**Challenge Requirements Document**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Wave** |  |  | **Trainer Name** | **Hossam Adel** |
| **Week Number** |  |  | **Challenge Name** | **Basic Com Module** |
| **Duration** |  |  | **Type** | **Project** |

**Team Size**

2

## **SW/HW environment**:

UART driver based on the ISR and Dev Board.

## **Restrictions (peripherals, configurations, what to use and not to use):**

We going to Design the software using whatever we had in our HW to get the highest throughput.

**Challenge Requirements**

**AGILE\_REQ1:**

Implement BCM ( Basic Com Module), Module has a capability to receive and send stream of data without reach 100% cpu load.

**AGILE\_REQ2:**

Implement BCM\_Init by following the next table :

|  |  |
| --- | --- |
| Function Name | BCM\_Init |
| Syntax | EnmBCMError\_t BCM\_Init (const BCM\_ConfigType \* ConfigPtr ) |
| Sync/Async | Synchronous |
| Reentrancy | Non Reentrant |
| Parameters (in): | ConfigPtr Pointer to a selected configuration structure |
| Parameters (out): | None |
| Parameters (inOut): | None |
| Return: | EnmBCMError\_t | one of predefine enumeration number |

**AGILE\_REQ3:**

The function BCM\_Init shall initialized the hardware UART module.

**AGILE\_REQ4:**

Implement BCM\_DeInit by following the next table :

|  |  |
| --- | --- |
| Function Name | BCM\_DeInit |
| Syntax | EnmBCMError\_t BCM\_DeInit ( void ) |
| Sync/Async | Synchronous |
| Reentrancy | Non Reentrant |
| Parameters (in): | None |
| Parameters (out): | None |
| Parameters (inOut): | None |
| Return: | EnmBCMError\_t | one of predefine enumeration number |

**AGILE\_REQ5:**

The function BCM\_DeInit shall uninitialized the hardware UART module.

**AGILE\_REQ6:**

Implement BCM\_RxMainFunction by following the next table :

<Part of my training let the Sprinters imaging how to implement the function from scratch>

Hint: This table need to be filled be the Sprinters.

|  |  |
| --- | --- |
| Function Name | BCM\_RxDispatch |
| Syntax |  |
| Sync/Async |  |
| Reentrancy |  |
| Parameters (in): |  |
| Parameters (out): |  |
| Parameters (inOut): |  |
| Return: |  |

**AGILE\_REQ7:**

The function BCM\_RxMainFunction shall Run the basic logic code to move the check the status machine code of the Receiving bytes and reconstruct the packet to extract only the user data.

This function part of the super loop code.

**AGILE\_REQ8:**

Implement BCM\_TxMainFunction by following the next table :

<Part of my training let the Sprinters imaging how to implement the function from scratch>

Hint: This table need to be filled be the Sprinters.

|  |  |
| --- | --- |
| Function Name | BCM\_TxDispatch |
| Syntax |  |
| Sync/Async |  |
| Reentrancy |  |
| Parameters (in): |  |
| Parameters (out): |  |
| Parameters (inOut): |  |
| Return: |  |

**AGILE\_REQ9:**

The function BCM\_TxMainFunction shall Run the basic logic code to move the check the status machine code of the transmitting bytes and construct the packet (Header – Data – Check SUM), and the Header contain Command and the data size.

This function part of the super loop code.

**AGILE\_REQ10:**

Implement BCM\_Send by following the next table :

<Part of my training let the Sprinters imaging how to implement the function from scratch>

Hint: This table need to be filled be the Sprinters.

|  |  |
| --- | --- |
| Function Name | BCM\_Send |
| Syntax |  |
| Sync/Async |  |
| Reentrancy |  |
| Parameters (in): |  |
| Parameters (out): |  |
| Parameters (inOut): |  |
| Return: |  |

**AGILE\_REQ11:**

The function BCM\_Send shall invoked by the user to send the stream of data.

This function part of the super loop code.

**AGILE\_REQ12:**

Implement BCM\_Receive by following the next table :

<Part of my training let the Sprinters imaging how to implement the function from scratch>

Hint: This table need to be filled be the Sprinters.

|  |  |
| --- | --- |
| Function Name | BCM\_Receive |
| Syntax |  |
| Sync/Async |  |
| Reentrancy |  |
| Parameters (in): |  |
| Parameters (out): |  |
| Parameters (inOut): |  |
| Return: |  |

**AGILE\_REQ13:**

The function BCM\_Receive shall invoked by the user to receive the stream of data.

This function part of the super loop code.