Sensors & Transducers, Vol. 155, Issue 8, August 2013, pp. 98-106

www.sensorsportal.com

Sensors & Transducers

http://www.sensorsportal.com

Construction of Smart Sensor Networks Data System Based on Integration Formalized BCCS Model

*** Chunlin Kuang, Guofang Kuang**
College of Information Technology, Luoyang Normal University, Henan Luo Yang, 471022, China
* Tel.: 0086+15937925882
E-mail: kuangchunlinls@163.com

Received: 8 June 2013 /Accepted: 12 August 2013 /Published: 20 August 2013

Abstract: Smart sensor means having information detection, information processing, information storage, logical thinking and judgment capabilities of the sensor. In this paper, CCS and B pass-by-value method and language, by giving a lightweight description language to achieve the BCCS model can be used to describe concurrent or distributed systems. The paper presents construction smart sensor networks data system based on integration Formalized BCCS model. The experiment shows that the B method combined with CCS on the powerful formal description of actual smart sensor networks data system capacity. This smart sensor can calculate the value of a physical quantity by using compensation sensor network data in order to improve intelligence. *Copyright © 2013 IFSA.*

Keywords: Smart sensors, B method, Value-passing CCS, BCCS, Sensors network.

1. Introduction

Smart sensors not only visual, touch, hearing, smell, taste function, and should have a memory, learning, thinking, reasoning and judgment "brain" capability. The former is accomplished by conventional sensors. Here, the functional configuration of conventional sensors include sensitive components, and analog conditioning circuit (ADC), an objective that sensitive components will be described with the physical state of the environment or the characteristic parameters of circuit elements into or state parameter, conditioning circuit parameters of the circuit converted into a voltage signal and normalized to meet the dynamic range of the ADC. ADC output intelligent processor should be intelligent digital signal processing.

Smart sensor means having information detection, information processing, information storage, and logical thinking and judgment capabilities of the sensor. It not only has a variety of conventional sensors function, but also with data processing, fault diagnosis, nonlinear processing, self-tuning, self-tuning and human-machine communication and other features. It is microelectronics, micro-computer technology and detection technology combined with the product. Early intelligent sensor is the sensor output signal is processed and converted by the interface to the microprocessor after the arithmetic processing part.

Our research combines methods and language of value-passing CCS and B given a lightweight describe language, achieve the BCCS model can be used to describe the concurrent or distributed systems. On the basis of the return value CCS operational semantics, combined with B on the definition of the abstract data structure, and it is the system constraints and the functional processing given BCCS operational semantics [1].

The BCCS model make up the traditional single method separates the various facets of the information systems inadequate. The BCCS model,

Article number 1292

two important facets of information systems has been unified, with B to describe the function, these functions are defined by the operation and status of modeling, with value-passing CCS to describe the behavior of the system characteristics, such as concurrency, real-time, the distribution of B described the function to define the value of the value-passing CCS. Expect to take the BCCS model, can reduce the difficulty of formal methods to describe and verify large concurrent software systems, while improving the reliability of concurrent or distributed software development.

Smart sensor further includes means to push. The design of this type of sensor has begun to change methods, not as much as a simple hardware configuration, the need for a simple software processing channel Movement, the corresponding output signal is a digital signal; from the "live part line" concept made after the field bus-based measurement and control system has been widely used, corresponding to the sensor design also proposed new requirements. From a development perspective, the future of independent single sensor will be less and less use of the occasion, more of multi-sensor system applications in order to achieve multi-parameter measurement and control of multiple objects. Measurement and control of the exchange of information is at the bottom line of the main unit through the field to complete. Data exchange is mainly through the Intranet and other networks to achieve. The paper presents construction smart sensor networks data system based on integration Formalized BCCS model.

2. BCCS Syntax and Operational Semantics Based on CCS and B Method

2.1. Research of CCS and B Method

Concurrent increase in the semantics of basic B, the original B method of operation (Operations) clause to event (EVENTS) clause to increase the dynamic clause (DYNAMICS), seen purely as a B abstract machine (AMN) is an interactive system. So Event B is similar to the Petri net description of dynamic, complicated process tool, but also inherited the function of B in the data description, and its basic structure, but also to some extent affected the simplicity of the method B, ease of use, more importantly, is such that the B in the description of the sequential model, the method becomes complicated and incomprehensible.

BCCS model uses the idea of integrated value-passing CCS and B language, with the B language to extend the ability of the CCS data and function definitions, and B method extended through CCS clear that the ability of the operational semantics and describe concurrent systems, forming amore expressive formal development method to describe the problem from a more facets [2]. BCCS model describes the syntax of the language, but it lacks the support of the semantic, combined with B on the definition of the abstract data structure on the basis of the return value CCS operational semantics constraints and the functional processing of the system, given the operation of the BCCS semantic, further portrayed BCCS model.

Smart sensors according to the given environmental conditions conventional sensors and a priori knowledge, the processor using the digital calculation, the hardware automatically compensate for conventional sensors linear, nonlinear and drift caused by environmental factors and signal distortion, to best recovery test signal [3]. Calculation method implemented in software, hardware defect compensation to achieve the purpose of the software.

The B method originally proposed by J.-R. Abrial, it is a theory and method of the formal development of computer systems used in the real-time domain has been applied. B, mentioned in this article is actually short for B technology in software engineering methods and tools set, it contains the B method tool B, B Tool Box is a sound practice-oriented software process based on the mathematical theory of technology B used symbols and supports most of the software process: requirements analysis, specification, software design, implementation and maintenance of the step-by-step structure of layered software along with step-by-step validation and verification of the guiding principles of the B method [4]. The B tool box includes a lot of tools, all the tools integrated in a window-based software development environment, these tools can be integrated to run automatically, you can support the entire process of the use of the B method development software tool B is gradually constructed to support software the confirmation process available static analysis, dynamic analysis using simulation technology, integrated theorem proving correctness proof.

B method using a simple pseudo programming language to describe the demand model, and software design and implementation, this language is an abstract machine symbols (AMN, Abstract Machine Notes), AMN type detection support the specification, dynamic verification of mathematical proof ensure that the design process indeed ended.

The BCCS system, in order to achieve lightweight, only need a pass value of a subset of the CCS, and therefore value-passing CCS to do some cutting (Let's Take C_CCS to represent the cutting CCS). Let C_CCS detailed description given several constructors. C_CCS and gives the operational semantics, operational semantics and basic CCS operational semantics similar. CCS introduction to the standard we can also see that it is based on two grammar classes, a set of processes, and a set of actions. Us and said two grammar C_CCS set. Here is syntax defined C_CCS, as is shown by Fig. 1.

$$
\begin{aligned}
&CCS\_prog\_cut && = Agent\_Decl\_cut \ldots Agent\_Decl\_cut \\
&Agent\_Decl\_cut && = C\_Agent\_cut \doteq Agent\_cut \\
&Agent\_cut \quad | && = C\_Agent\_cut \mid \\
& && \quad Action\_cut.Agent\_cut \mid \\
& && \quad \overline{Action\_cut}.Agent\_cut \mid \\
& && \quad \tau.Agent\_cut \mid \\
& && \quad Agent\_cut + Agent\_cut \mid \\
& && \quad Agent\_cut \mid Agent\_cut \mid \\
& && \quad Agent\_cut \setminus resSet\_cut \mid \\
& && \quad \boldsymbol{0} \\
&resSet\_cut && = \{Action\_cut, \ldots, Action\_cut\}
\end{aligned}
$$

Fig. 1. The defined C_CCS the syntax.

Finally, take a look at the dynamic semantics of CCS, CCS is a function of language, so the concept of the state not. People interested, if an action is performed before the process of expression is, then what will become of expression after the implementation of the action process. Actions usually mark, which, possibly with the parameter value refers to the action [5]. Mark indicates that the implementation of the action. This is essentially a reduction action; these rules can be a reduction rule, as is shown by equations.

$$a(x_1,\ldots,x_n).e \xrightarrow{\partial(v_1,\ldots,v_n)} e[v_1 x_1,\ldots,v_n / x_n] \quad (1)$$

$$\overline{a}(f_1,\ldots,f_n).e \xrightarrow{\bar{\partial}(f_1,\ldots,f_n)} e \quad (2)$$

$$\tau.e \xrightarrow{\tau} e \quad (3)$$

$$\frac{e_1 \xrightarrow{\mu v} e_1'}{e_1 + e \xrightarrow{\mu v} e_1'} \quad (4)$$

$$\frac{e_2 \xrightarrow{\mu v} e_2'}{e_1 + e_2 \xrightarrow{\mu v} e_2'} \quad (5)$$

$$\frac{e_1 \xrightarrow{\mu v} e_1'}{e_1 \mid e_2 \xrightarrow{\mu v} e_1' e_2} \quad (6)$$

$$\frac{e_2 \xrightarrow{\mu v} e_2'}{e_1 \mid e_2 \xrightarrow{\mu v} e_1 e_2'} \quad (7)$$

$$\frac{e_1 \xrightarrow{\mu v} e_1', e_2 \xrightarrow{\mu v} e_2'}{e_1 \mid e_2 \xrightarrow{\mu v} e_1' e_2'} \quad (8)$$

Through the use of a hybrid model-based language is VDM formal methods in the process algebra CCS. It constructs a mixed-language based on CCS and VDM to CCS-defined value. All operations on the data in this language VDM to define and call by value-passing CCS, the semantic object is Milner defined CCS semantics instead of VDM.

CCS is a functional language, its basic ingredient is called process (or stroke), process expressions containing free variables, the combination of the process is the process, the combination of depth is also arbitrary [6]. CCS has a pair of communication primitive groups label set, which is a set of names, and the anti-set of names $\overline{\Delta} = \{\overline{\alpha} \in \Delta\}$, which are complementary, and the set is called the label set, its elements unified to represent $\Diamond = \Delta \bigcup \overline{\Delta}$, on a self-remapping, indicated by $\overline{\overline{\lambda}} = \lambda$.

B tight management control over the entire software process description stage early in software development specifications, developers will control the user needs to verify the consistency of the test system demand model, at each stage of development, each design description and specification requirements document control its consistency, the lowest level of the design components from the pre-defined and reliable assembly of parts library, and translated into executable code, reliable predefined parts (many parts for the B tool-sets generated by testing or reuse application), the new lower layer member can be further added to these added components may be provided by the existence of reliable components and development, and to test their properties.

The constant part of the environment is with the definition of the B environment. With the handling of the environment and the process in the of Bruns early formal operational semantics "parameter residual problem, it is because the process to introduce the values of local variables are allowed memory caused by the environment. BCCS model environment to describe this constant environment, one is described by B the global variables, and the other is a local variable of a process, which ensures that the two does not occur previously mentioned, due to the "Parameter residue" the cause of the conflict.

ZCCS this article is a hybrid integrated formal methods on the basis of value-passing CCS, to borrow a Z language model (schema) and schema text to extend the value-passing CCS value. The advantage of this approach and our article, and it is the idea of a clear, easy-to-use structure. But due to lack of Z Language modular capacity, it is difficult to identify the impact of a state model for all modes of operation, can not support the reuse of specifications, the Z language difficult to be handled directly by the computer and other defects, and limits the practical application of ZCCS in large systems. In order to overcome these deficiencies of the Z language, we use the B method and language instead of Z and CCS do hybrid BCCS model, so ensure that the next premise of ZCCS advantages, B and language is also very good to solve due to a defect of the Z language itself insufficient of the ZCCS model.

Smart sensors the sensor during use all the problems involved includes. Shown sensitivity, zero drift, calibration, etc., even if a single physical measurement sensors must also have a standard interface to obtain the required information (such as temperature, humidity, etc.) in order to achieve and calibration of measured physical quantity. On many occasions, plus the output display unit, this single Smart sensor system has already covered the concept of traditional instruments.

2.2. BCCS Syntax and Operational Semantics

The BCCS model is more similar to the syntax and semantics of value-passing CCS. In the BCCS, the value of the value-passing CCS expression to describe B, which include: process and input action parameter declaration; parametric process and output action arguments (assignment); condition statement predicate expression. In order to better describe the syntax of the BCCS, its operational semantics, we need to define the BCCS syntax met language.

The input parameters for the action in the Bruns value-passing CCS is not global, but also for any action with the Senate, not to do a symbolic statement, nor the value of these symbols can be obtained for a correct description. In BCCS model, the parameters can be defined in any output action.

Deduction is assuming that you want to prove a conjecture, for example, it is rarely isolated to you, in most cases, the need to prove something (also known as the premise or assertion) relies on certain assumptions, the proof can assume that they are true, some when the suspect is under proof, also said that by the assumptions to deduce conclusions recorded as the following formula.

$$HYP \vdash P \tag{9}$$

$$\overline{P \vdash P} \tag{10}$$

$$\frac{HYP \vdash P \quad PHY\ P \subseteq HYP'}{HYP' \vdash P} \tag{11}$$

$$\frac{P \subseteq HYP}{HYP \vdash P} \tag{12}$$

$$\frac{HYP \vdash P \quad PHY, P \subseteq Q}{HYP \vdash Q} \tag{13}$$

The process can be similar to the definition of Bruns in CCS, with B to describe the grammar class B described. In a description of the list, you can introduce a collection of axioms description to define constants for the conventional definition and context. In BCCS, before the introduction into the process with B to define the structures previously mentioned.

A very important value-passing is CCS operator. Action drive, the state of the process can be constantly changing, we discuss the action parameter expression and next, we look at the CCS action itself what should in BCCS model expression.

Another problem is that in an environment with the handling of the environment and the process must be recorded multiple concurrent state of the process variables. This means that if the variable is shared by multiple parallel processes, so it is necessary to perform the conversion different process to ensure that the only variable. In the BCCS, based on the above-mentioned environment constructed into a binary tree, in order to better simulate with operator | or \ process environment, and the environment of different processes in different branches, on the front of the mentioned problem, which is reflected in the operational semantics of inference rules.

Traditional value-passing CCS, the representation of the action is a very abstract concept, and do not have a detailed description of the list. In the within the BCCS model, we put the CCS action through an implicit function converted to a B abstract machine defined operation (OPERATION) [7]. The implicit meanings mentioned here BCCS model description of a concurrent system, you can direct the operations defined in the B abstract machine as CCS action. Here there is a problem in the abstract machine B categories, but not on the input and output events CCS in the operation of the input and output strict classification. There are two ways to solve this problem, a semantic representation of extended abstract machine B, B abstract machine classification of input and output events, the second method is to add a descriptor events using B abstract machine specify the operation as an input or output action.

B inside a few very important binary relation, we study involving several were to be introduced, for each operator, first introduced its syntax, then a non-formal explanation to each operator. We first introduce a binary relation from one set to another set (), and then introduce a relationship of inverse ($\leftrightarrow$), also known as anti-relationship, the next is the relationship between the domain (dom) and range (ran), then the relationship between composite (;) and constant relationship (id) and finally, the relationship between the various restrictions ($\triangleleft, \triangleright$). The table is not the same each other, and the situation p, q, s and t are as follows (Table 1).

The BCCS The operational semantics based Bruns research in the semantics of value-passing CCS and B semantic foundation up to construct. Defined value in BCCS general description from B, rather than by value CCS values defined, so that the expression that defines the content richer, of course, if CCS can define, we do not limit it. A clear description of the BCCS operational semantics given

26 operating rules, used to describe the behavior may occur in all BCCS description. Operational semantics allows us to better understand the BCCS, and ensure that the BCCS syntax correctness, completeness and consistency.

Table 1. Binary relation defined table.

Grammar	Definition
$u \leftrightarrow v$	$\mathrm{P}(u \times v)$
p^{-1}	$\{b,a \mid (b,a) \in v \times u \wedge (a,b) \in p\}$
$dom(p)$	$\{a \mid a \in u \wedge \exists b \cdot (b \in v \wedge (a,b) \in p)\}$
$ran(p)$	$dom(p^{-1})$
$p\,;q$	$\{a,c \mid (a,c) \in u \times w \wedge \exists b \cdot (b \in v \wedge (a,b) \in p \wedge (b,c) \in q)\}$
$q \circ p$	$p\,;q$
$id(u)$	$\{a,b \mid (a,b) \in u \times u \wedge a = b\}$
$s \lhd p$	$id(s); p\ \{a,b \mid (a,b) \in p \wedge a \in s\}$
$p \rhd t$	$p; id(t)\ \{a,b \mid (a,b) \in p \wedge b \in t\}$

The environment is similar to the constant part B on the definition of environment [8]. Bruns early formal operational semantics with the handling of the environment and the process, as well as on the expression of the reference rules appear the same problem is the parameter residual "This is because the process of the introduction of the value of the local variable is allowed the memory caused in the environment. If the process follows the formula is the equation 14.

$$in(x:\{0,1,2\})A \tag{14}$$

Three free appears likely to have different values in the elements of the set of constraints may be derived from the constant introduction of a process parameter, before migrating to this point in the process, the other two parameters are the individual input action binding. Through this example, you can clearly see that the component of both processes in their own environment, the constraint set also has its own environment. Let us assume that a process form, we must have an environment to describe the status of its variables, with its constant environment, to provide a value for the free variable, as is shown in Fig. 2.

How to define the object described in the description of the list just beginning moments. If you do not have this information as the basis behind the objects described would be meaningless. Here, we assume that all statements are closed, that is, it should not contain any free variables. We can consider that the initial environment is empty that is, starting from a particular environment; you can use a non-formal methods.

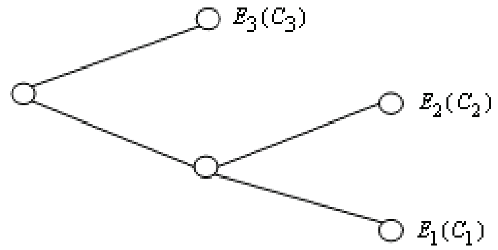

Fig. 2. The defined C_CCS the syntax.

We give the axiomatic structure is based on the simplified classical set theory, set theory and mathematical fields, we prefer the formal application, so are more concerned with the notation. Suppose for all BCCS grammar class can collection description, and each grammar fragment corresponds to a unique element of this collection, based on set theory and operational semantics is put the grammar variables described as a value expression [9]. In addition, it is assumed that a collection of the range of each variable, so the premise of the rule of each reference is $v \in S$.

BCCS described in Part B may be constructed to the global variable environment, CCS portion may be configured to a process environment, the descriptor list can be generated environment, where "local variable environment corresponding to an empty (this is because the initial process is not the environment branch so is empty).

Parameters of the process of constant, if there is an environment in which the local environment is described by the B abstract machine simple environment, global variable environment overloaded extension from the context of the environment, the value of the expression and characteristics of tulles the same value, which belongs to a local environment can migrate through action to reload, constants and parameters in the process environment bindings to the process, you can migrate to the after action.

SFSB instance pool, the EJB container provides a method similar to the operating system in a "context switch" approach for SFSB. In order to limit the SFSB number of instances running in memory, the EJB container needs to be SFSB session state saved to hard disk or other storage source, a process known as aestivation, SFSB instance customers call it up again , passivity session state back to the SFSB instance, a process known as activation (activation).Note that activate SFSB the instance is not necessarily SFSB instance is originally used, however, the new instance of the SFSB session state is still SFSB instance is activated session state.

This system has two components, sampling (samples) components periodically sampled radius and angle of the object, and accept conversion trigger (convert) to the conversion component (converter). Converter component to accept and it is converting them to a Cartesian coordinate output. The sampling component sampling radius and angle can be in any order. Conversion component in addition to the polar coordinates to rectangular coordinates output, but also remember the last time the converted value for the next conversion before receiving a repeat (repeat) trigger output was recently Cartesian coordinate values.

Ordinary input action prefix, for any without operator | and \ local environment is a simple environment (do not call the function), this environment is the environment of a B abstract machine described here is not (environment tree).This rule means that the process environment can be migrated to the action (input).

$$\frac{\{\alpha\} = ran(Inv \circ id(\{\alpha\}))}{<<BE, BE^{'}, AE>, \bar{a}.F> \xrightarrow{\alpha} <<BE, BE^{'}, AE>, F>} \qquad (15)$$

There are three types of distinct reference to the rules defined in the value-passing CCS description of the meaning of the first class describe how inherited value-passing CCS extended environment, where the environment contains a set of processes and a set of constants, symbol sets, action sets [10]. The second category describes how by converting a (environment, process) migrate to another (environment, process).The third category describes how to represent the value of the expression in the CCS.

The semantics of the CCS semantics and BCCS biggest difference is the third type of reference rules. Its expression is in the BCCS B given directly. Another difference is in BCCS introduced on the environment, in the discussion with parametric input operation, automatically includes any need to determine the type information of the range of parameters, since the BCCS B abstract machine the description portion and is defined on its basis's already made a judge.

A smart sensor node is composed of three parts: the traditional sense of the sensor, the network interface and a processing unit. According to different requirements, these three components can be used together to form different composite chips, it can be a single type. First, the sensor measuring a physical quantity to be converted to electrical signals by A/D converted digital signal through the microprocessor data processing (filtering, calibration) will send the results to the network, and the network data exchange is not the network interface module.

3. Construction Smart Sensor Networks Data System Based on Integration Formalized BCCS Model

Smart sensors to have a certain judgment, analysis and information processing capabilities, according to the work of the whole system regulating the various parts of the data transfer with the host computer, the system operates in a low power state and the optimal transmission efficiency optimized state. Currently there are several ways to achieve intelligent sensors, one is the common sensor and a microprocessor with digital bus interface coupled with signal conditioning circuitry combined together as a whole constitutes an intelligent sensor system, this non-integrated smart sensor is field bus control system, driven by the rapid development of the situation to develop. Another is the integration of smart sensors; this smart sensor system is sensitive components, signal conditioning circuitry and microprocessor unit integrated in a chip composition. In addition it was also taken semi-integrated manner, these several components into a few chips to form to facilitate their use of smart sensors.

Smart sensor system is the use of micro-machining technology and LSI technology, the use of silicon as a basic material to produce sensitive components, signal conditioning circuits and micro processing units and integrate them in a chip composition. This makes the intelligent sensor reaches the miniaturization, structural integration, thereby improving the accuracy and stability. Sensitive components constitute arrays, with the corresponding image processing software; you can achieve a number of graphic imaging and form image sensors.

The basic idea of the operating semantic description language abstract way the effect of the implementation of each component, avoid the semantics described depends on the language used in the specific computer. With BCCS operational semantics can better understand the BCCS model, to ensure the correctness of the syntax, completeness and consistency.

EJB container defines a method B abstract machine, portray some operations (actions), these operations are basically restore some of the ways or APIs EJB3, reflects Beans may have experienced in their life cycle event (action). With these events in the life-cycle process, the use of CCS can easily describe the EJB3.0 SFSB distribution of its concurrency features. By a formal description of the SFSB lifecycle, in the EJB3 specification of the SFSB lifecycle process more clear and consistent, showing BCCS powerful formal description of the actual system capacity.

This design of smart sensor system has the following characteristics: (1) the use of commonly used industrial site Mudbugs communication protocol, the programmable controller and communicate with other devices, using the RS-485 bus structure, get rid of the previous multi-line multi-point type sensor connection structure, simplifying the number of signal cable industrial site, standardized site layout, ease of installation, commissioning; (2) real-time, accurate and stable digital signal acquisition and 4-20 mA analog signal; (3) in order to avoid interference from external electrical signals, and does not affect the use of high communication baud communication effects, using high-speed opt coupler 6N137 the MAX485 with microcontroller PIC16F687 optical isolation. Select both isolated power module for more stable, lower power consumption.

From smart sensor network applications, its general use in the automation field of measurement and control level, between the Ministries of cable needed to connect through live together. For different applications, now there are many different bus standard protocols. To ensure that the design of the sensor is difficult to fully meet these protocols, interface issues which must be considered. This is a smart sensor network sensor biggest difference with the ordinary.

The use of formal methods for software development for the design, should be used in the discussion of "ideal" process algebra to describe concurrent structure and communication, but must rely on the description of the model-oriented method to declare the operation behavior. This paper presents a hybrid B language and process algebra CCS to build a BCCS model. In this model, we are given a language, basically mentioned in the first question can be resolved, and that value-passing CCS can provide rich description of data, while a partial solution to the second problem, that is the same model within demand for describing the behavior of the model-based demand, as the BCCS implemented software environment in the current study did not give, so the second question is just on the level described [11].

Let's further describe the dynamic part of the SFSB life cycle. Instance SFSB has not yet been instantiated that is no longer the system memory which does not exist. Useful resources through the introduction of a new instance are injected into the data members of the Bean, Bean class defines the @PostConstruct callback method, and the container will call the method.

$$ini\_SFSB \equiv newInstance().rdy1\_SFSB(nb) + 0$$

$rdy1\_SFSB(nb) \equiv$ if opt_di then

$dependencyInjection.rdy2\_SFSB(dib)$

else

$rename.rdy3\_SFSB(nb)$

$rdy2\_SFSB(nb) \equiv$ If opt_pc then

$postconstruct.rdy\_SFSB(beanIns)$ else

$rename.rdy\_SFSB(beanIns)$

$rdy3\_SFSB(nb) \equiv$ if opt_pc then

$postconstruct.rdy\_SFSB(beanIns)$ else

$rename.rdy\_SFSB(beanIns)$

Instance in the ready state, he can be free to accept a call request from the client; can be transactional or non-transactional nature. Timeout is removed or call @PrePassivate method the corresponding entering not instantiate or passivity.

Smart sensor network data can be easily detected by real-time processing of large amounts of data may also need to store them. The main purpose of storing large amounts of information is to prepare for later queries, this type of information include the history and information about the device probing analysis of the results of indexing, intelligent sensor with a digital communication interface, this interface can be directly through the computer to which it belongs communications and liaison and exchange of information. In addition, intelligent sensor information management program is very simple and convenient, for example, the detection system can be remote control, or the lock mode, work, and the measured data can be sent to the remote users.

$$\frac{BCCS\_Prog|-BCCS\_Env, \quad B|-BE \quad CCS|-AE}{BCCS|-<BE,\{\},AE>} \qquad (16)$$

With the sensor technology and computer network technology, including micro-mechanical systems and the associated connector, the signal processing technology, the development of network interface smart sensors, and put them on the network and for distributed control has become possible measure and gradually heating up [12]. The Smart sensor network can be composed of multiple high-precision, powerful control network. And low cost, installation and configuration, upgrades and maintenance. Smart sensor networks to make it more complete monitoring capabilities are widely used in automotive, industrial process control, health care, environmental monitoring and so on.

Assuming that the shape, such as a process or just a constant environment (i.e. the environment), but is in a constant environment, and the below-mentioned process brought. This can be achieved by a rule, with the non-formalized in terms of the method is: if the operator with a | or \ and the local environment of the process to migrate to a new process with the new local environment are possible, with the local environmental process can be done, which is behind the operational semantics rules Rule21.

Because of a timeout when the instance is in the passive state, the same will be removed, of course, can also be accepted the method @PostActivate re-return to the ready state.

The final with a SFSB's life cycle can be defined with the following equation. Because a SFSB is equivalent to a client on a server extension, then the following equation can also be regarded as the state in which the client.

Smart sensor system can be measured under feelings and follow certain rules to convert the signal device or devices, usually by sensitive components and conversion components, seen from the system, the sensor is a detection device being measured can feel information, and can feel detecting information according to certain rules converted into electrical or other forms of information output required to meet the information transmission, processing, storage, display, recording and control requirements. The importance of the sensor is embodied in its automation, scientific testing, measurement accounting, monitoring and diagnosis systems such as the basic link, as is shown by Fig. 3.

The paper presents construction smart sensor networks data system based on integration Formalized BCCS model. As computer hardware and software technology penetration, the microprocessor and sensors can be combined to produce with a certain data processing capabilities, and can self-test, self-calibration, self-compensation of a new generation sensors, smart sensors. Smart sensor network data processing occurs on the sensor development had a profound impact. Smart sensor network communication technology gradually matured and penetrated into all walks of life, all kinds of high reliability, low power, low cost, micro-volume network interface chip is developed.

The BCCS model is given the operational semantics to build a relatively complete the BCCS theoretical model. In this model, it is co-organized by the B and CCS a clear framework for the description of the data and behavior of complex systems. However, the relative B language and CCS speaking of BCCS the syntax is slightly more complex. BCCS in more detail characterizes the operation, but relatively CCS speaking, these operations will produce a more complex process. If the result of the operation is to be determined, will be relatively better, however, if the operation result is not determined, this problem will become more prominent.

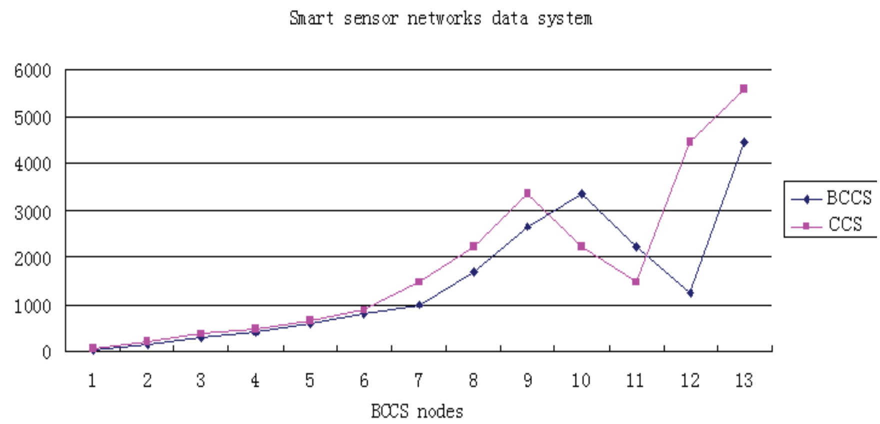

Fig. 3. Comparison results of smart sensor networks data system based on Formalized BCCS with CCS.

4. Conclusions

The main features of Smart sensor with high accuracy; has a good dynamic response speed; high reliability and stability; high SNR and high resolution; adaptive capacity; prices low, cost. Smart sensor to achieve the main way to achieve non-integrated: the traditional sensors, signal conditioning circuits, microprocessors and other combinations which constitute the system as a whole. Is an original technology based on the most economical and efficient way. The final realization of the basic norms is formal description on the EJB3 SFSB life cycle. EJB container defines a B abstract machine, and at the same time to portray some operations (actions), these operations are basically restoring the EJB3 some way (or APIs) reflects Beans may have experienced in the course of its life cycle event (Action). With these events in the life-cycle process, the use of CCS can easily describe the characteristics of the EJB3 SFSB the distribution of its concurrency. By the formal description of the SFSB life cycle so that the EJB3 specification SFSB the lifecycle process more clear and consistent, and also shows the B method combines CCS powerful formal description of the actual system capacity.

References

[1]. J. A. McDermid and L. M. Barroca Formal methods: Use and relevance for the development of safety critical systems, *Computer Journal*, 35, 6, 1992.

[2]. Zhou Shen Grong, Zheng Guo Liang, Comparison of Z and VDM with B, *Computer Science*, 29, 10, 2002, pp. 136-138.

[3.] Dan Dan Cui, Cai Qian Zhang, The Novel Model of Fusion Temperature Error Compensation System by Neural Network and Smart Sensors, *JCIT*, Vol. 8, No. 5, 2013, pp. 379 - 387.

[4]. G. Smith, J. Derrick, Specification, Refinement and Verification of Concurrent Systems—An Integration of Object-Z and CSP, *Formal Methods in System Design*, 18, 2001, pp. 249-284.

[5]. Deng Yao-Hua, Han Wei, Liao Qing-Fu, Wu Li-Ming, Regression Calculation Model of Flexible Material Processing Deformation based on Distributed Sensors Measurement, *AISS*, Vol. 3, No. 11, 2011, pp. 208 - 212.

[6]. Graeme Smith and John Derrick, Specification, Refinement and Verification of Concurrent Systems--

-An Integration of Object-Z and CSP, *Formal Methods in System Design*, 18, 2001, pp. 249-284.

[7]. M. Butler, M. Leuschel, Combining CSP and B for Specification and Property Verification, in *Proceedings of the Formal Methods*, (LNCS 3582), 2005, pp. 221-236.

[8]. Liu Meng, Sui Hulin, Wang Jun and Ma Qingbo, Firefighter Indoor Dead-Reckoning System with Inertial Sensors, *JDCTA*, Vol. 5, No. 12, 2011, pp. 259 - 265.

[9]. Michael J. Butler, Csp2b: A Practical Approach to Combining CSP and B, *Formal Asp. Comput*, 12, 3, 2000, pp. 182-198.

[10]. Bo Meng, A Formal Logic Framework for Receipt-freeness in Internet Voting Protocol, *Journal of Computers*, Vol. 4, No 3, 2009, pp. 521-530.

[11]. M. Butler, Csp2b: A practical approach to Combining CSP and B, *Formal Aspects of Computing*, 12, 3, 2000, pp. 182–198.

[12]. Liu Yang, Xing Jianping, Wang Ran, Wu Hua, Wu Xiaoming, Liang Haozhe, CF_OSSDL: An Optimum Space Step Distance Localization Algorithm of Wireless Sensors with Correction Factor in Ship Area Networks1, *AISS*, Vol. 4, No. 12, 2012, pp. 214-223.