

	Data	Signatures	
Author:	29/03/2020		Title: Guia do Desenvolvedor Sistema EAR - Hardware
Verif.:			
Aprov.:			

1. Escopo

O objetivo deste documento é descrever o sistema EAR sob o ponto de vista de um desenvolvedor do hardware e como base para o desenvolvedor de software entender o sistema EAR. Para a mecânica e o “Firmware” serão criados documentos específicos. Este documento reúne especificações, recomendações e sugere componentes existentes e disponíveis para compor o sistema. Este documento também procura incluir documentos emitidos anteriormente para facilitar o acesso às informações.

Estão previstas ampliações de escopo do projeto para suportar respiração assistida, Somente para os pacientes em fase de recuperação. A eletrônica do projeto EAR já contempla expansões para esta função e outras.

Este projeto pode ser alterado a qualquer momento, sem aviso prévio.

Disclaimer: não assumimos nenhuma responsabilidade sobre qualquer dispositivo construído a partir das informações deste documento e orientamos para que os dispositivos sejam projetados na estrita observância às normas locais, que sigam as recomendações de especialistas e que sejam homologados pelas agências reguladoras em todas as fases.

2. Controle do Documento

- 29/03/2020: Emissão R00 – Português BR

3. Referências

- Open Source COVID19 Medical Supplies facebook:
<https://www.facebook.com/groups/670932227050506/>
- COVID-19 Air BRASIL - Fast production of assisted ventilation devices
<https://www.facebook.com/groups/235476464265909/>
- RDSV200320 Emergency Automatic Respirator Specification R00 – Preliminary
- RDSV200322 EAR Hardware Proposition Draft R00 – PTBR
- MIT E-Vent – “Key Ventilation Specifications”

4. Considerações Iniciais

Abreviaturas e termos utilizadas em todo o documento:

- EAR: Emergency Automotic Respirator
- FDC: Fim de Curso

- BAG: bolsa de ar, pode ser a do AMBU ou a do FIO2
- AMBU: Equipamento médico existente, utilizamos somente o BAG para comprimir ar para o paciente
- FIO2: Fraction of Inspired O2 define a proporção de O2 no ar inspirado pelo paciente
- ECU: Electronic Control Unit
- PEEP: Pressão positiva expiratória final. Para eletrônicos e SW developers, se trata de um “Offset DC” na pressão do ar do pulmão do paciente. A pressão nunca cai a menos de 5-10cm de H2O.
- HEPA: tecnologia empregada em filtros de ar com alta eficiência na separação de partículas (wikipedia)
- NPN: polaridade da conexão do sensor ou atuador, comuta ou é comutado pelo GND (Low Side)
- SPI: Serial Peripheral Interface, interface serial síncrona, baseada em registradores de deslocamento
- WIP: Work in Progress, trabalho não finalizado
- GPIO: General Purpose IO, ou porta de uso geral do arduíno
- SOA: Safe operating area, ou área de operação segura do componente

Existem várias soluções de automação de AMBU, porque mais uma? A razão é simples: todas as automações de AMBU foram baseadas em materiais de disponibilidade em centros de pesquisa, não em municípios de menor renda. Sim, os 60% de municípios brasileiros que não dispõem de ventilador mecânico provavelmente não disporão dos insumos para produzi-los ou mantê-los neste primeiro momento.

O foco então se trata de um equipamento para regiões sem recursos, principalmente em países emergentes. Trata-se então do já difundido conceito de ter uma mão mecânica para apertar um AMBU. Esta mão mecânica é composta de um mecanismo de compressão mecânico movido por um motor de máquinas de vidro de automóveis, que já conhecemos e utilizamos em escala em produtos. Conhecemos os modos de falha do motor e os estamos tratando. **O motor de vidros tem mecânica adequada para suportar reversões em cada ciclo, necessárias para o controle da quantidade de ar e da pressão aplicada ao paciente. Isto é possível devido ao mecanismo de sem-fim acoplado a uma engrenagem e um coupler de borracha interno. Já testamos um acionador mecânico que comprime um AMBU fechado e abre a válvula de segurança alimentado em 12V. AMBUs abertos conseguimos comprimir com 4Vdc.**

Para o controle do sistema foi desenvolvida um módulo de controle – a **ECU EAR** - que pode ser montada e construída por qualquer pessoa que fabrique artesanalmente equipamentos eletrônicos. Existem versões com placas de circuito impresso face simples, para que possa ser produzida até por processos artesanais. Para alteração dos parâmetros, esta placa tem controles por potenciômetros para permitir uso fácil pelos profissionais de saúde. O microcontrolador é um Arduíno, **similar ao da solução do MIT**, que tem a maior base de desenvolvedores no mundo. Esta placa tem várias proteções, inclusive contra curto-circuito no motor, que entra em curto após uso prolongado. Esta placa também controla a velocidade do motor, definindo a pressão aplicada no AMBU. O uso do motor é reversível, para possibilitar o controle de deslocamento de compressão do AMBU, permitindo a variação do ar inspirado, ou volume tidal. A placa também possui várias entradas e saídas para acionamento direto de relés e leitura de sensores NPN.

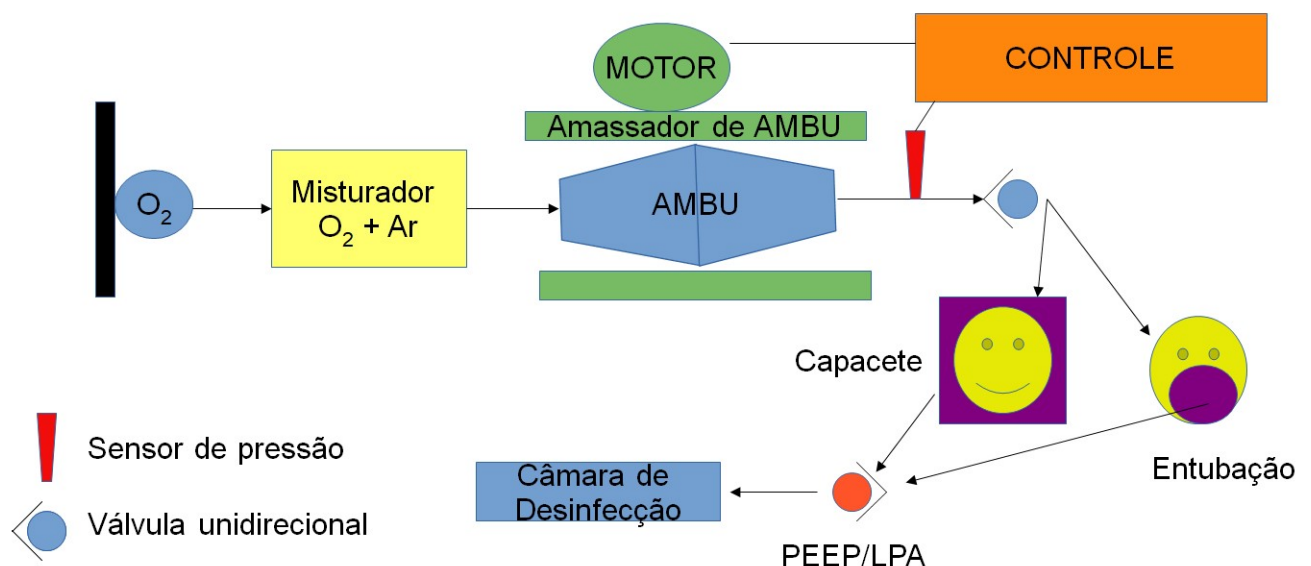
O “firmware” a que se refere esta especificação é, a princípio, executado no Arduino Pro Mini. Se o hardware for diferente, por favor buscar a documentação do hardware para sua versão de placa.

5. Requisitos Básicos

Requisitos básicos a serem atendidos, de acordo com: MIT-KVS: Key Ventilation Specification do MIT

- BPM (respirações por minuto):
 - MIT-KVS: 8-30BPM
- Volume Tydal - quantidade de ar a ser administrado ao paciente:
 - MIT-KVS: 200-800ml baseado no peso do paciente
- I/E ratio (relação tempo de inspiração / tempo de expiração):
 - MIT-KVS: começar entre 1:2, melhor se ajustável entre 1:1 e 1:4
- Pressão máxima:
 - limitada a 40cm H₂O em qualquer condição, recomendada válvula passiva de alívio fixa em 40cm H₂O
 - pressão de platô limitada a, no máximo, 30 cmH₂O
 - PEEP é necessário de 5-15 cm de H₂O, muitos pacientes precisam 10-15 cm de H₂O

6. Visão Global do Sistema EAR



Esboço do sistema EAR, desenho by Suzuki

O processo controlado pode ser implementado da seguinte forma:

- Mistura por válvulas mecânicas especializadas ou eletroválvulas pneumáticas a partir de duas fontes: O2 comprimido e ar comprimido. A mistura pode ser controlada pela ECU

– Compressão da **Bolsa de AMBU ADULTO por um mecanismo motorizado com reversão**. O controle de volume Tydal é realizado mediante controle de curso e a pressão máxima é realizada por controle da velocidade do motor. Estamos trabalhando em conjunto com desenvolvedores para termos bolsas para reposição, pois prevemos que estourem pelo uso contínuo. Em caso de rompimento de bolsa, quebra de motor ou dano de algum componente mecânico todo o conjunto compressor de AMBU pode ser substituído em menos de 5 segundos.

– **Acoplamento ao paciente, método não invasivo: usa máscara de AMBU com válvula unidirecional adicional para evitar retorno ao AMBU do ar contaminado** e encaminha para uma câmara descontaminante o ar expirado pelo paciente. Controle de Peep mandatório. Capacete com pressão negativa se disponível. Respiração assistida prevista no escopo do projeto.

- **Acoplamento ao paciente – método invasivo:** nesta etapa está prevista ventilação não assistida com uso de mangueira intratraqueal em pacientes sedados. Haverá monitoração de pressão positiva com alerta para detecção de expiração conflitante pelo paciente.

Utilizamos máscara de AMBU com mangueira e válvula para evitar retorno ao AMBU do ar contaminado e que encaminhe para uma câmara descontaminante o ar expirado pelo paciente. Peep mandatório. Capacete com pressão negativa se disponível.

- Controle de PEEP

Haverá **controle de PEEP por filtro de barreira HEPA ou similar**, associado ou não a uma restrição mecânica para controle de pressão de PEEP. Caso não existe filtro de barreira, será mandatória exaustão ou câmara de descontaminação.

Obs: haverá um medidor permanente de pressão de PEEP, mecânico ou eletrônico.

- Exaustão do ar contaminado

Esta etapa será mandatória se o ar expirado pelo paciente não passar por filtro de barreira. Podem ser aplicadas as soluções abaixo isoladamente ou em conjunto.

- **Passagem forçada por descontaminantes químicos ou mecânicos.** Podem ser utilizadas soluções com sabão, que aderem às gorduras do Covid19, ou água sanitária (fonte abundante de hipoclorito de sódio ou cálcio) em concentrações adequadas.

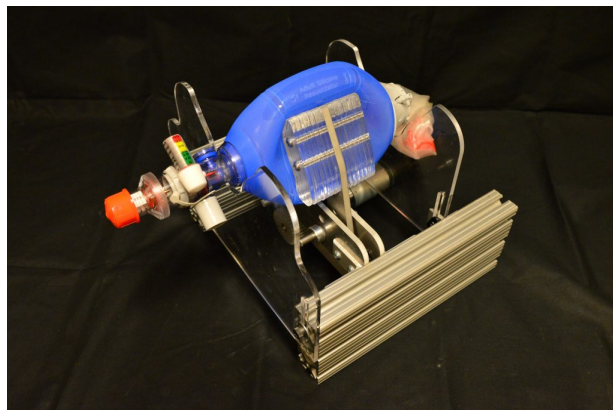
- **Aquecedores por queima.** Para submeter o vírus a altas temperaturas. Podem ser utilizadas serpentinas para aquecimento de água onde o ar passaria dentro, usar o ar expirado em chaminés com queima de madeira ou gás, junkers, etc

- **Irradiação UV:** o Covid19 poderia ser submetido a UV de alta intensidade. Seria necessário um tempo de exposição de muitos minutos mesmo com fontes de alta intensidade

Todo o setup acima pode ser alimentado por baterias automotivas carregadas pela rede AC, para prover energia mesmo que caia a rede AC **ou por uma fonte DC de 12V, 30 ampères ou maior, destas utilizadas para alimentar iluminação de Leds**, encontráveis em casas de eletrônica ou iluminação.

7. Visão Geral do Hardware Controlado pelo Sistema EAR

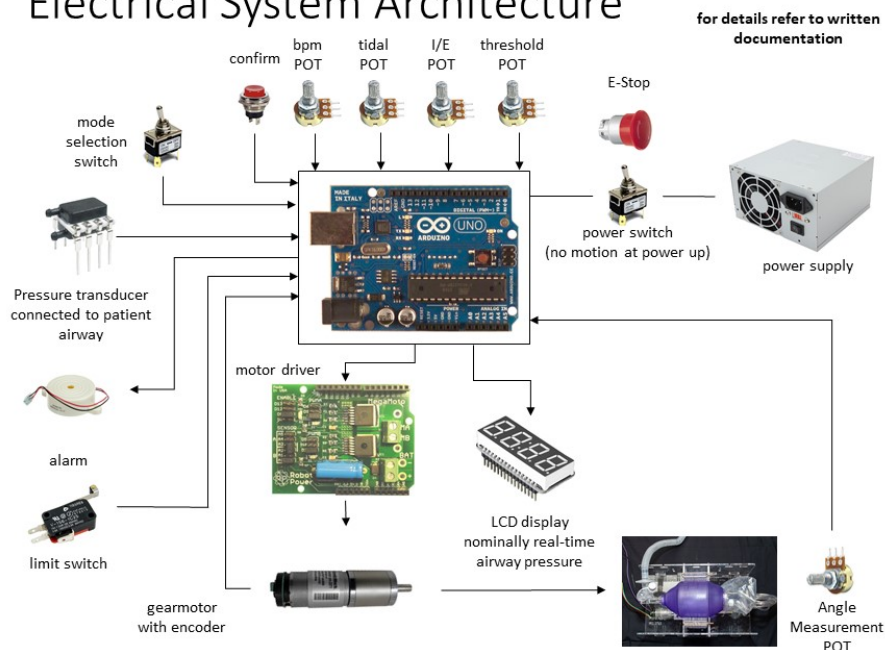
O Projeto EAR nasceu com conceito de compressão de AMBU pneumática, mas o conceito foi abandonado porque exigiria ar comprimido e porque o volume de ar não pode ser controlado facilmente. O conceito foi trocado para compressão mecânica de AMBU.



Compressor de AMBU projeto MIT E-Vent

A arquitetura do sistema é muito similar ao do projeto MIT E-Vent, que conheci alguns dias depois de esboçar o Projeto EAR, muitas coisas são idênticas, aproveito a imagem do MIT que é muito bem feita.

Electrical System Architecture



Arquitetura do projeto MIT E-vent

Listamos abaixo as diferenças principais da arquitetura do E-Vent e o EAR


- a ECU EAR usa Arduino Pro Mini, por questões de disponibilidade local. Pode usar a princípio qualquer Arduino, mas tem que fazer uma PCB adaptadora para acoplar na ECU EAR ou ligar por fios, somente para desenvolvimento.

- o projeto EAR prevê a possibilidade de fazer controle de FIO2 a partir de eletroválvulas pneumáticas de caminhão, mediante controle de tempo de acionamento individual, com armazenamento em uma almofada plástica de vinyl, doravante denominada simplesmente BAG. Enchemos parcialmente o BAG (ex: 80% ou menos) que será esvaziado apenas parcialmente (ex: máximo 30%) em cada admissão do AMBU. Existem sensores de FDC para inibir as válvulas se o BAG estiver a 80%. Para que o EAR possa aplicar a quantidade correta, existe um modo de configuração onde definimos os tempos de acionamento de cada válvula pelos mesmos pots de operação, pelo menos nesta fase inicial. Sugerimos marcas visuais no painel dos pots para memorizar esta configuração.

- o motor do EAR é um motor BDC de máquinas de vidro de automóveis, do tipo utilizado no Pálio e Gol, o motor robusto com maior disponibilidade local. Acredito que a vida útil é superior ao do E-Vent que parece ser um motor BDC com engrenagens planetárias enquanto o motor de vidros é um sem-fim com coupler interno, próprio para reversões.

O motor de vidros tem mecânica espelhada para lado direito e esquerdo, prever a montagem de qualquer tipo no hardware.

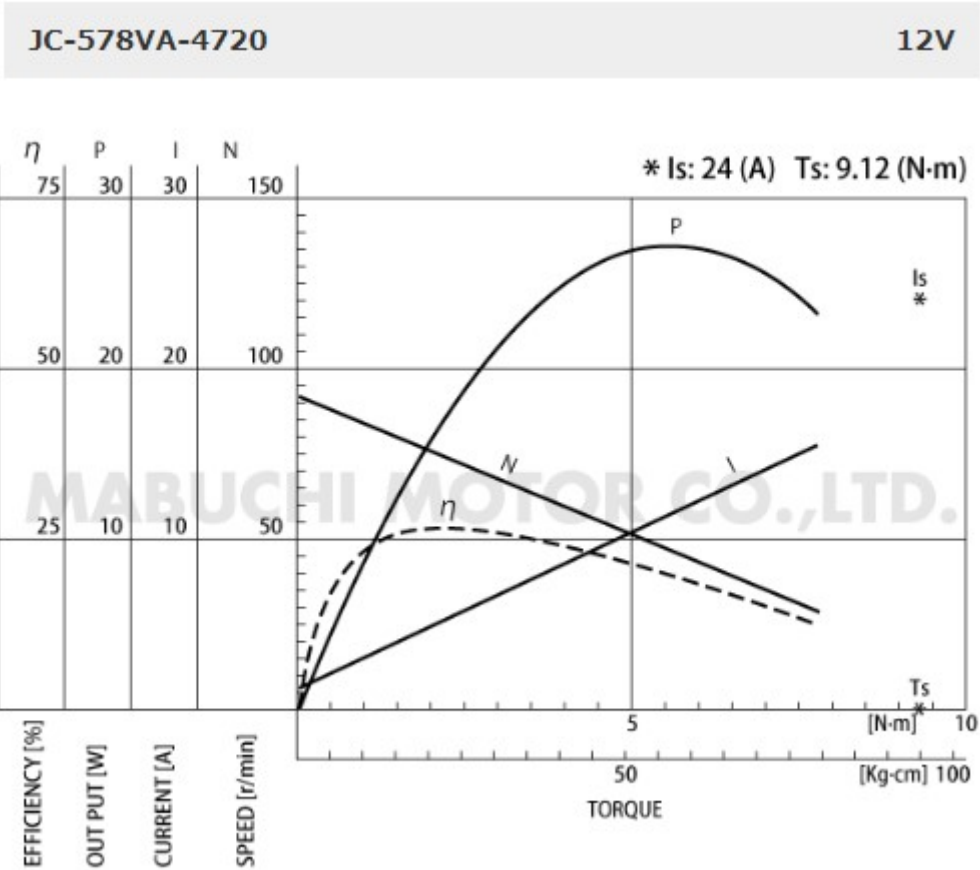
O motor de vidros não dispõe de encoder, deve ser adicionado externamente um sensor. Uma possibilidade é o “speed sensor” utilizado em automóveis. Caso a adaptação mecânica não seja possível, verificar o número de pulsos por revolução, extrair o ímã multipolar, colá-lo em alguma parte girante e posicionar o sensor hall de forma adequada.

MODEL	VOLTAGE	NO LOAD		STALL		WEIGHT	
	RATED	SPEED	CURRENT	TORQUE		g	
	V	r/min	A	N·m	kg·cm		A
 ▶ JC-578VA-4720 Standard Specifications	12	92	1.3	9.1	93.0	24.0	508



▶ JC-578VA-4720
Standard Specifications

Motor do EAR: Mabuchi, utilizado nos carros de passeio



Curva de Torque



Outra opção de acompanhar o curso do motor (neste caso do braço compressor) é utilizar uma bóia de tanque automotivo para prover posicionamento. A resolução é baixa, especialmente nos pontos correspondentes a $\frac{1}{4}$, $\frac{1}{2}$ e $\frac{3}{4}$, onde existem deformações para passar nas inspeções de recebimento das montadoras. A resistência é, a grosso modo, linear. Não sugerimos a utilização de potenciômetros devido ao rápido desgaste da pista.

- o projeto EAR não dispõe de monitor de pressão neste primeiro momento, sugerimos um monitor mecânico ou eletrônico. Pode ser agregado um display ao projeto posteriormente, via serial ou SPI. O sensor de pressão está previsto, mas com P/N TBD.

- para acionar o motor não utilizamos o motor driver shield, por disponibilidade. Usamos relés na versão mais simples e na versão mais complexa utilizamos mosfets discretos, dos tipos utilizados em armas Airsoft, ECUs e amplificadores automotivos. Quando usamos relés controlamos a pressão máxima através de uma fonte externa, ajustável, que varia a tensão aplicada ao motor. Os mosfets trabalham com PWM variável.

- No E-Vent temos a mistura de Ar e O₂ (FIO₂) realizada externamente e aplicada diretamente ao AMBU. No EAR temos a mistura realizada pelo EAR, mediante controle de válvulas pneumáticas de caminhões 12V.

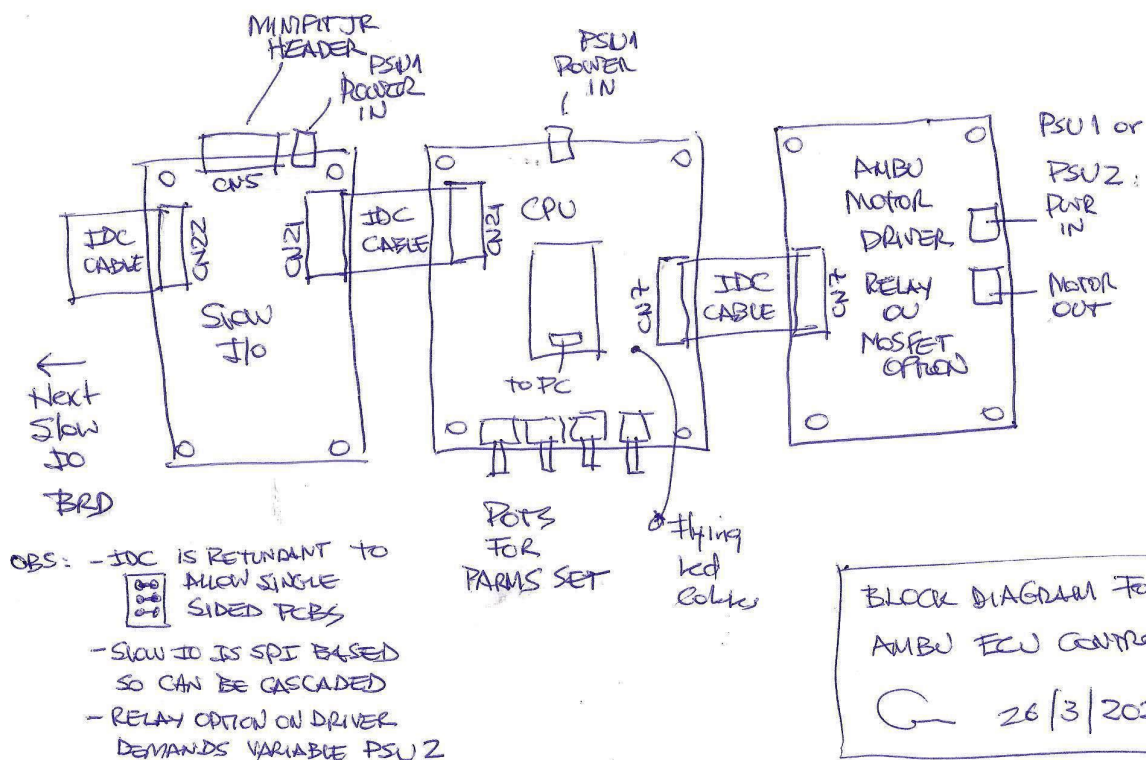
- o Projeto EAR tem expansão de IOs, por módulos. Cada módulo proporciona 8 entradas para sensores NPN e 8 saídas para relés com acionamento NPN.

Vamos então discorrer sobre os diversos componentes do hardware. Serão acrescentadas referências de fabricantes ou distribuidores locais, na medida do possível.

7.1. ECU EAR

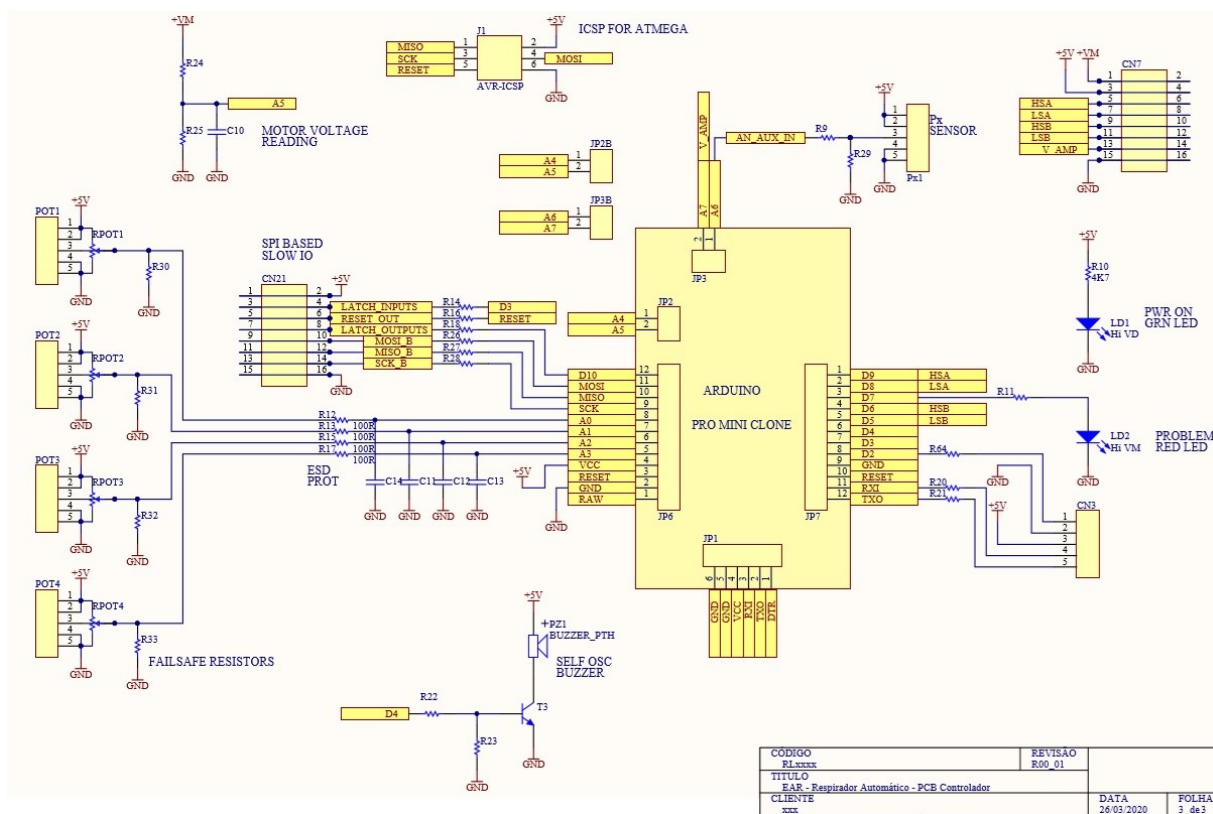
A ECU EAR, por se tratar de um projeto inclusivo, foi projetada com placas de circuito impresso face simples, com tecnologia PTH.

Em função de disponibilidade de PCBs em placas 10x10cm nos varejo, a ECU EAR foi dividida em 3 placas: CPU, IO Lento e Driver do Motor



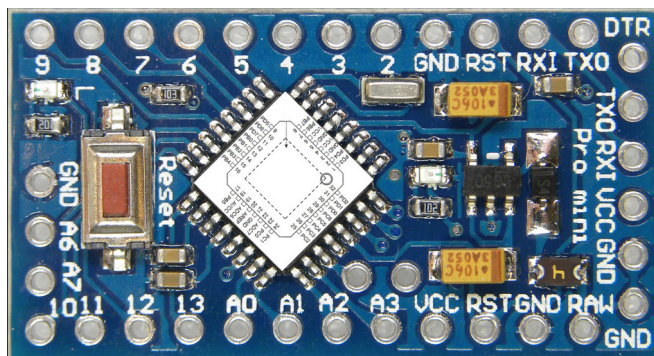
7.1.1. CPU ECU EAR

A CPU ECU contém o Arduino, a fonte de 5V interna, potenciômetros, leds e conectores para as demais placas. A conexão entre as placas é realizada por IDCs (flat cables) que podem ser obtidos de computadores.



7.1.1.1. Arduino

Para informações, favor consultar os documentos de referência. O Arduino referência utilizado para este projeto foi um clone do Arduino Pro Mini 5V 16 MHz, mostrado abaixo.



Outros Arduinos podem ser utilizados, desde que respeitadas as funcionalidades e o firmware seja contruído ou adaptado para a variação escolhida. Para contornar a baixa disponibilidade de IOs do Arduino Pro Mini, implantamos um expensor de IO baseado em SPI, implementado por CIs 74HC595 e 74HC165. Este expensor em sua versão mais básica disponibiliza 8 saídas para

acionamento NPN com capacidade para acionamento direto de relés e 8 entradas para sensores NPN, contato seco de relés. A interface de expansão pode ser multiplicada, mas é limitada na prática pela degradação dos sinais SPI. Já implementamos anteriormente com sucesso 16+16 IOs.

Tabela de PINOUT e funcionalidade do Arduino. Todos os sinais GPIO são ativos em +5V exceto se especificado em contrário.

PINO	FUNÇÃO PINO	OBSERVAÇÕES
TXO	TXO	Tx Bootloader, monitor serial ou TxD Rede
RXI	RXI	Rx Bootloader, monitor serial ou RxD Rede
RST	/RESET	Reset geral, para acionamento usar open collector NPN
GND	GND	GND
D2	GPIO	Reservado Rede_RTS
D3	GPIO	Latch Inputs expensor de IO – SH /LD 74HC165
D4	GPIO	Buzzer OUT
D5	GPIO	Motor Low Side B
D6	PWM	PWM Motor Hi Side B
D7	GPIO	Checklamp LED Vermelho
D8	GPIO	Motor Low Side A
D9	PWM	PWM Motor Hi Side A
D10	GPIO	Latch Outputs expensor de IO – RCK 74HC595 +Edge
D11	MOSI	MOSI expensor de IO
D12	MISO	MISO expensor de IO
D13	SCK	SCK expensor de IO – normalmente ligado a um LED
A0	AN0	Leitura Analógica POT1
A1	AN1	Leitura Analógica POT2
A2	AN2	Leitura Analógica POT3
A3	AN3	Leitura Analógica POT4 ou futura entrada Multiplex AN
VCC	VCC	+5V
RST	/RESET	Reset geral, para acionamento usar open collector NPN
GND	GND	GND
RAW	RAW	Não Conectado – Recomendado remover o regulador interno do Arduino.

A7	AN7	Leitura Analógica Corrente do Motor
A6	AN6	Leitura Analógica Elemento Externo (ex: Sensor Pressão)
A5	AN5	Leitura V_Motor após PPTC
A4	AN4	Reservado

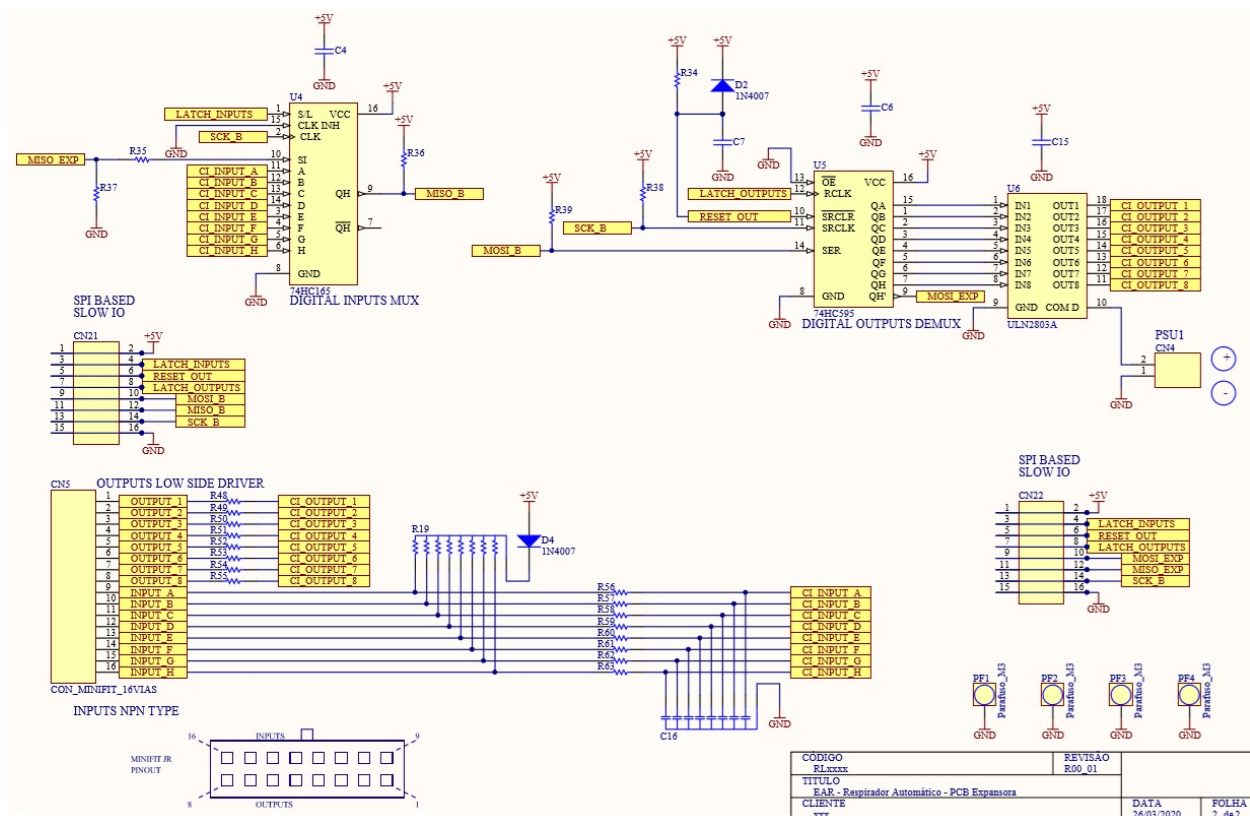
O pinout para conexão para interface serial/PC via conversor USB TTL será utilizado diretamente para gravação a partir de Bootloader. Optamos por este tipo de Arduino sem conversor USB TTL embutido porque podemos implementar posteriormente uma rede local com fio, provavelmente com isolamento galvânica, para interligar os módulos Arduino.

Como não há padronização nos clones de Arduino quanto ao pinout de A4 a A7, estes sinais serão disponibilizados na placa fora do footprint do Arduino, para conexão mediante fios que sugerimos que sejam realizados com “wire-wrap” por baixo da placa

7.1.2. Módulos de IOs Lentos

- o Projeto EAR tem expansão de IOs lentos, via SPI, por módulos. Cada módulo proporciona 8 entradas para sensores NPN e 8 saídas para relés com acionamento NPN. Denominamos lentos porque não serão frequentemente lidos ou atualizados, as taxas vão depender da implementação de software específica, mas penso que devem ficar no **range de 1 a 10 ms entre cada atualização. Um dos gargalos importantes é a leitura dos sensores que promovem o desligamento dos motores, quanto maior o tempo entre leituras, mais o motor irá se deslocar.** Cuidado deve ser tomado para ter bufferização de todas estas saídas em software, porque as leituras das entradas sobreescrevem, SEMPRE, as saídas. Embora as saídas necessitem de um sinal de “Latch_Outputs” para serem efetivadas, **um pequeno ruído elétrico pode promover isto.**

No primeiro módulo de IO, denominado IO1, de uso mandatório, várias destas entradas estão alocadas, seguem exemplos, a serem posteriormente detalhados:



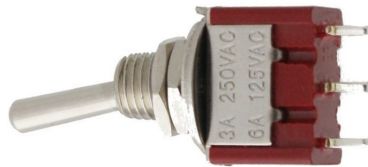
Já existem atribuições pré-definidas para entradas e saídas. Caso seja necessário trabalhar com entradas e saídas fora do até então previsto, por necessidades específicas, recomendamos que sejam utilizadas as últimas entradas e saídas existentes.

7.1.2.1. Entradas INx

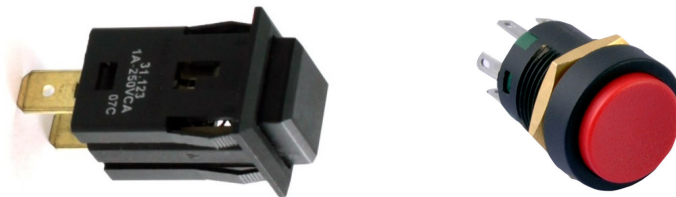
As entradas lentas são entradas NPN, comutadas a GND, níveis de tensão HCMOS (se usado 74HC165) ou TTL (se usado 74HCT165). Podem estar abertas ou conectadas a GND. Tensões acima de +5V não são recomendadas e podem causar queima do 74HC(T)165, apesar de que existem resistores para limitar a corrente dos diodos de clamp das entradas do 74HC(T)165.

Entradas já definidas:

- IN1A: Chave SPST ou SPDT CONFIG/OPERATION, que define se estamos configurando ou operando o sistema. Segue um exemplo.



- IN1B: pushbutton START/SET, define se começa (operação) ou aceita config (configuração). Seguem exemplos.



- IN1C: FDC (Fim de Curso) motor recolhido (AMBU em repouso), sugerimos utilizar duas chaves alavanca em série ou sensor hall + ímã. Cuidado: a operação pode ser diferente. Sugerimos que o estado seja programável porque o Hall será sensível na presença do ímã e as chaves podem operar assim também, mas se trabalhassem NF poderíamos perceber problemas com o cabo. Segue um exemplo.



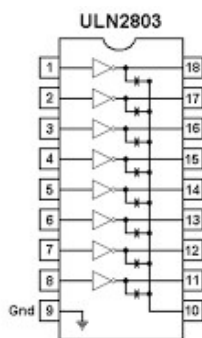
- IN1D: FDC motor expandido (AMBU totalmente comprimido), sugerimos utilizar duas chaves alavanca em série ou sensor hall + ímã. Cuidado: a operação pode ser diferente.

Sugerimos que o estado seja programável porque o Hall será sensível na presença do ímã e as chaves podem operar assim também, mas se trabalhassem NF poderíamos perceber problemas com o cabo.

- IN1E: FDC BAG FIO2 Ar+O2 expandido (BAG quase cheio), sugerimos utilizar duas chaves alavanca em série ou sensor hall + ímã. Cuidado: a operação pode ser diferente. Sugerimos que o estado seja programável porque o Hall será sensível na presença do ímã e as chaves podem operar assim também, mas se trabalhassem NF poderíamos perceber problemas com o cabo.

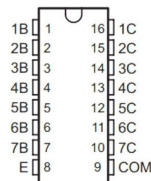
7.1.2.2. Saídas OUTx-x

As saídas foram implementadas com ULN2803 ou ULN2003, que são drivers bipolares padrão com a maior disponibilidade. Permitem acionamento direto de relés, com as bobinas ligadas a V+ (tensão de alimentação positiva) e a saída correspondente. Estes drivers são especificados para 500mA, mas a potência tem que ser observada. Caso a corrente seja insuficiente, podemos colocar drivers em piggyback ou, melhor ainda, utilizar relés auxiliares.

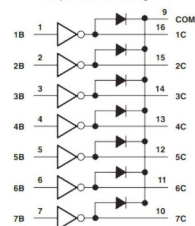


ULN2003

D, N, NS, and PW Package
16-Pin SOIC, PDIP, SO, and TSSOP
Top View



Simplified Block Diagram



Dispositivos acima de 12V (ex: 24V) podem ser utilizados, mas não podem ser aplicados diretamente nas saídas OUTX-X porque estas saídas tem diodos de “kickback” para a fonte de 12V. Neste caso, sugerimos utilizar uma fonte extra de 24V e um relé auxiliar de 12V para comutar o dispositivo em 24V. Existem muitos relés auxiliares de fabricação local. Segue exemplo.



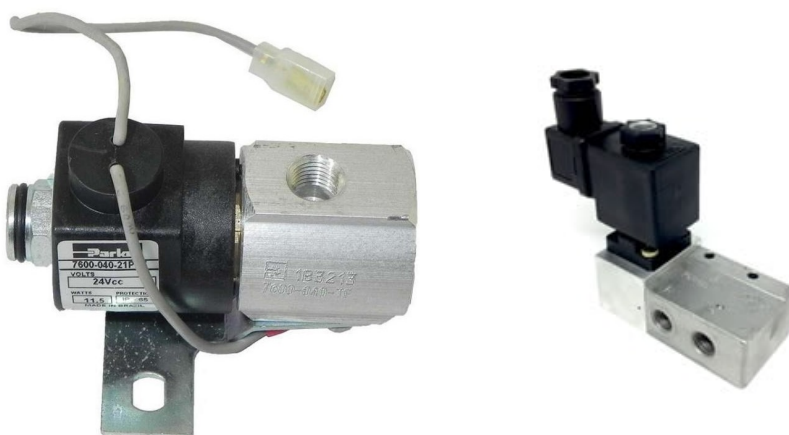
Saídas já definidas:

- OUT1-1 reservado. Pode não ser implementado no hardware, em função de disponibilidade do ULN2803, pode ser substituído pelo ULN2003 que tem somente 7 saídas.

- OUT1-2 acionador de sirene, ex: sirene automotiva 12V FKS ou similar. Necessário abafador mecânico adicional. Ex: almofadas aplicadas ao cone.



- OUT1-3 acionador eletroválvula O2 (para controle de FIO2), eletroválvula para ar comprimido de caminhões 12V. Ref Fluair 104-SL 1/4 12VCC, Thermoval tem diversos tipos, Parker tem várias possibilidades.



- OUT1-4 acionador eletroválvula Ar comprimido (para controle de FIO2), eletroválvula para ar comprimido de caminhões 12V, mesma válvula que a anterior. As saídas OUT1-3 e OUT1-4 podem também ser monitoradas externamente e eventualmente podem ser elementos de um sistema de watchdog muito lento, que poderia desligar todo o sistema.

7.1.2.3. Potenciômetros para ajustes

Os potenciômetros para o projeto EAR são os mais comuns encontrados, lineares, 10Kohms, encontráveis sob o nome genérico de B10K. O valor do potenciômetro não é crítico, valores na faixa de 220 Ohms a 47K devem funcionar. Como o Pot está ligado em +5V, GND e o wiper vai para o ADC, a leitura é ratiométrica. Nota ao pessoal de SW: não assumam que são lineares, porque o modelo montado será o encontrado, linear ou log, 1K, 5K, 10K, 22K... mas não importa muito porque quem fecha o laço de controle é o operador humano, que vai corrigir não-linearidades girando o botão.



Estes potenciômetros possuem dupla função, dependendo do modo de trabalho. O modo é definido por IN1A em GND (Configuração) ou aberto (Operação).

Significado dos potenciômetros em modo Operação:

- P1 define BPM, respirações por minuto
- P2 define volume Tidal, volume de ar+O₂ bombeado ao paciente
- P3 define taxa I/E, quociente tempo inspiração/tempo expiração
- P4 define limite de pressão máxima aplicada

Na operação o curso dos potenciômetros é restrito em 20% a 80% por questões de confiabilidade. O FMEA preliminar mostrou que os potenciômetros podem falhar e estas falhas provocam leituras próximas a 0 ou 100% (5V). O comportamento nas faixas de 0-10% e 80-100% é TBD. Opções seriam o último valor válido lido ou presets fixos pré programados (ex: 50%, fixo). De qualquer forma, alarmes serão gerados no BIP local e na sirene externa OUT1-1. Bloqueios mecânicos de curso podem ser considerados, como parafusos limitando o bico de “chickenhead pots”. Isto evita que em pânico o profissional de saúde use o potenciômetro fora da área útil. Recomendamos efetuar marcações de range máximo e mínimo, se possível.



Significado dos potenciômetros em modo Configuração:

- P1 BPM (fixo)
- P2 TBD (fixo?) Volume Tidal ?
- P3 define o tempo de acionamento da válvula de ar comprimido para controle de FIO₂ (0-200ms), partindo do princípio que o tempo de acionamento da válvula de O₂ é fixo, correspondente a 400ml (400ms). Lembrando que leitura zero seria 20% de curso.
- P4 TBD (fixo?)

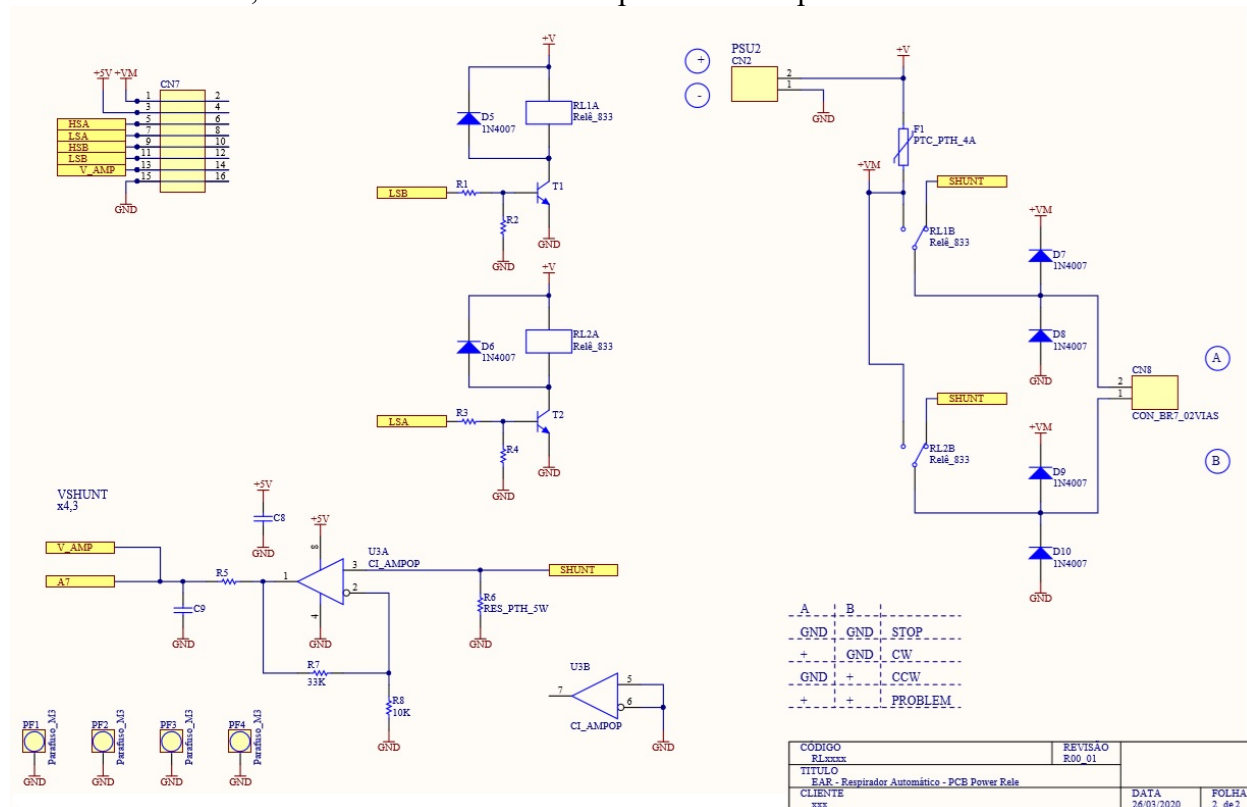
TBD: Quando em modo Configuração, os Pots serão aceitos mediante um comando “SET” gerado a partir de um botão ligado entre IN1B e GND. Talvez o botão altere dinamicamente o significado dos potenciômetros de múltipla função. Exemplo: apertando o botão P3 passa a definir o tempo de acionamento da válvula de ar comprimido para controle de FIO₂, armazena este valor em memória não volátil e ao soltar o botão retorna a função original. OU os pots só são atualizados em função do modo mediante um “enable”. Isto não é intuitivo para o profissional de saúde, o melhor seria se pudéssemos ter mais potenciômetros.

7.1.3. Driver do Motor

O driver do motor faz o acionamento do motor e retorna informação de corrente e tensão do motor para a CPU ECU EAR. Foram previstos dois modos de acionamento, via relés e mosfets. Em ambos os casos, a corrente do motor pode ser lida pela CPU mediante um shunt de 100mR, 2512 ou PTH, e uma amplificação por um amplificador operacional. A proteção de curtos é híbrida, ou seja, opera por hardware por períodos pequenos de tempo (alguns milissegundos para os mosfets) e se faz necessário que o software aborte acionamento em caso de curto prolongado.

7.1.3.1. Driver do Motor utilizando relés

O driver que utiliza relés é o mais simples, mas exige uma fonte de alimentação adicional, que pode ser uma fonte de bancada. Possui limitações, como a corrente do motor é definida somente pela tensão de alimentação da fonte adicional, ou seja, não há controle dinâmico de corrente, nem controle da pressão resultante. As vantagens são a simplicidade e facilidade de obtenção e troca dos relés padronizados T73. Recomendamos relés de fabricantes homologados ISO/TS 16949, mas na falta destes, usar relés como o da foto e prever menor período entre trocas.



Relé T73

7.1.3.2. Driver do Motor utilizando mosfets

Este driver ainda está em WIP. Utiliza mosfets canal N, low level drive, preferencialmente automotivos. A versão apresentada usa mosfets canal P e N, estamos trabalhando para utilizar somente canal N, em DPAK ou TO220. O controle é por PWM no HiSide e GPIO no low side da Ponte H. A tensão do shunt é amplificada para envio à CPU. Em caso de curtos, é importante: detectar, limitar o tempo em curto de acordo com a SOA do mosfet, registrar a falha, sinalizar o problema, desligar se exceder o tempo programado, esperar esfriar e tentar religar novamente. Maiores detalhes serão apresentados no EAR System Developer's SW Guide, a ser publicado.

