(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
9 October 2008 (09.10.2008)

PCT

(10) International Publication Number
WO 2008/119732 A1

(51) International Patent Classification:
G06F 15/17 (2006.01)

(21) International Application Number:
PCT/EP2008/053630

(22) International Filing Date: 27 March 2008 (27.03.2008)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
0706134.4 29 March 2007 (29.03.2007) GB

(71) Applicant *(for all designated States except US)*: **NOKIA CORPORATION** [FI/FI]; Keilalahdentie 4, FIN-02150 Espoo (FI).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **KUUSILINNA, Kimmo** [FI/FI]; Nyyrikintie 8 B 11, FIN-33540 Tampere (FI). **NIKARA, Jari** [FI/FI]; Ahvenisraitti 24 B 28, FIN-33720 Tampere (FI). **LIUHA, Petri** [FI/FI]; Jäkäläkatu 30, FIN-33820 Tampere (FI).

(74) Agents: **DERRY, Paul** et al.; Venner Shipley LLP, 20 Little Britain, London EC1A 7DH (GB).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI,

[Continued on next page]

(54) Title: A MODULAR DEVICE COMPONENT

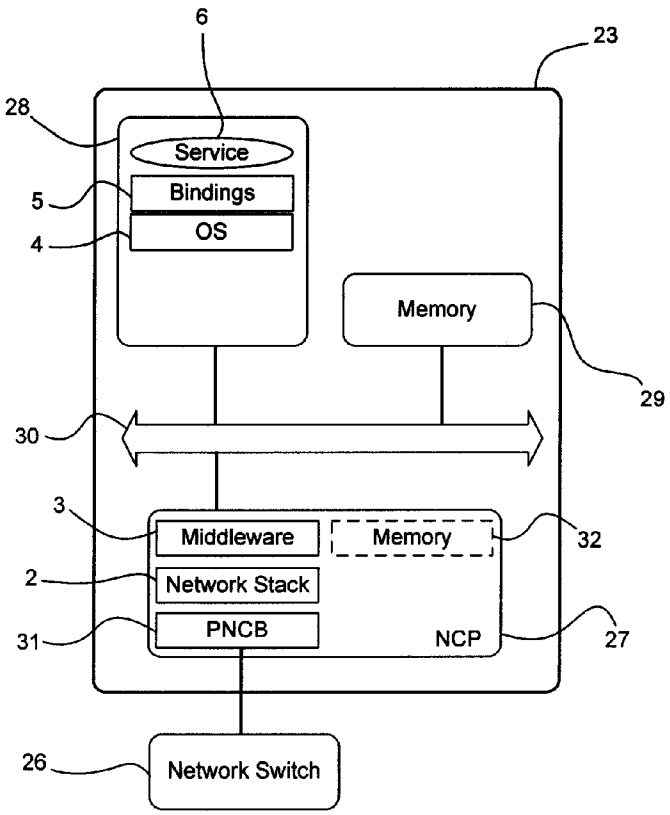

Fig. 4

(57) Abstract: A component (23) of an electronic device comprises a network connection processor (27), which comprises a physical network connection block (31) to receive data from and transmit data to a network and a first data processor (27) configured to process data arriving at the network connection processor (27), and a second data processor (28) configured to process data received from the network connection processor (27).

WO 2008/119732 A1

FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MT, NL, NO, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report*

— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments*

A Modular Device Component

Field of the Invention

The presented invention relates to a component of an electrical device and more particularly to the construction of the component and the way in which it connects to a network.

Background of the Invention

In many companies with complex high technology products, the renewal of product architectures can be a very arduous process. It is common that a successful technical asset in the beginning turns out to be a significant barrier to renewing the company. This results in legacy gradually taking control over the decisions and real changes not being made until the situation becomes critical.

Therefore, in recent times, such companies have been investigating solutions to this problem. One such solution is the use of modular architectures, wherein a device consists of a number of components, each responsible for a different application or service, connected by a network interconnect. An example of such a modular architecture is that of Network on a Terminal Architecture (NoTA).

In NoTA each component has mapped to it a service/services or application/applications and a communication stack, with each communications stack connected to the network interconnect. Figure 1 shows two interconnected NoTA communications stacks that logically enable two services to communicate. The first communication stack 1 comprises a first network stack 2, first middleware 3, a first operating system 4 and first bindings 5, and enables the first service 6 to communicate with a second service 7. Likewise, the second communication stack 8 comprises a second network stack 9, second middleware 10, a second operating system 11 and second bindings 12, and enables the second service 7 to communicate with the first service 6. The two communication stacks are connected through a network 13.

In this case the term network stack may relate to the first four layers of the Open Systems Interconnection Basic Reference Model (OSI Model). The first four layers of the OSI model are often known as the physical layer, the data link layer, the network layer and the transport layer. These four layers deal with the transport of data between applications and services and the network.

Middleware generally relates to software layers that mediate interaction between applications or services so that they can exchange data. The provision of middleware in a system also simplifies application development, by providing functions, such as masking the heterogeneity and distribution of the underlying hardware and operating systems, and the hiding of low-level programming details.

A NoTA hardware organisation of an electronic device may comprise a number of components connected through a network, each implementing a different application or service.

Each component may contain a microprocessor to which an application or service, and a NoTA communications stack (as in Figure 1) may be mapped. Each component may further comprise a physical network connection block to receive data from the network switch and a memory to store data. The microprocessor, the physical network connection block and the memory may all be connected to a BUS or another kind of communication network.

One main problem that exists with conventional modular architectures is that the components contain many implementation layers (such as those in Figure 1) with complex functionality. Therefore, the automatic implementation of these layers tends to result in a large code footprint, slow execution due to the component processor having to process the network traffic loads, and low power efficiency. Hence, such modular architectures tend not to be very competitive when compared to dedicated communication implementations with fewer channels.

Furthermore, generally, contemporary modular components are not truly independent, but instead tend to have interdependencies with other components in the system. This prevents, for example, optimal power management, due to more than one component being required to perform one task that might ideally be performed by a single component.

Another issue with contemporary modular components is that often the component processor is also responsible for moving incoming data to the memory. This means that the processor is unavailable for performing other more important tasks during this time.

Besides the renewal of architectures, another important issue that affects companies with complex high technology products is how to enable vendors to bring their intellectual property (IP) to a particular architecture. Traditionally, this has been difficult due to the confidentiality of future device architectures. Moreover, if a vendor develops IP for an existing architecture they may be too late, because new designs are likely to utilise new architectures.

Summary of the Invention

The present invention provides a component of an electronic device comprising a network connection processor, which comprises a physical network connection block to receive data from and transmit data to a network and a first data processor configured to process data arriving at the network connection processor, and a second data processor configured to process data received from the network connection processor.

The first data processor may be configured to process the data by implementing middleware and network stack functionalities. This improves the efficiency of the second data processor because the network interconnect traffic is dealt with by the network connection processor and therefore enables the second data processor to perform other tasks. Furthermore, the second data processor need not be over-dimensioned for network traffic.

The network connection processor may be decouplable from the component. This enables legacy systems to be adapted to the new architecture simply by incorporating the network connection processor. Similarly, this also enables forward integration whereby as long as the component includes a network connection processor, it can be connected to the network.

The second data processor may be configured to implement an application or service.

The component may further comprise a transport system configured to transport data between the network connection processor and the secondary data processor.

The component may have internal memory. This enables incoming data to be stored prior to being processed by the second data processor.

The transport system may be further configured to transport data between the network connection processor and the memory, and between the second data processor and the memory.

The component may have external memory, the external memory being exclusive to the component so as to reduce interdependency between components.

The network connection processor may further comprise memory. The memory may comprise circular buffers. This improves the overall performance of the network connection processor by enabling received data to be stored prior to processing.

The second data processor may be a field programmable gate array. This enables the component to be programmable and therefore able to implement a number of different services or applications as desired.

The transport system may be a BUS.

The present invention also provides an electronic device of modular construction comprising a plurality of components as described above. This enables the device to perform a number of different services or applications, with each being mapped to a different component.

The plurality of components may be connected by the network, and the electronic device may be a mobile phone.

The first data processor may be configured to implement firewall functionalities.

The present invention further provides a method comprising receiving data from a network, processing the received data, transporting the processed data to a memory, storing the processed data, transporting the processed data from the memory to a second data processor and further processing the processed data.

Processing the received data may comprise implementing network stack and middleware functionalities.

Further processing the processed data may comprise implementing an application or service.

Processing the received data may comprise implementing firewall functionalities.

Brief Description of the Drawings

Embodiments of the present invention will now be described, by way of example only, with reference to the accompanying drawings in which:

Figure 1 shows two interconnected NoTA communications stacks.
Figure 2 shows a mobile telephone terminal.
Figure 3 shows an exemplary modular network hardware organisation of the mobile telephone phone terminal of Figure 2.

Figure 4 shows a component of the modular network hardware organisation of Figure 3.

Figure 5 shows the flow of data within the component of Figure 4.

Figure 6 shows a conventional component of modular network hardware organisation.

Figure 7 shows the flow of data within the component of Figure 6.

Figure 8 shows a conceptual memory allocation for the network connection processor memory.

Figure 9 shows a component of the modular network hardware architecture wherein the service is mapped to a field programmable gate array.

Figure 10 shows a network connection processor as the network connection point of a complex computing architecture.

Figure 11 shows a network connection processor configured to implement a firewall.

Detailed Description

Figure 2 shows a mobile phone terminal. Visible are an aerial 14, a speaker 15, a display 16, a keypad 17 and an audio receiver 18.

Figure 3 shows an exemplary modular network hardware organisation of the mobile telephone depicted in Figure 2. The mobile phone comprises a plurality of components each relating to a different function of the mobile phone. The components comprise the aerial 14, the speaker 15, the display 16, the keypad 17, the audio receiver 18, a memory 19, a clock 20, a SIM card reader 21, an MP3 player 22, a video player 23, an image viewer 24, a modem 25. Each component is connected to a network switch 26.

Data may be passed from one component to another, via the network switch 26. For example, data received by the aerial 14 may be passed via the network switch 26 to the speaker 15 and likewise data received from the audio receiver 18 may be passed via the network switch 26 to the aerial 14.

Figure 4 shows an example of a component of the modular network hardware organisation of Figure 3. The component may be, for example, the video viewer 23. The component 23 comprises a network connection processor (NCP) 27, a component processor 28, a memory 29 and an internal connection such as a BUS 30. The network connection processor (NCP) 27 is a hardware processor.

The network connection processor 27 has an integrated physical network connection block (PNCB) 31, which is connected to the network switch 26 of the hardware organisation. The physical network connection block 31 may receive incoming data from the network switch 26, and conversely the network switch 26 may receive outgoing data from physical network connection block 31.

Mapped to the network connection processor 27 are network stack 2 and middleware 3 functionalities. These functionalities are responsible for the decoding of the incoming data and the determination of the destination of the incoming data, such as whether the data be passed to the memory 29 for retrieval at a later time, or whether it be passed to the memory 29 and on to the component processor 28 for immediate processing. Furthermore they are also responsible for, among other things, the transformation of the incoming data into a format that is recognized by the component processor 28. The network connection processor 27 may also be implemented so as to enforce memory locality, whereby it denies access to the component memory by a second component. This reduces the interdependency between components.

The presence of the network connection processor 27 means that the component processor 28 need not perform the above functions and thus the component processor 28 is free to perform other tasks.

It may be advisable, for performance reasons, for the network connection processor 27 to have internal memory 32. It may be used as buffers for DMA-like situations, temporal storage for protocol processing, temporal and configuration storage for the middleware 3, and as an instruction and/or data cache for the network connection processor 27. However, the network connection processor 27 may still

operate successfully without an internal memory. Instead, the network connection processor 27 could utilise the component memory 29.

The component processor 28 may be a microchip, which has mapped to it the service or application 6, the bindings 5, and operating system (OS) software 4. The component processor receives data of a recognized format from the network connection processor 27 and implements the service or the application 6.

The processing power of the network connection processor 27 may be relatively low in comparison to that of the component processor 27 due to the limited functionality it is required to implement.

It should be noted that although the component memory 29 has been depicted as embedded memory, this is only a logical restriction. The memory 29 may be physically outside of the component 23, but it is important that the memory is logically not accessible by anyone else other than the component 23 itself.

The provision of a network connection processor 27 in a component of modular-network hardware architecture is desirable due to the programmer-friendly interface provided towards the component processor, flexibility in DMA and other memory management schemes, and the possibility for software or firmware updates. The efficiency of the network connection processor 27 can be further enhanced with the use of application (or service) specific instructions, thus making the network connection processor 27 an Application Specific Instruction-set Processor (ASIP).

Furthermore, a network connection processor facilitates the implementation of highly modular systems. This may result in increased power efficiency, because where a particular function might have required two or more components to be operational in a previous, less modular system, a highly modular system may be able carry out the function with just one dedicated component.

The provision of a network connection processor 27 also enables unknown vendors to implement their IP, i.e. their software and its underlying functionality, with a

particular architecture. In conventional systems, in order to incorporate IP into a particular architecture, it is necessary for the vendor to know and understand the architecture so that they can integrate their IP with that architecture. This is sometimes a problem as a producer of the architecture may be reluctant to commit to purchase and thus disclose the details of the architecture without having sufficiently tested the vendor's product. The above-described embodiments allow that, instead of revealing the details of the architecture to an unknown vendor, the producer of the architecture is able to give the vendor a network connection processor 27, which the vendor can integrate with their IP to form a new component. The producer can then simply plug in the new component to existing prototyping environments in order to test it.

The provision of a network connection processor 27 in a component also provides significant improvements in performance when compared to conventional component implementation. Figure 5 shows the flow of data into and within a component, such as that in Figure 4. In this example, a data packet, comprising three individual data pieces, is incoming from the network and the component processor requires the first data piece for immediate processing. First, the data packet (d1, d2, d3) is received s1 from the network stack 26 by the network connection processor 27, where the destinations of the individual data pieces are determined. The network connection processor 27 then sends the whole data packet to the BUS 30 s2. The data packet is then transferred from the BUS 30 to the memory 29 s3. At this point, the required first data piece (d1) is passed from the memory 29 to the BUS 30 s4, and from the BUS 30 to the component processor 28 s5 for processing.

Figure 6 shows a conventional component of a modular network architecture, without a network connection processor. The component 33 comprises a component processor 34, a memory 35, a physical network connection block 36 and a BUS 37. The physical network connection block 36 is connected to the network switch 26. Mapped to the component processor are the service 6, bindings 5, an operating system 4, middleware 3 and the network stack 2.

Figure 7 shows the flow of data into and within the component of Figure 6. Again a data packet comprising three individual data pieces is incoming from the network and the component processor requires the first data piece for immediate processing.

First, the data packet (d1, d2, d3) is received t1 by the physical network connection block 36 from the network switch 26. The component processor then transfers the first piece of data (d1) from the physical network connection block 36 to the BUS 37 t2, and then from the BUS 37 to itself t3. The component processor 34 then determines the destination of the data piece. The data piece is then transferred back to the BUS 37 t4 and on to the memory 35 for storage t5. This process is then repeated for the other two data pieces. The second data piece (d2) is transferred from the physical network block 36 to the BUS 37 t6, from the BUS 37 to the component processor 34 t7, from the component processor 34 to the BUS 37 t8, and finally from the BUS 37 to the memory 35 for storage t9. The same operations occur with the third data piece and comprises a further four data transfers (t10 to t13). When all three data pieces have be transferred to the memory, the first data piece is recalled from the memory 35 to the BUS 37 t14, and then onto the component processor 34 for processing t15.

From a comparison of Figures 5 and 7, it is clear that the provision of the network connection processor 27 dramatically reduces the data traffic within the component.

Figure 8 depicts the conceptual memory allocation for the network connection processor memory 32. The code and stack 38 portion may be used to store instructions and variables for the operating system, the network stack, the middleware, and any low-level control software that the network connection processor requires in order to operate. The memory may further comprise circular buffers which comprise a first transmit buffer 39, a second transmit buffer 40 and a third transmit buffer 41, and also a first receive buffer 42, a second receive buffer 43 and a third receive buffer 44. The heap 45 controls the allocation of the buffers

The number of buffers does not need to be high, but it is advantageous to have several so that they can be rotated. That is, when a receive buffer becomes full, it is

converted into a transmit buffer and it becomes read-only. After the contents of the transmit buffer have been transferred off the network connection processor, the allocated memory space of the buffer may be freed.

The use of the network connection processor is not limited to the example of Figure 4. For example, if the service or application is very small, such as a single sensor, but a connection to the network is still desired, the service or application may be mapped to the network connection processor rather than to a separate component processor.

Similarly, hardware-centric IP can also be integrated with a network connection processor. Depending on the complexity of the service, the network connection processor may need to decode the service messages or the hardware may be sufficiently intelligent to do it on its own. Field programmable gate arrays could be used to provide a programmable hardware service. By providing the field programmable gate array with new configuration bitstreams, it could be used as a different service every time. Figure 9 illustrates a component 46 comprising a field programmable gate array 47, a memory 48, a BUS 49 and a network connection processor 27 (as in figure 4).

Figure 10 shows a network connection processor as the connection point of a complex computing architecture. A component 50 comprises a network connection processor 27, a component processor 51 with dedicated memory 52, a hardware accelerator 53, component memory 54, a BUS 55 and a bridge 56. The network connection processor 27 is connected to the network switch 26. The bridge 56 is connected to the BUS 55 and the peripheral BUS 57, which may be the interconnect of another component or even an entire network.

In addition to the features described above, the network connection processor may also be configured to implement firewall functionalities. Figure 11 shows a network connection processor including such a capability. In addition to the features of the above-described embodiments, the network connection processor 27 comprises a firewall 58 interposed between the network stack 2 and the PNCB 31.

The firewall functionality may be implemented by a network layer firewall. A network layer filter may also be known as a packet filter. As such, the firewall may be configured to filter out incoming data from unauthorized sources. The firewall may alternatively or additionally be configured to filter out incoming data that does not comply with predetermined allowed data patterns. This allows data that matches the predetermined allowed patterns and/or data incoming from authorized sources to pass. Other data is discarded. A user of the hardware organisation may define the authorised sources and/or allowed data patterns. Alternatively, the authorised sources and/or allowed data patterns may be pre-stored in the hardware organisation. The firewall may comprise a stateful network layer firewall or a stateless network layer firewall.

The firewall functionalities may alternatively be implemented by a firewall other than of the network layer firewall type.

It should be realised that the foregoing examples should not be construed as limiting. Other variations and modifications will be apparent to persons skilled in the art upon reading the present application. Moreover, the disclosure of the present application should be understood to include any novel features or any novel combination of features either explicitly or implicitly disclosed herein or any generalisation thereof and during the prosecution of the present application or of any application derived therefrom, new claims may be formulated to cover any such features and/or combination of such features.

Claims

1. A component of an electronic device comprising:

a network connection processor, which comprises a physical network connection block to receive data from and transmit data to a network and a first data processor configured to process data arriving at the network connection processor; and

a second data processor configured to process data received from the network connection processor.

2. The component according to claim 1, wherein the first data processor is configured to process the data by implementing middleware and network stack functionalities.

3. The component according to either one of claims 1 or 2, wherein the network connection processor may be decouplable from the component.

4. The component according to any one of claims 1 to 3, wherein the second data processor is configured to implement an application or service.

5. The component according to any one of claims 1 to 4 further comprising a transport system configured to transport data between the network connection processor and the second data processor.

6. The component according to any one of claims 1 to 5, wherein the component has internal memory.

7. The component according to claim 6, wherein the transport system is further configured to transport data between the network connection processor and the memory, and between the second data processor and the memory.

8. The component according to any one of claims 1 to 5, wherein the component has external memory, the external memory being exclusive to the component.

9. The component according to any one of claims 1 to 8, wherein the network connection processor further comprises memory.

10. The component according to any one of claims 6 to 9 wherein the memory comprises circular buffers.

11. The component according to any one of claims 1 to 10, wherein the second data processor is a field programmable gate array.

12. The component according to any one of claims 5 to 11, wherein the transport system is a BUS.

13. An electronic device of modular construction comprising a plurality of components, each of the plurality of components according to any one of claims 1 to 12.

14. The electronic device according to claim 13, wherein the plurality of components are connected by the network.

15. An electronic device according either one of claims 13 or 14, wherein the electronic device is a mobile phone.

16. A method comprising:
receiving data from a network;
processing the received data;
transporting the processed data to a memory;
storing the processed data;
transporting the processed data from the memory to a second data processor; and

further processing the processed data.

17. A method according to claim 15, wherein processing the received data comprises implementing network stack and middleware functionalities.

18. A method according to either one of claims 15 or 16, wherein further processing the processed data comprises implementing an application or service.

19. A component according to any of claims 1 to 12, wherein the first data processor is configured to implement a firewall.

20. A method according to any one of claims 16 to 18, wherein processing the received data comprises implementing a firewall.

21. A component of an electronic device comprising:

a network connection processor, which comprises a physical network connection block to receive data from and transmit data to a network and a first processing means configured to process data arriving at the network connection processor; and

a second processing means configured to process data received from the network connection processor.

22. The component according to claim 21, wherein the first processing means is configured to process the data by implementing middleware and network stack functionalities.

23. The component according to either one of claims 21 or 22, wherein the network connection processor may be decouplable from the component.

24. The component according to any one of claims 21 to 23, wherein the second processing means is configured to implement an application or service.

25. The component according to any one of claims 21 to 24 further comprising a transport system configured to transport data between the network connection processor and the second processing means.

26. The component according to any one of claims 21 to 25, wherein the component has internal memory means.

27. The component according to claim 26, wherein the transport system is further configured to transport data between the network connection processor and the memory means, and between the second processing means and the memory means.

28. The component according to any one of claims 21 to 25, wherein the component has external memory means, the external memory means being exclusive to the component.

29. The component according to any one of claims 21 to 28, wherein the network connection processor further comprises memory means.

30. The component according to any one of claims 26 to 29 wherein the memory means comprises circular buffers.

31. The component according to any one of claims 21 to 30, wherein the second processing means is a field programmable gate array.

32. The component according to any one of claims 25 to 31, wherein the transport means is a BUS.

33. An electronic device of modular construction comprising a plurality of components, each of the plurality of components according to any one of claims 21 to 32.

34. The electronic device according to claim 33, wherein the plurality of components are connected by the network.

35. An electronic device according either one of claims 33 or 34, wherein the electronic device is a mobile phone.

36. A method comprising:
receiving data from a network;
processing the received data;
transporting the processed data to a memory means;
storing the processed data;
transporting the processed data from the memory means to a second processing means; and
further processing the processed data.

37. A method according to claim 35, wherein processing the received data comprises implementing network stack and middleware functionalities.

38. A method according to either one of claims 35 or 36, wherein further processing the processed data comprises implementing an application or service.

39. A component according to any of claims 21 to 32, wherein the first processing means is configured to implement a firewall.

40. A method according to any one of claims 36 to 38, wherein processing the received data comprises implementing a firewall.

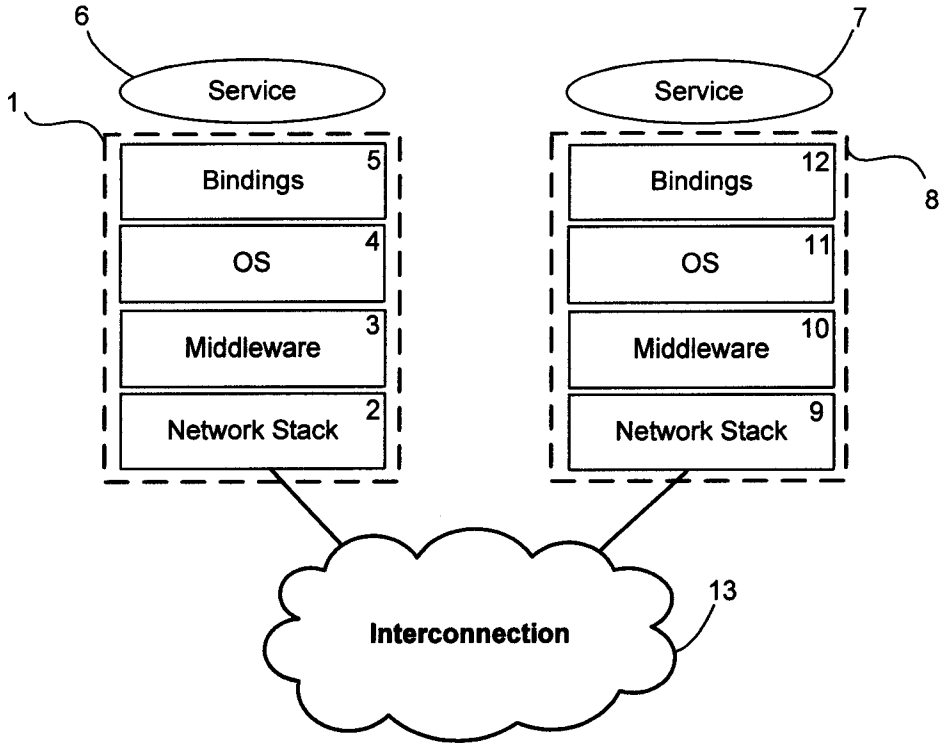
6
Service
7
Service
1
Bindings
5
OS
4
Middleware
3
Network Stack
2
Bindings
12
OS
11
Middleware
10
Network Stack
9
8
Interconnection
13

Fig. 1

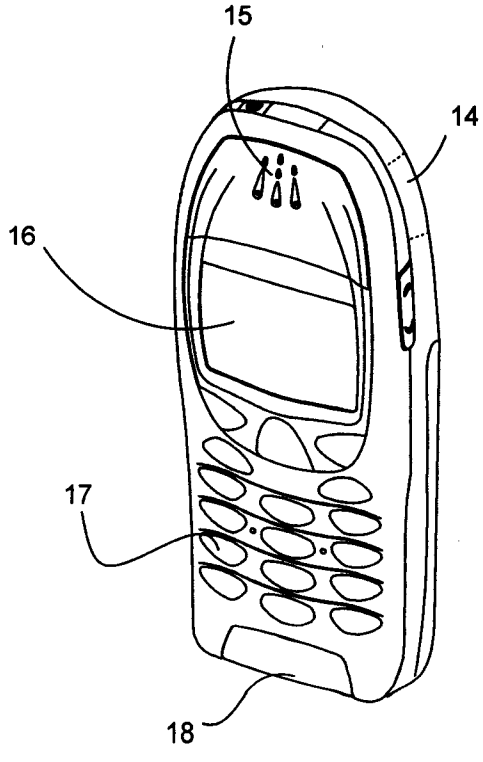
15
14
16
17
18

Fig. 2

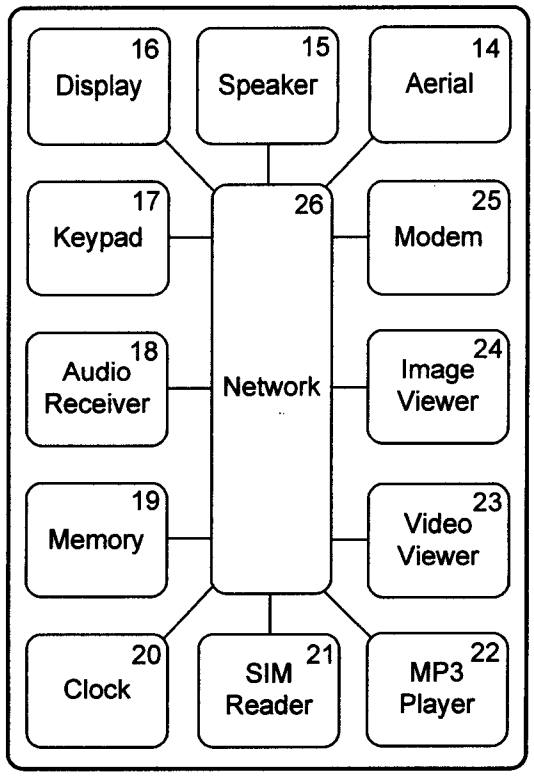
16
Display
15
Speaker
14
Aerial
17
Keypad
26
Network
25
Modem
18
Audio Receiver
24
Image Viewer
19
Memory
23
Video Viewer
20
Clock
21
SIM Reader
22
MP3 Player

Fig. 3

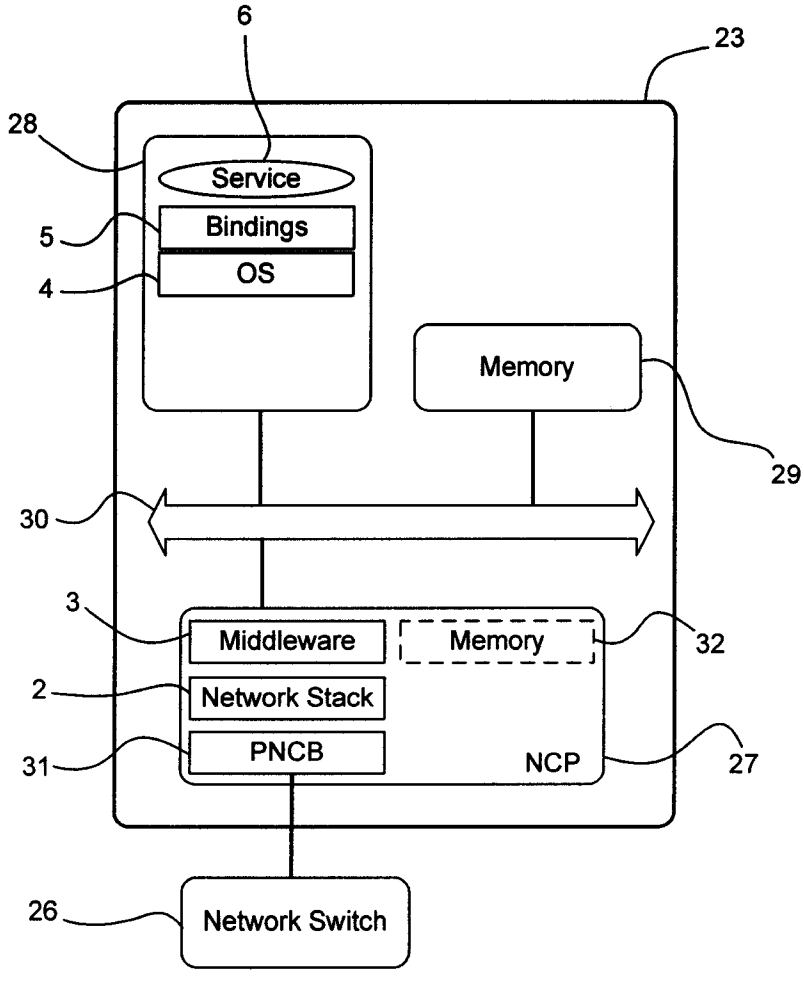
6
23
28
Service
5
Bindings
4
OS
Memory
29
30
3
Middleware
Memory
32
2
Network Stack
31
PNCB
NCP
27
26
Network Switch

Fig. 4

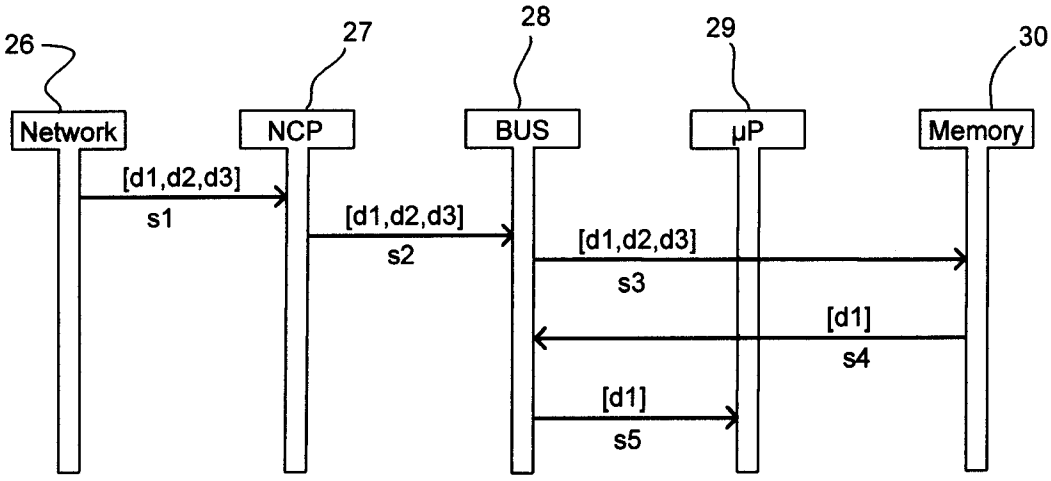
26
27
28
29
30
Network
NCP
BUS
µP
Memory
[d1,d2,d3]
s1
[d1,d2,d3]
s2
[d1,d2,d3]
s3
[d1]
s4
[d1]
s5

Fig. 5

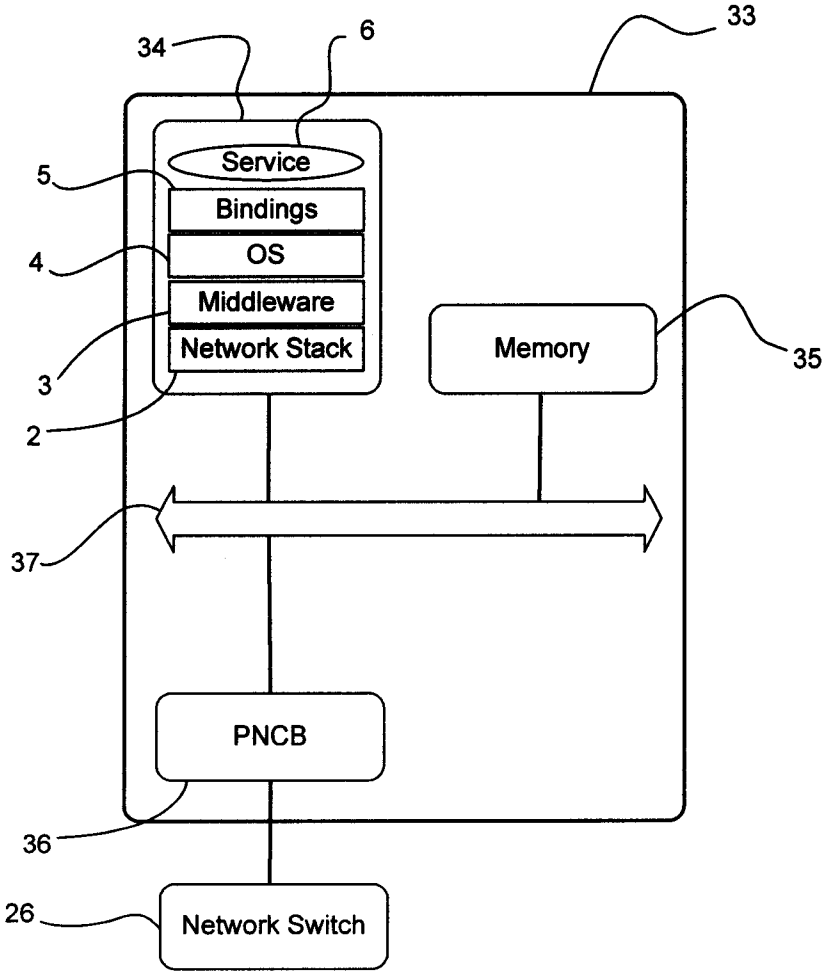
33
34
6
Service
5
Bindings
4
OS
Middleware
Network Stack
3
2
Memory
35
37
PNCB
36
26
Network Switch

Fig. 6

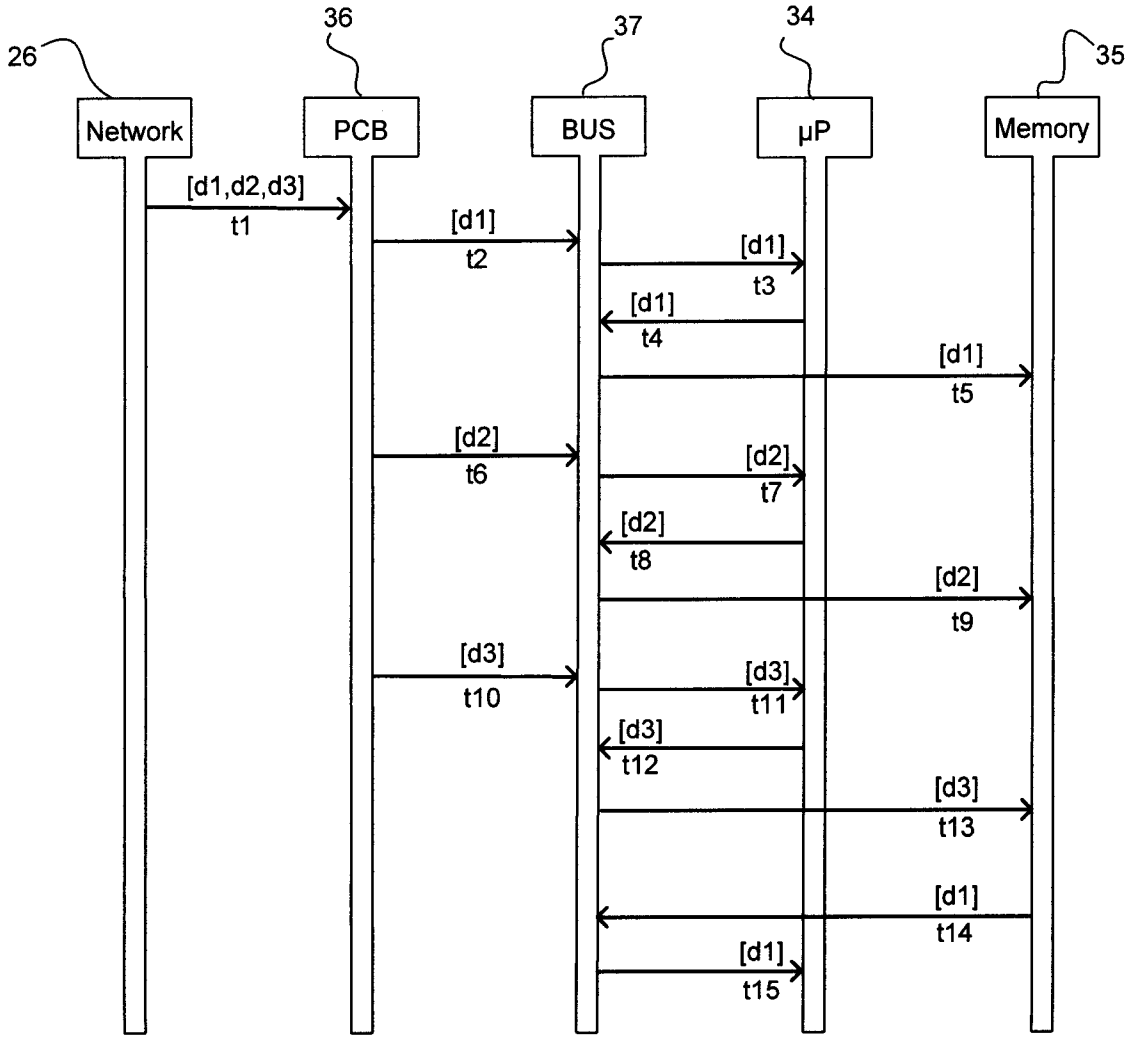
26
36
37
34
35
Network
PCB
BUS
µP
Memory
[d1,d2,d3]
t1
[d1]
t2
[d1]
t3
[d1]
t4
[d1]
t5
[d2]
t6
[d2]
t7
[d2]
t8
[d2]
t9
[d3]
t10
[d3]
t11
[d3]
t12
[d3]
t13
[d1]
t14
[d1]
t15

Fig. 7

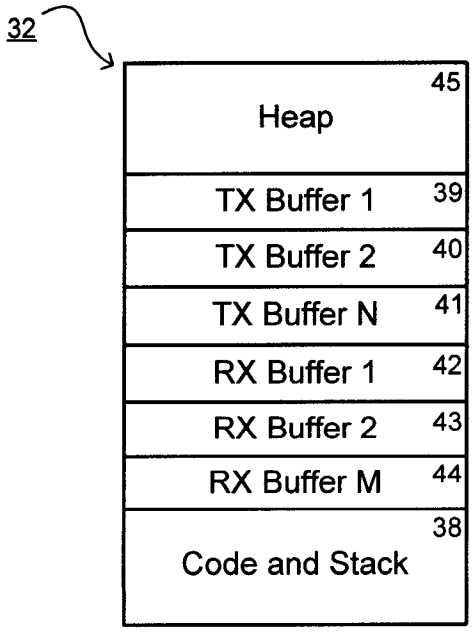
32
Heap
45
TX Buffer 1
39
TX Buffer 2
40
TX Buffer N
41
RX Buffer 1
42
RX Buffer 2
43
RX Buffer M
44
Code and Stack
38

Fig. 8

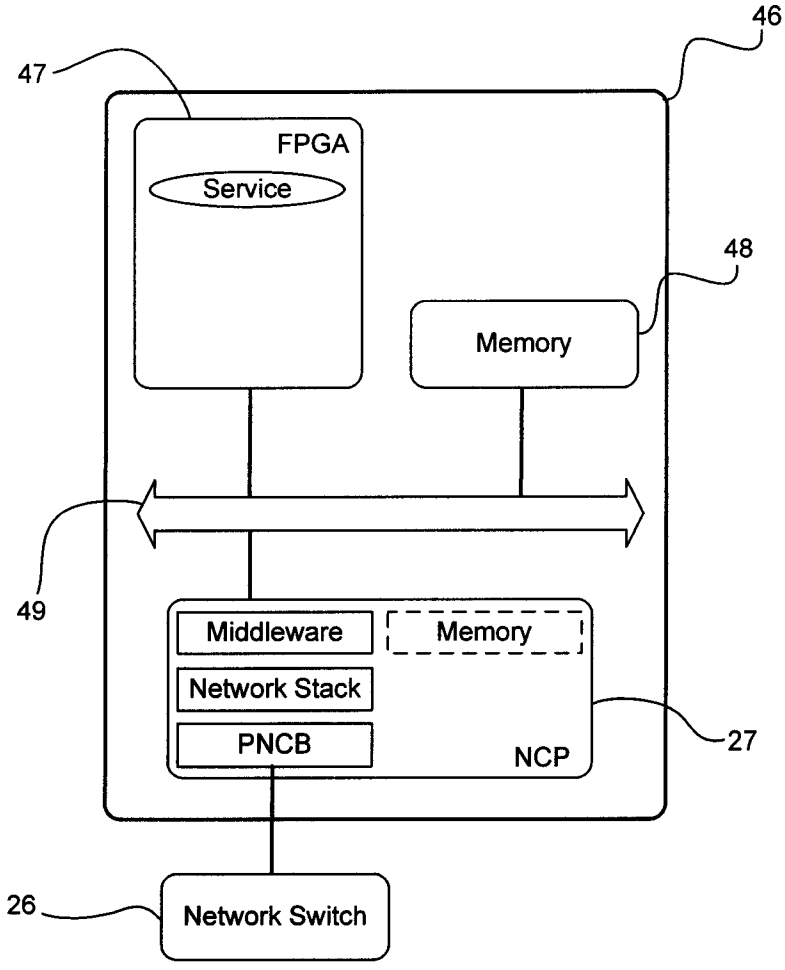
46
47
FPGA
Service
48
Memory
49
Middleware
Memory
Network Stack
PNCB
NCP
27
26
Network Switch

Fig. 9

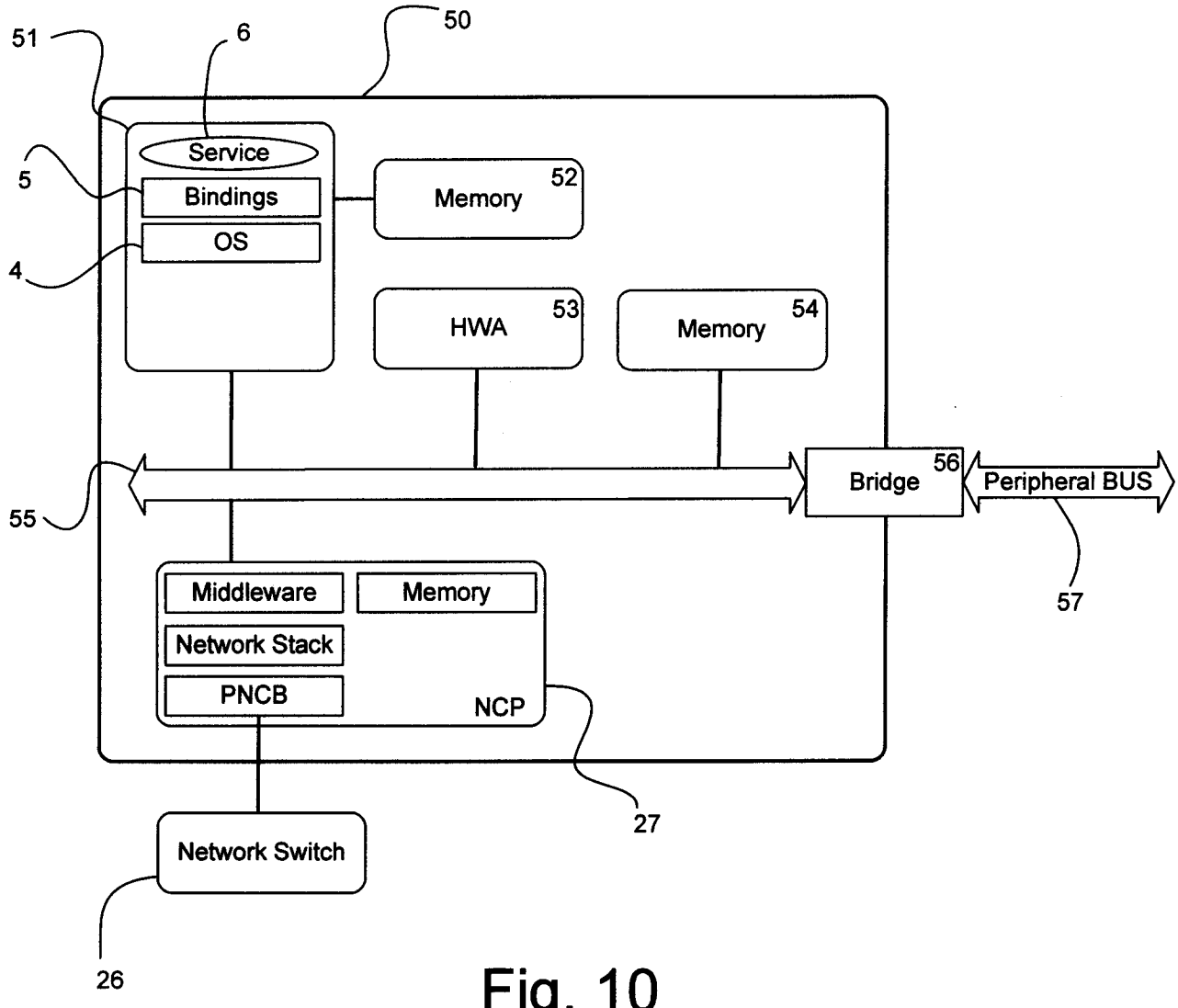
51
6
50
Service
5
Bindings
OS
4
Memory
52
HWA
53
Memory
54
Bridge
56
Peripheral BUS
55
57
Middleware
Memory
Network Stack
PNCB
NCP
27
Network Switch
26

Fig. 10

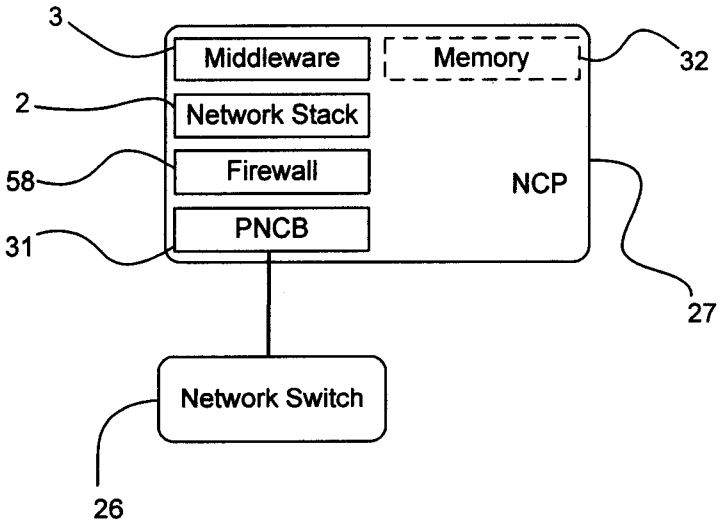
3
Middleware
Memory
32
2
Network Stack
58
Firewall
NCP
31
PNCB
27
Network Switch
26

Fig. 11

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2008/053630

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06F15/17

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06F H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal, WPI Data, INSPEC, COMPENDEX, IBM-TDB

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	DE 10 2004 035843 A1 (INFINEON TECHNOLOGIES AG [DE]) 16 February 2006 (2006-02-16) abstract paragraph [0003] - paragraph [0004] paragraph [0026] - paragraph [0027] paragraph [0035] - paragraph [0036] paragraph [0046] - paragraph [0047] paragraph [0065] paragraph [0084] - paragraph [0130] figures 4-7 ----- -/--	1-40

[X] Further documents are listed in the continuation of Box C.

[X] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
20 August 2008	27/08/2008

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Jonsson, Svante

2

Form PCT/ISA/210 (second sheet) (April 2005)

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2004/042483 A1 (ELZUR URI [US] ET AL ELZUR URI [US] ET AL) 4 March 2004 (2004-03-04) abstract paragraph [0007] paragraph [0014] - paragraph [0015] paragraph [0020] - paragraph [0022] paragraph [0049] - paragraph [0052] paragraph [0072]	16-18, 20, 36-38,40
X	US 2004/257986 A1 (JHA ASHUTOSH K [US] ET AL) 23 December 2004 (2004-12-23)	16-18, 20, 36-38,40
A	paragraph [0006] - paragraph [0008] paragraph [0039] - paragraph [0040] paragraph [0048] paragraph [0053] - paragraph [0055] paragraph [0068] - paragraph [0071]	10,30
A	US 2005/021871 A1 (GEORGIOU CHRISTOS J [US] ET AL) 27 January 2005 (2005-01-27) abstract paragraph [0038] - paragraph [0048]	1-40
A	SUORANTA R ED - KAMALA ET AL: "New Directions in Mobile Device Architectures" DIGITAL SYSTEM DESIGN: ARCHITECTURES, METHODS AND TOOLS, 2006. DSD 200 6. 9TH EUROMICRO CONFERENCE ON, IEEE, PI, 1 January 2006 (2006-01-01), pages 17-26, XP031102259 ISBN: 978-0-7695-2609-6 the whole document	1-40

2

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/EP2008/053630

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
DE 102004035843 A1	16-02-2006	WO 2006010353 A1	02-02-2006
		EP 1774434 A1	18-04-2007
		US 2007294509 A1	20-12-2007
US 2004042483 A1	04-03-2004	US 2008151922 A1	26-06-2008
US 2004257986 A1	23-12-2004	US 2008056124 A1	06-03-2008
US 2005021871 A1	27-01-2005	CN 1910571 A	07-02-2007
		JP 2005044361 A	17-02-2005
		KR 20050013076 A	02-02-2005
		TW 291104 B	11-12-2007
		US 2007239966 A1	11-10-2007
		US 2005021874 A1	27-01-2005