação em painéis de teste	49
6	Conclusão	51

Lista de Tabelas

2.1	Tipos de sinalização.	14
3.1	Lista de preços e as características padrões de cada microcontroladores. . .	29

Lista de Figuras

2.1	Corte de estrada total com desvio [16].	8
2.2	Corte de estrada parcial [16].	10
2.3	Corte de estrada parcial [11].	11
2.4	Esquema de implementação de sinalização temporária [16].	12
2.5	Painel de Mensagem Variável (PMV) de tipo a1 [12].	16
2.6	PMV de tipo a2 [12].	16
2.7	PMV de tipo a3 [12].	16
2.8	Painéis PLMU [12].	17
2.9	Painéis SLAV [12].	17
3.1	Gestão do painel. No lado esquerdo é necessário a presença de um técnico. No lado direito, é utilizado o acesso ao controlo remoto.	20
3.2	Esquema de solução proposta.	21
3.3	Soluções da via pública [9].	22
3.4	Exemplos de soluções da DMS[9].	23
3.5	Arquitetura de Pub/Sub do MQTT [15].	24
3.6	Placa de desenvolvimento Espressif32 (ESP32) [7].	26
3.7	Raspberry Pi Pico (versão W) [13].	27
3.8	Arduino MKR1000 WIFI [5].	27
3.9	Lichee Nano [17].	28
3.10	Módulo de cartão MicroSD [10].	30
3.11	Protótipo.	31

3.12	Arquitetura do sistema.	32
4.1	Exemplo de <i>GIF</i> normal	36
4.2	<i>GIF</i> convertido para clusters	36
4.3	Processo de <i>Upload</i>	38
4.4	Exemplo de erro de transmissão.	39
4.5	Receção e guardar os <i>GIFs</i>	40
4.6	Graphical user interface (GUI) com resposta do CRC.	40
4.7	Diretório de CRCs da memória	41
4.8	controlo e monitorização do estado do painel.	42
5.1	Enviado com sucesso.	44
5.2	Número de mensagens enviadas, recebidas e erros encontrados.	45
5.3	Tipos de ruídos artificialmente adicionados para avaliar o sistema proposto.	46
5.4	Erro de envio no GUI	47
5.5	Erro de envio no MQTT (Payload invertido).	47
5.6	Gráfico com o número de mensagens enviadas, recebidas e erros encontrados após a adição de ruídos artificiais.	48
5.7	Sinalização de limite de velocidade	49
5.8	Sinalização de perigo de pavimento escorregadio	49

Acrónimos e Siglas

CRC Cyclic Redundancy Check.

DMS Displays and Mobility Solutions.

ESP32 Espressif32.

ESTiG Escola Superior de Tecnologia e Gestão.

GIF Graphics Interchange Format.

GUI Graphical user interface.

IoT Internet of Things.

IPB Instituto Politécnico de Bragança.

MQTT Message Queuing Telemetry Transport.

PMV Painel de Mensagem Variável.

QoS Qualidade de Serviço.

RST Regulamento de Sinalização e Trânsito.

SPI Serial Peripheral Interface.

TCP/IP Transmission Control Protocol/Internet Protocol.

VMS Sinais de Mensagens Variáveis.

Capítulo 1.

Introdução

1.1. Contextualização do tema

A sinalização temporária não é uma questão levada a sério no âmbito de políticas, leis e regulamentos portugueses relativos a essa sinalização. Parece existir um contínuo desinteresse por parte dos empresários, das entidades responsáveis pela gestão do espaço público e mesmo dos utentes da via pública e uma conseqüente desvalorização desta questão. A segurança rodoviária em Portugal é afetada por uma coordenação inadequada entre várias entidades públicas e privadas responsáveis pela intervenção no sistema rodoviário e de segurança. Conforme o Plano Nacional de Prevenção Rodoviária, este problema é identificado em Portugal, especialmente em relação à falta de manutenção do sistema de sinalização rodoviária, tanto permanentes como temporários [16].

A sinalização temporária é um assunto que recebe pouca importância, por ser temporário. Apenas observando atentamente a localização dos sinais relativos à sinalização temporária que surgem na estrada e ler os regulamentos sobre sinais de trânsito para se tornarem evidentes as discrepâncias. As semelhanças são de facto poucas e por vezes preocupantes, levando a uma colocação incorreta do que pode ser grave, tanto para os trabalhadores como para os automobilistas e outros utilizadores da via pública. Nas autoestradas, por exemplo, onde as velocidades são elevadas, a colocação destes sinais deve ser cuidadosa e para cumprir a legislação em vigor. Mas em contextos urbanos que prevalece

a facilitação, devido às baixas velocidades e à natureza das intervenções, levando a um maior incumprimento. Quando, por exemplo, não se obedece à distância mínima exigida entre os sinais para uma determinada velocidade, a tarefa da condução não é feita em segurança, pois o condutor não tem capacidade para visualizar e assimilar a mensagem transmitida pelos sinais[1].

1.2. Objetivos

Neste trabalho, pretende-se realizar um trabalho colaborativo com uma empresa para desenvolvimento de uma solução real. A DMS, empresa líder no fornecimento de soluções de controlo de trânsito, desenvolve, entre outros produtos, PMV ou Sinais de Mensagens Variáveis (VMS). A manutenção ou construção de novas vias de tráfego, acarreta, muitas vezes, alterações da sinalização das vias existentes e/ou a introdução de sinalização temporária durante o período de construção/manutenção. A utilização de PMVs tradicionais nestes cenários é condicionada devido a várias dificuldades, em especial, às dificuldades físicas de movimentar e instalar equipamentos volumosos e pesados num curto espaço temporal.

Neste contexto, a DMS identificou uma necessidade relacionada com os PMVs móveis para a uma utilização temporária. O produto, detalhadamente na Secção **3.3.4**, em desenvolvimento, caracteriza-se por ser uma solução autónoma em termos energéticos, contando com fonte de alimentação capaz de alimentar todo o sistema (painel de LEDs, comunicações 4G, microcontrolador, etc). Dada a característica temporária da utilização destes dispositivos, a DMS pretende que o controlo seja feito por uma solução Web centralizada, onde os utilizadores possam configurar mensagens, estabelecer planos de afixação e verificar eventuais alarmes de uma forma simples, sem necessidade de equipamentos ou serviços adicionais de terceiros.

1.3. Organização e estrutura

O presente trabalho consiste num relatório dividido em seis capítulos.

No primeiro capítulo, é feita uma introdução ao tema, contextualizando-o e apresentando os objetivos do trabalho, além de descrever a organização e estrutura do relatório.

O segundo capítulo é dedicado à sinalização temporária, abordando os seus objetivos, os diferentes tipos de situações temporárias e as medidas adotadas para interromper o tráfego rodoviário. Também é discutida a implementação da sinalização, incluindo o uso de Painéis de Mensagem Variável (PMV) e Sinais de Mensagens Variáveis (VMS).

No terceiro capítulo, são exploradas as tecnologias e as suas aplicações relacionadas ao tema. São discutidos problemas e soluções, além de conceitos fundamentais como Internet das Coisas (IoT) ou tecnologias desenvolvidas pela Displays and Mobility Solutions (DMS).

O quarto capítulo apresenta o desenvolvimento do sistema, abordando diferentes estados, como o estado do `Upload`, o estado do `GetCRC` e o estado do `SetState` e `GetState`.

No quinto capítulo, são apresentados os resultados obtidos a partir da implementação do sistema, incluindo testes de envio de *GIFs* e introdução de erros.

No sexto e último capítulo, é apresentada a conclusão do estudo sobre sinalização temporária. Neste capítulo, são resumidas as principais descobertas e conclusões obtidas ao longo do trabalho. Além disso, são mencionadas algumas sugestões para trabalhos futuros, como a implementação de outras redes de comunicação e a exploração de diferentes microcontroladores.

1. Introdução

Capítulo 2.

A sinalização temporária

As sinalizações temporárias consistem num tipo específico de sinalização ideal para alertar e proteger os utentes da via pública quando existem obstáculos como acidentes ou trabalhos de conservação e manutenção. Da mesma forma, esta sinalização deve comunicar aos utilizadores da estrada situações de restrição ou proibição, essenciais para cada situação temporária.

2.1. Objetivos da sinalização temporária

A sinalização temporária pretende salvaguardar a segurança dos utentes e dos trabalhadores quando existem situações anómalas na estrada. No sentido de manter o fluxo de tráfego com a menor interferência possível, a sinalização temporária deve alertar os condutores sobre a existência de obstáculos, incentivando-os a adotar um comportamento adequado, conduzi-los pela área afetada e informando-os sobre o final da anomalia.

Assim, para segurança dos diversos utentes da via, a sinalização temporária deve contribuir para:

- *Advertir* os utilizadores da via com antecedência da existência de obras, ou obstáculos;

2. A sinalização temporária

- *Regularizar* a circulação do trânsito, a velocidade e outras condições para a segurando do mesmo local;
- *Canalizar* o fluxo de veículos e organizá-los para evitar movimentos possivelmente conflituosos;
- *Reduzir* o risco de acidentes e congestionamentos.
- *Transmitir* informações claras aos condutores;
- *Delimitar* a área de obras na via pública, não só protegendo os condutores, como também os trabalhadores.

2.2. Tipos de situações temporárias

Existem diferentes tipos de anomalias que podem surgir na via pública e que podem ser classificadas nos seguintes grupos: perigos temporários, trabalhos fixos e trabalhos móveis [8].

2.2.1. Perigos temporários

Os perigos temporários são considerados condicionamentos do trânsito provocados por acidentes, objetos caídos na via pública ou outras situações de caráter inesperado. Como o próprio nome indica, são perigos temporários que, à partida, não apresentam elevada demora da remoção da anomalia que provoca o perigo em questão.

2.2.2. Trabalhos fixos

Consideram-se trabalhos fixos os que condicionam o trânsito devido a trabalhos na estrada que são planeados devidamente antes da sua execução. Vários exemplos desse tipo, são trabalhos de infraestruturas, por exemplo, reparação de pavimento, abastecimento de água ou gás, ou até saneamento.

2.2.3. Trabalhos móveis

Os trabalhos móveis são trabalhos de reparação ou manutenção constituídos por um avanço contínuo nas zonas de trabalho, por exemplo, marcações de linhas rodoviárias ou observações de pavimentos. São trabalhos em constante movimento, mas possuem uma velocidade de progressão lenta o que os torna um perigo para a circulação sem devida informação.

2.3. Interrupção das vias rodoviárias

Para existir uma interrupção ou condicionamento das vias rodoviárias é necessário existir inconvenientes que impedem o tráfego normal, como referidos na Secção **2.2**. O encerramento total de uma estrada implica uma interrupção do trânsito que só deve ser utilizada em último caso. Geralmente, é uma ação de curta duração, pois induz sérias perturbações no trânsito. Existem algumas estratégias para lidar com o condicionamento da circulação da via rodoviária, tais como:

- O *encerramento total*, que pode ser do início ao fim do tempo de obras, como pode ser, por exemplo, apenas durante a noite ou fins de semana;
- *Encerramento parcial*;
- *Capacidade limitada de encerramento*;
- *Desvio de Trânsito*.

2.3.1. Encerramento total

O encerramento total, como o nome indica, consiste em bloquear completamente a via pública de modo a não existir passagem de automóveis para exercer um trabalho de manutenção ou reparo da mesma. Com este tipo de encerramento deve existir um desvio associado que contende a mobilidade necessária, para que os automobilistas que pretendem passar por essa zona tenham um local de circulação, como demonstrado na Figura 2.1 [16].

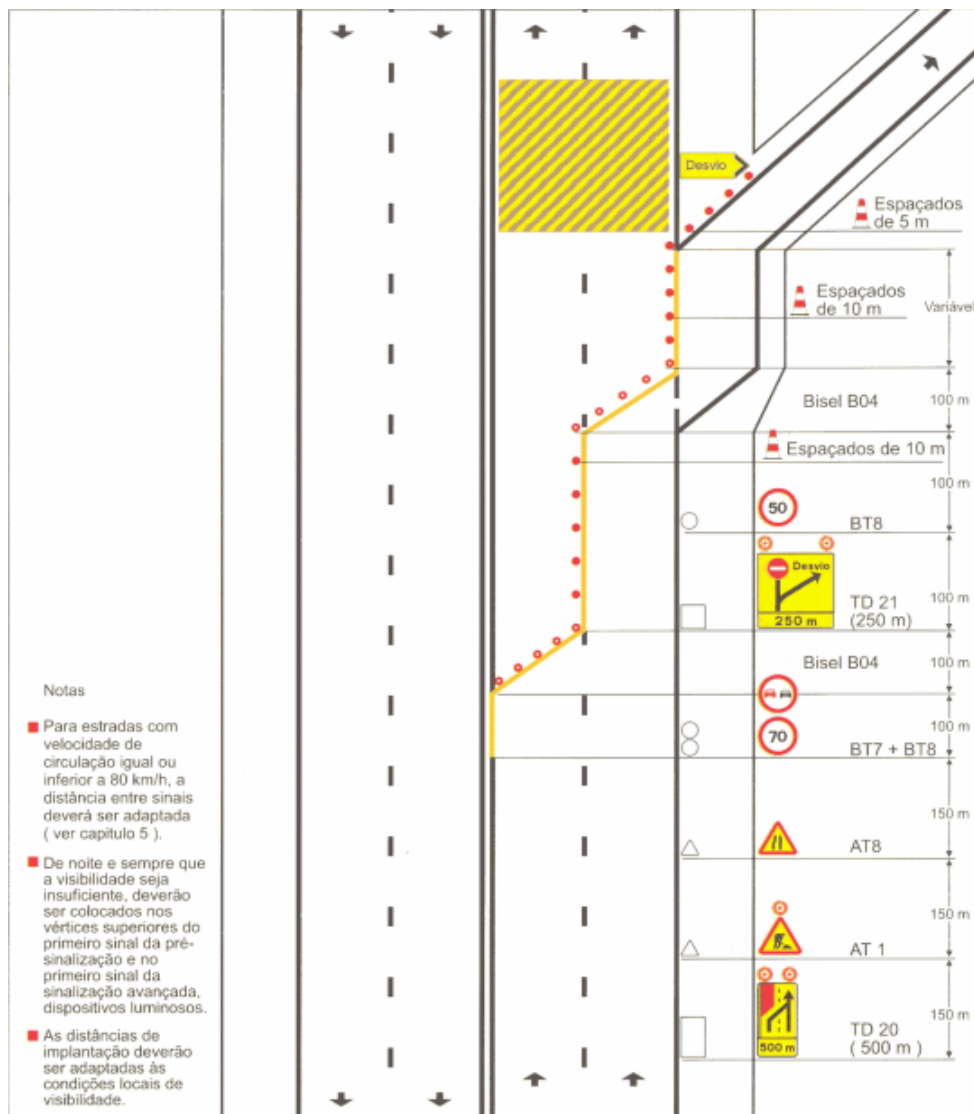


Figura 2.1: Corte de estrada total com desvio [16].

Na ocorrência deste tipo de interrupção da via, os peões podem, ou não, circular na mesma, conforme as condições de segurança oferecidas pelos empreiteiros encarregues da obra. Ao realizarem-se este tipo de encerramentos completos, reduz-se a taxa de acidente e aumenta a segurança de quem trabalha na obra e dos próprios condutores. No encerramento total existem alguns tipos de interrupções:

- **Encerramento a tempo inteiro** - Encerramento que consiste em bloquear a totalidade da via pública todos os dias da semana e até durante o período noturno;
- **Encerramento ao fim de semana** - Este encerramento é realizado apenas durante os períodos de fins de semana, sendo um encerramento total da via como no anterior. Normalmente, é utilizado este tipo de interrupção, em estradas onde que durante o fim de semana o volume de tráfego é muito menor que durante a semana. Frequentemente, estes trabalhos começam já na sexta-feira à noite, depois do período de ponta e continuam até segunda-feira de manhã, terminando antes da hora de ponta matinal;
- **Encerramento noturno** - A interrupção durante o período noturno é uma solução bastante comum, mas exigem focos luminosos devido à falta de luminosidade por se realizar no período noturno. Para além de exigir que se tenha mais cuidados devido à reduzida visibilidade, também causa poluição sonora. A operação de máquinas e veículos, durante o período noturno, é uma desvantagem para os moradores desta áreas, [1].

2.3.2. Encerramento parcial

O encerramento parcial é usado em trabalhos na via pública sempre que não é necessário cortar a estrada por completo. A via disponível, deve manter uma largura de 2,90 m para que, pelo menos, seja possível transitar um veículo de cada vez não bloqueando o trânsito ou criando desvios alternativos, exemplo na Figura 2.2.

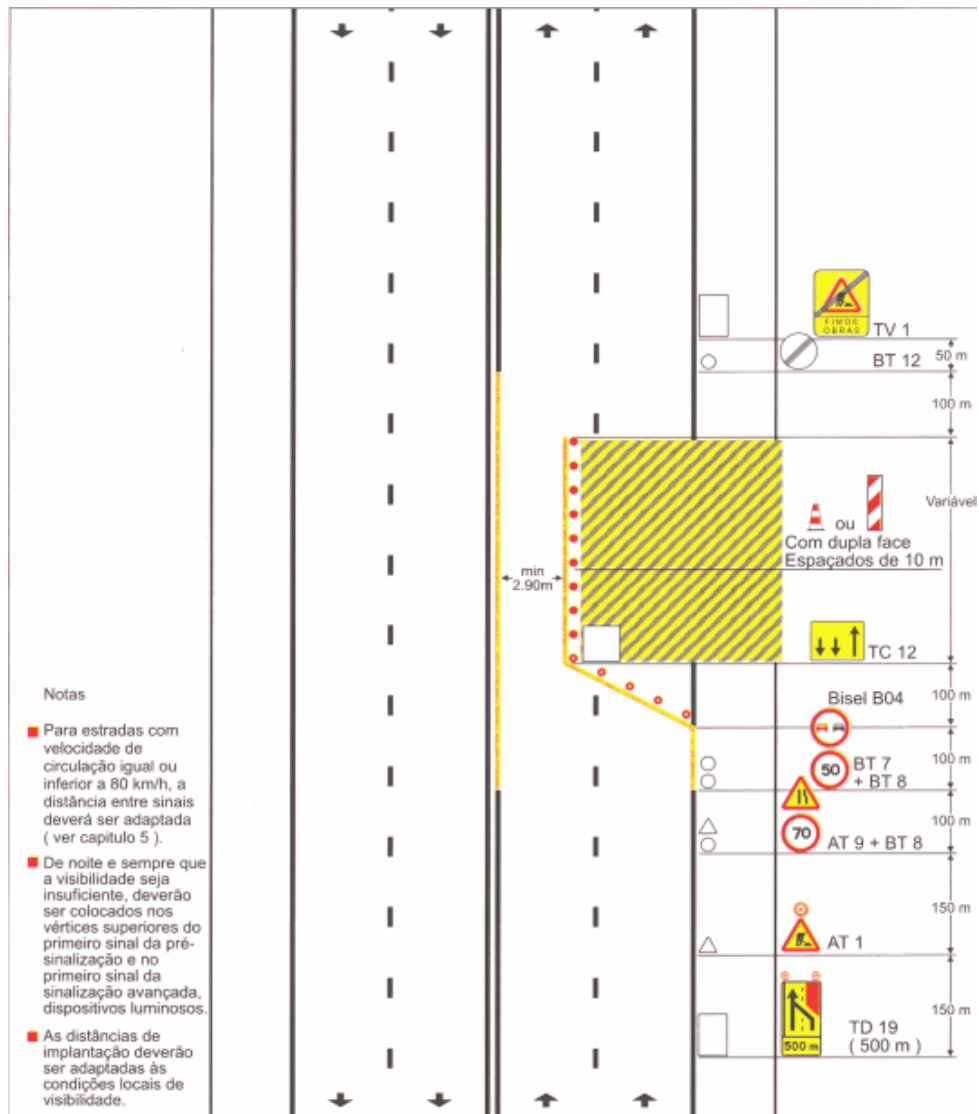


Figura 2.2: Corte de estrada parcial [16].

2.3.3. Capacidade limitada de encerramento

Existem algumas limitações neste tipo de interrupção de trânsito. Sendo este apenas um corte parcial da via pública, pode criar limitações no tipo de veículos que circulam nas áreas afetadas, possivelmente afetando veículos pesados, o que origina alterações a sua rota.

2.3.4. Desvio de trânsito

O desvio de trânsito consiste num percurso formado por, uma ou várias estradas, que, no conjunto, evitam a via interdita ao trânsito, devendo ser devidamente sinalizado. Exemplo de sinalização de desvio de trânsito na Figura 2.3.



Figura 2.3: Corte de estrada parcial [11].

2.4. Implementação

Considera-se que é uma zona regulada por sinalização temporária, uma zona com uma interrupção parcial das faixas de rodagem, uma implementação da sinalização ficaria compreendida entre 3 tipos: sinalização de aproximação, sinalização de posição e sinalização final, exemplo na Figura 2.4.

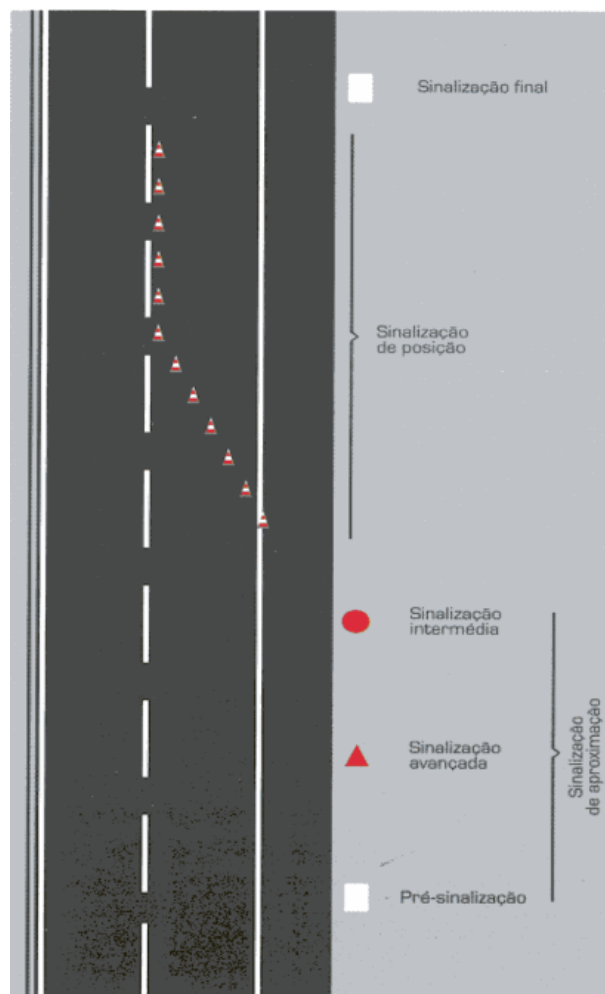


Figura 2.4: Esquema de implementação de sinalização temporária [16].

2.4.1. Sinalização de aproximação

A sinalização de aproximação é colocada sempre que existam obras ou obstáculos na via pública. Deste tipo de sinalização fazem parte a pré-sinalização, a sinalização avançada e a intermédia, considera-se que:

- **Pré-sinalização** - A pré-sinalização utiliza-se sempre que haja necessidade de fazer algum desvio ou mudança na via de circulação. O objetivo desta é alertar o condutor com a devida antecedência, de uma zona de perigo motivando a alteração do seu comportamento normal (ST7 e ST8a);
- **Sinalização avançada** - O objetivo da sinalização avançada e da intermédia é obrigar os condutores a redobrar a sua atenção e cautela, reduzir gradualmente a velocidade, evitar acidentes e permitir maior fluxo de tráfego em áreas restritas. A sinalização avançada é feita com recurso a sinais de perigo, é utilizada em casos de obras ou obstáculos ocasionais que, por natureza não implicam o condicionamento de trânsito e podem ser identificados por meio de sinalização de posição (A23 e A29);
- **Sinalização intermédia** - Após a sinalização avançada, existe a sinalização intermédia que usa sinais de proibição ou cedência de passagem. É estabelecido que os sinais intermédios devem ser utilizados sempre que as condições das estradas ou a natureza das obras e obstáculos exijam o uso de limites de velocidade, não ultrapassagem ou outras proibições (C13 e C14a);

2.4.2. Sinalização de posição






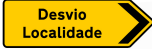
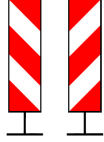



A sinalização de posição deve ser utilizada sempre que existam obras ocasionais ou obstruções na via. Isso garante a proteção de áreas restritas, por exemplo, devido aos trabalhos, acidentes, necessidade de assistência ou obstáculos na estrada sejam separados da via onde é permitido circular, permitindo a segurança dos trabalhadores e a facilidade de acesso às viaturas de socorro e assistência (ET5, ET6, ET10).

2. A sinalização temporária

2.4.3. Sinalização final

A sinalização final informa a condutores que a zona restrita acabou e eles devem aceder às condições normais de trânsito. O uso desta sinalização deve ser feita através dos sinais de proibição anteriormente impostos e do sinal de “Fim dos Trabalhos/obras” (ST14) [16].

Tabela 2.1: Tipos de sinalização.

A23  [18]	A29  [19]	ST7  [24]	C13  [20]	C14a  [21]
ST8a  [26]	ET5  [23]	ET6  [14]	ET10  [22]	ST14  [25]

2.5. Painel de Mensagem Variável (PMV) e Sinais de Mensagens Variáveis (VMS)

Os PMVs são uma forma avançada de tecnologia utilizada para exibir mensagens e informações em sinalizações e painéis. Compostos por uma matriz de luzes, geralmente LEDs, possuem a capacidade de exibir diferentes padrões e mensagens de forma dinâmica. Essa característica permite que os PMVs sejam amplamente utilizadas em diversos ambientes, como estradas, rodovias, estações de comboio, aeroportos e outros locais.

A principal vantagem dos PMVs reside na sua capacidade de transmitir informações de maneira ágil e adaptável. Graças à sua matriz de LEDs controláveis, é possível exibir mensagens personalizadas e ajustá-las instantaneamente para refletir dados e condições em constante mudança. Isso possibilita que os utilizadores sejam informados sobre o tráfego, acidentes, condições climáticas, horários de chegada e partida, entre outros [29].

Os PMVs e os VMS são termos que se referem ao mesmo tipo de tecnologia. Ambos são usados para exibir mensagens e informações dinâmicas em sinalizações e painéis. Compostos por uma matriz de luzes, geralmente LEDs, eles possuem a capacidade de exibir diferentes padrões e mensagens em tempo real.

Nas seguintes subsecções apresentam-se alguns tipos de painéis.

2.5.1. Painéis de mensagem variável (PMV)

Os PMV são dispositivos eletrônicos utilizados para exibir informações importantes e atualizadas, em estradas e autoestradas. Existem diferentes configurações de PMV, incluindo aquelas com uma ou duas áreas de apresentação de mensagens simbólicas, como sinais ou símbolos, e uma área separada para apresentação de mensagens de texto. Normalmente, a área de mensagens de texto possui três linhas com capacidade para 18 caracteres por linha, e as mensagens são centralizadas.

Também existem PMV que possuem uma única área que pode exibir sinais, símbolos e mensagens de texto. Estes tipos de PMV são conhecidos como painéis de matriz única

2. A sinalização temporária

e são classificados como PMV dos tipos **a1** a **a3**.

O tipo **a1** de PMV possui uma configuração que inclui uma área de apresentação de sinais e uma área de apresentação de mensagens de texto. Isso é ilustrado na Figura 2.5.

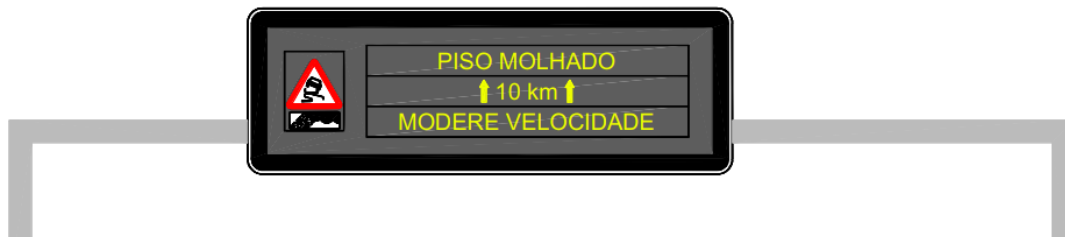


Figura 2.5: PMV de tipo **a1** [12].

O tipo **a2** de PMV possui duas áreas de apresentação de sinais e uma área de apresentação de mensagens de texto, conforme mostrado na Figura 2.6.



Figura 2.6: PMV de tipo **a2** [12].

O tipo **a3** de PMV possui uma única área de apresentação que pode exibir tanto sinais visuais como mensagens de texto. Esses PMVs, conhecidos como painéis de matriz única, são ilustrados na Figura 2.7.



Figura 2.7: PMV de tipo **a3** [12].

2.5.2. Painéis luminosos de mensagem única (PLMU)

Os PLMU possuem uma única área destinada à exibição de um único sinal ou, em alguns casos, um sinal acompanhado de um painel adicional.

Esses painéis são ilustrados na Figura 2.8. Os sinais exibidos pelos PLMU têm uma colocação lateral e devem ser repetidos no lado esquerdo. Com esta disposição garante-se que os condutores recebem as informações necessárias de forma clara e consistente, contribuindo para a segurança e orientação adequada nas vias.



Figura 2.8: Painéis PLMU [12].

2.5.3. Sinais luminosos de afetação de vias (SLAV)

Os SLAV são compostos por sinais luminosos instalados acima de cada faixa de tráfego. Esses sinais podem exibir três cores diferentes: luz verde (seta vertical), luz amarela (seta inclinada a 45°) e luz vermelha (cruz vermelha). Esses sinais são representados na Figura 2.9. A luz verde indica permissão para seguir, a luz amarela intermitente alerta para precaução e a luz vermelha significa proibição de passagem.



Figura 2.9: Painéis SLAV [12].

2. A sinalização temporária

Capítulo 3.

Tecnologias e aplicação

3.1. Problema e solução

No contexto atual, garantir um controle de tráfego eficiente nas vias públicas é crucial para a segurança dos condutores e peões. Em situações como obras, acidentes ou eventos temporários, uma sinalização apropriada desempenha um papel fundamental ao orientar os condutores. Os painéis informativos são inestimáveis para fornecer informações sobre as condições do tráfego, eventos especiais e alertas críticos aos condutores. Contudo, a tarefa de atualizá-los pode apresentar desafios que exigem soluções eficientes e práticas. Um dos desafios enfrentados é enviar um técnico ao local do painel para atualizar os alertas (como no lado esquerdo da Figura 3.1).

Dependendo da localização geográfica do painel e das condições climáticas, enviar um técnico pode ser um processo caro e demorado. Além disso, a logística de transporte e a segurança do técnico podem-se tornar preocupações adicionais quando o painel está localizado em áreas remotas ou de difícil acesso.

Uma solução para minimizar esse problema é adotar tecnologias de comunicação remota (lado direito da Figura 3.1). Em vez de se depender da presença física de um técnico, sistemas de controle remoto podem ser usados para atualizar e monitorizar os painéis. Por meio de *Internet of Things (IoT)*, é possível enviar comandos de atualização para os painéis digitais de forma rápida e eficiente. Nesse contexto, algumas abordagens utilizam

3. Tecnologias e aplicação

sistemas baseados em comunicações dedicadas, nos quais, é comum encontrarem-se alguns projetos que utilizam redes 4G.

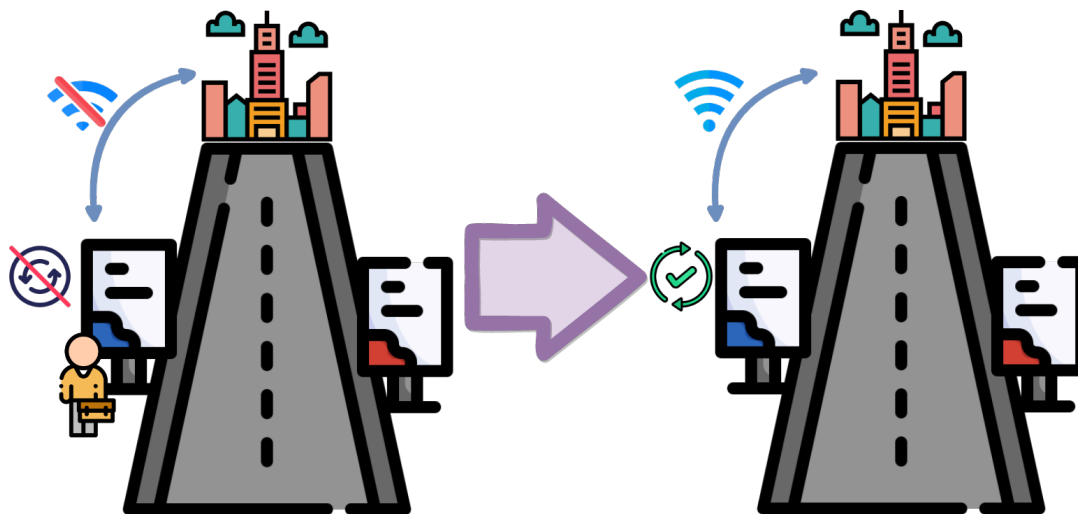


Figura 3.1: Gestão do painel. No lado esquerdo é necessário a presença de um técnico. No lado direito, é utilizado o acesso ao controle remoto.

Neste âmbito, é necessário desenvolver uma solução que utilize um painel informativo controlado por um ESP32 para receber informações via MQTT. Essa solução permite um controle flexível e em tempo real das informações exibidas no painel, tornando-o, assim, numa base de controlo de tráfego temporário eficaz.

A solução proposta é baseada num sistema de sinalização temporária composto por um painel informativo de LED controlado por um microcontrolador ESP32. A utilização do protocolo MQTT permite uma comunicação leve e eficiente entre o painel e o servidor responsável por enviar os comandos (Figura 3.2).

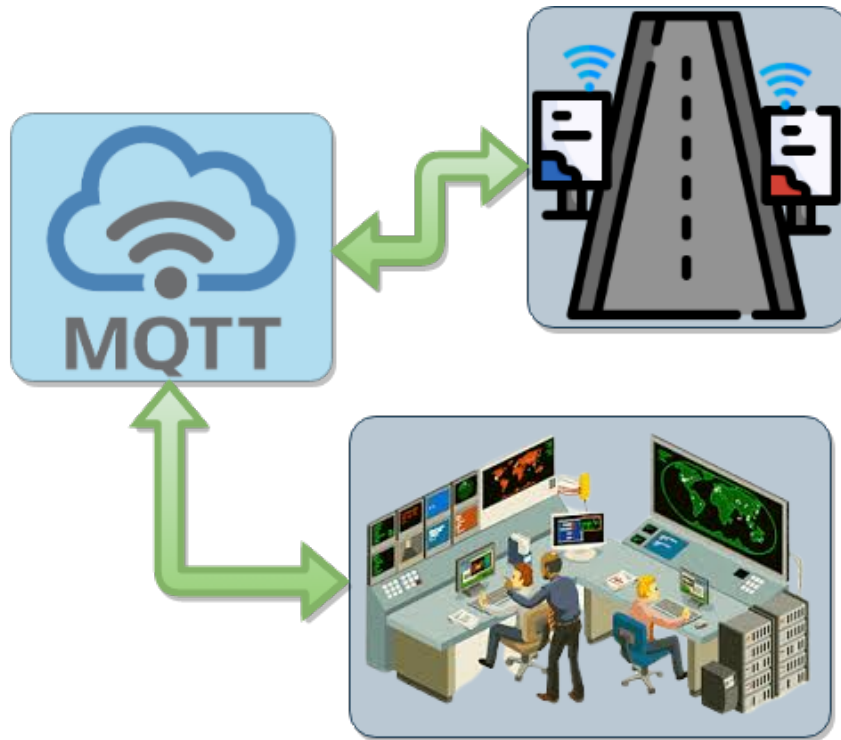


Figura 3.2: Esquema de solução proposta.

O painel informativo possui uma matriz de LEDs que permite a exibição de mensagens claras e visíveis aos condutores. Através do ESP32, o painel é capaz de se conectar ao broker de MQTT e receber payloads contendo informações específicas sobre o controle do tráfego necessário num determinado momento. Além das mensagens de texto, essa solução também permite o envio de *GIFs* (imagens) para serem exibidos aos condutores, tornando a sinalização ainda mais impactante e visualmente atrativa.

Com a capacidade de exibir *GIFs* animados, o painel informativo proporciona uma forma dinâmica de transmitir informações aos condutores. Por exemplo, em casos de obras, ao invés de apenas exibir a mensagem “Obra em Andamento”, é possível exibir um *GIF* mais interativo e visualmente agradável aos condutores, pois transmite mais informação e cativa a sua atenção de forma mais efetiva, auxiliando na compreensão e respeito às sinalizações temporárias para o aviso das obras ou outro tipo de interrupção da via pública.

3.2. Base dos conceitos

3.2.1. Tecnologias da DMS

A DMS é uma empresa dedicada ao desenvolvimento de soluções, entre outras, para o controlo do tráfego rodoviário. A empresa é especializada em fornecer tecnologias avançadas de sinalização e comunicação visual para melhorar a eficiência e a segurança nas vias públicas. No caso de tecnologia orientada a vias públicas e trânsito, podemos encontrar soluções de VMS fixos e móveis, soluções de infraestruturas de túneis e pontes, portagens, etc (Figura 3.3).

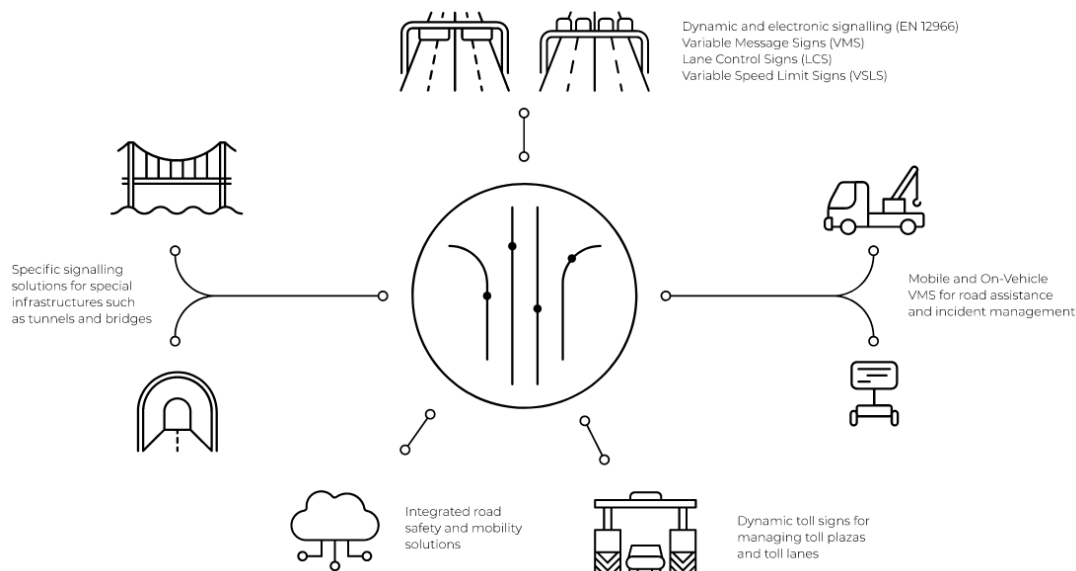


Figura 3.3: Soluções da via pública [9].

Os displays são capazes de exibir informações vitais, como mensagens de trânsito, alertas de segurança, orientações e outras informações relevantes para os condutores, sendo projetados para garantir uma visibilidade clara e eficaz, mesmo em condições de iluminação adversas ou alta velocidade.

Existem diferentes tipos de displays, tais como, displays de matriz de pontos, VMS, displays de seta direcional, displays de velocidade e muito mais. Cada um desses produtos é projetado de forma a desempenhar funções específicas, visando a melhoria da segurança

e a fluidez do tráfego nas estradas [9] (Figura 3.4).



Figura 3.4: Exemplos de soluções da DMS[9].

3.2.2. Internet das Coisas (IoT)

Kevin Ashton, pesquisador britânico do Massachusetts Institute of Technology, é considerado o primeiro especialista a usar o termo Internet das Coisas em 1999. De acordo com Ashton, o termo IoT é baseado na ideia que é possível testemunhar o momento em que duas redes separadas, ou seja, a rede de comunicação humana e o mundo das coisas, se encontram. Um ponto onde, não seja possível apenas utilizar um computador, mas onde o próprio “computador se use” de forma autónoma. As “coisas” estão cada vez mais e mais conectadas, de forma inteligente, e não apenas para “sentir” o mundo ao seu redor,

também para interagir com ele [6].

Sabemos que neste século, existe alguma variedade de tecnologias de comunicação IoT. Muitas delas continuam em desenvolvimento e em aperfeiçoamento. Para este caso, o uso da comunicação MQTT é necessário, pois é uma comunicação simples de pequeno processamento e é ideal para o desenvolvimento da tecnologia.

3.3. Análise técnica da tecnologia

3.3.1. MQTT

A comunicação MQTT é um protocolo de mensagens leve para sensores e pequenos dispositivos móveis, otimizado para redes *Transmission Control Protocol/Internet Protocol (TCP/IP)*. O esquema de troca de mensagens é baseado no modelo Publicador-Subscritor, que é extremamente simples e portátil. Essa arquitetura, exemplificada na Figura 3.5, torna o MQTT altamente escalável e eficiente, permitindo a comunicação assíncrona entre vários dispositivos de forma distribuída.

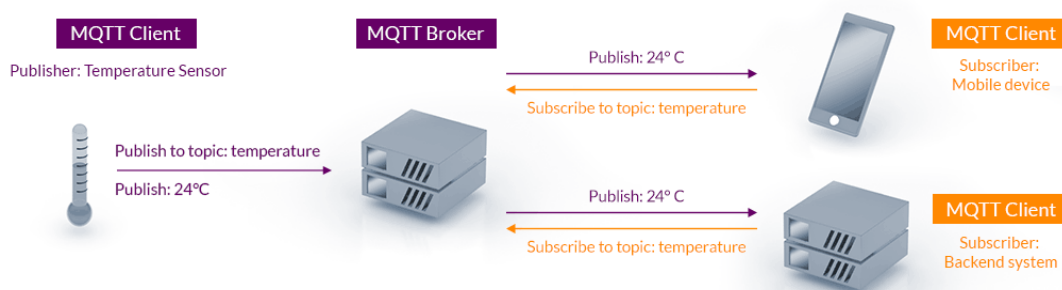


Figura 3.5: Arquitetura de Pub/Sub do MQTT [15].

Algumas características importantes do MQTT incluem:

- Leve e eficiente: O MQTT é projetado para ser eficiente em termos de recursos, tornando-o adequado para dispositivos com capacidades limitadas de processamento, memória e largura de banda;

- Baixo consumo de energia: O protocolo foi otimizado para minimizar o consumo de energia, o que é especialmente importante em dispositivos alimentados por bateria;
- Confiabilidade: O MQTT suporta mecanismos de Qualidade de Serviço (QoS) que garantem a entrega das mensagens, mesmo em redes instáveis. Existem três níveis de QoS: QoS 0 (entrega no máximo uma vez), QoS 1 (entrega pelo menos uma vez) e QoS 2 (entrega exatamente uma vez);
- Conectividade: O MQTT permite a comunicação bidirecional, permitindo que os dispositivos enviem e recebam mensagens de forma eficiente;
- Suporte a tópicos: Os dispositivos podem publicar mensagens em tópicos e inscrever-se em tópicos específicos para receber mensagens relevantes. Isso permite uma arquitetura flexível e escalável para a troca de informações.

O MQTT é usado em cenários de IoT, onde há a necessidade de comunicação eficiente e confiável entre dispositivos distribuídos. É implementado em plataformas de IoT e é suportado por uma variedade de linguagens de programação e bibliotecas [15].

3.3.2. Microcontroladores

ESP32

O ESP32 (Figura 3.6) é um microcontrolador de baixo custo e baixo consumo de energia desenvolvido pela Espressif Systems. É muito utilizado no campo do IoT devido às suas capacidades de conectividade sem fio, alto desempenho e recursos avançados. Uma das principais características do ESP32 é a sua conectividade sem fios. Suporta redes Wi-Fi, o que permite a comunicação com outros dispositivos conectados à rede local ou à Internet. Além disso, possui suporte nativo para Bluetooth, tornando-o adequado para aplicações que exigem comunicação sem fio de curto alcance, como controle remoto, monitorização de sensores e comunicação com dispositivos móveis [27]. De forma geral, o ESP32 destaca-se pela combinação de recursos, desempenho e custo acessível, tornando-se a opção preferida para o projeto.



Figura 3.6: Placa de desenvolvimento ESP32 [7].

O ESP32 foi selecionado como prova de conceito para demonstrar que é possível implementar o sistema com um microcontrolador de baixo custo. Contudo, existem outras opções que podem ser exploradas para projetos que exigem recursos adicionais ou em que seja possível investir um pouco mais em *hardware*.

Aqui estão algumas outras opções que podem ser consideradas para o desenvolvimento do sistema:

- Raspberry Pi Pico (versão W)
- Arduino MKR1000
- Sipeed Lichee Nano

Raspberry Pi Pico (versão W)

O Raspberry Pi Pico (Figura 3.7) é uma placa de desenvolvimento de baixo custo da família Raspberry Pi. Embora seja mais conhecido como um microcontrolador de propósito geral, ele também pode ser utilizado para projetos de IoT. A versão W do Pico possui conectividade Wi-Fi, o que o torna uma opção interessante para este tipo de sistemas.

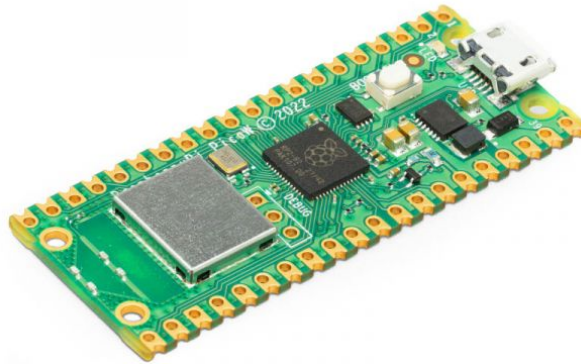


Figura 3.7: Raspberry Pi Pico (versão W) [13].

Arduino MKR1000

O Arduino MKR1000 (Figura 3.8) é um microcontrolador baseado na plataforma Arduino, combina um microcontrolador SAMD21 com conectividade Wi-Fi. Embora tenha um custo um mais elevado em comparação com o ESP32, é uma opção popular para projetos de IoT que exigem integração com a plataforma Arduino.



Figura 3.8: Arduino MKR1000 WIFI [5].

Sipeed Lichee Nano

O Sipeed Lichee Nano (Figura 3.9) é um microcontrolador compacto e de baixo custo com um módulo Wi-Fi integrado. Embora não seja tão conhecido como outros microcontroladores mencionados, pode ser uma opção a ser considerada para projetos com restrições de tamanho e orçamento.

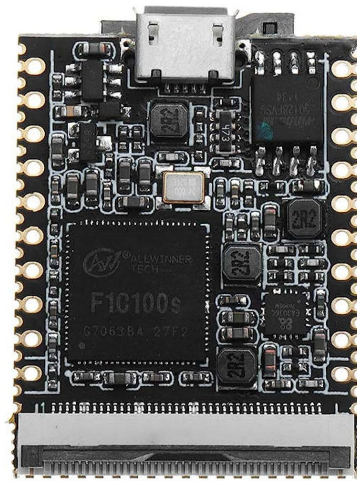


Figura 3.9: Lichee Nano [17].

Comparação entre microcontroladores

Com base nas informações fornecidas, na Tabela 3.1, e que todos os dispositivos possuem conexão Wi-Fi, capacidade de se conectar a um broker MQTT para compartilhar dados e também têm capacidade de se conectar ao painel de LEDs. Isso significa que todos os dispositivos envolvidos na aplicação possuem os recursos necessários para implementar a solução proposta.

Tabela 3.1: Lista de preços e as características padrões de cada microcontroladores.

Microcontroladores	CPU	RAM	Flash	Preço *
ESP32	240 MHz	320 KB	4 MB	2.48 €
Arduino MKR1000 WiFi	48 MHz	32 KB	256 KB	41.14 €
Raspberry Pi Pico (W version)	133 MHz	264 KB	2 MB	7.49 €
Sipeed Lichee Nano (WiFi module)	900 MHz	256 KB	16 MB	10.70 €

* Os custos são baseados em cada um dos websites oficiais de cada um dos microcontroladores.

3.3.3. Módulo de cartão MicroSD

O módulo de cartão micro SD (Figura 3.10) possui características que o tornam fácil de usar com microcontroladores, como a versatilidade de montagem e programação. Também os cartões micro SD padrão operam com uma voltagem de 3,3 V, o que significa que não é possível conectá-los diretamente a circuitos com lógica de 5V, uma vez que qualquer tensão acima de 3,6V pode danificar permanentemente o cartão. No entanto, os módulos de micro SD já vêm preparados para alimentação de 5V, pois possuem reguladores de tensão embutidos [10]. No contexto deste projeto, o cartão SD é extremamente útil, pois o ESP32 não possui uma memória volátil capaz de armazenar um *GIF* completo. Portanto, podemos utilizar o cartão SD para gravar *GIFs* na sua memória e, posteriormente, realizar a leitura do mesmo. Isso permite que o ESP32 manipule o *GIF* de forma eficiente e realize as operações necessárias.

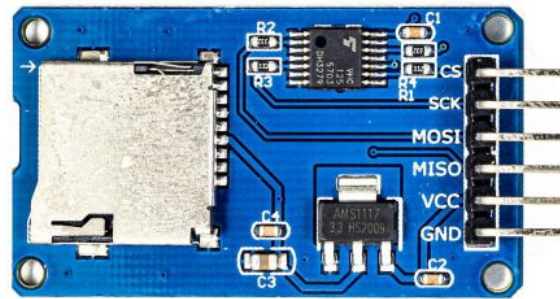


Photo by ElectroPeak

Figura 3.10: Módulo de cartão MicroSD [10].

3.3.4. Descrição do protótipo

O protótipo é baseado num microcontrolador ESP32, que desempenha um papel central na operação do painel. O ESP32 foi escolhido como o microcontrolador principal devido à sua conectividade, incluindo suporte a WiFi, e o seu poder computacional, pois possui capacidade de processamento suficiente para lidar com as necessidades de processamento e controle da matriz de LEDs. Também é complementado por um cartão SD para armazenamento de dados, pois a sua memória volátil não é suficiente para gerir um *GIF* de uma só vez. A matriz de LEDs constitui o elemento principal do painel, sendo responsável pela apresentação clara e impactante das informações. No entanto, de forma a garantir o funcionamento e o brilho adequado dos LEDs, uma fonte de alimentação externa de 5V é utilizada. Tal deve-se ao facto do ESP32 não possui capacidade para fornecer energia suficiente a todos os LEDs simultaneamente, mas tendo a comunicação feita por *Serial Peripheral Interface (SPI)* para o controlo do painel.

Este painel possui uma configuração de LEDs diferente da configuração convencional. Normalmente, o mapeamento dos LEDs num display é feito por linhas, o que significa que as imagens são carregadas para o display linha por linha, de cima para baixo ou vice-versa. No entanto, nesse caso, o display é mapeado em clusters de 4x4 de forma a reduzir a impedância e facilitar o desempenho dos PCB.

Essa abordagem de mapeamento em clusters de 4x4 LEDs ajuda a melhorar a eficiência e o desempenho do display. Ao agrupar vários LEDs em clusters, a corrente elétrica pode ser distribuída de forma mais eficiente, reduzindo a impedância e melhorando a qualidade da iluminação dos LEDs. Isso também pode permitir um controle mais preciso dos LEDs e uma melhor uniformidade de brilho em todo o display.

Na Figura 3.11 está apresentado o protótipo do projeto.

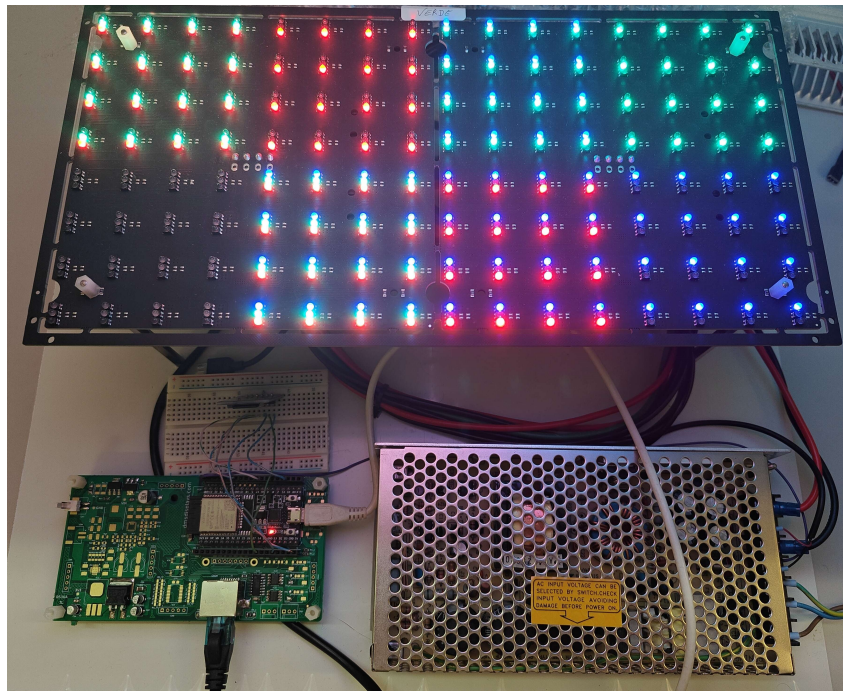


Figura 3.11: Protótipo.

O funcionamento do protótipo é baseado no envio ou controle dos *GIFs* carregados para o painel. A informação é enviada por MQTT para o microcontrolador, onde ocorre o Upload do *GIF* caso ele não esteja presente na memória. Além disso, pode ser enviado um comando para indicar qual *GIF* que deve ser utilizado da memória.

3. Tecnologias e aplicação

Entre o microcontrolador ESP32 e a memória do cartão SD, ocorre a leitura e escrita de arquivos, utilizados para exibir os *GIFs* no painel. Isso permite que os *GIFs* armazenados no cartão SD sejam reproduzidos no painel de forma controlada pelo microcontrolador. Esta arquitetura está representada na Figura 3.12.

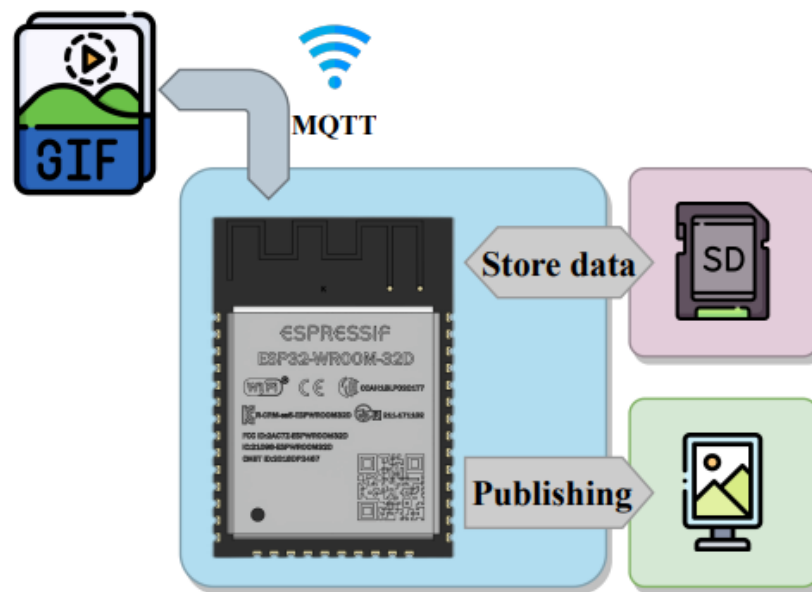


Figura 3.12: Arquitetura do sistema.

Capítulo 4.

Desenvolvimento

O projeto está dividido em diferentes partes, cada uma com uma funcionalidade específica. O controlo remoto dos painéis é abrangente e inclui o envio de *GIFs*, o estado de **Upload** é responsável pelo envio dos arquivos que não estão na memória SD do ESP32.

Também é realizada a comparação dos arquivos presentes, no cartão SD e os da base de dados, esse é o estado do *GetCRC*, cuja função é confirmar se o arquivo na memória é o correto. Existem dois estados semelhantes: o estado do **GetState**, que confirma qual *GIF* está a ser exibido no painel, e o estado do **SetState**, responsável por definir qual o *GIF* a ser exibido no painel. Posteriormente, foram desenvolvidos testes de envio e deteção de erros, os quais serão explicados na implementação dos testes.

4.1. Estado do Upload

O **Upload** é uma das fases mais complexas do sistema, envolvendo uma série de etapas para ser concluído com sucesso. Durante esse estado, são executados diversos processos que serão explorados detalhadamente ao longo desta Secção.

Dentre eles, destaca-se a conversão dos *GIFs* lineares para clusters, o cálculo do CRC, a divisão dos dados dos *GIFs* para o envio, a utilização de formatos JSON, a receção de payloads *MQTT* e o armazenamento no cartão SD por partes, visando o uso posterior desses dados para exibição no painel.

4.1.1. **ImgToCluster**

Esse processamento da imagem é necessário devido à configuração do display, que possui clusters de LEDs. A biblioteca utilizada para o carregamento das imagens para os LEDs segue o método convencional, que normalmente considera o mapeamento por linhas. No entanto, devido à configuração específica do display com clusters de LEDs, é necessário realizar esse processamento adicional à imagem. Ao dividir a imagem em clusters de LEDs, o processamento visa garantir que a informação correta seja enviada para cada grupo de LEDs correspondente. Isso envolve mapear os pixels da imagem para as coordenadas corretas nos clusters de LEDs. Através deste processamento, é possível enviar a informação correta de cor e brilho para cada LED num cluster específico, garantindo uma exibição adequada da imagem no display.

Para demonstrar esse processo, no algoritmo **1** está apresentada a função responsável pela conversão do *GIF*. A função recebe uma imagem de entrada e realiza a conversão dos píxeis lineares para um arranjo de clusters de 4x4.

Algorithm 1 img2ImgClustr

```
1: function IMG2IMGCLUSTER(imgIn)
2:   outImg ← CRIARIMAGEMRGB(imgIn.tamanho, corDeFundo=(0, 0, 0))
3:   pix ← OBTERRDADOSPIXELS(outImg)
4:   width, height ← OBTERTAMANHOIMAGEM(imgIn)
5:   for xc in range(4) do
6:     for yc in range(4) do
7:       for l in range(0, altura / 8) do
8:         for x in range(0, largura, 4) do
9:           for y in range(1 * 8, (1 + 1) * 8, 4) do
10:            rgbXY ← OBTERRPIXEL(imgIn, (x + xc, y + yc))
11:            pix[xOut%largura, yOut//largura] ← rgbXY
12:            xOut ← xOut + 1
13:            yOut ← yOut + 1
14:          end for
15:        end for
16:      end for
17:    end for
18:  end for
19:  return outImg
20: end function
```

Como exemplo deste processo na Figura 4.1 apresenta-se um exemplo de *GIF* utilizado para os painéis e na Figura 4.2 apresenta-se a imagem após ter sido convertida para clusters.



Figura 4.1: Exemplo de *GIF* normal

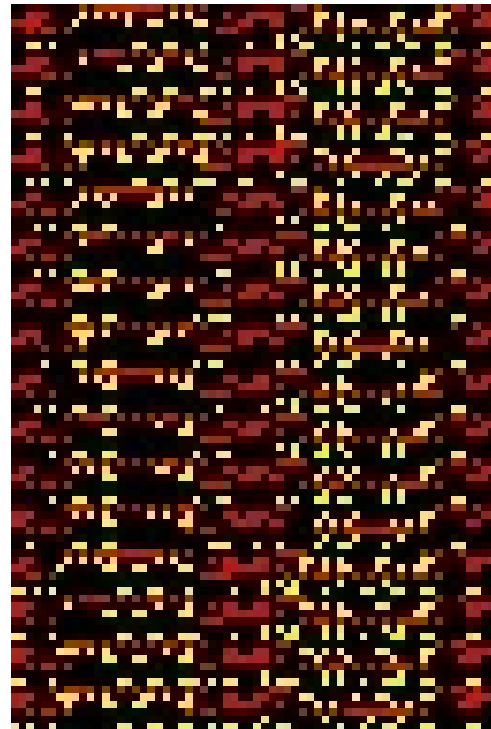


Figura 4.2: *GIF* convertido para clusters

4.1.2. Cálculo do CRC

Com a utilização dos *GIFs* processados anteriormente, inicia-se o tratamento dos dados para o envio, que inclui o cálculo do CRC CCITT de 16 bits para todo o arquivo de imagem. O CRC CCITT é uma variante do algoritmo CRC que segue as recomendações do Comité Consultivo para Telegrafia e Telefonia Internacional (CCITT), atualmente conhecido como ITU-T, para a geração do código de verificação.

Este cálculo serve como um checksum para verificar a integridade dos dados da imagem durante a transmissão. O algoritmo CRC-16-CCITT utiliza o polinómio demonstrado pela Equação 4.1 [3].

$$x^{16} + x^{12} + x^5 + 1 \quad \text{Eq. 4.1}$$

O processo de cálculo envolve a divisão dos dados de entrada por este polinómio, e o resto obtido é o checksum de CRC. Para esse propósito, na Equação 4.2, x representa a

sequência de bits que está a ser verificada, e a função `resto` calcula o resto da divisão de $x \times 2^{16}$ pelo polinómio usado no algoritmo CRC-16-CCITT. Ao aplicar o algoritmo CRC-16-CCITT, obtemos um valor inteiro que representa o conteúdo da imagem [3].

$$CRC - 16 - CCITT(x) = resto(x \times 2^{16}, polinómio) \quad \text{Eq. 4.2}$$

Posteriormente, os dados são recalculados para cada uma das partes do *GIF*, uma vez que ainda serão organizados para garantir que cada parte está correta quando chegarem ao microcontrolador.

4.1.3. Divisão em partes e formatação JSON

Para facilitar o processo de transmissão, o ficheiro total foi dividido em partes menores. No trabalho desenvolvido, cada parte possui 100 caracteres. Essa divisão em fragmentos de tamanho reduzido permite uma transferência e processamento eficientes no microcontrolador ESP32, levando em consideração as limitações mencionadas anteriormente. Cada parte terá o seu próprio valor de CRC calculado posteriormente no processo, como indicado anteriormente.

Para garantir a compatibilidade e facilidade de transmissão, cada parte do ficheiro foi convertido para o formato Base64. Embora a codificação Base64 aumente o comprimento dos caracteres em aproximadamente 1/3, optamos por esse formato como o formato de transmissão. A codificação em Base64 oferece uma representação textual dos dados binários, permitindo uma transferência confiável e direta por meio do protocolo MQTT [4].

Para facilitar a organização e identificação das partes do ficheiro, foi adotado um formato JSON. Cada parte é representada como um formato JSON contendo as seguintes informações: o nome do ficheiro, o número total de partes, o número da parte no ficheiro, o valor CRC da parte no ficheiro, o valor CRC do ficheiro completo e os dados associados à parte.

Ao utilizar este processo, possibilitamos uma transmissão confiável e eficiente dos

dados das imagens *GIF* por meio do *MQTT*. Os valores de CRC calculados fornecem um meio de garantir a integridade dos dados durante todo o processo de transmissão, enquanto a organização da carga útil e a representação em JSON facilitam a identificação adequada e o processamento de cada parte da carga útil. A Figura 4.3 representa esta metodologia do processo.

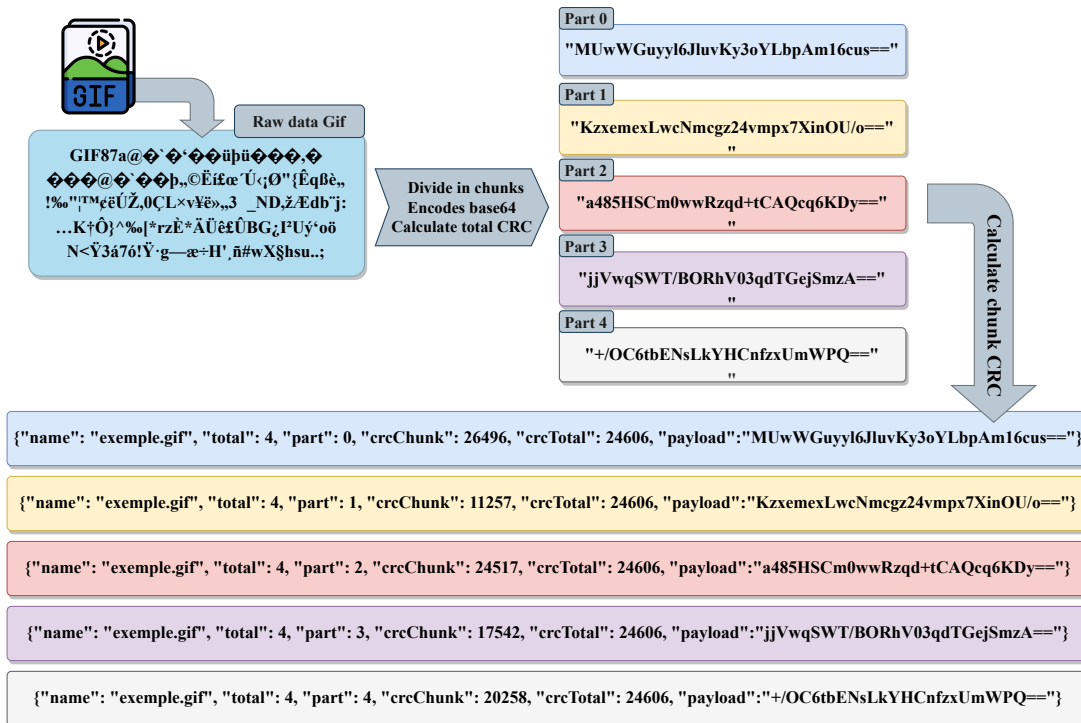


Figura 4.3: Processo de *Upload*.

4.1.4. Receção dos payload e armazenamento dos dados

A estratégia de descodificação do ficheiro, centra-se nos processos envolvidos em descodificar o formato Base64 e, em seguida, realizar as verificações de CRC. Em seguida, o ficheiro temporário é armazenado e, por fim, cada parte é combinada num único ficheiro e guardada num cartão SD ligado ao ESP32.

Após receber o ficheiro, o ESP32 inicia o processo de descodificação convertendo o ficheiro codificado em Base64 de volta para o seu formato original. Esta etapa permite recuperar os dados reais da imagem contidos no ficheiro. Em seguida, o ESP32 realiza

cálculos de CRC nos dados decodificados. Isto envolve calcular o valor de CRC de cada parte individual e compará-lo com o valor de CRC fornecido no ficheiro JSON. Se algum valor de CRC não corresponder, indica um erro de transmissão ou corrupção de dados, conforme indicado na Figura 4.4, e podem ser tomadas medidas apropriadas, como solicitar a retransmissão da parte afetada.

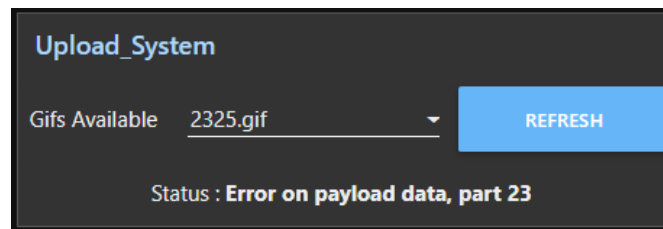


Figura 4.4: Exemplo de erro de transmissão.

Para simplificar a manipulação e reunião das partes do ficheiro, o ESP32 armazena temporariamente cada parte em ficheiros separados. Esses ficheiros temporários fornecem um mecanismo para organizar os dados até que o ficheiro completo seja reconstruído com sucesso. Se todas as verificações de CRC forem bem-sucedidas para todas as partes, indicando a integridade dos dados, o ESP32 procede à criação de um ficheiro temporário ao qual cada parte é anexada sequencialmente, mantendo a ordem correta.

Após a montagem do ficheiro completo, o ESP32 realiza uma verificação final do CRC no ficheiro reconstruído. Essa verificação garante a integridade da imagem na totalidade. Se o valor do CRC corresponder ao fornecido no ficheiro JSON, isso confirma a reconstrução bem-sucedida do ficheiro sem erros. Nesta etapa, o ESP32 renomeia o ficheiro usando o nome especificado no ficheiro JSON. O ficheiro renomeado está pronto para o armazenamento permanente. Para isso, um cartão SD é conectado ao microcontrolador ESP32. O ESP32 grava o ficheiro reconstruído no cartão SD, proporcionando uma solução confiável e duradoura de armazenamento.

Ao utilizar estas estratégias de decodificação e armazenamento, o ESP32 restaura efetivamente os dados da imagem do formato Base64, verifica a integridade dos dados por meio de verificações de CRC, organiza e une as partes do ficheiro, e finalmente armazena o ficheiro completo no cartão SD conectado. A Figura 4.5 ilustra o fluxo do processo de

4. Desenvolvimento

transmissão e descodificação de mensagens, representado em forma de diagrama. Nesse processo, a descodificação ocorre em etapas, sendo que cada parte é executada à medida que o payload é recebido, seguindo a sequência de receção dos payloads.

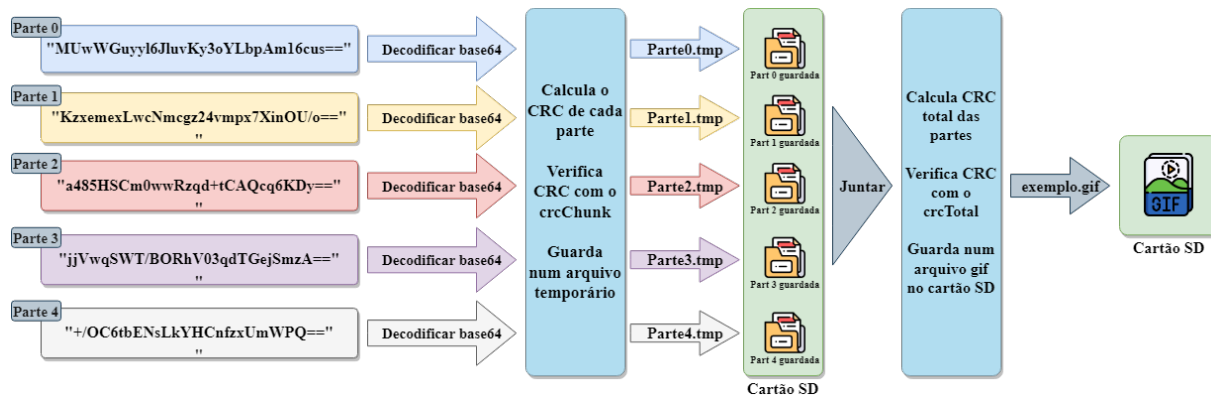


Figura 4.5: Recepção e guardar os GIFs.

4.2. Estado do GetCRC

A fase `GetCRC` é uma etapa de verificação de ficheiros. Nesta fase, a base de dados faz um pedido para obter o valor do CRC do ficheiro X armazenado no microcontrolador, de modo a confirmar se o ficheiro na memória está correto ou não. Um exemplo da GUI ao solicitar o CRC do arquivo “0.gif”, com a resposta do sistema de verificação da memória, é apresentada na Figura 4.6.

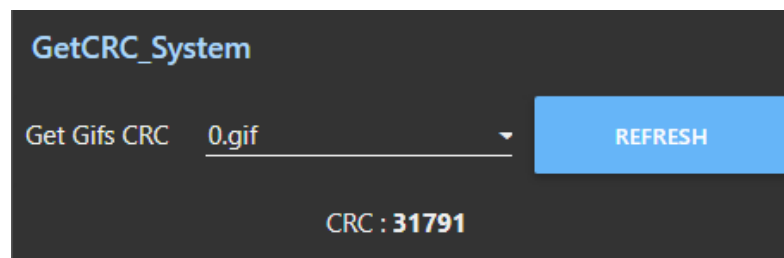
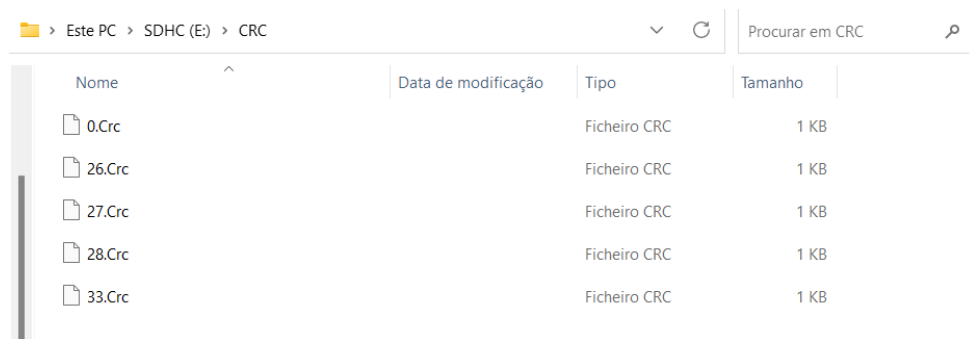


Figura 4.6: GUI com resposta do CRC.

Durante a inicialização do microcontrolador, é implementada uma função para ler e calcular o CRC de cada ficheiro *GIF* armazenado na memória. Essa função cria um ficheiro num diretório de CRCs em que o conteúdo é o valor do CRC do ficheiro calculado (Figura 4.7). Este ficheiro também pode ser criado na fase do *Upload* após a receção do ficheiro completo e sem erros.



Nome	Data de modificação	Tipo	Tamanho
0.Crc		Ficheiro CRC	1 KB
26.Crc		Ficheiro CRC	1 KB
27.Crc		Ficheiro CRC	1 KB
28.Crc		Ficheiro CRC	1 KB
33.Crc		Ficheiro CRC	1 KB

Figura 4.7: Diretório de CRCs da memória

4.3. Estado do `SetState` e `GetState`

O `SetState` é uma fase de controlo dos painéis. Nesta fase, ocorre o controlo remoto do sistema, onde são definidos os *GIFs* a serem exibidos nos painéis. É nessa etapa que são enviadas as instruções para selecionar e reproduzir os *GIFs* desejados nos painéis, permitindo o controlo remoto do conteúdo exibido.

Durante a inicialização do microcontrolador, o painel é configurado com um *GIF* padrão onde todos os LEDs estão desligados. Essa configuração é feita para evitar a exibição de informações incorretas caso ocorra uma falta de energia no microcontrolador. Dessa forma, o painel permanece num estado neutro até que seja estabelecida uma conexão estável e os dados corretos sejam transmitidos para exibição no painel.

O `GetState` é usado para obter o estado atual do painel. Nessa fase, uma solicitação é feita para obter informações sobre o estado atual dos LEDs, ou seja, quais *GIFs* estão a ser reproduzidos. Essa informação é essencial para monitorizar e controlar o funcionamento dos painéis, permitindo a visualização do estado atual dos LEDs e garantindo uma

comunicação bidirecional entre o sistema e os painéis.

Na Figura 4.8, é possível observar o método de aquisição do estado em que o painel está e qual o *GIF* que está a ser exibido (“28.gif” a ser reproduzido) e também o método de colocar o *GIF* vai ser exibido no painel(“28.gif” foi selecionado para ser exibido.).

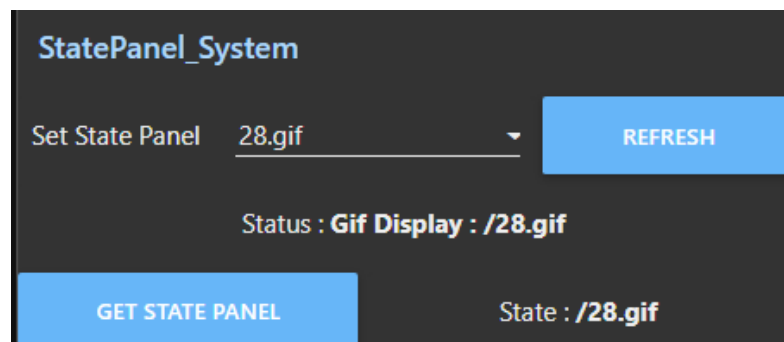


Figura 4.8: controlo e monitorização do estado do painel.

Capítulo 5.

Resultados do sistema

5.1. Implementação de testes

Foram realizados testes para avaliar o desempenho e eficácia do sistema desenvolvido para enviar imagens via MQTT, utilizando um conjunto de dados composto por *GIFs*. O conjunto de dados é composto por *GIFs* com resoluções, tamanhos de arquivo e complexidades diferentes. Essa versão do conjunto de dados possui aproximadamente 102.000 arquivos no formato de URL, apresentando tamanhos de arquivo heterogêneos [2].

Os testes tiveram como principais objetivos validar o envio de múltiplos arquivos e a capacidade de lidar com diferentes tamanhos, tanto com ruído como sem ruído. Para isso, foi desenvolvido um script em Python que realiza o *download* de cada arquivo *GIF*, redimensiona-os para 75x75 píxeis e limitados até 0,1 segundo. Em seguida, uma seleção aleatória de 100 *GIFs* foi feita. Esses 100 *GIFs* foram selecionados aleatoriamente por outro script em Python e possuem tamanhos que variam de 3 a 2077 kilobytes.

Durante os testes, cada um dos 100 *GIFs* foi selecionado individualmente para ser transmitido sequencialmente através do sistema implementado, utilizando a comunicação MQTT para enviar os dados ao destino. No final de cada teste, a integridade das imagens recebidas foram verificadas utilizando um programa de verificação de arquivos binários. Esse programa permitiu comparar os 100 *GIFs* escolhidos aleatoriamente (arquivos enviados) com os arquivos gravados no cartão SD após o teste (arquivos recebidos). A validação

do sistema proposto foi conduzida em duas fases, seguindo a descrição a seguir em cada uma delas.

5.1.1. Envio de 100 *GIFs*

Este primeiro teste envolve o envio dos 100 *GIFs* selecionados aleatoriamente sem manipulação intencional dos dados, ou seja, sem simulação de ruído. No caso de existir algum erro de envio, o utilizador é avisado pelo GUI e pode fazer o *upload* novamente, como, por exemplo, na Figura 4.4. Mas quando o *Upload* é bem-sucedido é indicado na GUI, Figura 5.1

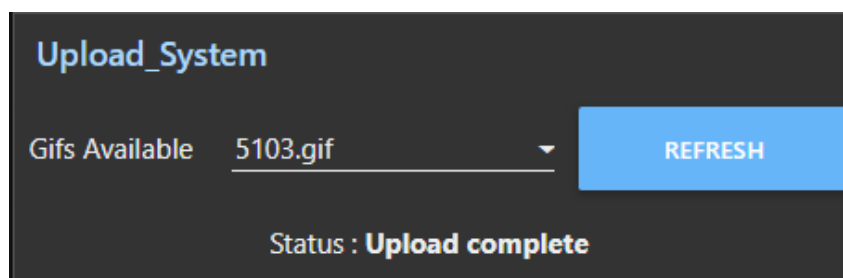


Figura 5.1: Enviado com sucesso.

Após o envio de todos os 100 *GIFs*, conforme mencionado anteriormente, o programa de verificação de arquivos binários compara cada um dos arquivos transmitidos com aqueles recebidos (cartão SD). A Figura 5.2 exibe o gráfico obtido após este primeiro teste, que apresenta os valores totais de envio, recepção e erros.

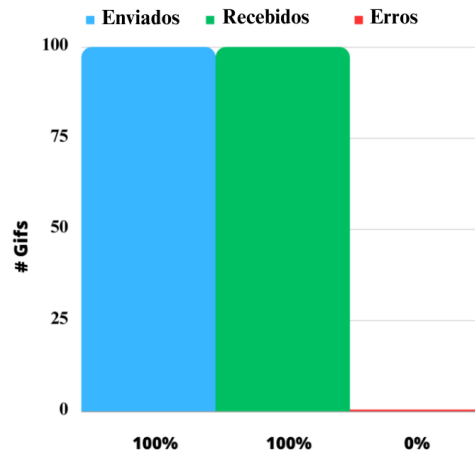


Figura 5.2: Número de mensagens enviadas, recebidas e erros encontrados.

5.1.2. Introdução de erros

O segundo teste consiste em enviar a mesma quantidade de *GIFs*, porém com a introdução de ruído nos *GIFs* aleatórios. Para fazer frente a preocupação do sistema errar, foi realizado um procedimento de teste para avaliar a robustez do sistema e a eficácia da implementação do CRC.

O procedimento de teste envolveu a simulação de diferentes cenários para avaliar a capacidade do sistema em detetar erros introduzidos intencionalmente nos dados transmitidos. A Figura 5.3 ilustra graficamente a manipulação intencional dos dados e a seguir estão as descrições deles:

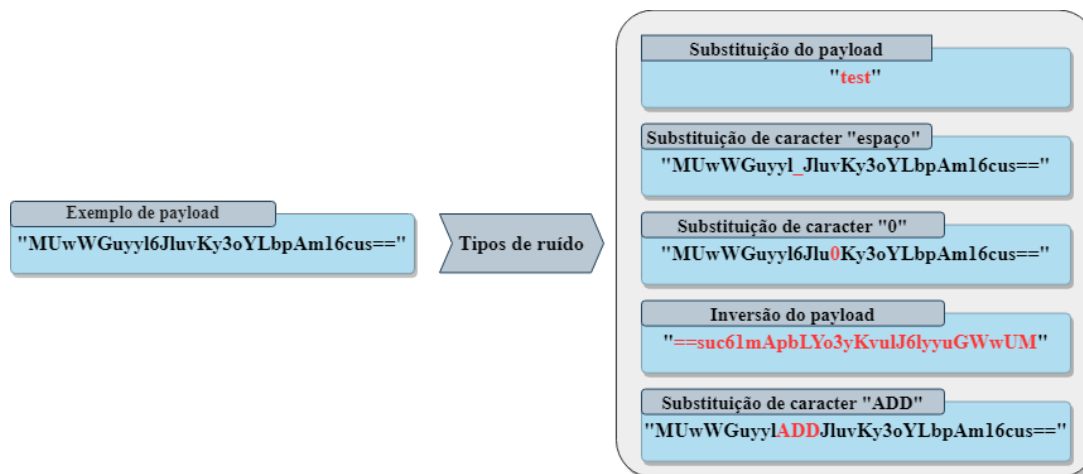


Figura 5.3: Tipos de ruídos artificialmente adicionados para avaliar o sistema proposto.

- **Substituição do payload:** Neste teste, o payload foi deliberadamente removido de uma das partes do *GIF*. O objetivo foi avaliar se o CRC poderia detetar a ausência dos dados e gerar um erro;
- **Substituição de Caracter:** Para avaliar a resiliência do sistema à corrupção de dados, uma única letra numa parte do payload foi substituída intencionalmente. A letra foi substituída por outra letra, número ou até mesmo deixada em branco. Esperava-se que o cálculo do CRC detete a alteração nos dados, sinalizando um erro;
- **Inversão de Payload:** A capacidade do sistema de identificar a inversão do payload foi testada através da inversão intencional do payload de uma das partes do *GIF*. A verificação do CRC era esperada que detetasse essa inversão, indicando um erro de transmissão;
- **Substituição de Caracter por "ADD":** Para avaliar a sensibilidade do sistema à expansão de dados, letras adicionais foram introduzidas numa parte do payload. Espera-se que a validação do CRC detetasse a discrepância no comprimento dos dados, indicando um erro.

A deteção de erros foi feita pelo sistema e notifica o utilizador da GUI sobre o erro ocorrido. Para além de informar sobre o erro, o sistema também indica qual foi o payload

específico ou parte do ficheiro que apresentou o erro, permitindo que sejam tomadas medidas em relação a isso. Esta informação detalhada auxilia na identificação e correção dos problemas durante a transmissão dos dados.

Na Figura 5.4 apresenta o GUI com um erro de Upload, indicando onde se encontra o erro, e na Figura 5.5 demonstra-se os payloads reais com o payload modificado logo a baixo

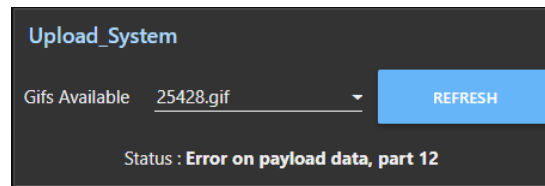


Figura 5.4: Erro de envio no GUI

```
{ "name": "25428.gif", "total": 132, "part": 10, "crcBlock": 28645, "crcTotal": 29232, "payload": "raskTZIUT5s0Z+zgabN2bl66d0v+2tvX7
x+1o+u8nYu4cZ432u66KrUXTyq8dS7d3WZHTpo2d/PWnYu6CL23IgFKLP8pscJIk0nuECGS1I7bKVPn2Emj5i2etXHS7umjx6/3fA==" }
1e21df61-7d79-4b43-86eb-9b3b50c3b582
{"name": "25428.gif", "total": 132, "part": 11, "crcBlock": 22152, "crcTotal": 29232, "payload": "wnM9VZxN46ijjU7zhKTQ07
hJVM436Xjzm2XlJBNPPPj0s0IAOKCBQwIL+DAEFGscoh47LJhCSogooMINPNRws05f8+jjnzTaWSPNR+x8ZA004KxjDTfgfFXc0+Ns5w442JTjjQ==" }
1e21df61-7d79-4b43-86eb-9b3b50c3b582
{"name": "25428.gif", "total": 132, "part": 11, "crcBlock": 22152, "crcTotal": 29232, "payload": "wnM9VZxN46ijjU7zhKTQ07hJVM436Xjzm
2XlJBNPPPj0s0IAOKCBQwIL+DAEFGscoh47LJhCSogooMINPNRws05f8+jjnzTaWSPNR+x8ZA004KxjDTfgfFXc0+Ns5w442JTjjQ==" }
97a10637-5ddd-4263-808a-bb3cdd004c75
{"name": "25428.gif", "total": 132, "part": 12, "crcBlock": 60655, "crcTotal": 29232, "payload": "M9CcY6CUlaiCgwAcePhABD
WcMQUTUKBBCHqQPKII6qsc8op2rwjjTXL2JNPPPj05gwz57iGDTvSreOmM96kIw0347zT3EPpcOMNg9hY4w02ykCjzjqzUaKPEQRYMMQRcDAxA==" }
97a10637-5ddd-4263-808a-bb3cdd004c75
{"name": "25428.gif", "total": 132, "part": 12, "crcBlock": 60655, "crcTotal": 29232, "payload": "=="AxAdEQRQMMYRQEPKaUzqjzjCky20w4Y
h9gNMOcpPE3Tz7430wIk69Mm0erSvTDGi75zwg50jPPPNJ2LXTjjwr2po8csq6IiKPkqHCCBBKUTUQMwDBAHpEcAwgCialUC6Yc9M"}
Error on payload data, part 12
351e0b2e-52cb-4264-974f-15f5f83f3dc1
{"name": "25428.gif", "total": 132, "part": 13, "crcBlock": 19317, "crcTotal": 29232, "payload": "EFMYMgUUUqSRhiqQHHJIN2
VC0iY21pn/RSc//xCDzHHUXMd000q4Aw2g5VADJ3YSwV00Ntqkw412GhDzTDMQIcNKu4Iih4UTFhgARNgTrHGGVBAMcUVkjxiCGqmIPII9rAQw==" }
351e0b2e-52cb-4264-974f-15f5f83f3dc1
{"name": "25428.gif", "total": 132, "part": 13, "crcBlock": 19317, "crcTotal": 29232, "payload": "EFMYMgUUUqSRhiqQHHJIN2VC0iY21pn/R
Sc//xCDzHHUXMd000q4Aw2g5VADJ3YSwV00Ntqkw412GhDzTDMQIcNKu4Iih4UTFhgARNgTrHGGVBAMcUVkjxiCGqmIPII9rAQw==" }
def556fa-d6e2-4c9c-88cd-92b6d6be6f0d
{"name": "25428.gif", "total": 132, "part": 14, "crcBlock": 5431, "crcTotal": 29232, "payload": "CSrW0CnVPv8EE8yj86XDFW7
UWHeOYOM8RVk8x2qjzjWMAuNMcqQww42qZQD6hMocDBBCEgwgYQUZ4DLhBTkHmLIGpSocogj16WCiJtV4PMSvcMEA440v4bD5IHQKDN0v4aJdA==" }
def556fa-d6e2-4c9c-88cd-92b6d6be6f0d
```

Figura 5.5: Erro de envio no MQTT (Payload invertido).

A implementação dos CRCs no sistema de transmissão de *GIFs* mostrou-se bem-sucedida na deteção de erros introduzidos durante o processo de teste. Para cada um dos cinco tipos de ruídos mencionados anteriormente na Figura 5.3, foram testados 20 *GIFs* selecionados aleatoriamente. Os cálculos dos CRCs identificaram com precisão as inconsistências nos payloads, retornando mensagens de aviso indicando qual a parte dos payloads estava incorreta. Os dados apresentados na Figura 5.6 fornecem uma visão abrangente dos valores totais enviados, recebidos e erros detetados durante o teste com

5. Resultados do sistema

ruído intencional. Isso demonstra a capacidade do sistema em garantir a integridade dos dados durante a transmissão.

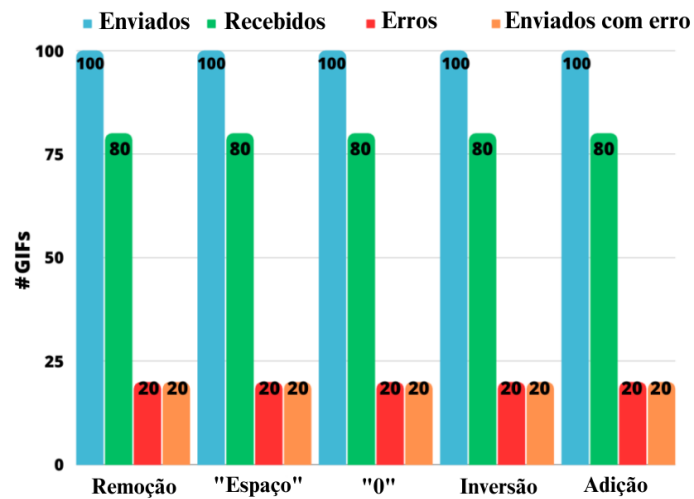


Figura 5.6: Gráfico com o número de mensagens enviadas, recebidas e erros encontrados após a adição de ruídos artificiais.

5.1.3. Implementação em painéis de teste

Depois dos testes anteriores realizados num protótipo, foi realizado um teste adicional num sistema usando painéis de mensagem variável, que são geralmente instalados em vias públicas. O teste envolveu a exibição de *GIFs* nos painéis, com o objetivo de verificar a transmissão correta das imagens e o controle individual de cada LED no painel.

Nas Figuras 5.7 e 5.8, são apresentadas fotos dos painéis com alguns *GIFs* em execução.



Figura 5.7: Sinalização de limite de velocidade



Figura 5.8: Sinalização de perigo de pavimento escorregadio

Foi realizado um vídeo que está disponível em [28], para demonstrar o funcionamento do sistema. Nesse vídeo, foram testados todos os estados do sistema, proporcionando uma visão abrangente do seu desempenho e operação.

Capítulo 6.

Conclusão

Este projeto apresentou um método de transferência de imagens utilizando o MQTT. A imagem original foi dividida em várias mensagens e foi aplicada uma comunicação segura e confidencial. Esse procedimento minimizou o uso da largura de banda da rede, levando em consideração as limitações de comunicação de alguns dispositivos utilizados nesse contexto. Além disso, um mecanismo de verificação de erros, como o CRC, foi implementado para detectar erros durante a transmissão. O sistema desenvolvido foi aplicado num ambiente real, utilizando painéis de mensagem variável, a fim de verificar o controle remoto da exibição de imagens conforme necessário.

O sistema também foi testado quanto à sua capacidade de lidar com diferentes tamanhos de *GIFs*. Esses testes, utilizando um conjunto diversificado de *GIFs*, permitiram avaliar a robustez e eficiência do sistema desenvolvido para o envio de imagens via MQTT. Os resultados obtidos forneceram informações valiosas sobre o desempenho, capacidade e limitações do sistema no que diz respeito à transmissão dos dados.

Como trabalho futuro, há planos de implementar a metodologia proposta em diferentes tipos de painéis. Além disso, sugere-se explorar outras tecnologias de comunicação, como LoRA ou CoAP, para permitir o controle remoto em diversos locais. Uma área de foco adicional é a implementação de medidas de segurança adicionais, como o uso de chaves de criptografia sem a necessidade de enviá-las nos payloads. Isso ajudaria a fortalecer a segurança do sistema, protegendo os dados transmitidos contra acesso não autorizado.

6. Conclusão

Bibliografia

- [1] Daniela de Jesus Correia Gomes et al. «Sinalização temporária em ambiente urbano: manual de boas práticas». Em: (2009).
- [2] Yuncheng Li et al. «TGIF: A New Dataset and Benchmark on Animated GIF Description». Em: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Jun. de 2016.
- [3] Mucong Chi, Dazhong He e Jun Liu. «Fast Software-based Table-less Algorithm for CRC Generation». Em: *2018 21st International Symposium on Wireless Personal Multimedia Communications (WPMC)*. 2018, pp. 544–549. DOI: [10.1109/WPMC.2018.8713103](https://doi.org/10.1109/WPMC.2018.8713103).
- [4] Ching-Chuan Wei, Pei-Yi Su e Shu-Ting Chen. «Comparison of the LoRa image transmission efficiency based on different encoding methods». Em: *International Journal of Information and Electronics Engineering* 10.1 (2020), pp. 1–4.
- [5] Arduino. *Arduino MKR1000 WIFI*. <https://store-usa.arduino.cc/products/arduino-mkr1000-wifi>. Acedido em junho. 2023.
- [6] Kevin Ashton. *Entrevista exclusiva com o criador do termo Internet das Coisas*. <http://www.finep.gov.br/noticias/todas-noticias/4446-kevin-ashton-entrevista-exclusiva-com-o-criador-do-termo-internet-das-coisas>. Acedido em abril. 2023.

- [7] botnroll.com. *ESP32*. <https://www.botnroll.com/pt/ethernet-wi-fi/3554-placa-de-desenvolvimento-esp32-esp32-wroom-32d-compat-vel.html>. Acedido em maio. 2023.
- [8] Rui Couto. *Revisão do Manual de Sinalização Temporária*. https://apcap.pt/wp-content/uploads/2021/02/Template_2APCAP_Sinalizacao-temporaria_Rui-Couto.pdf. Acedido em maio. 2023.
- [9] DMS. *Road and Traffic*. <https://dmsdisplays.com/solutions/road-traffic/>. Acedido em maio. 2023.
- [10] ElectroPeak. *MICRO SD TF card memory*. <https://electropeak.com/micro-sd-tf-card-adapter-module>. Acedido em junho. 2023.
- [11] escola de condução Invicta. *Sinalização Temporária*. <https://www.invicta.pt/codigo/sinais.asp?id=14>. Acedido em maio. 2023.
- [12] escola de condução Invicta. *Sinalização Temporária*. <https://www.imt-ip.pt/sites/IMTT/Portugues/InfraestruturasRodoviaras/InovacaoNormalizacao/Divulgao%20Tcnica/InstrucaoTecnicaUtilizacaoSinalizacaoMensagemVariavel.pdf>. Acedido em junho. 2023.
- [13] Mauser. *Raspberry Pi Pico W*. https://mauser.pt/catalog/product_info.php?products_id=095-0597&src=raspberrypi. Acedido em junho. 2023.
- [14] meiocorte. *ET6 – Cones*. <https://meiocorte.pt/sinais/sinalizacao-vertical/sinalizacao-temporaria/dispositivos-complementares/et6/>. Acedido em maio. 2023.
- [15] MQTT. *MQTT*. <https://mqtt.org/>. Acedido em abril. 2023.
- [16] Infraestruturas de Portugal. *Manual de sinalização temporária*. <https://servicos.infraestruturasdeportugal.pt/pdfs/infraestruturas/Manual%20de%20Sinaliza%C3%A7%C3%A3o%20Tempor%C3%A1ria%20JAE%201997%20-%20Tomo%20II%20-%20Estrada.pdf>. Acedido em abril. 2023.

- [17] seeedstudio. *Lichee Nano*. <https://www.seeedstudio.com/Sipeed-Lichee-Nano-Linux-Development-Board-16M-Flash-WiFi-Version-p-2893.html>. Acedido em junho. 2023.
- [18] sinalnorte. *A23 – Trabalhos na via*. <https://sinalnorte.com/produto/sinalizacao-vertical-perigo-a23/>. Acedido em maio. 2023.
- [19] sinalnorte. *A29 – Outros perigos*. <https://sinalnorte.com/produto/sinalizacao-vertical-perigo-a29/>. Acedido em maio. 2023.
- [20] sinalnorte. *C13 – Proibição de exceder a velocidade máxima*. <https://sinalnorte.com/produto/sinalizacao-vertical-regulamentacao-proibicao-c13/>. Acedido em maio. 2023.
- [21] sinalnorte. *C14a – Proibição de ultrapassar*. <https://sinalnorte.com/produto/sinalizacao-vertical-regulamentacao-proibicao-c14a/>. Acedido em maio. 2023.
- [22] sinalnorte. *ET10 – Perfil Móvel em Plástico*. <https://sinalnorte.com/produto/acessorios-complementares-et10-pmp/>. Acedido em maio. 2023.
- [23] sinalnorte. *ET5 – Balizas de alinhamento*. <https://sinalnorte.com/produto/sinalizacao-vertical-temporaria-et5-balizas-alinhamento/>. Acedido em maio. 2023.
- [24] sinalnorte. *TD7 – ST7*. <https://sinalnorte.com/produto/sinalizacao-vertical-temporaria-td7-st7/>. Acedido em maio. 2023.
- [25] sinalnorte. *TV1 – ST14*. <https://sinalnorte.com/produto/sinalizacao-vertical-temporaria-tv1-st14/>. Acedido em maio. 2023.
- [26] sinalnorte. *TV4 – ST8a/b*. <https://sinalnorte.com/produto/sinalizacao-vertical-temporaria-tv4-st8a-st8b/>. Acedido em maio. 2023.
- [27] Espressif Systems. *ESP32*. <https://www.espressif.com/en/products/socs/esp32>. Acedido em maio. 2023.

Bibliografia

- [28] David Valente. *MQTT Remote Control of Indicator Panels*. <https://youtu.be/NThibquqSV0>. Acedido em julho. 2023.
- [29] Henrique Frank W Puhlmann. «PMV (Painel de Mensagens Variáveis) com NT-CIP». Em: ().