

Trabalho Prático Laboratorial

Identificação de condutor

Laboratórios Integrados 2



Mestrado em Engenharia

Electrónica e de Computadores



José Henrique Brito

1. Introdução

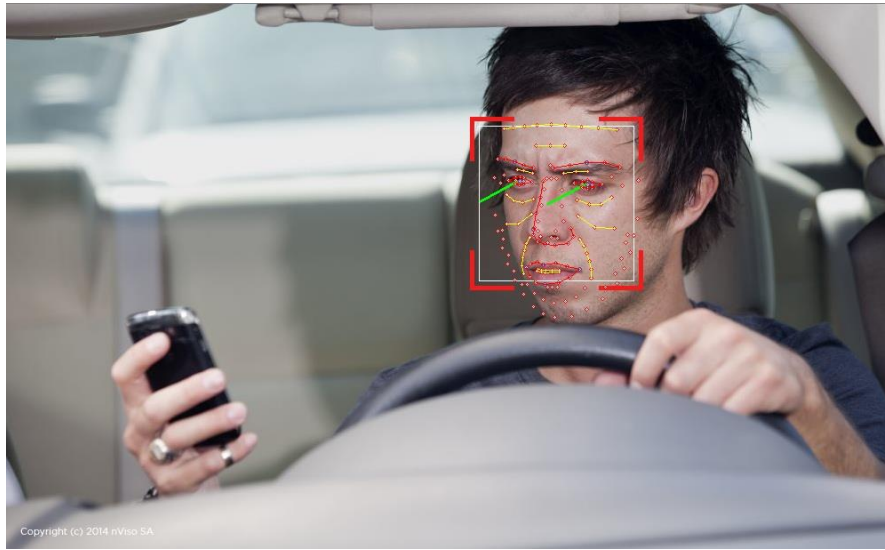


Figura 1 – Reconhecimento do condutor

A indústria automóvel procura constantemente novas formas de adaptar o funcionamento dos seus produtos às conveniências e conforto do cliente. O objectivo final deste projecto é o desenvolvimento e implementação de um sistema que permita que um automóvel reconheça a identidade do seu condutor, o que permitirá o ajuste automático de parâmetros do interior do automóvel (posição do banco, volume do rádio, etc.) às preferências do condutor que estiver a conduzi-lo em cada momento.

A verificação da identidade do condutor será efectuada através de visão por computador, com recurso a imagens obtidas por uma câmara apontada à face deste, tipicamente montada junto ao retrovisor do automóvel. O módulo de processamento de imagem deverá comparar a face presente na imagem com faces de referência existentes numa base de dados com os condutores previamente introduzidos no sistema. O processamento de imagem será implementado em PC. A comunicação entre o PC e os restantes dispositivos do automóvel será efectuada através do barramento CAN, um standard de comunicação comum nos automóveis actuais.

Este trabalho compreende portanto o desenvolvimento de um sistema integrado que engloba projecto de *hardware* e *software*. Neste projecto, a aplicação que correr no PC implementa funções de processamento e análise de imagem e comunica com um dispositivo através de USB, que por sua vez permite a comunicação com uma rede industrial CAN. O esquema global do sistema é o da Figura 2.

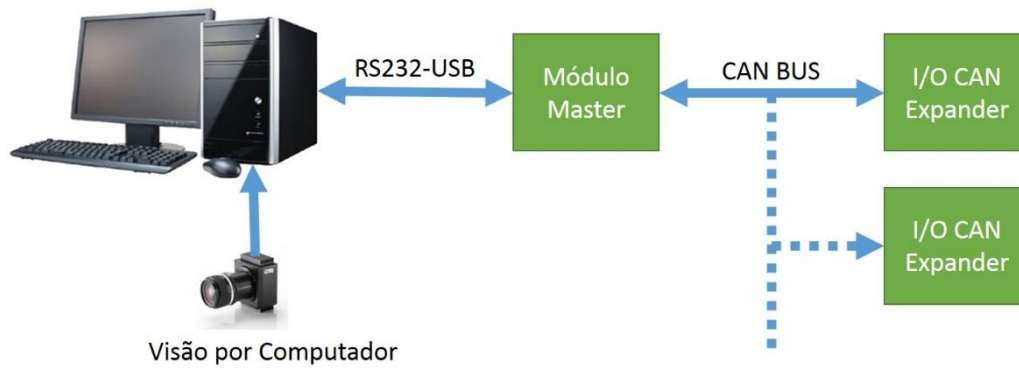


Figura 2 – Arquitectura do Sistema

1.1. Áreas de conhecimento

- Visão por computador.
- Sistemas embebidos.
- Redes industriais.
- Projecto, desenho e montagem de placas de circuito impresso.

1.2. Grupos

- Devem ser formados grupos de 4 elementos
 - 2 elementos para o bloco de *software*
 - 2 elementos para o bloco de *hardware*

1.3. Duração

- 30 Horas de contacto
- 160 Horas de trabalho

2. Descrição e requisitos dos módulos do sistema

O sistema é constituído por dois blocos principais. O primeiro bloco, baseado em *software*, é responsável pelo interface com o utilizador, gestão dos dados e configuração do sistema, processamento e análise de imagem e comunicação dos dados através de USB. O segundo bloco, baseado em sistemas embebidos, é responsável pela comunicação, através de uma rede industrial CAN, dos dados obtidos no módulo de *software*.

O bloco de *software* deverá ser constituído por:

- Módulo de interface gráfico com o utilizador
- Módulo de aquisição de imagem, obtida por uma câmara, detecção e reconhecimento de faces na imagem
- Módulo de comunicação de dados por protocolo série através de USB.

O bloco baseado em sistemas embebidos incluirá o desenvolvimento de dois tipos módulos de *hardware*:

- Módulo Master, que estará ligado ao PC por USB, responsável pela gestão e transmissão dos dados resultantes da aplicação de processamento e análise de imagem, através de uma rede de comunicação industrial baseada no protocolo CAN BUS.
- Módulo de expansão I/O, que estará ligado ao barramento CAN, traduzirá nas suas saídas os dados resultantes da análise de imagem, transmitidos pelo módulo master.

2.1. Módulo Master CAN (*hardware*)

Este módulo de *hardware* será utilizado para realizar a interface entre o PC a rede CAN.

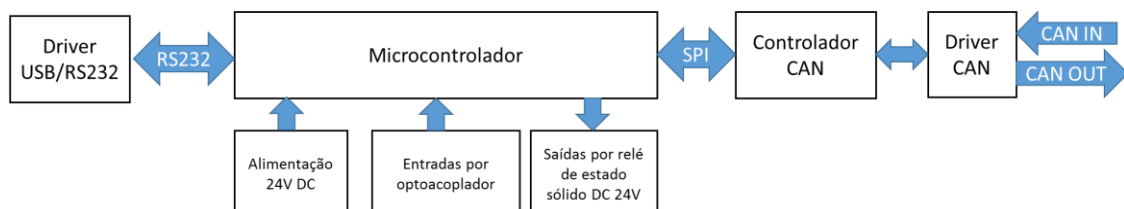


Figura 3 – Módulo Master

Para tal, no projecto deverão ser consideradas as seguintes características:

- Alimentação por fonte externa 12/24 V DC;

- Ligação ao PC por USB – deverá para isso incorporar um conversor USB/RS232 (FT232R ou similar);
- Transmissão de dados através do protocolo CAN – deverá para isso integrar um microcontrolador (Atmel 328), que comunicará com o PC através do protocolo RS232, e com um controlador Stand-Alone CAN (MCP2515) através da interface SPI.
- Regularização dos níveis de tensão para rede CAN através de um driver CAN (MCP2551SN);
- Disponibilização de um conjunto de **6 entradas** (por opto-acoplador LTV-356T ou similar) e **2 saídas** (por relé de estado sólido - CPC1002N ou similar), com indicação luminosa do estado.

2.2. Módulo de expansão I/O CAN (*hardware*)

Este módulo de *hardware* será utilizado como módulo de expansão da rede CAN.

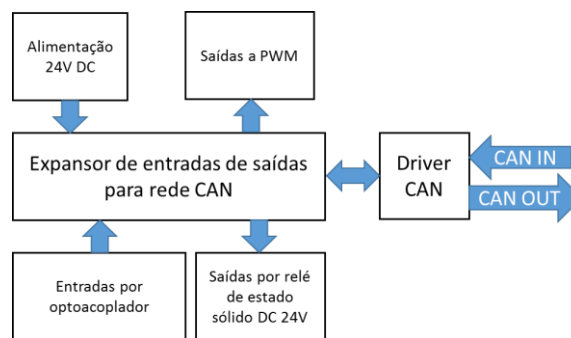


Figura 4 – Módulo de expansão I/O

No projecto deverão ser consideradas as seguintes características:

- Alimentação por fonte externa 12/24 V DC;
- Implementação do protocolo CAN através de um bloco expansor de entradas e saídas MCP25020 ou MCP25050. A regularização dos níveis de tensão para rede CAN deverá ser estabelecida através de um driver CAN (MCP2551SN);
- Disponibilização de um conjunto de **2 entradas** (por opto-acoplador LTV-356T ou similar) e **6 saídas** (por relé de estado sólido - CPC1002N ou similar) com indicação luminosa do estado.

2.3. Módulo de Comunicação PC <-> Módulo Master CAN (*software*)

É necessário definir um protocolo de comunicação, entre o PC e o módulo Master, que permita:

- Enviar, do PC para o Módulo Master, o resultado da identificação de faces, com identificação de quais os módulos de expansão I/O para onde deve ser enviado
- Enviar, do PC para o Módulo Master, um pedido de identificação de todos os módulos de expansão I/O ligados à rede CAN.
- Enviar, do Módulo Master para o PC, um pedido de criação/alteração de registo de face na base de dados. Quando uma das entradas do módulo Master é activada, a posição correspondente da base de dados deve passar a referir-se à face detectada nesse momento (caso alguma face esteja a ser detectada).

2.4. Módulo de Comunicação Módulo Master CAN <-> Módulo de expansão I/O CAN (*software/firmware*)

É necessário definir um módulo de comunicação no módulo Master e a configuração do módulo de expansão, que permita:

- Configurar os registos do controlador Stand-Alone CAN para que este funcione no modo pretendido;
- Converter as mensagens recebidas do PC no protocolo CAN;
- Transmitir as mensagens através da rede CAN;
- Consultar o código de identificação de todos os módulos de expansão I/O ligados à rede CAN.

Para programar os registos do módulo de expansão pode ser utilizado o *software* MCP250xxProgrammer.exe. Recomenda-se a utilização do programador PICKit 2.

2.5. Módulo de processamento de imagem: identificação e reconhecimento de faces

As principais tarefas associadas ao processamento de imagem são:

- 1) Identificar o melhor setup para aquisição da imagem (câmara, iluminação, óptica, etc);
- 2) Calibrar a distorção radial e tangencial provocada pela lente;
- 3) Detectar a(s) face(s) na imagem, escolhendo a principal
- 4) Segmentar a face
- 5) Efectuar uma extracção de características da face detectada
 - a) Definir quais as características relevantes a extrair

- 6) Comparar características da face detectada com características de faces existentes na base de dados

Os dados a enviar para a rede CAN são:

- Se foi identificada uma face
- A identidade da face identificada;

2.6. Módulo de interface com o utilizador

Deverá ser desenvolvida uma interface com o utilizador que permita uma fácil e intuitiva interação com o sistema. Esta deverá respeitar os seguintes requisitos:

- O interface deverá indicar o estado da ligação ao Módulo Master
- O interface deverá poder listar todos os módulos ligados à rede CAN
- A imagem da câmara deverá aparecer em tempo real no ecrã o PC
- A deteccção de faces deverá ser assinalada na imagem
- A identificação de faces deverá ser assinada na imagem
- A não detecccção de faces deverá ser assinada numa ou mais saídas digitais de um ou mais módulos de expansão I/O; a(s) saída(s) dos módulos a activar deverão ser configuráveis
- A identificação de faces deverá ser assinada numa ou mais saídas digitais de um ou mais módulos de expansão I/O; a(s) saída(s) dos módulos a activar deverão ser configuráveis
- A deteccção de faces sem identificação deverá ser assinada numa ou mais saídas digitais de um ou mais módulos de expansão I/O; a(s) saída(s) dos módulos a activar deverão ser configuráveis
- O interface deverá permitir a criação, edição e remoção de registos de faces na base de dados; para cada face deverá ser possível configurar quais as saídas de quais módulos de expansão I/O deverão ser activados/desactivados
- O interface deverá permitir a recepção de pedidos de criação de novo registo de face na base de dados provenientes do módulo master, com a face identificada no momento;

3. Desenvolvimento

Durante o desenvolvimento do projecto, todos os documentos, desenhos, esquemáticos, código fonte, etc., deverão ser geridos através de um sistema de gestão de versões, do tipo Git. O servidor na *cloud* a utilizar poderá ser o GitHub, ou uma alternativa que suporte repositórios privados como o Gitlab ou o BitBucket.

Os componentes a utilizar no desenvolvimento dos PCBs podem ser do tipo through-hole ou SMD. O desenvolvimento do *hardware* deverá começar pela realização de um protótipo em *breadboard* e só depois pelo desenho e implementação de uma placa PCB. Para o desenho do esquemático e edição do *layout* sugere-se a utilização do *software* EAGLE.

A(s) linguagem(ns) de programação, as ferramentas de desenvolvimento e as bibliotecas externas são livres.

3.1. Principais componentes e equipamentos

- PC
- Câmaras Ethernet ou USB
- Sistemas de iluminação: dome, back light, array de leds, anel de leds.
- Ópticas, Filtros
- Microcontrolador Atmel 328 ou similar
- Controlador Stand-Alone CAN MCP2515
- CAN I/O expander MCP25020 ou MCP25050
- Driver CAN MCP2551SN
- Optoacoplador LTV-356T ou similar
- Relé de estado sólido DC CPC1002N ou similar
- Conversor USB/RS232 FT232R
- Varistor
- Regulador de tensão
- Ferrite bead BK0603HS330-T

4. Avaliação

A avaliação é individual e incluirá uma defesa oral. Ao longo do projecto serão avaliados os seguintes itens:

- Trabalho preparatório (obrigatório) e performance do aluno no decorrer das aulas; (10%)
- Funcionamento individual de cada um dos módulos desenvolvidos e do sistema na sua globalidade:
 - Módulo de interface com o utilizador (*software*);
 - Módulo Master CAN (*hardware/software*);
 - Módulo de expansão I/O CAN (*hardware/software*);
 - Módulo de comunicação PC <-> Módulo Master CAN (*software*);
 - Módulo de comunicação Módulo Master CAN <-> Módulo de expansão I/O CAN (*software*);
 - Módulo de Processamento e análise de imagem;
 - Relatório Final;
 - Apresentação oral;

5. Bibliografia

- Tutorial Eagle: <https://www.youtube.com/playlist?list=PLA3877F8BF4576577>
- Datasheet Microcontrolador Atmel 328
- Datasheet CAN I/O expander MCP25020.
- Datasheet Controlador Stand-Alone CAN com interface SPI (Serial Peripheral Interface) MCP2515
- Datasheet Driver CAN MCP2551SN
- Datasheet Conversor USB/RS232 FT232R
- Datasheet Optoacoplador LTV-356T ou similar
- Datasheet Relé de estado sólido DC CPC1002N ou similar
- Burning the Bootloader Atmega328:
<http://arduino.cc/en/Tutorial/ArduinoToBreadboard>
- USB FTDI hardware guide:
http://www.ftdichip.com/Support/Documents/AppNotes/AN_146_USB_Hardware_Design_Guidelines_for_FTDI_ICs.pdf
- CAN + Arduino area network: <http://modelrail.otenko.com/arduino/arduino-controller-area-network-can>
- CAN bus shield: http://www.seeedstudio.com/wiki/CAN-BUS_Shield
- Programação MCP250xx: http://liionbms.com/php/programming_MCP250xx.php
- Instalação do OpenCV em Windows:
http://docs.opencv.org/3.1.0/d3/d52/tutorial_windows_install.html#gsc.tab=0