

**FUNDAÇÃO ESCOLA TÉCNICA LIBERATO SALZANO VIEIRA DA CUNHA**  
**CURSO TÉCNICO DE ELETRÔNICA**

**ANTÔNIO LIMA DELLA FLORA, 01**  
**HENRIQUE UTZIG, 14**

**TURMA 4411**

**Softstarter**

**Disciplinas: Potência & Micros**

**Professores: Irineu Ronconni & Marcos Zuccolotto**

**Novo Hamburgo, 11 de setembro de 2018**



## **INTRODUÇÃO**

O presente projeto foi desenvolvido para a disciplina de Eletrônica de Potência e Sistemas Microprocessados II, e consiste na criação de um softstarter microcontrolado para o acionamento de cargas monofásicas.

## **PARTES DO PROJETO**

### **MICROCONTROLADOR**

Para o projeto foi utilizado um STM32F411, por ser o microcontrolador trabalhado no curso.

### **Detector de zero**

O detector de pulso irá gerar um pulso sempre que a senóide da rede passar por 0 volts. O sinal é, então, inserido no pino PA1, que é ligado ao Timer 2.

### **Disparo**

Para fazer a lógica de disparo, foi configurado o pino PA1 como trigger do Timer 2, de modo que ele inicie a contagem junto à rede. Foi configurado um PSC de 84, de modo que o timer incremente 1 a cada microssegundo. Aliado a isso, o código altera o valor de ARR e CCR conforme o ângulo de disparo necessário, criando, assim, um sinal PPM.

Para fazer o disparo, foi utilizado o MOC3021, que é um optoacoplador com um triac, facilitando assim o disparo do nosso triac principal, o TIC226. Como encontramos problemas no início dos testes, conferimos o datasheet do componente e ele recomenda adicionar um snubber quando forem utilizadas cargas indutivas.

### **Isolação**

A fim de garantir uma maior segurança, todas as partes conectadas à rede estão isoladas opticamente, seja por meio de um 4N26 ou um MOC3021.

### **Relé de Bypass**

O relé de bypass é ativado por uma saída do ARM que vai para 1 quando a carga está totalmente ligada. Ao ligá-lo, ele irá curto-circuitar o triac, impedindo, assim, que o TIC dispare. Caso a corrente exceda 200% o relé irá desligar, permitindo o desligamento da carga.

## **Medição de corrente**

Para realizar a medição de corrente foi utilizado um transformador de corrente com uma relação de 1000 para 1. O transformador foi conectado em paralelo com um resistor de 100ohms. A tensão observada no resistor era proporcional à corrente. Contudo, a tensão era muito baixa, fazendo com que fosse necessário utilizar um retificador de precisão para amplificar e retificar o sinal sem a queda dos diodos.

Depois de retificado o sinal é conectado ao PA4, que é uma das entradas analógicas do ARM. A fim de determinar a corrente RMS, foi utilizado DMA e um timer, para fazer a medida de tensão 16 vezes a cada semiciclo. Esse valor era armazenado 32 vezes e feita a média dos mesmos, e no final era obtida a tensão RMS do sinal, que era proporcional à corrente.

Como a corrente nominal do motor utilizado era conhecida, fizemos um sistema de indicação através de LEDs. O verde significa uma corrente abaixo de 150%, o amarelo entre 150 e 200% e o vermelho acima de 200%.

## **Medição de RPM**

A fim de medir a rotação do motor por minuto foi utilizado um disco com uma fenda e uma barreira de infravermelho, para produzir um pulso a cada revolução do motor. Esses pulsos eram contabilizados por um timer que teve o seu contador conectado a um pino e, a cada 1 segundo, tinha o seu registrador lido e apagado, de maneira a contar novamente.

## **GITHUB**

Para tornar esse projeto algo durável e que não fosse desaparecer após a apresentação, o grupo teve a ideia de utilizar o github, de modo a ir postando o seu código na plataforma após cada alteração. Isso mostrou-se muito interessante, pois nos permitiu ver nosso próprio progresso. O link do nosso projeto é o seguinte: <https://github.com/dfantonio/Softstarter-using-ARM-STM32F411> e o da interface serial é <https://github.com/henriqueutzig/GUI-Softstarter>.

## **IHM**

A interface de usuário foi desenvolvida em C# utilizando a IDE Visual Studio. Nessa interface temos os parâmetros e conexão serial com o microcontrolador, o ajuste do tempo de subida e descida e o monitoramento da Irms, Vrms e RPM do motor. Além disso ainda possuímos uma caixa de texto para debugar caso o

microcontrolador mande um erro via serial e uma caixa de comando serial também para debug.

Softstarter auf Deutsch

Serial

COMs:  Baudrate:

Tempos de descida e subida

UP  s DOWN  s

Vrms lms RPM

Vrms =  lms =  RPM =

configure a porta COM