**Module Requirements of “CheckStatus()”:**

There is a package consisting of two systems---System1, System2. A packet of data is acquired from this package. Acquired data from this package is stored in an array Sys1count[], Sys2count[] for System1 and System2 respectively and treated as current sample. Data is acquired from each system in 3 axes i.e. X, Y & Z.

Hence,

Sys1count[0] represents System1 data in x axis

Sys1count[1] represents System1 data in y axis

Sys1count[2] represents System1 data in z axis

Sys2count[0] represents System2 data in x axis

Sys2count[1] represents System2 data in y axis

Sys2count[2] represents System2 data in z axis

**Scope: Intention is to check the package validity (which combines both sys1 and sys2) and the data validity of the system1 only. Specifications for data validity logics for system2 are not briefed, hence not required to consider these logics of the code for verification.**

For the data acquired from the package, two levels of checks are done on this data. In the first level, data is validated w.r.t package parameters. Then second level of validation is carried out for both the systems at axes level. **This acquisition & validation will be carried out in 3 different working modes 11, 33 & 44.**

1. Package Validity Checks
2. System Data Validity Checks
3. **Package validity Checks:** This validity should be carried out only in modes 11 & 44 only.

Data received from this package is validated using following global input parameters:

* Time out
* Header
* Checksum
* Data Validity

**Action after validating package (logics):**

* + - If the package is found failed w.r.t above parameters, replace current sample with old sample for both systems.
    - If this package validity failure happens for 20 or more times continuously, declare permanent failure of System1 & System2 in the “overall status word”.

1. **System1 Data Validity checks (Axes wise):** This validity should be carried out in modes 11, 33 & 44. There are two types of data validity checks
2. Absolute limit validity:

Definition: If the present sample value is greater 0.625 or less than -0.625, then it is treated as invalid, otherwise valid.

* + - * Intermittent failure declaration logic: If absolute limit failure for the current sample is observed, indicate intermittent failure for the current sample in the “intermittent status word” for the current sample. (Refer ‘intermittent status word” to represent this failure)
      * Permanent failure declaration logic: If absolute limit failure happens for 10 or more times continuously, declare permanent failure of corresponding axes of System1 in “Sys1FailFlag” and “overall status word”. Refer Note 1 for System1-x axes failure, and refer “overall status word” to represent this failure). Conditions to conduct this check are detailed as:

In Mode 11 && 44 , Package validity(of the current cycle) is to be checked. If Package is valid, then only this check needs to be done

In Mode33, Package validity is not required to be checked.

**Note1: For System1-X axes, there is a backup system called System1-G. Therefore, absolute limit permanent failure is to be declared for System1-X axis in “overall status word”, if switchover to System1-G has not happened. (Status of switch over to System1-G is represented by input global flag “Switchover”)**

1. Relative limit validity:

Definition: If the difference between previous sample and present sample count is greater than 0.0489 then the data is invalid, otherwise valid.

* + - * Intermittent failure declaration logic: If relative limit failure for the current sample is observed, indicate intermittent failure for the current sample in the “intermittent status word”. (Refer ‘intermittent status word” to represent this failure)
* **Input parameters:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.No. | Parameter name | Data Type | Global/Local | Usage |
|  | TimeoutFlag | unsigned int | Global | 1. Indicates that time out occured in acquiring data from package 2. No time out |
|  | HeaderValidFlag | unsigned int | Global | 1. Indicates header is invalid 2. Indicates header is valid |
|  | chksumValidFlag | unsigned int | Global | 1. Indicates checksum is invalid 2. Indicates checksum is valid |
|  | DataValid | Unsigned int | Global | 1. Indicates data is invalid 2. Indicates data is valid |
|  | mode\_word | Unsigned int | Global | Represents working modes for the system. Can take only 11, 33, 44. |
|  | Sys1count[3] | double | Global | Current sample of data acquired from System1 for 3 axes |
|  | Sys1count\_old[3] | double | Global | Previous sample of data acquired from System1 for 3 axes |
|  | Switchover | Unsigned int | Global | 0-Indicates that switchover has not happened  1- Indicates switch over has happened |

* **Output parameters:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.No. | Parameter name | Data Type | Global/Local | Usage |
|  | Sys1FailFlag [3] | Unsigned int | Global | Indicates permanent failure of System1. |
|  | Sys2FailFlag[3] | Unsigned int | Global | Indicates permanent failure of System2. |
|  | Interstatus | unsigned int | Global | Status word reflecting the occurrences of intermittent failures of all axes of System1 & System2 (detailed below) |
|  | overall\_status | unsigned int | Global | Status word reflecting the occurrences of permanent failures of all axes of System1 & System2 (detailed below) |

* **Status words indication:**
  + - Overall status word:

Indicate all the permanent failures for both systems axes parameters (X, Y & Z)

* + - Intermediate status word:

Indicate all the intermittent failures of package validity, absolute limit and relative limit in appropriate bits for both systems axes parameters (X, Y & Z).

### 

### Overall Status Word (Indicating Permanent failure of System1 & System2)

|  |  |  |
| --- | --- | --- |
| **Bit No.** | **Parameter** | **Description** |
| MSB 15 | -- | -- |
| 14 | -- | - |
| 13 | -- | - |
| 12 | -- | - |
| 11 | Sys1-Z failure in case of package failure(checking for 20 times) or exceeding absolute limits for at least 10 times | 1 – NOT OK 0 – OK |
| 10 | Sys1-Y failure in case of package failure(checking for 20 times) or exceeding absolute limits for at least 10 times | 1 – NOT OK 0 – OK |
| 09\*\* | *Sys1-G in case of pulses exceeding limit for atleast 10 times* | *1 – NOT OK 0 – OK* |
| 08 | Sys1-X failure in case of package failure(checking for 20 times) or exceeding absolute limits for at least 10 times | 1 – NOT OK 0 – OK |
| 07\*\* | Sys2-Z failure in case of package failure(checking for 20 times) *or exceeding absolute limits for at least 10 times* | 1 – NOT OK 0 – OK |
| 06\*\* | Sys2-Y failure in case of package failure(checking for 20 times) *or exceeding absolute limits for at least 10 times* | 1 – NOT OK 0 – OK |
| 05\*\* | Sys2-X failure in case of package failure(checking for 20 times) *or exceeding absolute limits for at least 10 times* | 1 – NOT OK 0 – OK |
| 04 | -- | -- |
| 03 | -- | -- |
| 02 | -- | -- |
| 01 | -- | -- |
| LSB 00 | -- | -- |

\*\* Conditions represented in italics are not in the scope of the present specifications

### Intermittent Status Word (Indicating intermittent failure of System1 & System2)

|  |  |  |
| --- | --- | --- |
| **Bit No.** | **Parameter** | **Description** |
| MSB 15 | **--** | **-** |
| 14 | **--** | **-** |
| 13 | **--** | **-** |
| 12\*\* | *Sys2–Z relative pulses* | *0-OK 1-NOT OK* |
| 11\*\* | *Sys2 –Y relative pulses* | *0-OK 1-NOT OK* |
| 10\*\* | *Sys2–X relative pulses* | *0-OK 1-NOT OK* |
| 09\*\* | *Sys2 –Z absolute pulses* | *0-OK 1-NOT OK* |
| 08\*\* | *Sys2-Y absolute pulses* | *0-OK 1-NOT OK* |
| 07\*\* | *Sys2-X absolute pulses* | *0-OK 1-NOT OK* |
| 06 | Sys1-Z relative pulses | 0-OK 1-NOT OK |
| 05 | Sys1-Y relative pulses | 0-OK 1-NOT OK |
| 04 | Sys1-X relative pulses | 0-OK 1-NOT OK |
| 03\*\* | *Sys1\_G absolute pulses* | *0-OK 1-NOT OK* |
| 02 | Sys1-Z absolute pulses | 0-OK 1-NOT OK |
| 01 | Sys1-Y absolute pulses | 0-OK 1-NOT OK |
| LSB 00 | Sys1-X absolute pulses | 0-OK 1-NOT OK |

\*\* Conditions represented in italics are not in the scope of the present specifications