Assignment 2

Test-Driven Development – Unity

|  |  |
| --- | --- |
| Nome: | n.mec: |
| Inês Leite | 98490 |
| Bruno Teixeira | 98244 |

Testes realizados:

-Funcionalidade do comando do tipo “P” e envio dos corretos valores;

- Identificação da ocorrência de erro quando a string de comando está vazia;

-Identificação do comprimento máximo da string permitida pela UART;

-Identificação da utilização de um comando invalido;

-Identificação da ausência do inicializador do comando “#”;

-Identificação da ausência do terminador do comando “!”;

-Identificação da ausência de um ou mais parâmetros para o comando “P”;

-Identificação de erro no parâmetro CheckSum;

-Funcionalidade do comando tipo “S” e respetivos erros;

-Teste da funcionalidade de estruturar um comando apartir de uma string e os seus respetivos erros;

- Funcionalidade do comando “A” relativo a processamento por texto;

- Devolução de erro na função stringDebug quando a string do comando está vazia;

Soluções para o sucesso dos testes realizados:

-Para processar o erro de ter uma “string” vazia foi criado uma variável, inicializada a zero, esta é incrementada à medida que é recebido um novo “character” por forma de não só ter conhecimento do comprimento da “string” ao longo do processo, mas também impedir que sejam incrementados mais “characters” do que os permitidos. Com esta abordagem ambos os testes de “string” vazia e comprimento máximo foram bem-sucedidos.

-A identificação de um comando valido é possível ser realizado verificando a segunda posição da “string” armazenada. Se nesta posição não possui um comando aceite pela UART, é devolvido um erro.

-Uma abordagem semelhante à anterior foi seguida em relação à verificação do inicializador e terminador do comando.

-Para verificar o numero de parâmetros realizou-se uma subtração entre uma variável temporária que armazena a posição do inicializador e outra com a posição do terminador, caso o resultado não fosse o esperado, significaria que este não estava corretamente configurado.

-Para a verificação do CheckSum foi comparado o valor recebido com uma soma feita dos parâmetros recebidos, caso não coincidisse devolveria um erro.

-Através das prevenções realizadas anteriormente e como o comando “S” apenas necessita do inicializador, comando e terminador não foi necessário realizar alterações para que este teste fosse bem-sucedido.

-Caso a string introduzida for maior do que o tamanho máximo permitido menos o espaço necessário para o EOF e o SOF devolve um erro, caso contrário é construído o comando correspondente aos dados.

- Ao utilizar a função stringDebug verifica-se que ao passar os parâmetros desejados estes encontram-se corretamente incrementados no array, em seguida é usado uma função com o intuito de automatizar o cálculo do parâmetro checksum e este também ser armazenado da mesma forma.

-Garantiu-se que caso o comando esteja vazio não se pode fazer stringDebug e devolve o respetivo erro;

Funcionalidades acrescentadas:

-Suporte para poder construir um comando a partir de uma string;

-Processamento por texto e respetivas funções auxiliares;

-Suporte para envio de comando de texto;

-Função de debug relativo ao visionamento do array a ser enviado;

-Automatização do cálculo do checkSum;