

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE INFORMÁTICA

MARCELO TEIDER LOPES
STEFAN CAMPANA FUCHS

DOCUMENTO DE PROJETO

APS

CURITIBA

2014

MARCELO TEIDER LOPES
STEFAN CAMPANA FUCHS

DOCUMENTO DE PROJETO

APS de Sistemas Embarcados apresentada ao Departamento Acadêmico de Informática da Universidade Tecnológica Federal do Paraná como requisito parcial para obtenção dos títulos de “Bacharel em Sistemas de Informação” e “Engenheiro em Computação”.

Orientador: Prof. Douglas Paulo Bertrand Renaux

CURITIBA

2014

LISTA DE FIGURAS

FIGURA 1 – Diagrama da arquitetura funcional do sistema.	5
---	---

SUMÁRIO

1	ARQUITETURA FUNCIONAL	5
1.1	ALOCAÇÃO DAS FUNÇÕES EM HARDWARE E SOFTWARE	6
2	ARQUITETURA FÍSICA	7
2.1	USO DE MEMÓRIA	7
2.2	DIAGRAMA DE OBJETOS	7
3	PROJETO DOS COMPONENTES	8

1 ARQUITETURA FUNCIONAL

O diagrama da Figura 1 expõe uma representação da arquitetura funcional do sistema. As setas representam o sentido dos fluxos de dados entre cada função do sistema.

“Comunicação com o computador” envolve a UART (física e driver), e o protocolo de comunicação do sistema. Esta função é utilizada para a comunicação com o simulador de elevador, que é executada em um computador. “Recebe entrada dos botões” recebe requisições dos botões internos e externos de andares, feitas pelos usuários do elevador. A função “Controla luzes dos botões” é responsável por ligar ou desligar as luzes dos botões de acordo com as requisições recebidas dos usuários e de acordo com o movimento do elevador. “Enfileira requisições” é responsável por organizar as requisições de botões na ordem em que devem ser atendidas. “Controla movimento das portas” efetua a abertura e fechamento das portas, de acordo com a lógica de funcionamento do elevador. “Controla movimento do elevador” é responsável por fazer o elevador subir ou descer para atender às requisições enfileiradas.

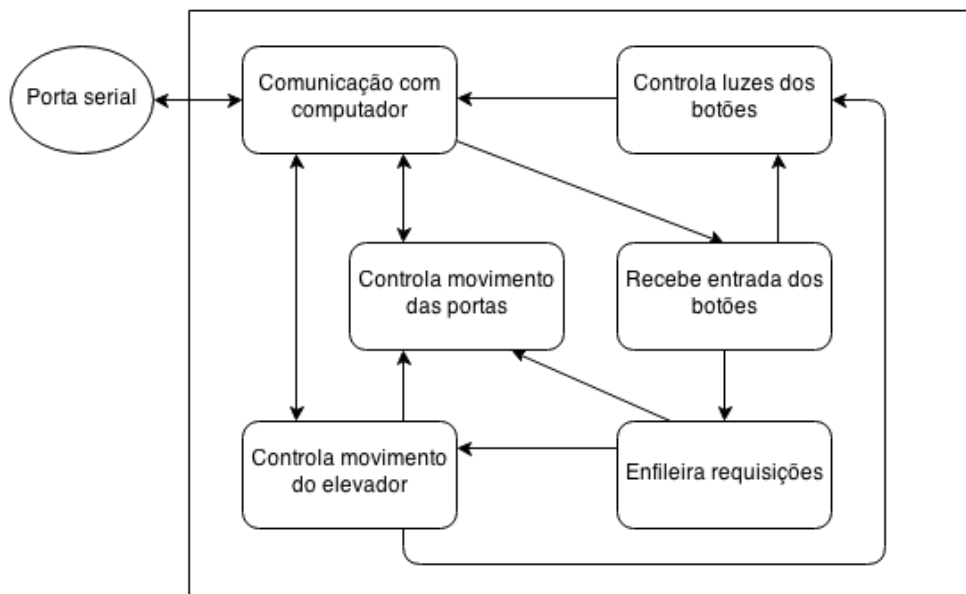


Figura 1: Diagrama da arquitetura funcional do sistema.

1.1 ALOCAÇÃO DAS FUNÇÕES EM HARDWARE E SOFTWARE

Tabela 1: Alocação das funções em Hardware e Software.

Bloco	Hardware	Software
Comunicação com o computador	X	X
Recebe entrada dos botões	-	X
Controla luzes dos botões	-	X
Enfileira requisições	-	X
Controla movimento das portas	-	X
Controla movimento do elevador	-	X

A função de comunicação com o computador constitui-se tanto de hardware quanto de software. A parte de hardware está relacionada ao periférico UART0 do Kit LPC1768. O software relaciona-se ao driver desenvolvido para inicializar e utilizar este periférico, além do protocolo de comunicação do sistema.

O restante das funções é constituída apenas de software, pois controlam a lógica de operação do sistema, não fazendo diretamente o uso de periféricos.

2 ARQUITETURA FÍSICA

2.1 USO DE MEMÓRIA

A Tabela 2 apresenta o uso estimado das memórias RAM e flash do Kit LPC1768 para o desenvolvimento do sistema. Considerou-se na estimativa apenas os componentes que já estavam disponíveis ao projeto anteriormente (CMSIS RTOS e driver da UART). Pela análise, como a maior parte das memórias permanece livre, considera-se que a quantidade é suficiente para o desenvolvimento dos componentes restantes do sistema.

Tabela 2: Uso estimado das memórias do Kit LPC1768.

	RAM (64 kB máx.)	Flash (512 kB máx.)
CMSIS RTOS	2600 B	5400 B
Driver da UART	150 B	800 B
Total ocupado	2750 B (2.68 kB)	6200 B (6.05 kB)
Total Livre	61.32 kB (95.81%)	505.95 kB (98.82%)

2.2 DIAGRAMA DE OBJETOS

3 PROJETO DOS COMPONENTES

- UART: já presente no Kit LPC1768;
- Driver UART: implementado anteriormente, seguindo o especificado em estudo da plataforma;
- ISR UART: implementado em conjunto com o driver da UART;
- Buffer UART: implementado em conjunto com o driver da UART;
- Controlador do elevador: Monitora constantemente os sensores do elevador, e envia comandos aos atuadores de acordo com a lógica de operação do elevador.
- Protocolo de comunicação: Recebe informações do buffer da UART, interpreta o seu significado e repassa-as para a função de tratamento do controlador do elevador; Recebe comandos do controlador, converte-os para uma linguagem aceita pelo simulador e repassa-os para o buffer da UART.