



universidade de aveiro

departamento de eletrónica, telecomunicações e informática

Curso 8309 - Mestrado Integrado em Engenharia Eletrónica e Telecomunicações

Disciplina 41505 – Sistemas Embutidos e de Tempo Real

Ano 2022/23
letivo

Autores:

88919 Rodolfo Reis Oliveira

Turma P1 Grupo 7

Data [28/03/2023]

Docente Paulo Bacelar Reis Pedreiras

Módulo que Processa Comandos Recebidos via UART e Módulo Unity para Teste

Módulo que processa comandos recebidos via UART que permite a deteção de erros que possam existir no comando lido, este módulo apenas conhece dois tipos de comando, tipo P e S. As funcionalidades implementadas por este módulo são: deteção de *string* cheia e vazia, deteção do formato do comando se é o esperado, *# CMD DATA CS !*, deteção de se o conteúdo recebido é o mesmo que o enviado, através da comparação com o byte *checksum*, deteção de comando incompleto, por exemplo, o comando P espera que o DATA tenha 3 bytes para os parâmetros PID, e deteção de um comando introduzido erradamente.

Já o módulo Unity tem como objetivo implementar diferentes envios de comandos que possam testar se o módulo acima está a detetar os erros e os acertos corretamente.

Ajustes e funcionalidades adicionadas Feitos ao Módulo 'cmdproc.c'

- Detecção de um comando diferente de P e S e retorno de '-2', comando invalido foi encontrado, e *reset* da *string*;
- Detecção do formato correto, se a *string* começa com # e termina com !, e se a *string* tem mais do que um # ou !, caso não seja o correto retorna '-4';
- Detecção de *string* incompleta, no comando P espera-se pelo menos uma *string* de tamanho 7, no comando S espera-se pelo menos uma *string* de tamanho 4, se tiver incompleto retorna '-1';
- Detecção de *Checksum*, caso o conteúdo recebido, *CMD+DATA*, não seja igual ao enviado, *CS*, retorna '-3';

Módulo Unity Test 'testcmdproc.c'

- Test_cmdproc_StringFull:
 - Teste para verificar se é detetado que a *string* está cheia, para isso são inseridos 10 elementos através da função newCmdChar() e da qual é retornado sempre o valor 0, de seguida insere-se mais 1 elemento e é esperase que retorne '-1';
- Test_cmdproc_StringEmpty:
 - Teste para verificar se é detetado que a *string* está vazia, chamando a função cmdProcessor() e espera-se o retorno de '-1';
- Test_cmdproc_StringFormat:
 - Teste para verificar se é detetado uma *string* que não segue o formato # *CMD DATA CS !*, verificando se o primeiro e último elemento são respetivamente # e !, para isso primeiro inserimos uma *string* sem começar com # e chamando a função cmdProcessor() espera-se o retorno '-4', posteriormente, reseta-se a *string* para intrudozir uma nova que começa com # mas não termine com ! e chamando a função cmdProcessor() espera-se um retorno '-4';
 - Se na *string* forem inseridos mais do que um # e/ou ! também deve retornar '-4' quando se chama a função cmdProcessor();
- Test_cmdproc_PCScerto:
 - Inserimos uma *string* de formato correto para o comando P e com o *CS* inserido corretamente, ao chamar cmdProcessor() espera-se o retorno '0';
- Test_cmdproc_PCSerrado:
 - Inserimos uma *string* de formato correto para o comando P e com *CS* errado, ao chamar cmdProcessor() espera-se o retorno '-3';
- Test_cmdproc_SCScerto:
 - Inserimos uma *string* de formato correto para o comando S e com *CS* correto, ao chamar cmdProcessor() espera-se o retorno '0';
- Test_cmdproc_SCScerrado:
 - Inserimos uma *string* de formato correto para o comando S e com *CS* errado, ao chamar cmdProcessor() espera-se o retorno '-3';
- Test_cmdproc_NoCommand:
 - Inserimos uma *string* onde temos um formato de comando P e em seguida S mas em vez do campo *CMD* estar P ou S, em cada caso, temos uma outra letra e ao chamar a função cmdProcessor() espera-se o retorno '-2' nos dois caso;
- Test_cmdproc_Pinc:

- Inserimos uma *string* de formato incompleto para o comando P, neste caso sem um dos parâmetros do controlador PID, como o tamanho esperado para uma string do comando P é de 7, caso seja inferior a função cmdProcessor() retorna '-1';
- Test_cmdproc_Sinc:
 - Inserimos uma *string* de formato incompleto para o comando S, neste caso sem o CS, como o tamanho esperado para uma string do comando S é de 4, caso seja inferior a função cmdProcessor() retorna '-1';