



Universidade Federal  
de Campina Grande

Universidade Federal de Campina Grande  
Departamento de Engenharia Elétrica  
Sistemas a Eventos Discretos  
Prof. Kyller Costa Gorgônio

**Documentação Projeto 1**  
**Sistema de controle de tráfego para um cruzamento de**  
**quatro vias utilizando a ferramenta Supremica**

Isac Gomes de Almeida Silva - 118210312  
Roberto da Silva Dourado Vasconcelos - 118210478

Campina Grande - PB, Abril de 2024

## Sumário

1	Introdução . . . . .	1
2	Visão Geral do Sistema . . . . .	2
	2.1 Estados . . . . .	3
	2.2 Eventos . . . . .	3
3	Projeto do Sistema Utilizando o Supremica . . . . .	4
4	Testes Simulador Supremica . . . . .	8

## **1 INTRODUÇÃO**

O Projeto se trata da implementação de um controle de tráfego para um cruzamento utilizando autômatos finitos. Com o foco em desenvolver um sistema eficiente que gerencie o fluxo de veículos e pedestres, minimizando o tempo de espera e evitando congestionamentos. Além de Garantir a segurança de todos os usuários da via, incluindo veículos e pedestres.

O sistema deve controlar os semáforos para veículos e pedestres e promover passagem segura pelas vias, evitando colisões e minimizando o tempo de espera. Ou seja, os semáforos não devem entrar em conflito Ex.(Amarelo/Amarelo , Verde/Verde, Verde/Amarelo).

## 2 VISÃO GERAL DO SISTEMA

O modelo a seguir foi imaginado a partir dos objetivos e especificações solicitadas. E usado como base para a implementação do sistema real.

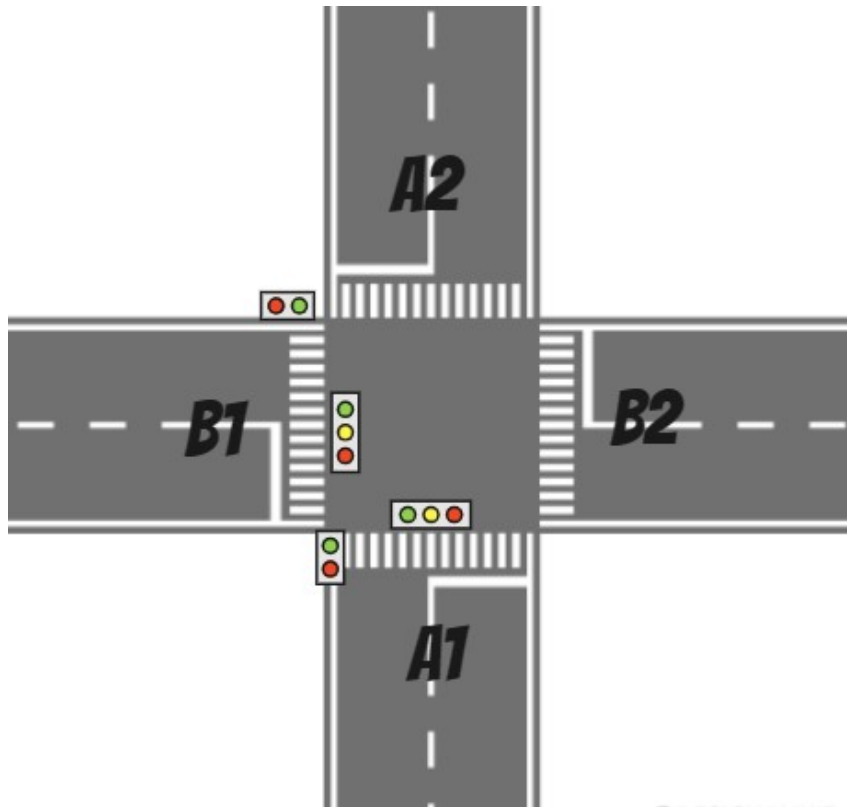


Figura 1 – Modelo do Cruzamento

Os carros podem se aproximar do cruzamento pelas vias A1 ou B1 e seguirem para as vias A2 ou B2. Por exemplo, eles podem vir de A1 e seguirem para A2, ou virem por A1 e irem para B2. Ou chegaram de B1 e irem para B2 (ou para A2). Eles sempre chegam por A1 ou B1 e sempre seguem para A2 ou B2. O sistema possui um total de 4 semáforos, 2 para carros (verde, amarelo, vermelho) e 2 para pedestres (verde e vermelho).

Como o sistema deve utilizar autômatos finitos para modelar o comportamento dos semáforos e o fluxo de tráfego e pedestres. O primeiro passo é definir os estados e eventos a serem criados

## 2.1 ESTADOS

Os estados são representados como verde, vermelho e amarelo. A ordem dos semáforos devem seguir o seguinte esquema, para atingir as metas de segurança.



Figura 2 – Estados Previstos

## 2.2 EVENTOS

Eventos controláveis: A Mudança de estado dos semáforos para veículos, alterar a luz de um semáforo de verde para amarelo, de amarelo para vermelho, ou de vermelho para verde. É uma mudança programada. Ativação/desativação dos semáforos para pedestres. Seguindo o oposto dos semáforos para carros. De forma periódica, também é controlável.

Eventos não controláveis: Solicitação do botão de pedestre, o pedestre pressionando o botão para solicitar a travessia. É um evento que não pode ser controlado.

Emergências e eventos inesperados, Situações como a passagem de veículos de emergência, acidentes ou falhas nos semáforos são exemplos de eventos não controláveis.

### 3 PROJETO DO SISTEMA UTILIZANDO O SUPREMICA

O Supremica foi o software utilizado para implementar e simular o projeto descrito a seguir. Se trata de uma poderosa ferramenta de software projetada para modelar, simular e verificar sistemas dinâmicos, como autômatos finitos, sistemas de controle e redes de Petri. Oferece uma interface intuitiva e uma variedade de recursos avançados para facilitar o desenvolvimento e a análise de sistemas complexos. É disponibilizado gratuitamente em <https://supremica.org/>.

O primeiro passo para criar os autômatos foi definir os estados (verde, amarelo e vermelho) para o caso apresentado abaixo. E os eventos, controláveis e não controláveis.

O temporizador que existe no cenário real foi simulado como sendo as passagens controláveis (GtoY, RtoG e YtoR) e as entradas não controláveis representam a solicitação feita por um pedestre no semáforo P1 ou uma emergência. A emergência leva o automato para um estado vermelho como segurança para todos no cruzamento.



Figura 3 – Eventos Semáforo 1

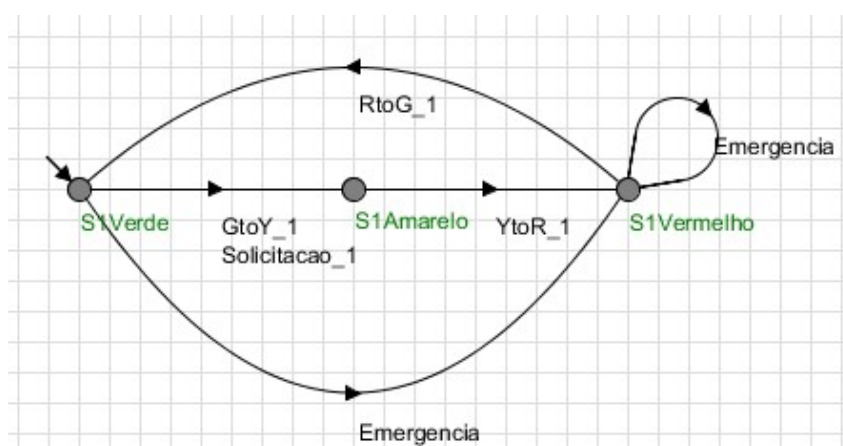


Figura 4 – Automato Semáforo 1

O automato para pedestre possui o seguinte formato. Estado vermelho nos momentos de emergência e quando o semáforo 2 passar para vermelho. verde no cenário contrário. Utilizando a tática de espelhar o semáforo 2 para o obter o funcionamento correto de pedestre 1, podemos obter com mais facilidade o estado vermelho,vermelho,vermelho,vermelho.

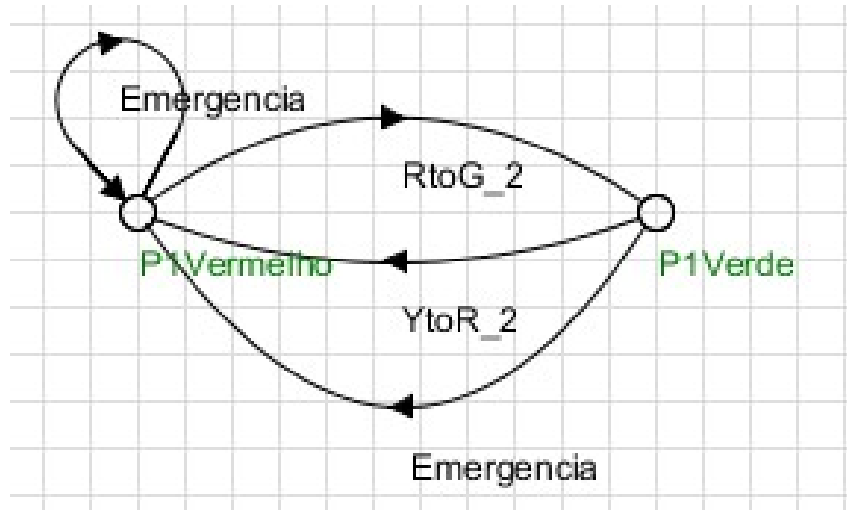


Figura 5 – Automato Semáforo Pedestre 1

Os autômatos para o Semáforo 2 e pedestre 2, possuem comportamento análogo. Iniciando em estados inversos a seus pares.

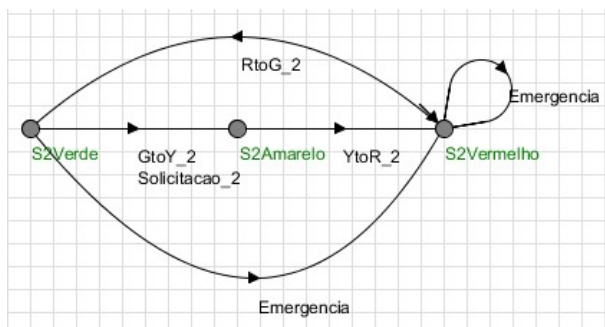


Figura 6 – Automato Semáforo 2

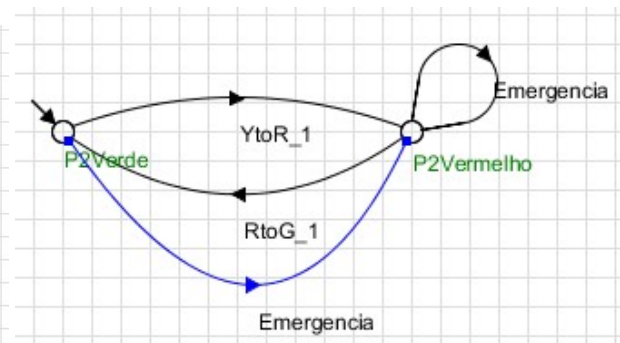


Figura 7 – Automato Pedestre 2

Com todos os autômatos criados, o próximo passo é realizar a síntese de supervisores. Que envolve a criação de um modelo formal que represente tanto a planta (o sistema a ser controlado, neste caso, o cruzamento) quanto as especificações de controle desejadas (regras de segurança, eficiência no fluxo de tráfego, etc.). Nessa etapa podemos bloquear os estados indesejados no sistema, aqueles que põem em risco a segurança dos usuários. Por exemplo, estados onde os dois semáforos estão verdes.

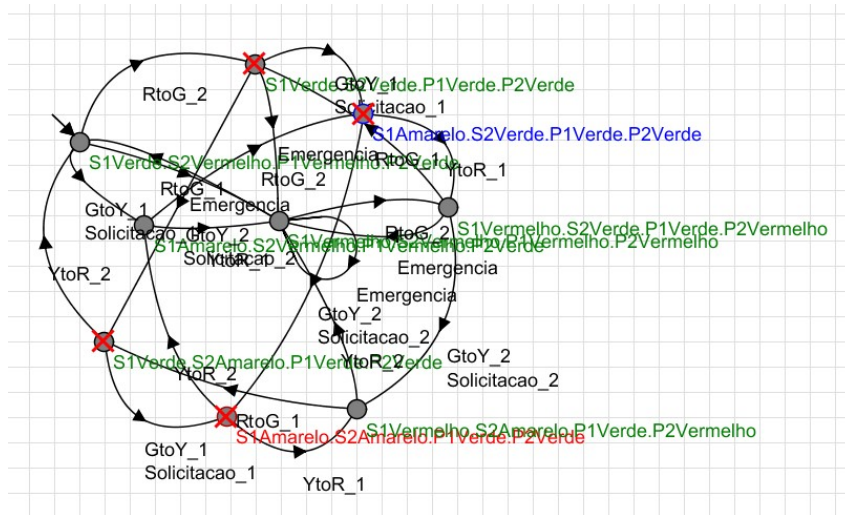


Figura 8 – Planta com estados forbidden

Após isso foi necessário criar uma variável buffer. Para garantir que nenhum dos dois semáforos entrassem em loop, sem liberar a passagem para o outro. Essas variáveis e guardas garantem que o semáforo 1 só pode ficar verde novamente após o semáforo 2 passar pelo ciclo também uma vez.

Editing variable 'buffer'
 ✕

Name:	buffer
Type:	0..1
Initial:	buffer == 0

Figura 9 – Criação da variável buffer

Obs. Ao criar a variável se mostrou necessário alterar o nome dos estados no supervisor. O que resolver o erro apresentado na figura 8.



Após essas etapas O supervisor final representando o sistema completo assume o seguinte formato.

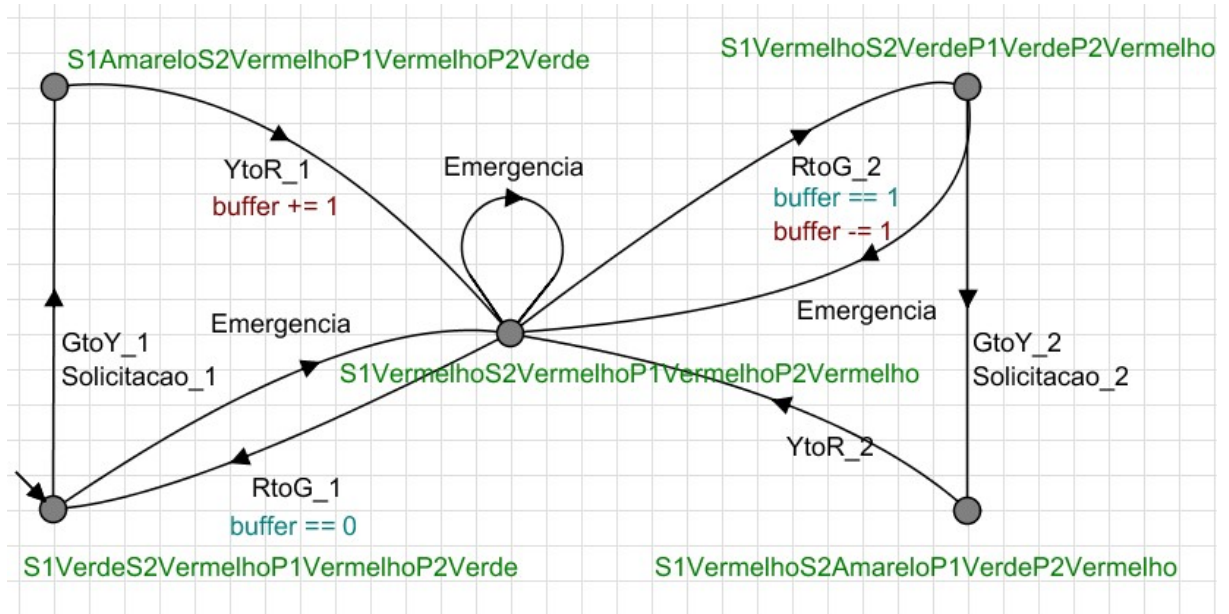


Figura 10 – Sistema Completo

#### 4 TESTES SIMULADOR SUPREMICA

Utilizando o suprmica além de fazer análise de diferentes formas. É possível simular os autômatos gerados. Abaixo temos apresentadas algumas imagens geradas ao longo dos testes de funcionamento.

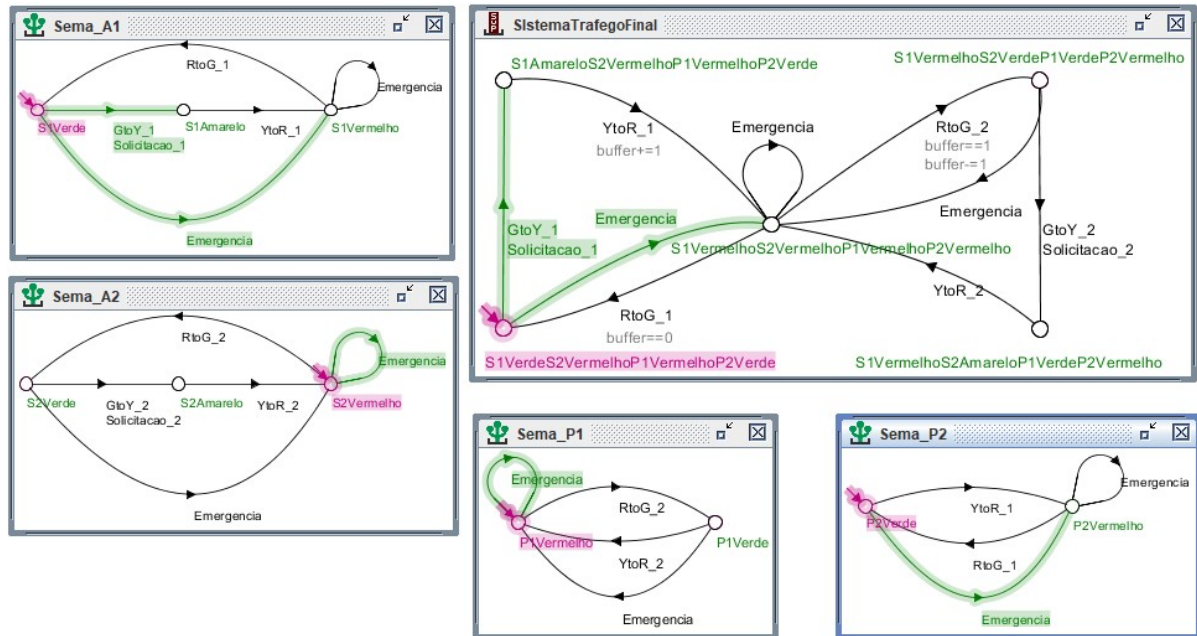


Figura 11 – Simulador Estados Iniciais

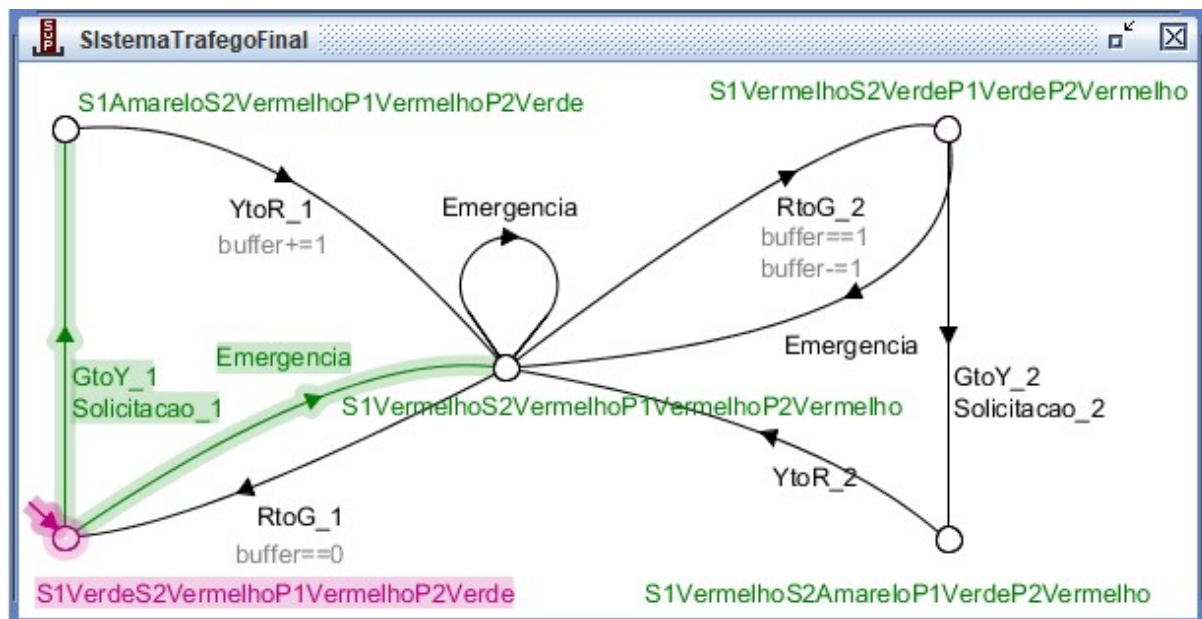


Figura 12 – Supervisor Estados Iniciais

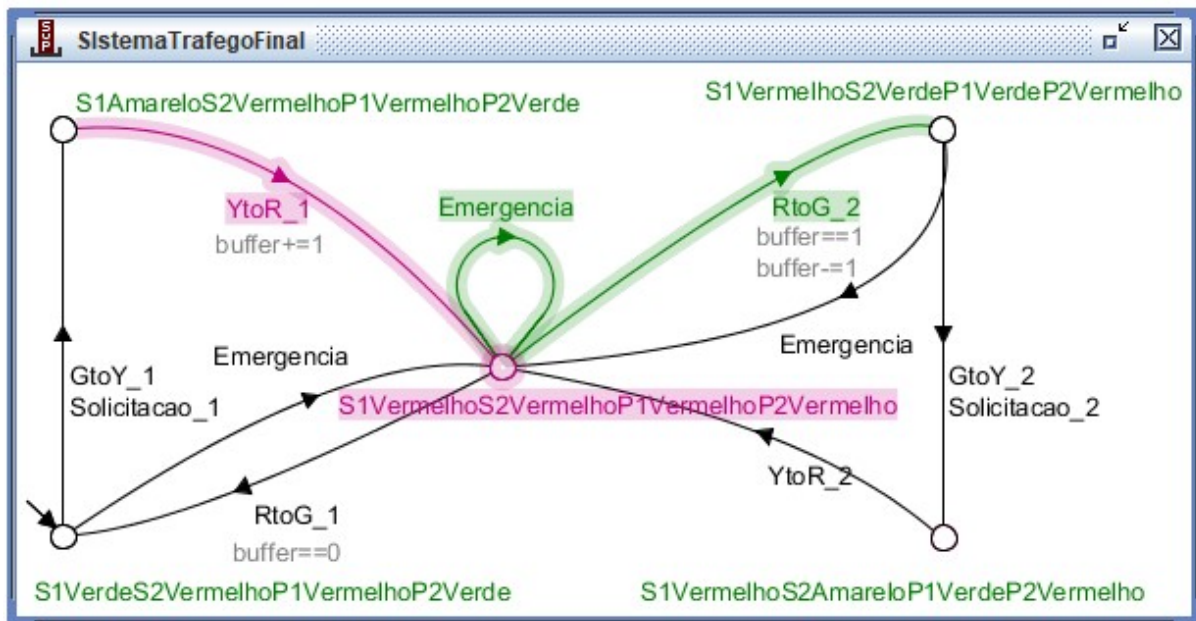


Figura 13 – Estado Intermediário de Segurança

Testes mais detalhados foram apresentados no vídeo enviado em anexo.