BRIEF SUMMARY OF THE INVENTION

The present invention provides a definite or fixed method for implementing a computing system software and safety mechanisms, and provides the clearly defined and specific unique criteria to measure both software constructions and the safety mechanisms for said software based on the data value and data timing attributes, said method is based on the exclusive disclosure: every computing system has two and only two systematic attributes: data value and data timing, and the system functionalities can be fully represented by the data comprising Input Data, Middle Data and Output Data illustrated in the drawing, in which the Output Data represent fully the system functionalities under the input data from the system black-box point of view, the Middle Data represent fully the middle functionalities that are transporting and transforming the Input Data to the Output Data. Each data has two, and only two attributes: Data Value and Data Timing, and the goals of system development are to derive the two attributes for each required output data correctly; and the purpose of said safety mechanisms of said system software is to detect every deviation of either said two attributes of said data between implemented ones and defined ones (that are defined as errors) that will impact said attributes of any said output data, and prevent said system software from impacting other systems caused by outputting deviated data from said software.

[00014] So, the development activities regarding to the system functionalities and safety, such as the value and data timing attributes realizations and systematic safety developments, will be complete, consistent, accurate and efficient if they are applied on the data and their relationships defined in said system.

BRIEF DESCRIPTION OF THE DRAWING

[00015] The drawing is an illustration of some embodiments of the disclosure. The preceding and other objects, features, and advantages of the disclosure will be apparent from the following description of embodiments.

DETAILED DESCRIPTION OF THE INVENTION

[00016] The invention discloses a specific and definite method to construct computing software based on said system requirements by focusing on only two attributes: Data Value and Data Timing, wherein said method consisting of:

a. defining and implementing a plurality of data based on said system's requirements, wherein each said data consists of only two systematic attributes consisting of data value attribute and data timing attribute, consisting only of 3 types of data:

1). one or more input data which are imported from input devices of said system; and

2). one or more middle data which are stored in memory devices of said system; and

3). one or more output data which are exported by output devices of said system; and

b. defining and implementing data calculations for each said output data using one or more said middle data and one or more said input data in said system, wherein each said calculation consists of:

1). one or more mathematic expressions; or

2). one or more logic expressions; or

3). one or more experience expressions; or

4). one or more artificial intelligence expressions; or

5). derivations of one or more expressions above; and

c. executing said implemented calculations to derive both said systematic attributes for each said output data which are development goals of said software; and

d. developing safety mechanisms for each said output data to prevent said software from outputting deviated data if either attribute of said data deviates from their definitions;

wherein said computer system consisting of:

a. one or more input devices for inputting said input data into said computer system from outside using input transmission protocol; and

b. one or more output devices for outputting said output data out of said computer system from inside using output transmission protocol; and

c. one or more memory devices for storing said middle data that are generated during said calculations; and

d. one or more Arithmetic Logic Unit devices for doing said calculations; and

e. one or more data transmission links between said devices; and

f. a management system for managing said devices and said links.

[00017] Computing system operation concept

[00018] For every computing system, it can be described using elements illustrated in the drawing consisting of a plurality of output data, a plurality of input data, a plurality of middle data and the calculations that are represented by formulas of $f1, \ldots, fn$, among which, one calculation is defined to derive each output data using one or more input data and one or more middle data. Wherein said calculations consist of not only the mathematic calculation but also any methods to derive the output data, such as logic expression, a fuzz expression, an experience or AI expression, or any combinations and derivations of those mentioned expressions. The description above illustrated in the drawing is the system operation concept, wherein said data

and said calculations are mandatory in every computing system development because they establish the relationships between the output data and input data via the middle data, and they must be defined at the beginning of the development. If any said data or any said calculations is not defined accurately, explicitly and completely, then the system development or the required software constructions will be infeasible. And the following development steps including the software constructions and safety mechanisms' developments will be done based on those defined data and relationships, so, (1) the development is consistent, (2) the development will cover fully the required information, (3) the development will not include any other information or steps that are not defined above.

[00019] Each data in a computing system has two, and only two attributes: Data Value and Data Timing, and the goals of system development are to derive the two attributes for each defined output data.

[00020] For one of embodiments in the drawing, the system operation concept based on the output data, the input data and the middle data can be defined in detail using formulas below:

Output Data 1 = ƒ1 (Input Data 11, …, Input Data 1i, Middle Data 11, …, Middle 1j);

Output Data 2 = ƒ2 (Input Data 21, …, Input Data 2l, Middle Data 21, …, Middle 2p);

•••

Output Data n= ƒn (Input Data n1, …, Input Data nq, Middle Data 1n, …, Middle nr).

Among the formulas above, m, n, k, i, j, l, p, q, r all are integers with the relationships:

1<= i, l, q <=m; 1<=j, p, r <= k; and all the input data groups consisting of the group of Input Data 11, …, Input Data 1i, and the group of Input Data 21, …, Input Data 2l, …, and the group of Input Data n1, …, Input Data nq are subsets of the input data group consisting of Input Data

1, … Input Data m; and all the middle data groups consisting of the group of Middle Data 11, …, Middle Data 1j, and the group of Middle Data 21, …, Middle Data 2p, …, and the group of Middle Data n1, …, Middle Data nr are subsets of the middle data group consisting of Middle Data 1, … Middle Data k.

[00021] Wherein said Output Data 1 is derived from all the operated data in the calculation represented by the formula of ƒ1 consisting of Input Data 11, …, Input Data 1i, Middle Data 11, …, Middle 1j; and using the same way to derive the Output Data 2, …, Output Data n.

[00022] The derivation method above can be done recursively to every data in the system concept that need to be decomposed further into decompositions as the development progresses. For example, if the Middle Data 11 needs to be decomposed into such expression: Middle Data 11 = ƒm11 (Input Data 111, … Input Data 11i, Middle Data 111, …, Middle Data 11j), wherein said ƒm11 is the calculation to derive the Middle data 11, said input data group of Input Data 111, …, Input Data 11i is a subset of input data group of Input Data 11, …, Input Data 1i, said middle data group of Middle Data 111, …, Middle Data 11j is a subset of middle data group of Middle Data 11, …, Middle Data 1j. Then the derivation method for the Middle Data 11 will be done by applying the derivation processes above to the expression of Middle Data 11 = ƒm11 (Input Data 111, … Input Data 11i, Middle Data 111, …, Middle Data 11j). So that, the data value attribute of each defined output data and each needed middle data can be derived using the processes above.

[00023] In the system operation concept above, the middle data are defined to store and share middle calculating results to support the output data calculations, so the values of said middle data are defined according to the calculations, and the timing attributes of said middle data are indirectly defined by said output data's transmission protocol, because said output data

have the defined exporting times in their export transmission protocols, which must include said input data available time, said middle data available time, said calculation durations and said output data's exporting time durations, among those time attributes or time durations, all are defined by either their transmission protocols or their calculation durations except said middle data's timing attributes, so once said output data's timing attributes are defined by their transmission protocols, then said middle data's timing attributes can be derived from the known attributes or time durations of other data; the calculations are defined to transform and transport the input data and middle data to derive the output data.

[00024] Taking the system as a black box, the software output data which are the system output data as well represent the system external behaviors that are the expected system functionalities under the input data. From the system operation concept, the output data depend totally on the input data, the middle data and the calculations; and anything else that is not in the calculations will not have any effect to the output data.

[00025] The safety requirement analysis can be done based on the system operation concept. For example, if the Output Data 1 in the system operation concept above is a safety relevant data that will be related to a safety device, then the operations defined by the $f1$ and all the operated data in the $f1$ are safety relevant.

[00026] So, the development of the system safety and functionalities will be efficient and specific if said development focuses on only the elements that have effects to the output data. To define the timing attributes for each said output data, the maximum data value change frequency of said input data should be considered, so that the said output data's exporting frequency should not be more than said frequency; said output data's timing attributes should be derived from said output data's export frequency plus some system overheads time. And each said middle data in

said calculation should be assigned a suitable time slot as said data's timing attributes according to said output data's timing attribute. Then each said middle data should be assigned to a suitable device in said computer system according said data's both attributes, that is: said assigned device's functional capacity and performance should meet said data's attributes. In this way, said software can be constructed according to said input, middle and output data's attributes.

[00027] From analysis above, in every computing system under development, it will be complete, specific and accurate that the development is to derive the two data attributes of each defined output data: data value and data timing based on the relevant input data and middle data. To make said input data ready, the development needs:

a. defining input transmission protocol for each said input data, wherein said protocol defines both systematic attributes of said input data; and

b. allocating a suitable input device and executing said inputting for each said input data, wherein said device meets said input transmission protocol;

wherein each said input data value is defined as value received from said input device; each said input data timing is defined as time when said data is imported from said input device and available to be used by said Arithmetic Logic Unit.

[00028] To make said middle data ready, the development needs:

a. constructing defined generation logic for each said middle data; and

b. constructing transmission protocol for each said middle data to transmit said middle data between said memory and said Arithmetic Logic Unit, wherein said protocol defines said data two systematic attributes; and

c. allocating a suitable memory device for each said middle data, wherein said device meets said middle data transmission protocol; and

d. executing said generation for each said middle data and storing said data into said memory device using said transmission protocol.

wherein each said middle data value is defined as generated value during said calculation stored in said memory device; each said middle data timing is defined as time duration reading from said memory device to be available to be used by said Arithmetic Logic Unit. To make said output data ready, the development needs:

a. constructing defined calculations for each said output data; and

b. constructing exporting transmission protocol for each said output data, wherein timing attribute of said protocol is equal or more than said output data's calculated timing attribute; and

c. transmitting all said input data in said calculation to said Arithmetic Logic Unit using said links by said management system; and

d. transmitting all said middle data in said calculation to said Arithmetic Logic Unit using said middle data transmission protocol; and

e. executing said constructed calculation for each said output data; and

f. allocating a suitable output device for each said output data, wherein said device meets said output data exporting transmission protocol; and

g. transmitting said output data from said Arithmetic Logic Unit to said output device using said links by said management system for each said output data; and

h. exporting said output data from said output device for each said output data using said exporting transmission protocol.

wherein each said output data value is derived from said implemented calculation; each said output data timing is calculated as time summary of:

(1). time duration of making all said input and middle data in said output data's calculation to be available to be used by said Arithmetic Logic Unit in said calculation; plus

(2). duration that is used by said Arithmetic Logic Unit to execute said calculation including transmission time that is used by said management system to transmit all said input and middle data in said output data's calculation from their locations to said Arithmetic Logic Unit using said links; plus

(3). duration that said Arithmetic Logic Unit transmits said calculation result data as said output data to said output device; plus

(4). duration that said output device exports said output data from said system.

The timing calculation above does not consider any interruption. In the reality, most computing systems must handle the interruptions, such as the interruptions from the high priority functions, waiting for responses from other functions, waiting for the input data from the input device functions.

[00029] In a computing system, there are two types of interruptions involved in the data timing calculations: the first one is waiting for needed data which are transferred from other functions to be ready; the second one is the interruptions from high priority functions.

[00030] Data timing attribute calculation considering interruptions is the sum of said data timing without considering interruption plus the interruption durations, in which, each interruption duration calculation is the same as a data timing calculation. So that, the data timing attributes of each required output data can be derived using the process above.

[00031] There are two part time durations in each data timing attribute calculation consisting of operation duration and transfer duration, in which, said operation duration is the

time used by of said Arithmetic Logic Unit to do the operations defined in said calculation, for example, for said Output Data 1 in the system operation concept, the operation duration is the time used by the Arithmetic Logic Unit (ALU) to operate all the operated data in the formula defined in the $f1$; said transfer duration is the time to transfer said data from the source location to the destination location.

[00032] There are only two types of data transfer in a computing system consisting of synchronous data transfer and asynchronous data transfer.

[00033] Synchronous data transfer has the following types:

a. synchronous function call with parameters: the transferred data will be available immediately to the receiving function, so the duration is zero.

b. synchronous communication (either serial or parallel), the data available time that is the data transfer duration can be calculated based on clock information in the protocol.

[00034] Asynchronous data transfer has the following types:

a. Asynchronous function call: the duration should be calculated as the designed timeout value.

b. Asynchronous memory sharing, i.e., the contents availability in the shared memory must be known by periodic polling, whose duration is the designed timeout value.

c. Asynchronous communication: the duration is calculated based on the timing defined by the transmission protocol.

[00035] The benefits of using the disclosed embodiments to develop a computing system consist of making use the definitions from the system operation concept, and the whole development process above and the development measurement criteria are clearly and